Achieving Packet-Level Quality of Service Through Scheduling in Multirate WLANs

Yuan Yuan, Daqing Gu, Williams Arbaugh and Jinyun Zhang

TR2004-117 December 2004

Abstract

Wireless packet scheduling has been a popular paradigm to achieve packet-level quality of services in terms of fairness and throughput in the presence of channel errors. However, the current design does not anticipate the multi-rate capability offered by the IEEE 802.11a/b/g physical layer, thus suffering significant performance degradation in 802.11 WLANs. In this paper, we propose Multirate Wireless Fair Scheduling (WMFS). In MWFS, each flow is granted temporal fair share of the channel, in contrast to the throughput fair share adopted by existing algorithms. Therefore, each flow receives services in proportion to its perceived transmission rate, and high-rate flows are able to opportunistically exploit their good channel conditions and receive more services. MWFS also renovates the compensation model in order to allow for error-prone flows to catch up, thus ensuring fairness for all flows over error-prone channels. We demonstrate the effectiveness of MWFS through both simulations and analysis. Especially WMFS achieves system throughput 159% of state-to-the-art scheduling algorithms in simulated scenarios.

IEEE Vehicular Technology Conference (VTC) Fall 2004

Achieving Packet-Level Quality of Service Through Scheduling in Multirate WLANs

Yuan Yuan[1], Daqing Gu[2], Williams Arbaugh[1], Jinyun Zhang[2]

[1]Computer Science Department, University of Maryland, College Park, MD 20740, USA
[2]Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA

Keywords: QoS, Scheduling, IEEE 802.11, Wireless Channels, Multirate Physical Layer

Abstract

Wireless packet scheduling has been a popular paradigm to achieve packet-level quality of services in terms of fairness and throughput in the presence of channel errors. However, the current design does not anticipate the multi-rate capability offered by the IEEE 802.11a/b/g physical layer, thus suffering significant performance degradation in 802.11 WLANs. In this paper, we propose Multirate Wireless Fair Scheduling (WMFS). In MWFS, each flow is granted temporal fair share of the channel, in contrast to the throughput fair share adopted by existing algorithms. Therefore, each flow receives services in proportion to its perceived transmission rate, and high-rate flows are able to opportunistically exploit their good channel conditions and receive more services. MWFS also renovates the compensation model in order to allow for error-prone flows to catch up, thus ensuring fairness for all flows over error-prone channels. We demonstrate the effectiveness of MWFS through both simulations and analysis. Especially WMFS achieves system throughput 159% of state-to-the-art scheduling algorithms in simulated scenarios.

1 Introduction

In recent years, wireless LAN technology based on the IEEE 802.11 standard has become increasingly popular to provide users untethered Internet access. In order to improve radio spectrum utilization, the IEEE 802.11a/b/g specifications offer a physical-layer multi-rate capability [1]. Specifically, in IEEE 802.11b, users can transmit at one of the four rate options 1, 2, 5.5, and 11Mbps, whereas in 802.11a, eight rate choices are allowed at 6, 9, 12, . . . , and 54Mbps. With such physical-layer enhancements in place, a host can select the best transmission rate depending on its perceived channel quality measured by the signal-to-noise ratio (SNR). If used properly, this new option can greatly improve the system throughput and effectively support communication-intensive multimedia and data applications.

The multirate option poses new challenges for network protocol design in the context of wireless packet scheduling [2]. Packet scheduling, notably fair queueing, has long been a popular paradigm [2]-[7] to provide packet-level quality of services (QoS) in terms of throughput, delay and fair sharing, thus enabling both delay-sensitive and throughput-sensitive applications. Wireless packet scheduling [2] further addresses the issue of location-dependent channel errors in wireless networks, and shields short-term error bursts from packet flows. However, the state-of-the-art wireless fair scheduling typically assumes a single, fixed transmission rate for *all* users. It does not anticipate multiple rate options across users. Fair queueing algorithms designed for single-rate environment hence suffer from significant throughput reduction in the current 802.11a/b/g WLANs. In fact, even the fairness notion may not be justified anymore.

