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

# Distributed Opportunistic Scheduling in Multihop Wireless Ad Hoc Networks

Yijiang Sun, Victor O.K. Li and Ka-Cheong Leung
Department of Electrical and Electronic Engineering
The University of Hong Kong, Hong Kong SAR, China
Email: {yjsun, vli, kcleung}@eee.hku.hk

*Abstract*—**In this paper, we introduce a framework for distributed opportunistic scheduling in multihop wireless ad hoc networks. With the proposed framework, one can take a scheduling algorithm originally designed for infrastructure-based wireless networks and adapt it to multihop ad hoc networks. The framework includes a wireless link state estimation mechanism, a medium access control (MAC) protocols and a MAC load control mechanism. The proposed link state estimation mechanism accounts for the latest results of packet transmissions on each wireless link. To improve robustness and provide service isolation during channel errors, the MAC protocol should not make any packet retransmissions but only report the transmission result to the scheduler. We modify IEEE 802.11 to fulfill these requirements. The MAC load control mechanism improves the system robustness. With link state information and the modified IEEE 802.11 MAC, we use BGFS-EBA, an opportunistic scheduling algorithm for infrastructured wireless networks, as an example to demonstrate how such an algorithm is converted into its distributed version within the proposed framework. The simulation results show that our proposed method can provide robust outcome fairness in the presence of channel errors.**

## I. Introduction

A wireless ad hoc network consists of a number of nodes communicating with each other on wireless links without infrastructure support. A multihop ad hoc network is an ad hoc network in which the packets of a traffic flow are relayed by one or more intermediate nodes before they reach the destination. To support different types of multimedia applications, providing various quality of service (QoS) guarantees for multihop flows is an important issue in wireless ad hoc networks. In this paper, we focus on the issue of providing robust service isolation and outcome fairness through opportunistic packet scheduling.

Some recent work [1], [2], [3] have been proposed for fair packet scheduling in wireless ad hoc networks. In the rest of this section, we first survey the related work on error compensation and opportunistic scheduling in multihop ad hoc networks. Then, we describe the problem we are facing and our major contributions on resolving it.

### A. Related Work

In [4], a fair scheduling algorithm, namely, TBCP (Timestamp Based Compensation Protocol), is proposed to account for channel errors. By employing a TDMA[1]-based system, TBCP is designed to adapt the start-time fair queueing (SFQ) scheme [5] into the ad hoc environment. Nodes exchange their service tags among two-hop neighbors at the beginning of each frame. The time slot allocation is decided based on the service tags. A flow experiencing channel errors will be compensated automatically since a packet which has been sent unsuccessfully tends to have a smaller service tag and a higher priority. Although collisions among neighboring nodes are inevitable since they do not have exactly the same information, TBCP does not handle collisions due to conflicts in the transmission schedules computed by different nodes. TBCP does not utilize the link status information to improve performance.

The opportunistic scheduling algorithms schedule packets by taking the link status into account. In wireless ad hoc networks, there are two main classes of opportunistic transmission algorithms as discussed in [6]. The first class of algorithms dynamically adapts the transmission rate to the quality of an individual wireless link in order to obtain a throughput gain [7], [8]. The basic idea is to transmit more packets at a higher rate when the channel condition is good. The other class of algorithms exploit multi-path diversity [6], [9], [10], [11]. The idea is to use multiple paths to forward packets at the network layer and opportunistically select the next-hop station with a good wireless link at the MAC layer.

In [10], an anycast extension for IEEE 802.11 [12] is proposed. Instead of sending an RTS (Request to Send) packet for on-demand channel reservation, an MRTS (Multicast RTS) packet is multicast from the transmitter to all possible next-hop receivers. A receiver responds with a CTS (Clear to Send) packet on hearing an MRTS packet. Collisions are avoided by assigning different priority orders in the MRTS packet. On hearing a CTS packet from any receiver, the transmitter starts to transmit a DATA packet, which will suppress any further CTS transmissions from other receivers. A successful transmission conforms to a full MRTS-CTS-DATA-ACK handshake protocol.

