Buffer-Aware and Traffic-Dependent Packet Scheduling in Wireless OFDM Networks

Jinri Huang, Zhisheng Niu
State Key Lab on Microwave and Digital Communications
Department of Electronic Engineering, Tsinghua University,
Beijing, 100084, P.R. China,
Telephone: +86-10-62794328 Fax: +86-10-62773634

Abstract—Most current research on wireless packet scheduling assumes infinite buffer space and fixed packet length, which is impractical for real systems. In this paper, we propose a practical packet scheduling algorithm for wireless packet-switched OFDM systems, called Buffer-Aware and Traffic-Dependent (BATD) scheduler, which consists of two interactive parts: PHY-layer adaptive modulation and coding (AMC) technology on each subcarrier and MAC-layer opportunistic scheduling. Aiming at decreasing the buffer overflow probability while keeping the system throughput as large as possible, the scheduling decision is made based on the information of not only channel conditions but also buffer status as well as traffic characteristics. In addition, the scheduler is able to guarantee certain fairness among users by introducing the so-called “history service function”. Simulation results show that under practical constraints of finite buffer space and variable packet length, the proposed algorithm outperforms traditional packet scheduling schemes significantly in terms of packet loss rate due to buffer overflow and the total system throughput.

I. INTRODUCTION

Providing diverse QoS for heterogeneous classes of traffic is one of the major challenges for future wireless networks. At different layers of a network, QoS requirement appears in different forms. For example, at packet level, QoS appears in terms of packet delay, data rate, delay jitter and so on. Packet scheduling is an important mechanism to guarantee QoS at this level.

Packet scheduling was first developed in wireline environment and then extended to wireless networks. The essence of these schemes lies in that the transmission of packets is simply deferred when the channel is in deep fading and then compensated when the link qualities recover. The limitation of these scheduling algorithms is that channels are modelled as either “good” or “bad”, which is too simple to characterize realistic channels, especially for data services. A good survey of these schemes can be found in [1]. Recently, new schemes of packet scheduling for wireless networks have emerged, which make use of the independence among channels of multi users [2]. By scheduling the users when their channel conditions are favorable, these opportunistic scheduling schemes can achieve multiuser diversity and maximize system throughput. However, despite of their significance, most schemes have two drawbacks. Firstly, infinite buffer space is assumed. That is, any arrived packet can be buffered and any packet dropping due to buffer overflow will not happen. Secondly, most of the approaches assume the packet length is fixed, which is unrealistic. Packets of different length occupy different buffer space and have different requirements for system resources such as bandwidth. For practical applications, packet scheduling should be aware of the buffer status as well as packet characteristics.

While packet scheduling provides QoS guarantee at packet level, resource allocation at PHY layer can meet QoS requirement at bit level and further improves system performance, especially for OFDM networks, which supply an additional degree-of-freedom, i.e., subcarrier space. Recently, the interest in resource allocation for OFDM systems has transferred from confining to single layer (i.e., PHY layer) to cross-layer design. Especially, joint MAC-PHY resource allocation and scheduling has attracted much research interests [3-7]. For example, in [3], Zhang proposed a cross-layer link-adaptive largest-weighted-throughput (LWT) scheduling algorithm for real-time applications under a realistic continuous and frequency-selective fading channel model. In [5], Ryu designed a UEPS scheduling algorithm to support both real-time and non-real-time traffics in OFDMA wireless systems. Despite the performance gain these schemes have gotten, all of them fail to take buffer status as well as packet length into account. They leave the problem mentioned above (i.e., finite buffer space and variable packet length) untouched.

In this paper, we are motivated to propose a cross-layer buffer-aware and traffic-dependent (BATD) packet scheduling for downlink transmission of non-real-time traffic in wireless packet-switched OFDM networks. In contrast to [3-7], our algorithm deals with more practical case in which the buffer space is finite and different packets have different length.