In this paper, we propose MWFS, a wireless fair scheduler that leverages the multirate capability offered by WLANs based on 802.11b/a/g and supports both data and multimedia applications. A key innovation of MWFS is to re-define fairness and compensation in *temporal* shares and depart from throughput-based fairness and compensation. In MWFS, each backlogged flow will receive a fair share in terms of transmission time slices. Hosts under good quality channels, thus selecting high rates, will transmit more packets than hosts under bad quality channels. The distinction between time-based and throughput-based fairness and compensation is critical in multirate networks. MWFS can provide flows with dramatically different throughputs as governed by their channel conditions, but all flows will achieve approximately identical time shares. For example, a flow operating at 11Mbps will obtain roughly 5.5 times the throughput of a flow operating at 2Mbps, but these two flows will both access the channel approximately identical share of the time. In summary, MWFS is able to opportunistically exploit high quality channels when they occur via transmissions of packets in proportion to their high data rates. Through both analysis and simulations, we confirm the effectiveness of MWFS design. The results show that MWFS is able to improve overall throughput upto 159% over the current single-rate scheduling algorithm and individual throughput by up to 550% in simulated scenarios.

2 Problem Illustration

We consider a packet-switched wireless LAN based on IEEE 802.11b/a/g. Scheduling of packet transmissions is performed at each Access Point (AP). Even though all the mobiles and the AP share the same channel, errors are location dependent due to fades, interferences, etc.

2.1 Single-Rate Wireless Packet Scheduler

The state-of-the-art wireless scheduling solutions [2] are proposed to work in a single-rate scenario where all flows perceive the same channel capacity at all times. We first illustrate how wireless fair queueing works in a single-rate scenario. For simplicity, we use the simplest wireless fluid fair queueing model of [6] to identify issues. Consider three continually backlogged flows to be served by the same AP during the time interval [0,2], with flow weight $r_1 = r_2 = r_3$. Each flow is expected to receive services in proportion to its flow weight, denoted by r_i for flow i, if its channel condition (i.e., transmission rate and error patterns) is identical to others. In the single-rate scenario, say, all three flows perceive the same base transmission rate $C = 1$. Then, the fairness and compensation models work as follows.

The fairness model dictates that in the absence of channel errors, each flow receives its fair share of service, defined in terms of throughput, in proportion to its flow weight. In the example, each flow will receive services (in bytes/second) $C \cdot (2-0) \cdot \frac{1}{3} = \frac{2}{3}$ over the time interval $[0, 2]$.

Now consider the error-prone case, in which flow 1 and flow 2 have error-free channels, while flow 3 perceives channel errors during interval [0,1). By applying wireless fair scheduling over the time periods [0,1) and [1,2], we arrive at the following fair channel capacity allocation over each interval:

$$W_1[0,1) = W_2[0,1) = 1/2,$$

$$W_1[1,2] = W_2[1,2] = 1/6, W_3[1,2] = 2/3.$$

That is, flow 3 will relinquish its allocated service worth $\frac{1}{3}$ during [0,1] since it perceives channel error, and let flows 1 and 2 transmit first. However, during [1,2], each of flows 1 and 2 will give up $\frac{1}{6}$ to compensate flow 3. Overall, in the time window [0,2], the allocation is

$$W_1[0,2] = W_2[0,2] = W_3[0,2] = 2/3.$$

which still satisfies fair sharing for all three flows.

