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

Admission Control for Providing QoS in Wireless Mesh Networks

Qiang Shen[†‡], Xuming Fang[†]
[†]Provincial Key Lab of Information Coding and Transmission
Southwest Jiaotong University, Chengdu, 610031 P. R. China
q.shen@ufl.edu, xmfang@home.swjtu.edu.cn

Pan Li[‡] and Yuguang Fang[‡]
[‡]Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611 USA
{lipanleo@, fang@ece.}ufl.edu

***Abstract*—An admission control algorithm should be properly designed to guarantee the Quality of Service (QoS) in wireless mesh networks (WMNs). Based on channel busyness ratio, an admission control algorithm (ACA) is proposed to provide QoS for realtime and non-realtime traffic. For realtime traffic, all the nodes on a route make the admission control decision based on the estimation of available bandwidth. For non-realtime traffic, a rate adaption algorithm is proposed to adjust the sending rates of the source nodes to prevent a network from entering a saturated status. Finally, we demonstrate the effectiveness by simulations in NS-2.**

I. INTRODUCTION

Wireless mesh networks (WMNs) have become a critical part of the Internet. They are widely deployed in many scenarios, such as campus networking, community networking, and so on. However, how to provide proper QoS for multimedia traffic is a crucial problem that has not been well addressed in the existing literature.

Recently, a few schemes based on admission control [1-5] have been proposed to provide QoS in wireless ad hoc or mesh networks. In INSIGNIA [1], in-band signaling allows it to quickly restore flow state when topology changes occur. In SWAN [2], the admission control mechanism collects the bandwidth information by a node listening to all the transmissions in its transmission range. Unfortunately, probing introduces lots of overhead and may not obtain an accurate value if a probe is lost.

The Contention-aware Admission Control Protocol (CACP) [3] and Perceptive Admission Control (PAC) [4] are both protocols that enable a high QoS by limiting the flows in the networks. However, CACP has significant overhead since packet transmission using high power affects the ongoing transmission significantly. For PAC, the extension of the sensing range will decrease the spatial reuse, and it will lead to some incorrect rejection decisions. Meanwhile, both CACP and PAC assume the available bandwidth has a fixed linear relationship to that idle channel time. In [5], Wei et al. propose a call admission control method for wireless mesh networks and their scheme is based on the interference capacity in chain topology. Zhai et al. [6,7] propose the original definition of channel busyness ratio in wireless local area networks (WLANs), and based on the channel busyness ratio, they propose a call admission and rate control scheme for VoIP and the best effort traffic, respectively.

In this paper we propose an admission control algorithm (ACA) to provide QoS in wireless mesh networks. ACA treats various traffic in different ways. It offers an admission decision to realtime traffic, but provides a sending rate for non-realtime traffic. As wireless resource is scarce, ACA also adjusts all the sending rates of ongoing non-realtime flows to maintain the ongoing realtime flows' QoS.

The rest of the paper is organized as follows, Section II presents the estimation of available bandwidth. Section III describes the proposed admission control algorithm in detail. After that, Section IV shows the effectiveness of ACA by simulations. Finally, Section V concludes our work.

II. ESTIMATION OF AVAILABLE BANDWIDTH

Since it has been shown that channel busyness ratio provides an efficient way to estimate the resource in WLANs [6,7], we attempt to study this issue in WMNs in this section.

A. Channel busyness ratio in WLANs

As described in [6], a slot could be an empty one, one with a successful packet transmission, or one with a collision. Let p_i, p_s, p_c be the probabilities that the observed slot is one of those three kinds, respectively. Also, let T_{suc} and T_{col} be the average time periods associated with one successful transmission and collision, respectively. In the case that RTS/CTS mechanism is used, we have

$$T_{suc} = T_{rts} + T_{cts} + T_{data} + T_{ack} + 3T_{sifs} + T_{difs}$$
$$T_{col} = T_{rts} + T_{cts\_timeout} + T_{difs} = T_{rts} + T_{eifs}$$