In [6], multi-user diversity is exploited to improve energy efficiency as well as network throughput and fairness through two mechanisms, namely, CRA (Cooperative Rate Adaptation) and COS (Cooperative and Opportunistic Scheduling). The information for links connecting to the neighboring nodes is estimated locally and piggybacked within GRTS (Group RTS, similar to MRTS in [10]) and CTS packets. Differing from [10], GRTS and multiple CTS packets are not used to

This work is supported in part by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Grant No. HKU 7148/06E.

[1] TDMA stands for Time Division Multiple Access.

implement MAC layer anycast, but to facilitate the inter-node information exchange.

### B. Our Contributions and Organization of the Paper

To the best of our knowledge, none of the previous work takes error compensation, link status estimation and fair scheduling into consideration simultaneously for distributed scheduling in multihop ad hoc networks. In this work, we propose a framework, named "Robust Opportunistic Scheduling for Ad Hoc Networks" (ROSA), with which a scheduling algorithm originally designed for infrastructured wireless networks can be adapted to multihop ad hoc networks. The adapted algorithm performs distributed scheduling opportunistically by utilizing the link status information provided by ROSA. We also improve the robustness of the system by limiting the traffic load at the MAC layer.

The rest of the paper is organized as follows. Section II describes the system model and the assumptions used throughout the paper. Section III describes the ROSA framework in detail. Section IV presents the performance results through simulation in ns-2 [13]. We conclude our work in Section V.

## II. System Model and Assumptions

We consider a multihop wireless ad hoc network. Nodes communicate over the same channel. A node cannot transmit and receive packets simultaneously. A collision happens when a receiver is in the transmission ranges of multiple transmitters. Wireless links are error-prone and the occurrences of channel errors are not negligible.

Instead of using static flow weights, the QoS requirement of an end-to-end flow specifies the desired service rate. An admission control mechanism is used to grant the desired QoS requirements. The desired service rate is propagated to all the intermediate nodes along the path. This may be accomplished by piggy-backing the desired rate on each packet of the flow.

For ease of presentation, we assume that a contention-based MAC scheme is used, although this is not a requirement of ROSA. Since the packet transmission schedule is computed at each node locally based on incomplete and conflicting network information, collisions are inevitable. However, with the admission control mechanism no flow shall offer a traffic load above the admitted service rate. We assume that the collision rates are statistically stable and predictable [14].

The state of a wireless link is estimated to be either *good* or *bad*. A packet sent on a *good* link has a much higher probability of success than that on a *bad* link. The link conditions are independent of each other. Unsuccessful transmissions are due to either channel errors or packet collisions. The transmitter has no means to know the cause of an unsuccessful transmission.

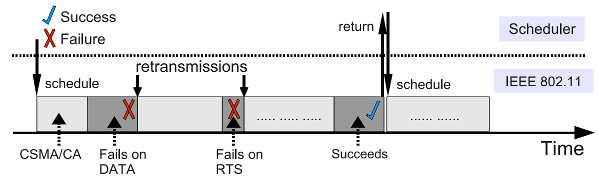
Lastly, we assume all flows are properly routed. We do not consider routing issues in this paper.

## III. The ROSA Framework

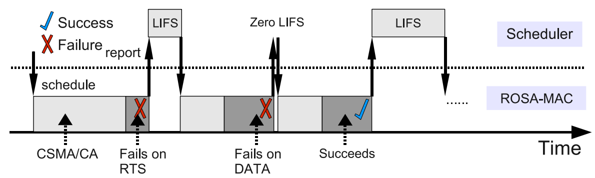
As stated in Section I-B, the goal of ROSA is to adapt scheduling algorithms originally designed for infrastructured wireless networks to a distributed algorithm in multihop ad hoc networks.

### A. MAC Layer Blocking Problem and Solution

First, we observe that the basic scheduling model in ad hoc networks is different from that in an infrastructured network. In an infrastructured wireless network, all packets are scheduled by a base station. When a packet transmission fails, the scheduler can decide either to retransmit it or to transmit another packet from another flow.




(a) The IEEE 802.11 MAC and the Scheduler.




(b) The Modified IEEE 802.11 MAC and the Scheduler.

Fig. 1. Modifications Made to the IEEE 802.11 MAC Protocol.