2.2 Multirate Scenario

We now show its problems in the multi-rate case. Let the transmission rates of flows 1, 2, and 3 during $[0, 2]$ be $C_1 = 1$, $C_2 = 2$, and $C_3 = 11$, respectively. Applying the fairness model specified by the wireless fair queueing algorithm, each flow must receive identical throughput during [0,2]. Since the rate ratios of these three flows are $C_1 : C_2 : C_3 = 1 : 2 : 11$, the time allocated for these flows must observe $T_1 : T_2 : T_3 = 1 : \frac{1}{2} : \frac{1}{11}$ (i.e., inverse proportional to the transmission rate) in order to ensure identical throughput for all flows. Therefore, flow 3 will be equivalently served with $2 \cdot \frac{1/11}{1+1/2+1/11} = \frac{4}{35}$ time units during [0,2]. Then, each flow receives services $W_1[0,2] = W_2[0,2] = W_3[0,2] = \frac{44}{35}$ if all flows perceive clean channels all the time. The overall channel throughput during [0,2] is $\frac{132}{35} \approx 3.77$. The problem with this throughput fairness model is that, even though the transmission rate of flow 3 is 11 times of flow 1, it only receives $\frac{1}{11}$ of the transmission time received by flow 1 to ensure throughput fairness. This unnecessarily penalizes flows under good-quality channels and reduces the overall system throughput.

In contrast, if each flow is granted a fair *temporal* share, then each flow is served with $\frac{2}{3}$ time units during $[0, 2]$. The overall system throughput will be $(1 + 2 + 11) \cdot \frac{2}{3} = \frac{28}{3}$, strictly larger than $\frac{132}{35}$. In fact, the overall throughput is 247% of the throughput achieved by the algorithm above. Fundamentally, flows under good-quality channels will be able to transmit the same amount of time share as flows under bad-quality channels. The system is thus able to opportunistically exploit the high transmission rates of flows under good channels.

Now we look at the compensation model in the presence of channel errors. If we apply the compensation model of the single-rate wireless fair queueing, then it is easy to calculate that:

$$W_1[0,1) = W_2[0,1) = 2/3,$$

$$W_1[1,2] = W_2[1,2] = 62/105, W_3[1,2] = 44/35.$$

Again, in order to achieve fair compensation in terms of throughput, low-rate flows 1 and 2 are allocated more time slices than flow 3. However, if we adopt compensation based on time shares, flow 3 will receive $\frac{1}{3}$ time units for compensation during [1,2]. Then, the aggregate service flow 3 receives is $W_3[0,2] = 11 \cdot \frac{2}{3} = \frac{22}{3}$. The overall throughput for all three flows will be 9.33, rather than 3.77.

This simple example illustrates the limitations of wireless fair queueing in the multirate case. In general, due to rate difference at each host, server allocations designed to be fair in terms of throughput may be inconsistent with temporal fairness. As a result, high-rate flows under good channel conditions will be unnecessarily penalized to give more transmission time units for low-rate flows under bad-quality channels. The overall channel throughput is significantly reduced and the benefits of multi-rate physical layer are severely mitigated.

3 MWFS: Multirate Wireless Fair Scheduling

To exploit the multirate capability provided by the physical layer, we depart from the throughput-oriented fairness and compensation models, originally proposed for the single-rate scenario, and adopt the notion of fair temporal shares. In essence, MWFS seeks to ensure temporal fairness in the presence of location-dependent channel errors. It has three main components:

- Error-Free Service Model, which defines the ideal fair service model for flows that transmit at different rates.

- Lead and Lag Model, which determines which flows are leading or lagging their error free service, and by how much.

- Compensation Model, which compensates for lagging flows at the expense of leading flows, thus addressing the key issue of location-dependent channel errors in wireless channel access.