The proposed scheduling scheme is composed of two interactive parts: MAC-layer opportunistic scheduling and PHY-layer adaptive modulation and coding (AMC) technology. The MAC scheduler schedules packet transmission based on an overall consideration of queue (buffer) status, link throughput that can be achieved and “history service” the users have gotten from the system. Meanwhile, at PHY layer, we adopt AMC technology to maximize per-link data rate, maintain the packet error rate (PER) at a sufficiently low level, and make the wireless channel appear to be almost error free to upper
layers. Thanks to the cross-layer design and the integrated consideration of buffer status and channel conditions, our scheme outperforms the traditional packet scheduling algorithms. The ultimate goal is to decrease packet dropping rate due to buffer overflow, improve total system throughput and supply certain fairness among users.

The remainder of this paper is organized as follows. In Section II, we introduce the system model of this proposed scheme. Then, the detailed cross-layer BATD packet scheduling scheme is to be presented in Section III. Section IV gives numerical results comparing with other traditional scheduling schemes to demonstrate the performance of our algorithm. Finally, we make conclusions in Section V.

II. SYSTEM MODEL

Figure 1 illustrates the system structure of our proposed scheme. We focus on downlink transmission of a packet-switched network with OFDM signaling. Therefore, it is the base station (BS) that makes scheduling decision for packet transmission. Suppose that there are \( N_c \) subcarriers and \( K \) active traffic flows in the current system. The packet arrival process is modeled as a Poisson process with independent rate \( \lambda_i \) (\( i = 1, 2, ..., K \)). Upon each packet arrival, the BS puts the packet into its corresponding buffer which has finite space of \( L_i \) (\( i = 1, 2, ..., K \)) bits. We further assume that different packets may have different length of bits. We deal with packet by packet transmission, which is reasonable since a packet is usually long enough so that it requires several OFDM symbols to carry. After finishing transmission of a packet, a HOL (Head Of Line) packet in buffers will be selected for next transmission by our BATD scheduler. We assume the base station has knowledge of instantaneous channel state information (CSI) at the beginning of each packet scheduling. With this CSI, the BS can implement adaptive modulation and coding (AMC) to maximize the throughput on each subcarrier. Furthermore, this throughput information can be passed to the BATD scheduler, which is the core of our cross-layer design scheme. Making scheduling decision based on the following factors: buffer status (i.e., spare buffer space), traffic characteristics (i.e., packet arrival rate and packet length), per-link condition and “history service”, the BATD scheduler selects a HOL packet and passes it to PHY layer for transmission.

![Transmitter structure at the base station](image)

In this paper, we focus on non-real-time data traffic, which can tolerate certain packet delay but is sensitive to packet loss. Packet loss occurs due to two factors: packet transmission error caused by erroneous channels and packet dropping because of buffer overflow, which happens when the spare space of the buffer is deficient upon new packet arrival. Similar to other scheduling schemes, we assume that ARQ (Automatic Repeat Request) is applied to recover the packet errors, and a packet is retransmitted until it is received successfully. However, traditional algorithms usually assume infinite buffer space. That is, there will not be any buffer overflow happened. While in our scheme, we deal with more practical situation of finite buffer space and integrate the buffer status into the scheduling decision to decrease the packet dropping rate. Furthermore, to supply certain fairness among users, we introduce a “history service function” \( f_i(t) \) for each session \( i \) to reflect the services it has gotten from the system until time \( t \), which will be further described in the following section.

III. JOINT MAC-PHY PACKET SCHEDULING ALGORITHM

In this section, we present our cross-layer packet scheduling algorithm in details. This scheme comprises two parts: MAC BATD packet scheduler and PHY AMC technique with OFDM signaling. First, adaptive modulation and coding (AMC) technique is implemented at PHY layer for each subcarrier to maximize the system throughput and then, at MAC layer, our BATD scheduling algorithm is presented.