Then, under the assumption that there is no hidden terminal, we can obtain [6]:

$$R_i = \frac{p_i\sigma}{p_i\sigma + p_s T_{suc} + p_c T_{col}}$$
$$R_b = 1 - R_i$$
$$R_s = \frac{p_s T_{suc}}{p_i\sigma + p_s T_{suc} + p_c T_{col}}$$

where σ is the length of a time slot, R_i is defined as the channel idleness ratio, R_b the channel busyness ratio, and R_s the channel utilization, respectively. In WLANs, the observed node can distinguish the successful transmission, collision and idle exactly. However, in WMNs, there will be transmissions both inside and outside its transmission range. In this case, we need to use a different approach to estimate the available bandwidth.

B. Estimation of packet successful transmission probability

Instead of assuming the successful transmission probability of RTS being the same as that of the whole packet, we take the successful transmission probabilities of RTS, CTS, DATA and ACK into consideration. Let p_{srts}, p_{scts}, p_{sdata} and p_{sack} be the successful transmission probabilities of RTS, CTS, DATA and ACK frames, respectively, and p_{crts}, p_{ccts}, p_{cdata} and p_{cack} be the collision probabilities of RTS, CTS, DATA and ACK frames, respectively. Also, we assume that n is the total number of nodes in the observed node's sensing range, n_1 the number of hidden terminals, and p_t the average transmission probability for each node. According to the characteristics of IEEE 802.11 MAC in multi-hop networks, we assume $p_{scts} = 1$ and $p_{sack} = 1$, namely, $p_{ccts} = 0$ and $p_{cack} = 0$.

Let p_{ss_i} denote the successful transmission probability in slot i, then

$$p_{srts} = \sum_{i=0}^{r-1} (1 - p_{srts1})^i p_{srts1}$$
$$p_{srts1} = \prod_{i=1}^{\lceil T_{rts}/T_{slot} \rceil} p_{ss_i}$$
$$p_{crts} = 1 - p_{srts}$$

where r is the $ShortRetryLimit$ defined as the retransmission times of RTS in 802.11, and the value in the standard is 7. Meanwhile, p_{srts1} is the successful transmission probability of a single RTS transmission. p_{ss_i} is shown as:

$$p_{ss_i} = \begin{cases} (1 - p_t)^{n+n_1-1} , & \text{i=1;} \\ (1 - p_t)^{n_1} , & 2 \leq \text{i} \leq \lceil T_{rts}/T_{slot} \rceil \end{cases}$$

Generally speaking, the sensing range will not be smaller than twice of the transmission range, and in this case the successful transmission probability of the DATA frame can be derived as follows:

$$p_{sdata} = \prod_{i=1}^{\lceil T_{data}/T_{slot} \rceil} p'_{ss_i}$$
$$p_{cdata} = 1 - p_{sdata}$$

where

$$p'_{ss_i} = \begin{cases} 1 , & if \ \text{i} \leq \lceil (T_{eifs} - T_{sifs})/T_{slot} \rceil; \\ (1 - p_t)^{n_1} , & \text{otherwise} \end{cases}$$

Finally, we can obtain the successful transmission probability of a whole packet, denoted as $p_{spacket}$:

$$p_{spacket} = p_{srts} p_{sdata}$$

C. Sensing-range bandwidth

The channel busyness ratio in WMNs is contributed by two parts: one is the transmission in its transmission range which is the same as that in WLANs, and the other part is the transmission in its sensing region excluding its transmission range.