The pseudo code of MWFS is shown in Figure 1.

```
ON PACKET P ARRIVING $queue_i$ OF $source_i$
    enqueue(p, $queue_i$);
    /* A is set of active traffic sources */
    if $i \notin A$
        $A = A -$ {i};
        $start\_tag_i = virtual\_time$;
        $finish\_tag_i$ = $start\_tag_i$
                + $sum_{i \in A}(r_i)/r_i * size(p)/speed_i$ ;
        $credit_i = 0$;
        $comp\_tag_i = 0$;

EXTRACTING PACKET FOR TRANSMITTING
    $l = min_{start\_tag_k}(k \in A)$;
    if $credit_l <= 0$ /* handling non-leading flows */
        if (l can send)
            PKT_SEND($l, l$)
        else
            /* l cannot send due to channel error */
            /* find flow with smallest start_tag to substitute i */
            g = $min_{start\_tag_k}(k \in A$, k_can_send)
            if (g exists)
                /* transmit for g, update tags,credit for both */
                PKT_SEND($g, l$);
            else /* All sources are in channel error */
                IDLE;
    else /* handling leading flows */
        /* decide whether let leading flow transmit or compensate */
        flag = compensation_flag(l); /*1: compensate, 0: normal */
        $g = min_{comp\_tag}(k|k \in A, credit_k < 0, k\_can\_send)$;
        if (l_can_send)and((!flag)or((flag)and(g !exists))
            PKT_SEND($l, l$);
        else if (g exists)
                PKT_SEND($g, l$);
            else /* All sources are in channel error */
                IDLE;
        /* check if some flow becomes inactive */
        /* adjust other flows in A accordingly */
        HANDLE_IDLE_FLOW;

PKT_SEND($g, l$)
/* send packet from $source_i$, update tags, credits for both */
    $p_l$ = dequeue($l$);
    /* update start and finish tag only for $l$ */
    $start\_tag_l = max(virtual_time, finish\_tag_l)$;
    $finish\_tag_l = start\_tag_l + sum_{i \in A}(r_i)/r_l * size(p_g)/speed_l$;
    if ($l \neq g$ )
        $credit_l + = size(p_g)/speed_l$;
        $credit_g - = size(p_g)/speed_l$;
        $comp\_tag_l = -size(p_g)/(speed_l * credit_l)$;
        $comp\_tag_g = -size(p_g)/(speed_g * credit_g)$;
```

Figure 1: Pseudo Code of MWFS

In MWFS, a packet with sequence number k of flow f arriving at time $A(t_k^f)$ is assigned two tags: a start tag S_k^f and a finish tag F_k^f, defined as follows:

$$S_k^f = \max\{V(A(t_k^f)), F_{k-1}^f\};$$
$$F_k^f = S_k^f + L_p/(r_f \cdot C_f(t)) \qquad (1)$$

The transmission rate of flow f at t is $C_f(t)$[1]. The start tag, as well as the finish tag of a packet, is normalized with respect to its current transmission rate. This is to allow for the flow that perceives good channel quality and transmits at higher rate to receive service in proportion to its current rate.

The lead and lag model specifies the temporal share (i.e., how many time units) each leading flow has to give up in the future, and how many time units a lagging flow will receive for compensation. Variable $credit_g$ in function PKT_SEND() keeps track of how much compensation lagging flow g needs. It is defined in term of time units to ensure fair temporal compensation.

For the compensation model, we preserve the feature of graceful compensation among lagging flows but address issues of multirate flows and arbitrary packet size. This is achieved by introducing a compensation tag defined as $\frac{L_g}{C_g(t)} \cdot \frac{1}{credit_g(t)}$ for lagging flow g, given that L_g is the head-of-the-line packet size. Then, the lagging flow with the smallest compensation tag is selected to receive the compensation time slice. In short, the compensation model seeks to allocate compensation time slices fairly among lagging flows.

An additional benefit of MWFS is its backward compatibility with WFS [6] in the single-rate scenario. If all flows perceive identical transmission rates, then MWFS degenerates to WFS. In fact, fairness and compensation models based on throughput share and temporal share will be equivalent in the single-rate case.

4 Simulation Evaluation

We now present simulation results to evaluate MWFS in various scenarios. We compare its performance with WFS [6]. Four type of traffics are considered in the simulations, i.e., FTP, CBR, Poisson and Markov-modulated Poisson Process (MMPP) sources. Packet size for each flow may vary. We use one-step prediction [3] to estimate the immediate future channel based on the current channel state (i.e., clean/dirty). This exploits the feature that channel errors are highly correlated over short time. Each simulation run lasts for 100000 units unless otherwise explicitly stated, and the results are averaged over 50 simulation runs.