However, in ad hoc networks, the situation is different with some MAC protocols that have retransmission mechanisms. Take IEEE 802.11 as an example. Fig. 1(a) shows a typical interaction between the scheduler and the underlying IEEE 802.11 MAC. When the scheduler gives the MAC layer a packet to transmit, IEEE 802.11 will first perform carrier-sensing, collision avoidance and exponential backoff (as shown in Fig. 1(a) with light gray blocks) before it actually sends an RTS (Request To Send) packet for that packet. The RTS-CTS-DATA-ACK protocol (as shown with dark gray blocks) may fail either on RTS or on DATA at the transmitter when the corresponding CTS or ACK is not received. In any case, IEEE 802.11 will backoff for a random period of time and try to retransmit that packet again. This retry process continues repeatedly until the packet is successfully transmitted or it is dropped as the retry limit is exceeded. Then, the scheduler selects the next packet and passes it to the MAC layer for transmission.

The problem with this model is that, while the MAC protocol keeps retransmitting a failed packet, the scheduler does not have any control over it. All retransmission schedules are made by the MAC layer. As a result, the scheduler running over such a MAC protocol does not have full control over what is transmitted. We term this phenomenon the MAC layer blocking problem, because one failed packet in the MAC layer will block the scheduler from scheduling other flows. The problem becomes severe when a link becomes *bad*, since packets transmitted on that link is highly likely to be retransmitted until it is dropped. This further blocks other packets, which may potentially be transmitted on *good* links, for a relatively long time.

To address this problem, the ROSA framework requires that the MAC protocol not to make any retransmissions automatically, but should report the result of each transmission to the scheduler. As IEEE 802.11 is becoming very popular in wireless ad hoc networks, we modify it to meet the above requirement. The modified version is named ROSA-MAC. Fig. 1(b) shows a series of typical interactions between the scheduler and ROSA-MAC. Except for some implementation details, the only difference between ROSA-MAC and IEEE 802.11 is that, in ROSA-MAC, a failed transmission is not retransmitted automatically. Instead, after a packet transmission attempt, the MAC layer just reports the result to the scheduler and waits for a new packet from it. Through this design, all scheduling decisions are made solely by the scheduler. The "LIFS" in Fig. 1(b) refers to "Load Control Inter-Frame Space," which is used to implement MAC load control. Refer to Section III-C for details.

### *B. Link State Estimation*

Link state estimation is an important element of opportunistic scheduling in infrastructured wireless networks as it provides the necessary information for the scheduler to exploit multiuser diversity. In order to adapt such algorithms to ad hoc networks, it is also necessary for the ROSA framework to provide a similar mechanism to differentiate between *good* and *bad* wireless links.

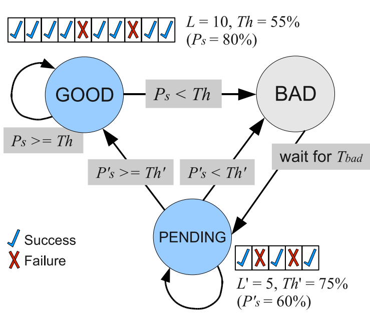



Fig. 2. Link State Estimator.

In ad hoc networks, there is no base station to act as the central controller or dedicated control channel to feedback the channel state. Due to these characteristics, we base our link state estimation mechanism on the transmission history of each link. A link state estimator (LSE) is employed to monitor each packet transmission and record the last $L$ transmission results for each wireless link. Each record entry is marked either as "Success" or "Failure". A link varies between three states: "GOOD", "BAD" or "PENDING", as shown in Fig. 2. LSE estimates the link state based on the packet success rate, $P_s$, which is defined as the fraction of the number of successful transmissions over the most recent $L$ transmissions, and a threshold $Th$, where $0<Th<1$. All records are initialized as successful and the link state is initialized as GOOD. The link state remains to be GOOD when $P_s \geq Th$. It transits from GOOD to BAD when $P_s < Th$. When a link goes BAD, it stays in the BAD state for a time period $T_{bad}$ before it transits to the PENDING state. The transmission records are not updated in the BAD state (in case a scheduling algorithm allows transmissions on a bad channel) and the PENDING state. In the PENDING state, LSE estimates the channel status by transmitting $L'$ packets. If the successful rate $P'_s$ is above the threshold $Th'$, the link status transits to the GOOD state[2]. Otherwise, it goes back to the BAD state and waits for another $T_{bad}$.