Considering the retransmission of RTS and the backoff time, the time duration of RTS-CTS transmission should be:

$$T_{rts+cts} = \sum_{i=0}^{r-1} (1 - p_{srts1})^i p_{srts1} (iT_{col} + T_{rts} + T_{cts} + T_{sifs})$$

The successful transmission time duration T_s and the collision transmission duration T_c in WMNs are as follows:

$$T_s = T_{rts+cts} + 2T_{sifs} + T_{data} + T_{ack} + T_{difs}$$
$$T_c = (1 - p_{srts}) r T_{col} + p_{srts} p_{cdata} T_s$$

Besides the change above, R_i, R_b and R_s are changed as follows:

$$R_i = \frac{p_i \sigma}{p_i \sigma + p_s T_s + p_c T_c}$$
$$R_b = 1 - R_i$$
$$R_s = \frac{p_s T_s}{p_i \sigma + p_s T_s + p_c T_c}$$

where, p_i, p_s and p_c is defined as follows:

$$p_i = (1 - p_t)^n$$
$$p_s = n p_t p_{spacket}$$
$$p_c = 1 - p_i - p_s$$

Finally, the normalized bandwidth in the observed node's sensing range, denoted by s, is expressed as follows:

$$s = R_s \times T_{data}/T_s$$

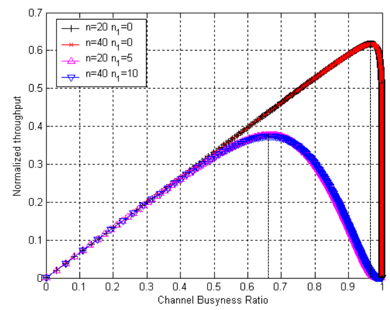


Fig. 1. Channel busyness ratio

Assuming the packet size is 512 bytes, we can obtain the numerical results illustrated in Fig. 1. We observe that the optimal operation points are different when $\frac{n_1}{n}$ changes. The case when n_1=0 corresponds to what is proposed in [6].

D. Estimation without neighbor information

As we know, it is difficult to determine the number of nodes in the observed node's sensing range and the number of hidden terminals. Fortunately, according to the above discussion, we can obtain that: p_{sdata} is a function of p_t and n_1; R_b is a function of p_t, n and n_1. If we assume n as a constant, p_{sdata} and R_b can be obtained by monitoring the communication in MAC layer, with such information, p_t and n_1 can be estimated. As shown in Fig. 1, the normalized throughput is only related to the $\frac{n_1}{n}$. Thus, we can get the maximum bandwidth, used bandwidth and available bandwidth, denoted by B_{max}, B_{use} and B_a, respectively. For example, if p_{sdata}=0 and R_b=0.75, the curve we obtain is the same as $n = 20$ and $n_1 = 0$ in Fig. 1. At this time, B_{max}=0.62, B_{use}=0.495 and B_a=0.125.

III. Admission Control Algorithm

ACA uses admission control and rate adaption scheme to maintain the network operating at an unsaturated status.

A. Realtime traffic

To provide minimum QoS for non-realtime traffic, we need to set an upper band, denoted by $B_{r_{max}}$, for the bandwidth consumed by realtime traffic. For example, we can set $B_{r_{max}}$ to 80% of threshold bandwidth, denoted by B_{th}, and B_{th} is 85% of the maximum bandwidth. Both of $B_{r_{max}}$ and B_{th} are adaptable to the traffic configuration of the specific network. In this way, we can guarantee that the non-realtime traffic can occupy at least 20% of the threshold bandwidth. (B_{max}-$B_{r_{max}}$) is reserved for the route discovery and some other control messages. Furthermore, we assume that the gateway is always the source or destination, so the proportion of various traffic can be handled properly by the gateway.

Bandwidth is the most important factor in determining whether QoS can be satisfied or not. So, three parameters, (B_{ave_i}, B_{peak_i}, Len_i), are used to describe the basic wireless resource requirement for flow i. B_{ave_i} is the average data rate of flow i, B_{peak_i} is the peak data rate, and Len_i is the average packet length in bits.

The total bandwidth occupied by all the admitted flows is recorded by ACA when a flow joins or leaves a network, and it is denoted by (B_{ave_a}, B_{peak_a}). Unfortunately, due to the route break and mobility, it is very expensive to refresh those information in time for the nodes other than sources and destinations. For instance, once a link is broken, the source cannot notify the nodes between the broken link and the destination, indicating that this flow's information cannot be used anymore. If that information is not cleared, expired flows will still occupy the resource. Hence, we can use (B_{ave_a}, B_{peak_a}) only at the gateway because they are always sources or destinations in WMNs.