4.1 Throughput gain in error-free channel

We consider six FTP flows in the error-free scenario to show the throughput improvement of MWFS over WFS. The comparison base is that each flow uses 2 Mbps transmission rate. In the multi-rate scenario, flows 1 and 2 transmit at 11 Mbps, flows 3 and 4 use 5.5 Mbps, and flows 5 and 6 still use 2 Mbps. We also vary the packet size of each flow in simula-

---

[1] This rate can be normalized with respect to the base rate.

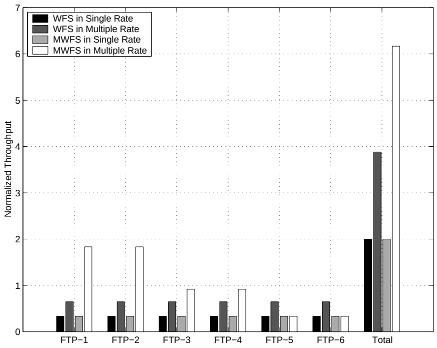

Figure 2: Throughput Gain of MWFS over WFS

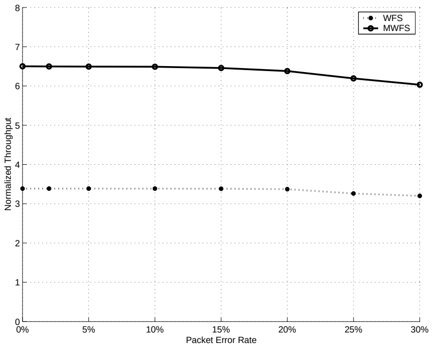

Figure 3: WFS vs. MWFS Throughput

Error Rate	FTP-1	FTP-2	FTP-3	FTP-4
0%	1.0000	1.0000	1.0000	1.0000
2%	0.9991	0.9991	1.0015	1.0005
5%	0.9982	0.9980	1.0019	1.0019
10%	0.9972	0.9979	1.0056	0.9977
15%	0.9915	0.9920	1.0094	0.9928
20%	0.9770	0.9779	1.0110	0.9940
25%	0.9513	0.9518	0.9604	0.9522
30%	0.9261	0.9254	0.9456	0.9294

Table 1: Normalized temporal share statistics for traffic sources

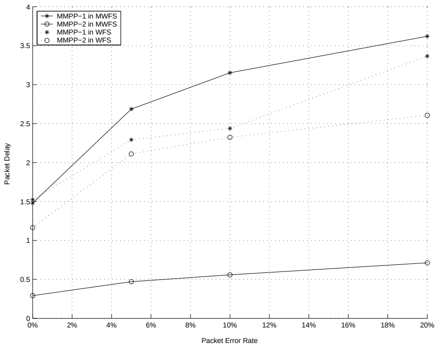

Figure 4: Packet delay performance for MMPP

tion runs. Figure 2 shows the per-flow throughput, as well as the overall throughput.

The figure shows that, MWFS achieves aggregate throughput 159% of WFS in the multi-rate scenario, while achieving throughput identical to WFS if all flows use the same rate. Significant throughput gain is also achieved on a per-flow basis, particularly for flows that use higher transmission rates. Throughput increases by 550% for flows 1 and 2, and increases by 270% for flows 3 and 4. Compared with the base case, WFS only increases 94% for each flow.

4.2 Throughput and fairness in error-prone channel

In this set of experiments, we study the effectiveness of the compensation model of MWFS in the presence of channel errors. The popular two-state discrete Markov Chain is employed to simulate channel errors. Four FTP source are used, and two flows (FTP-3,4) use base transmission rate 2.0 Mbps and the other two (FTP-1,2) transmit at 11 Mbps.

The throughput results for WFS and MWFS are depicted in Figure 3, where channel error varies from 0% up to 30%. We observe that as channel error increases to 20%, the throughput for both algorithms begins to suffer moderately because the probability that all flows are simultaneously error-prone increases. However, the overall throughput of MWFS remains approximately 87.5% to 92% higher than WFS.