A. PHY AMC technique

Adaptive modulation and coding (AMC) is being used in most 2.5/3G wireless networks to increase the transmission rate by exploiting the wireless channel variations [8]. The objective of AMC is to maximize the data rate by adjusting transmission parameters to the available CSI, while maintaining a prescribed bit error rate (BER) or packet error rate (PER) [9]. Let \( c_{ij} \) denote the maximum number of bits per OFDM symbol (per Hz) carried by the \( j \)th (\( j = 1, 2, ..., N_n \)) subcarrier of the \( i \)th (\( i = 1, 2, ..., K \)) user. Then the \( c_{ij} \) can be expressed as a function of the received signal noise ratio (SNR) \( \gamma_{ij} \) and target BER \( P_b \). According to [10], \( c_{ij} \) can be well approximated by

\[
c_{ij} \approx \log_2(1 + \Gamma \gamma_{ij})
\]

where \( \gamma_{ij} \) denotes the received SNR on the \( j \)th subcarrier of the \( i \)th user and \( \Gamma \) is a constant, called the SNR gap. For practical signal constellations (e.g., QAM), \( \Gamma \) is related to a given BER requirement. For example, for M-QAM modulation, \( \Gamma \) can be determined by [10]

\[
\Gamma = -1.5 \frac{1}{\ln(5P_b)}.
\]

Suppose there are totally \( N \) available transmission modes, each mode corresponding to one transmission rate, denoted by \( r_n \) (\( n = 1, 2, ..., N \)) (bits/symbol). For example, for \( 2^n \)-QAM \( (n = 1, 2, ..., \log_2(2^n)) \) modulation scheme, \( r_n \) can be calculated as \( r_n = \log(2^n) = n \) (bits/symbol). Then \( c_{ij} \) is in the set of \( r_n \), that is, \( c_{ij} \in \{r_1, r_2, ..., r_N\} \). With AMC, we partition the
SNR range into \( N + 1 \) nonoverlapping consecutive intervals, with boundary points denoted by \( \{ \gamma_1, \gamma_2, \ldots, \gamma_N \} \). By Eqn. (1), these boundary points can be calculated as (Note that \( c_{ij} \in \{ r_1, r_2, \ldots, r_N \} \))

\[
\gamma_n = \frac{(2^r - 1)}{\Gamma}, n = 1, 2, \ldots, N.
\]

The basic principle of AMC is that the transmission mode \( n \) will be chosen for the \( j \)-th subcarrier of the \( i \)-th user when \( \gamma_n \leq \gamma_{ij} < \gamma_{n+1} \) (\( n = 1, 2, \ldots, N - 1 \)). For any \( \gamma_{ij} \geq \gamma_N \), transmission mode \( N \) will be applied. To avoid deep channel fading and unnecessary transmission loss, no packet will be transmitted whenever \( \gamma_{ij} < \gamma_1 \).

Once AMC is implemented on each subcarrier for every use, the total data rate the \( i \)-th user can achieve is given by

\[
T_i = \sum_{j=1}^{N_s} c_{ij}, i = 1, 2, \ldots, K.
\]

This throughput information of each user is passed to the BATD packet scheduler at MAC layer, which is the core of our algorithm and will be further described next.

### B. BATD packet scheduling

Our buffer-aware and traffic-dependent (BATD) packet scheduling algorithm aims at achieving three objectives: 1) keep packet dropping rate as low as possible by means of taking buffer status into account. 2) maximize the total system throughput and 3) keep certain fairness among users in terms of “history service function” \( f_i(t) \) which will be defined in this section.

Let \( \Delta_i(t) \) denote the spare space of buffer \( i \) at time \( t \), which represents the time at the beginning of each packet scheduling. To reflect the amount of service the \( i \)-th user (i.e. session) has gotten from the system till time \( t \), we introduce a “history service function” \( f_i(t) \). The larger the \( f_i(t) \) is, the more service the system has supplied for this user. Thus, the difference of the history service functions, i.e., \( \Delta_{ij} = f_i(t) - f_j(t) \), indicates the fairness among users to some extent. The smaller the difference \( \Delta_{ij} \) is, the fairer the system is for all the users.

Thus, the packet scheduling endeavors to solve the following problem: given the buffer status \( \Delta_i(t) \), traffic characteristics (i.e., packet arrival rate \( \lambda_i \) and the length of packets), the achievable data throughput \( T_i \) and history service function \( f_i(t) \), which HOL (head-of-line) packet should be chosen for transmission so that the three objectives mentioned above can be achieved.