Estimation of bandwidth consumption for a single flow is another problem that needs to be addressed by admission control protocol. We assume the route to the destination is known before performing admission control. When we use AODV as the routing protocol, we can obtain the previous hop ($prehop$), next hop ($nexthop$), the number of hops (m_1) to the source (S) and the number of hops (m_2) to the destination (D). Since the sensing range is always between two and three times of the transmission range, we can estimate the number of hops (m) in the observed node's sensing range in the following way:

$$\begin{aligned} &if\ (m_1 > 2) \quad h_1 = 2\ else\ h_1 = m_1 \\ &if\ (m_2 > 2) \quad h_2 = 2\ else\ h_2 = m_2 \\ &m = h_1 + h_2 \end{aligned}$$

Based on those information, the bandwidth consumed by flow i can be estimated as follows:

$$\Gamma(B_{ave_i}) = mB_{ave_i}$$

In addition, B_{ave_a} is defined as the sum of different $\Gamma(B_{ave_i})$ for realtime flows in the observed node's buffer. In the same way, we can calculate B_{peak_a} corresponding to all B_{peak_i} for realtime flows.

When receiving a real-time connection request from the application layer, a node checks whether it has enough resource to establish this new flow. If so, it initiates an admission request to the destination to verify whether all the other nodes on the path have enough resource to accommodate this flow.

For ACA of all the nodes except for the gateway, the admission decision must base on the information collected by themselves. Those nodes cannot use B_{ave_a} and B_{peak_a} because they may be outdated. Instead, they can obtain the ratio of various traffic by monitoring the channel busyness ratio during a small period of time. Denote by R_{real} the contribution from realtime traffic to channel busyness ratio. Then, we must maintain

$$R_{real}B_{use} + \Gamma(B_{ave_{new}}) \leq B_{r_{max}} \quad (1)$$
$$R_{real}B_{use} + \Gamma(B_{peak_{new}}) \leq B_{th} \quad (2)$$

If both Eq. (1) and Eq. (2) are satisfied, ACA will accept this flow locally, and then forward this request to the next hop of the path. Otherwise, it initiates an admission reply with a rejection decision immediately without considering the other nodes along this path. If this admission request arrives at the destination, the destination will issue an admission reply with the final decision. After receiving the admission reply, the source node is notified the admission result. If this flow is accepted, the source will send packets stored in its buffer for the flow to the destination.

For ACA at the gateway, after receiving the admission request, if all the following constraints are satisfied, this application will be accepted locally:

$$\begin{aligned} B_{ave_a} + \Gamma(B_{ave_{new}}) &\leq B_{r_{max}} \\ B_{peak_a} + \Gamma(B_{ave_{new}}) &\leq B_{th} \end{aligned}$$

After a flow finishes, the source sends a termination request to the destination to release all the resource allocated to this flow. Apparently, Eq. (1) and Eq. (2) depend on the accuracy of R_{real}. Larger estimation of R_{real} leads to smaller ratio of realtime traffic. Since it is difficult to estimate R_{real} accurately, we assume the traffic ratio that a node monitors in its communication range is the same as the ratio in its sensing range. For the purpose of differentiating realtime packets and non-realtime packets, one reserved bit in MAC header is used.

The observed channel busyness ratio is contributed by three parts: one from realtime traffic with a decodable MAC header R_{b_1}, the other from non-realtime traffic with a decodable MAC header R_{b_2} and the third from undecodable MAC header, denoted by R_{b_3}, due to various reasons such as collisions or transmissions in its sensing range but not in its transmission range. The approximation of R_{real} is calculated as follows:

$$R_{real} \approx R_{b_1} \times \left(1 + \frac{R_{b_3}}{R_{b_1}+R_{b_2}}\right) = \frac{R_{b_1} \times R_b}{R_{b_1}+R_{b_2}}$$

where we assume R_{b_3} is composed of realtime traffic and non-realtime traffic according to the ratio of $\frac{R_{b_1}}{R_{b_2}}$.