Note that GOOD, BAD and PENDING are internal states within LSE. When the scheduler consults LSE for the link status, LSE responds *good* if the link's internal state is GOOD or PENDING, or *bad* if the link's internal state is BAD. $L$, $Th$, $L'$, $Th'$ and $T_{bad}$ are all tunable parameters of ROSA.

### *C. Medium Access Abuse Problem and Solution*

In infrastructured wireless networks, the scheduler can schedule not only outgoing (down-link) packets but also incoming (up-link) packets. However, in ad hoc networks, only outgoing packets are scheduled by the local scheduler. Incoming packets are scheduled by other nodes. The intended receiver will fail to receive an incoming packet when it is transmitting. In order to strike a balance between receiving packets from the previous hop and forwarding them to the next hop, a forwarding node should not transmit packets too frequently so that there is not enough time left for reception.

For one node, suppose that a collision happens with probability $p_c$ on each packet transmission for one of its wireless links. For the simplicity of analysis, we assume that no packet is dropped due to buffer overflow. Let $\bar{N}$ be the average number of transmission attempts made for each packet. If every packet received by a node is also forwarded to another node, $\bar{N}$ also represents the ratio of the outgoing traffic load over the incoming traffic load at the MAC layer. $\bar{N}$ can be calculated as:

$$\bar{N} = (1-p_c)\sum_{n=1}^{M-1} n \cdot p_c^{n-1} + M \cdot p_c^{M-1} \quad (1)$$

where $M$ is the retry limit.

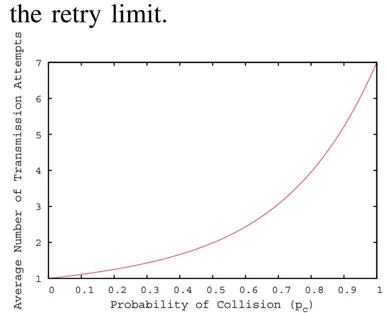



Fig. 3. Average Number of Transmission Attempts against Collision Rate.

Fig. 3 is the plotted curve of $\bar{N}$ versus $p_c$ according to (1) with $M = 7$, the default retry limit of the IEEE 802.11 MAC protocol. When the traffic load is not very high, $p_c$ is usually very small, that makes $\bar{N}$ only slightly larger than 1. This means that, when the wireless link is *good* and the collision rate is low, the outgoing traffic load should be only

[2]The temporary transmission history recorded during the PENDING state is also copied to the main record to reflect the latest changes.

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

slightly above the incoming traffic load. However, when some of the outgoing links turn *bad*, the outgoing traffic load may increase due to uncontrolled packet retransmissions and results in service degradation to all the passing flows.

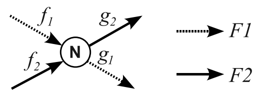



Fig. 4. An Ad Hoc Node Forwarding Two Flows.

We use a simple example as shown in Fig. 4 to illustrate this problem. Node N is forwarding two multihop flows $F1$ and $F2$. It receives packets for flow $F1$ and $F2$ on links $f_1$ and $f_2$, and transmit their packets on links $g_1$ and $g_2$, respectively. Suppose that $g_1$ turns *bad* at some time. Node N will access the medium more frequently to retransmit the failed packets on $g_1$. However, this also reduces the chance for N to receive packets successfully on links $f_1$ and $f_2$. As a result, the packet queue for $F2$ is cleared rapidly. After $F2$'s queue is emptied, all medium accesses are devoted to $F1$'s packets, which are unlikely to be sent successfully on the *bad* link $g_1$. This in turn worsens the situation further. We term this problem as the medium access abuse problem since it originates from the abusive use of the wireless medium.