To study the temporal fairness and effectiveness of compensation model, we record the normalized time share acquired by each flow in Table 1. When the channel is clean, each flow obtains an equal temporal share 1 unit. Note that as the channel error increases, the time units obtained by flows decrease, and this subsequently leads to throughput reduction, shown in Figure 3. However, the long-term temporal shares of all flows are still roughly preserved, thus showing that the compensation model is working. Even when channel error rate increases to 20%, the system throughput and time share gained by each flow only reduce slightly. This demonstrates that MWFS is still able to shield errors from flows and retain good overall throughput.

4.3 Packet delay in error-prone channel

We study the impact of error-prone channel on packet delay. Flows 1 and 2 are MMPP sources with packets arriving at the rate 1.0, and flows 3 and 4 are Poisson sources with packet arrival rate of 0.5. Two CBR flows with rate 1.0 are to emulate the background traffic. The transmission rates for MMPP-1, Poisson-1 and CBR-1 are set as 2 Mbps, and the other three sources use 11 Mbps. The error patterns are the same as in Section 4.2.

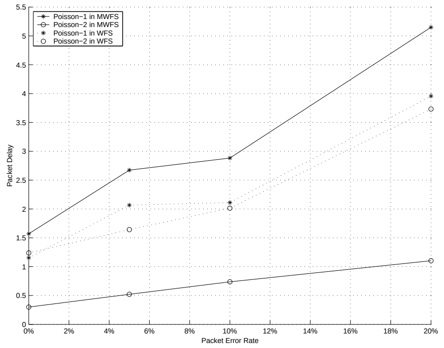

Figure 5: Packet delay performance for Poisson flows

The packet delay for MMPP and Poisson flows are depicted in Figure 4 5 respectively. It shows that, the delay experienced by each flow increases as channel error becomes severe. Because WFS ensures throughput fairness, the delays experienced by high-rate flows and low-rate flows are approximately the same. However, delay is different for low- and high-rate flows in MWFS. High-rate flows, MMPP-2 and Poisson-2, experience noticeably less delay than those low-rate flows. However, low-rate flows still have delay performance comparable to WFS. This shows that MWFS is able to provide certain degree of flow separation among high-rate and low-rate flows, such that high-rate flows will not be penalized or even paralyzed by low-rate flows.

5 Conclusion

The main contribution of this paper is a novel scheduling algorithm MWFS that ensures packet-level QoS in terms of minimum throughput, fair channel share, and maximum packet delay. Unlike wireless scheduling algorithms in the literature [2, 6], MWFS is able to opportunistically exploit multi-rate capability in physical layer and significantly improve overall channel throughput. In stead of providing throughput-based fairness, MWFS renovates fairness in term of time share, which allows flows in good channel condition to receive more services proportional to their higher rates. WMFS works with current 802.11 MAC and can handle variable packet size and idle flows. The simulation results confirm the effectiveness of WMFS in multi-rate wireless LANs.

References

[1] B. O'Hara and A. Petrick. IEEE 802.11 Handbook, A Designer's Companion. IEEE Press, 1999.

[2] T. Nandagopal, S. Lu, and V. Bharghavan. A Unified Architecture for the Design and Evaluation of Wireless Fair Queueing Algorithms. Proc. MOBICOM'99.

[3] S. Lu, V. Bharghavan, and R. Srikant. Fair scheduling in wireless packet networks. IEEE/ACM Transactions on Networking, Aug. 1999.

[4] T. Ng, I. Stoica, and H. Zhang. Packet fair queueing algorithms for wireless networks with location dependent errors. In Proceedings of IEEE INFOCOM'98, San Francisco, CA, May 1998.

[5] Parameswaran Ramanathan , Prathima Agrawal. Adapting packet fair queueing algorithms to wireless networks. Proceedings of the fourth annual ACM/IEEE international conference on Mobile computing and networking,

[6] S. Lu, T. Nandagopal, and V. Bharghavan. A Wireless Fair Service Algorithm for wireless packet networks. ACM MOBICOM 1998.

[7] Goyal, P., Vin, H. M., and Cheng H. Start-time Fair Queuing: A Scheduling Algorithm for Integrated Services Packet Switching Networks. IEEE/ACM Transactions on Networking, October 1997.