B. Non-realtime traffic

Rate adaption scheme is designed for the best effort traffic to adjust its sending rate according to the network status. When there is a non-realtime connection request, ACA is used to determine a suitable initial sending rate.

Obviously, an initial sending rate should be determined firstly for a non-realtime flow. If B_{use} is larger than B_{th} at any node on the path, namely, the path works on a saturated status, we will set the initial sending rate, $B_{nr_{i,j}}$ for flow i at node j, a default value, say one packet per second. Otherwise, all the nodes except the gateway will use the following equation to determine their local initial sending rates:

$$B_{nr_{i,j}} = \begin{cases} \Gamma^{-1}(B_{th} - B_{use}), & if\ \Gamma(B_{ave_i}) > B_{th} - B_{use} \\ B_{ave_i}, & otherwise \end{cases}$$

For the gateway, how to estimate its local initial sending rate is different from the other nodes. Recall that the bandwidth consumed by realtime traffic is (B_{ave_a},B_{peak_a}). Then, the maximum bandwidth that non-realtime traffic can occupy at the gateway, denoted by $B_{nr_{max}}$, is described as follows:

$$B_{nr_{max}} = \begin{cases} B_{th} - B_{peak_a}, & if\ B_{peak_a} < 0.8B_{th} \\ B_{th} - B_{r_{max}}, & otherwise \end{cases}$$

If the bandwidth consumed by all non-realtime traffic, denoted by $B_{nr_{con}}$, is smaller than $B_{nr_{max}}$, all their maximum application bandwidth will be allocated. $B_{nr_{con}}$ is defined as the sum of $\Gamma(B_{ave_i})$ corresponding to all the non-realtime flows in the gateway's buffer. Otherwise, we can calculate the initial sending rate, denoted by B_{nr_i}, for non-realtime flow i at the gateway according to the following rule:

$$B_{nr_{i,g}} = \frac{B_{nr_{max}}}{B_{nr_{con}}} B_{ave_i}$$

After all kinds of acceptance decisions are made, if the sending rate of other flows needs to be adjusted, the destination will send an adjustment notification to the source with a new sending rate. In this way, we can maintain a rough fairness among non-realtime flows. The initial rate should be the minimum sending rate among the local initial rate of all the nodes on the path from the source to destination:

$$B_{nr_i} = \min(B_{nr_{i,j}}, B_{nr_{i,g}})$$

Once the new non-realtime flow is established, related non-realtime flows will adjust their sending rates again if the wireless resource is tight. Namely, the initial sending rate can not guarantee that a network operates at an unsaturated status all the time. For example, if a node discovers B_{use} is larger than B_{th}, this node will send adjustment notifications to all the known sources.

We introduce a novel and simply way to adjust the sending rate of non-realtime flow as follows:

$$B_{nr_{new}} = \frac{R_{th} - R_{real}}{R_b - R_{real}} B_{nr_{old}} \tag{3}$$

where R_{th} is the channel busyness ratio corresponding to B_{th}. Unlike the derivation of R_{real} in the admission control scheme, it needs a different way to estimate it here. We just give a lower bound and an upper bound of R_{real} as follows:

$$R_{b_1} \le R_{real} \le R_{b_1} + R_{b_3}$$

To enforce a conservatively increasing and aggressively decreasing rule, we set R_{real} here as follows:

$$R_{real} = \begin{cases} R_{b_1}, & if\ R_b \le R_{th}; \\ R_{b_1} + R_{b_3}, & otherwise \end{cases}$$

All the nodes on the path are qualified to send the rate adjustment to decrease the sending rate of the source. However, only the destination is entitled to send the rate adjustment to increase it. To avoid frequent change of the sending rate due to changes of the channel busyness ratio, this rate adaption should be adapted periodically and all the parameters in Eq. (3) should be the average values over a predetermined period.