To address this problem, the ROSA framework uses a token bucket traffic regulator at the MAC layer of each node to limit the traffic load that the MAC layer puts onto the wireless medium. A token bucket of size $S_{tk}$ is filled with bit-by-bit tokens at a constant rate of $R_{tk}$ tokens per second. A packet transmission consumes $S_{pkt}$ tokens. When a packet is scheduled by the scheduler, it first checks the amount of tokens $N_{tk}$ in the bucket. If $N_{tk} < S_{pkt}$, the scheduler waits for a time period of $\frac{S_{pkt}-N_{tk}}{R_{tk}}$ by setting $LIFS$ to that value. After that, the token bucket will have enough tokens to transmit a packet. When $N_{tk} \geq S_{pkt}$, $LIFS$ may also be set to reduce the MAC layer traffic burstiness. $LIFS$ is calculated by:

$$LIFS = \begin{cases} \frac{S_{pkt}-N_{tk}}{R_{tk}} & N_{tk} < S_{pkt} \\ T_m \cdot (1 - \frac{th_m}{R_m}) & N_{tk} \geq S_{pkt}, R_m > th_m \\ 0 & N_{tk} \geq S_{pkt}, R_m \leq th_m \end{cases} \quad (2)$$

where $R_m$ is the average rate at which the scheduler passes packets to the MAC layer and $T_m$ is the length of the time window used to measure $R_m$. $T_m$ is split into $N_m$ time slots of length $t_m$, i.e., $T_m = N_m \cdot t_m$. For each time period $t_m$, $R_m$ is also updated to $N_{tx}/T_m$, where $N_{tx}$ is the number of packets that have been passed to the MAC layer during the current time window. When $R_m$ does not exceed its threshold $th_m$, $LIFS$ is set to zero in order to minimize the delay. When $R_m$ exceeds the threshold, $LIFS$ is set to $T_m \cdot (1 - th_m/R_m)$ in order to reduce $R_m$ back to $th_m$. After a time period $LIFS$, $N_{tx}$ is reduced by $N_{tx} \cdot LIFS/T_m = N_{tx} - T_m \cdot th_m$, and thus brings $R_m$ back to $th_m$. $S_{tk}$, $N_m$ and $t_m$ are tunable parameters. Typically, $t_m$ is less than the average packet transmission time and $T_m$ is about several tens of packet transmission times. $R_{tk}$ is the upper bound of $R_m$ since it is the token generation rate. $R_{tk}$ and $th_m$ should be provided by the admission control mechanism or certain bandwidth allocation mechanism [3]. Since $th_m$ is generally smaller than $R_{tk}$, $R_m$ may fluctuate around $th_m$ during the channel error period.

### D. Adapting BGFS-EBA to Distributed Environment

BGFS-EBA [15] (Bandwidth Guaranteed Fair Scheduling with Effective Excess Bandwidth Allocation) is an opportunistic scheduling algorithm for infrastructured wireless networks. It aims to provide bandwidth and delay guarantees for flows with an error-free link and allocate excess bandwidth among lagging flows (flows with achieved goodput smaller than its target share, typically caused by channel errors) in an equitable manner.

With the mechanisms provided by the ROSA framework, BGFS-EBA can be adapted to ad hoc networks by mapping its basic elements to those in the ad hoc environment. BGFS-EBA performs packet scheduling based on three basic parameters: the total available bandwidth $R$, the target rate $r_i$ of each flow $i$ and its link status $Lk_i$. In an ad hoc network, with the ROSA framework, $R$ is provided during the connection establishment phase. $r_i$ is mapped to the local value of the desired service rate of each end-to-end flow, and $Lk_i$ is provided by LSE. The adapted distributed algorithm is named OBGSA (Opportunistic Bandwidth Guaranteed Scheduling for Ad hoc networks). The internal algorithm of OBGSA is the same as BGFS-EBA. Interested readers can refer to [15] for details.

## IV. Simulation Results and Discussion

The simulation is performed with ns-2, in which the proposed link status estimation mechanism, ROSA-MAC protocol, and the OBGSA scheduling algorithm are all implemented. Link errors are caused by channel fading. A practical ns-2 extension for the well-known Ricean fading model is used [16]. The capacity of the wireless channel used in the simulation is 2 Mbps or 256 KBps. Table I lists the ROSA parameters used in the simulation.

TABLE I
Simulation Parameters of ROSA.

| $L$ | $th$ | $L'$ | $th'$ | $T_{bad}$ | $t_m$ | $N_m$ | $th_m$ | $R_{pkt}$ |
|---|---|---|---|---|---|---|---|---|
| 5 | 75% | 2 | 90% | 30ms | 10ms | 50 | 56KBps | 58KBps |

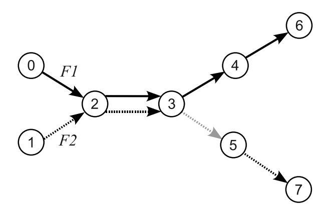
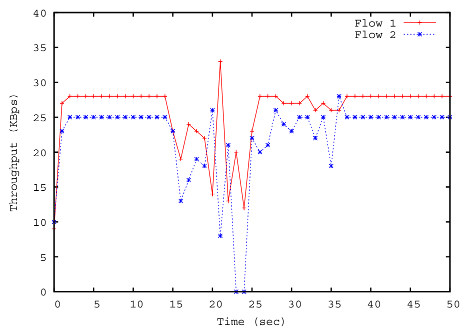



Fig. 5. A Multihop Ad Hoc Network with Two End-to-End Flows.

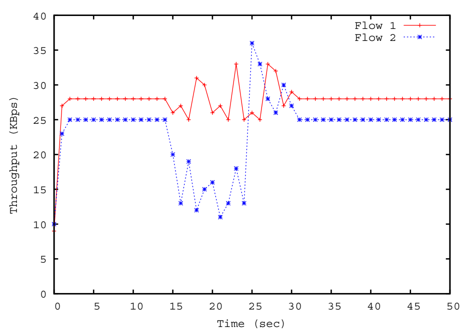
Fig. 5 shows the network topology used in the simulation. Two multihop constant bit rate (CBR) flows $F1$ and $F2$ are sent from node 0 to node 6 and from node 1 to node 7, respectively. The offered data loads for $F1$ and $F2$ are 28KBps and 25KBps, respectively. The transport protocol used is UDP.

The simulation lasts for 50 seconds. During the time period [15s, 25s], link $3 \rightarrow 5$ suffers link errors due to channel fading. This is simulated by temporarily replacing the Two Ray Ground propagation model of link $3 \rightarrow 5$ with the

Ricean fading model. The Ricean factor $K$ is set to 2 and the maximum Doppler frequency $f_m$ is set to $20Hz$. The buffer size for each flow at each node is 16 packets. The end-to-end throughput is measured at the destination node.




(a) Throughput with IEEE 802.11 + FIFO.




(b) Throughput with ROSA Framework.

Fig. 6. Performance of IEEE 802.11+FIFO and ROSA, with Fading Channel.

TABLE II
END-TO-END THROUGHPUTS WITH CHANNEL FADING.

| Flow | Offered load (KBps) | Avg. throughput during the affected period without ROSA (KBps) | Avg. throughput during the affected period with ROSA (KBps) |
|---|---|---|---|
| $F1$ | 28 | 20.54 (73.4%) | 27.94 (99.8%) |
| $F2$ | 25 | 15.10 (60.4%) | 21.31 (85.2%) |

Fig. 6(a) shows the end-to-end throughput curves for $F1$ and $F2$ using IEEE 802.11 with the FIFO scheduler. The throughputs of both flows fluctuate severely during the channel fading period. They become stable again at 39s. During the period with channel fading, the throughputs of both flows even drop to zero. Table II lists the average throughputs for $F1$ and $F2$ during the affected period. The throughputs for $F1$ and $F2$ drop to 73.4% and 60.4% of the offered loads, respectively. This is unfair for $F1$ since wireless links traversed by $F1$ are in *good* condition throughout the simulation.

Fig. 6(b) and Table II show that, with the ROSA framework, the affected period is shortened to [15s, 30s]. During the channel fading period, the throughput for $F1$ fluctuates mildly. The throughput for $F2$ decreases to 85.2% of the offered load but rises quickly when the channel recovers. The throughput for $F1$ stays at 99.8% of the offered load. This means that $F1$ is only very slightly affected by the fading channel.

The simulation results show that the ROSA framework not only improves the end-to-end throughput for flows that encounter channel errors, but also provides robust service isolation to flows with different and time-varying link conditions. The service degradation caused by channel errors on error-prone links is limited only to the flows that pass through those links. Therefore, the ROSA framework can provide robust throughput guarantees and outcome fairness to multihop flows in wireless ad hoc networks with channel errors.