IV. PERFORMANCE EVALUATION

In this section, we demonstrate by simulations in NS-2 that ACA manages the flows well to provide good QoS for all the admitted flows. We use IEEE 802.11 DCF as the MAC protocol and the AODV as the routing protocol.

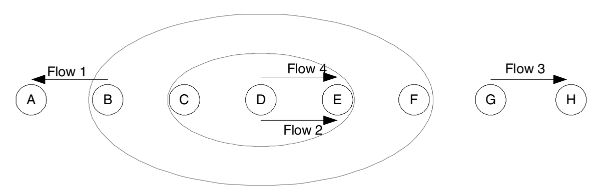
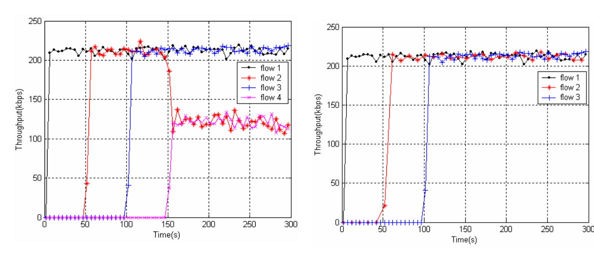


Fig. 2. A simple topology

A. Effectiveness of bandwidth estimation

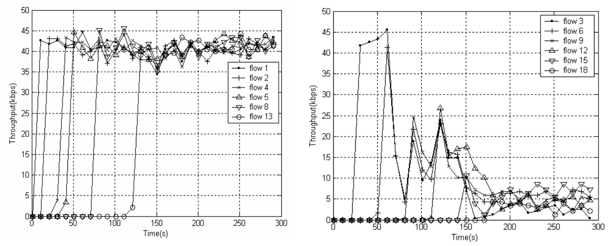
The topology shown in Fig. 2 is used to show the effectiveness of bandwidth estimation. We only use the realtime traffic in this scenario and assume each of the flow consumes 200 kbps, namely, 50 packets/s when the packet size is 512 bytes/packet. Meanwhile, flows from 1 to 4 are added to the network every 50 seconds.

If we use CACP [3], when flow 4 starts, the available bandwidth is 24.7% of the channel capability, namely, 482 kbps, it is reasonable for this network to accommodate flow 4. Fig. 3(a) shows the performance when all those four flows are



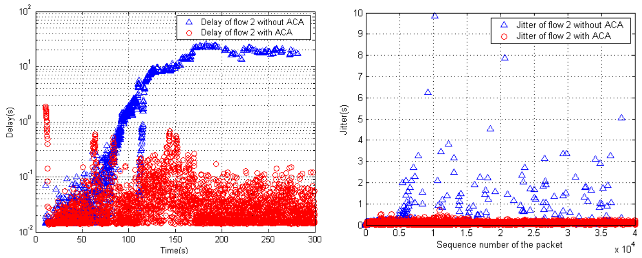
(a) Performance with CACP (b) Performance with ACA

Fig. 3. Throughput for topology in Fig. 2



(a) Realtime flows (b) Non-realtime flows

Fig. 4. Throughput in grid networks with ACA



(a) Delay (b) Delay jitter

Fig. 5. Performance of flow 2 in grid networks with ACA and without ACA

accepted. Apparently, the QoS of flow 4 is very low and the throughput for flow 3 is affected significantly. Namely, the previous admission control scheme cannot provide good QoS in this situation.