## V. CONCLUSION

In this paper, we present the ROSA framework for distributed opportunistic scheduling in multihop wireless ad hoc networks. The framework includes a link status estimation mechanism, the MAC protocol and a MAC load control mechanism. We use BGFS-EBA as an example and adapt it to ad hoc networks within the ROSA framework. The simulation results show that the adapted algorithm increases the system throughput and provides robust outcome fairness in the presence of channel errors. In the future, we shall adapt other scheduling algorithms designed for infrastructure-based wireless networks by transforming them to the distributed algorithms for to multihop ad hoc networks under the ROSA framework.

## REFERENCES

[1] H. Luo, J. Cheng, and S. Lu, "Self-Coordinating Localized Fair Queueing in Wireless Ad Hoc Networks," *IEEE Trans. on Mobile Computing*, vol. 3, no. 1, pp. 86–98, Jan. 2004.

[2] L. Tassiulas and S. Sarkar, "Maxmin Fair Scheduling in Wireless Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 163–173, Jan. 2005.

[3] B. Li, "End-to-End Fair Bandwidth Allocation in Multi-Hop Wireless Ad Hoc Networks," in *Proc. IEEE ICDCS 2005*, Columbus, Ohio, USA, Jun. 2005, pp. 471–480.

[4] H.-L. Chao and W. Liao, "Fair Scheduling in Mobile Ad Hoc Networks With Channel Errors," *IEEE Trans. on Wireless Communications*, vol. 4, no. 3, pp. 1254–1263, May 2005.

[5] P. Goyal, H. M. Vin, and H. Cheng, "Start-Time Fair Queueing: A Scheduling Algorithm for Integrated Services Packet Switching Networks," *IEEE/ACM Trans. on Networking*, vol. 5, pp. 690–704, 1997.

[6] Q. Zhang, Q. Chen, F. Yang, X. Shen, and Z. Niu, "Cooperative and Opportunistic Transmission for Wireless Ad Hoc Networks," *IEEE Trans. on Mobile Computing*, vol. 21, no. 1, pp. 14–20, Jan. 2007.

[7] A. Kamerman and L. Monteban, "Wavelan II: A High-Performance Wireless lan for the Unlicensed Band," *Bell Labs Technical Journal*, vol. 2, no. 3, pp. 118–133, Jul. 1997.

[8] G. Holland, N. Vaidya, and P. Bahl, "A Rate-Adaptive mac Protocol for Multi-Hop Wireless Networks," in *Proc. ACM MobiCom*, Jul. 2001, pp. 236–251.

[9] R. R. Choudhury and N. H. Vaidya, "Mac-Layer Anycasting in Ad Hoc Networks," *SIGCOMM C.C.R.*, vol. 34, no. 1, pp. 75–80, Jan. 2004.

[10] S. Jain and S. R. Das, "Exploiting Path Diversity in the Link Layer in Wireless Ad Hoc Networks," in *Proc. IEEE WoWMoM*, Jun. 2005, pp. 22–30.

[11] J. Wang, H. Zhai, Y. Fang, J. Shea, and D. Wu, "Omar: Utilizing Multiuser Diversity in Wireless Ad Hoc Networks," *IEEE Trans. on Mobile Computing*, vol. 5, no. 12, pp. 1764–1779, Dec. 2006.

[12] *IEEE 802.11 Wireless LAN Media Access Control (MAC) and Physical Layer (PHY) specifications*, IEEE Std., 1997.

[13] K. Fall and K. Varadhan, *The ns Manual (Formerly ns Notes and Documentation)*, The VINT Project, Feb. 2008.

[14] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Trans. on Selected Areas in Communication*, vol. 18, no. 3, pp. 535–574, Mar. 2000.

[15] Y. Cao and V. Li, "Wireless Packet Scheduling for Two-state Link Models," in *Proc. IEEE GLOBECOM*, vol. 1, Nov. 2002, pp. 819–823.

[16] R. Punnoose, P. Nikitin, and D. Stancil, "Efficient Simulation of Ricean Fading within a Packet Simulator," in *Proc. IEEE VTC 2000 Fall*, vol. 2, Sep. 2000, pp. 764–767.