In contrast, ACA detects the existence of the hidden terminals, and the maximum bandwidth that a node can achieve is much lower than the bandwidth without any hidden terminal. Taking node D for example, without hidden terminal, B_m is 62% of the basic data rate, and $B_{r_{max}}$ is 823 kbps at this time. By monitoring the communication of node 3, the successful transmission probability of data frames is 0.68, and $B_{r_{max}}$ is changed to 703 kbps. Note that, this does not mean we can accept flow 4, because that B_{use} is larger than $B_{r_{max}}$ at this time. The reason is that the retransmissions of the failing packets cost a lot of wireless resource, leading to a significant decrease of the available bandwidth. Hence, flow 4 is rejected and the QoS of other flows are guaranteed. The performances with ACA are shown in Fig. 3(b).

B. Effectiveness in grid networks

A grid topology is used to demonstrate the effectiveness of ACA in this subsection. We use grid topology with 7×7 nodes and the nodes can only communicate with their closest neighbors. There will be a new non-realtime flow every two realtime flows and the sources are chosen randomly. Meanwhile, each flow consumes 40 kbps and will be added to the network every 10 seconds. In addition, the total number of flows is twenty.

Throughput for realtime and non-realtime traffic is shown in Fig. 4. A stable throughput for each realtime flow is guaranteed once it is established. Note that, flow 7 is rejected but flow 8 is accepted because the bandwidth consumption and the location of flow 8 are different from that of flow 7. For the same reason, flow 13 is also accepted. The delay for flow 2 in each case is shown in Fig. 5(a). With ACA, the delay will be kept in a certain range. Note that, when we use ACA, the delay for the first few packets is very large due to the delay for the connection request and connection reply. In Fig. 5(b), delay jitter with ACA is also much better than that without ACA.

V. Conclusion

Providing QoS over wireless mesh networks is always a challenge. The support level of QoS really depends on the knowledge of the network resource and traffic situation. In this paper, we investigate a new admission control algorithm based on the bandwidth estimation in order to provide QoS for various traffic. Extensive simulation study shows that our proposed scheme indeed provides good QoS support while efficiently utilizing the residual resource for best effort data traffic.

Acknowledgment

The work of Qiang Shen and Xuming Fang was partially supported by the China Scholarship Council, NSFC under Grant 60772085 and the Open Research Fund of State Key Lab of ISN, Xidian University under Grant ISN8-01. The work of Yuguang Fang was partially supported by Chunghwa Telecom M-Taiwan Program M-Taoyuan Project and the National Science Council (NSC), R.O.C., under the NSC Visiting Professorship with contract number NSC-96-2811-E-002-010 while he was a visiting professor in the Department of Computer Science and Information Engineering at National Taiwan University.

References

[1] S. B. Lee, G. S. Ahn, X. Zhang, A. Campbell, "INSIGNIA: An IP-Based Quality of Service Framework for Mobile Ad Hoc Networks, Journal of Parallel and Distributed Computing," Special Issue on Wireless and Mobile Computing and Communications **60**(4):374-406, 2000.

[2] G. Ahn, A. Campbell, A. Veres and L. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks," in Proc. of Infocom'02, New York, 2002.

[3] Y. Yang, R. Kravets, "Contention-Aware Admission Control for Ad Hoc Networks," IEEE Transactions on Mobile Computing, **4**(4):363-377, 2005.

[4] I. D. Chakeres, E. M. Belding-Royer, J. P. Macker, "Perceptive admission control for wireless network quality of service," Ad Hoc Networks **5**(7):1129-1148, 2007.

[5] H. Wei, K. Kim, A. Kashyap, S. Ganguly, "On Admission of VoIP Calls Over Wireless Mesh Network," In Proc. of ICC Next Generation Mobile Networks, Istanbul, June 2006.

[6] H. Zhai, X. Chen, and Y. Fang, "A Call Admission and Rate Control Scheme for Multimedia Support over IEEE 802.11 Wireless LANs," ACM Wireless Networks, **12**(4):451-463, August 2006.

[7] H. Zhai, J. Wang, and Y. Fang, "Providing Statistical QoS guarantee for Voice over IP in the IEEE 802.11 wireless LANs," IEEE Wireless Communications, **13**(1):36-43, Feb. 2006.