In order to decrease the packet dropping rate due to buffer overlow, intuitively, we should schedule those queues when their spare buffer space is small or the size of the HOL packet is large, since the transmission of such packets means release of more buffer space, which prevents buffer overlow when new packets arrive. On the other hand, if the packet arrival rate \( \lambda_i \) is high, the corresponding queue should be given higher priority for scheduling. The reason is that frequent packet arrival tends to lead to buffer overlow if the service rate cannot be guaranteed. Furthermore, from the point of view of system throughput, those queues which have better channel condition (i.e., the achievable link throughput \( T_i \) is large), should be scheduled to maximize the system throughput. Finally, in order to guarantee certain fairness among users, according to the meaning of \( f_i(t) \), the scheduling priority of a queue should be inversely proportional to \( f_i(t) \). That is, the smaller the \( f_i(t) \) is, the more possible the \( i \)-th queue may be scheduled.

Based on the analysis above, we propose our BATD scheduling algorithm as follows.

After transmission of a packet, serve the queue \( i^* \) for next transmission which satisfies:

\[
i^* = \arg \max_i \frac{\lambda_i P_i(t) T_i(t)}{\Delta_i(t) f_i(t)}
\]

where \( \alpha \) is a positive constant, \( t \) represents the time at the beginning of current packet scheduling, \( P_i(t) \) denotes the HOL packet length of the \( i \)-th queue at time \( t \), \( \lambda_i \) is the Poisson packet arrival rate, \( T_i(t) \) indicates the achievable throughput for the \( i \)-th user at time \( t \), \( \Delta_i(t) \) represents the spare space of the \( i \)-th buffer at time \( t \) and \( f_i(t) \) is the history service function for the \( i \)-th user at time \( t \).

In (2), \( \lambda_i \), \( P_i(t) \) and \( \Delta_i(t) \) are related to the buffer status or traffic characteristics and are known to the scheduler, while the throughput information of \( T_i(t) \) can be passed by PHY layer to the scheduler (see Fig. 1). Therefore, the last step is to design the history service function \( f_i(t) \). We define \( f_i(t) \) as follows.

\[
f_i(t) = \begin{cases} (1 - \beta)f_i(t - 1) + \beta \frac{P_i(t - 1)}{T_i(t - 1)}, & \text{queue } i \text{ is served} \\ (1 - \beta)f_i(t - 1), & \text{otherwise} \end{cases}
\]

where \( f_i(0) = 0 \) and \( \beta \) is called the low pass filtering parameter, which has similar meaning to \( t_e \) in Proportional Fair (PF) scheduling scheme [11].

The scheduling rule described in Eqn. (2) and (3) is referred to as Buffer-Aware and Traffic-Dependent, since the priority of each HOL packet is given under the consideration of buffer status (i.e., spare buffer space \( \Delta_i(t) \)) as well as traffic characteristics (i.e., packet arrival rate \( \lambda_i \) and packet length \( P_i(t) \)). This critical feature makes preventing packet dropping due to buffer overflow and thus keeps the packet dropping rate as low as possible. In addition, our scheme possesses the merits of pure opportunistic (PO) scheduling to maximize the system throughput by taking \( T_i(t) \) into account. Last, the introduction of \( f_i(t) \) can keep certain fairness among users.

### IV. Numerical results

In this section, the performance of the proposed BATD algorithm is evaluated and compared with various traditional scheduling schemes for wireless networks. An OFDM system with 64 subcarriers and an OFDM symbol duration \( T_s = 200 \mu s \) is considered. We assume that each user’s subcarriers
undergo identical Rayleigh fading independently. There are 5 different constellations available for AMC at PHY layer. That is, BPSK, 4QAM, 16QAM, 64QAM and 256QAM. To observe the effect of the algorithm more clearly, we consider the case that there are just two sessions in the system. At MAC layer, the packets arrive according to a Poisson process with arrival rate $\lambda_i = 500$ (i = 1, 2) packets per second. The packets of the first session have fixed length of 100 bytes and the packet length of the second session is uniformly distributed in [80,120] bytes. The space of the second buffer is set long enough so that there will not be any packet dropping happened. We concentrate on the packet dropping rate of the first buffer.

Fig. 2 and Fig. 3 show the buffer overflow probability (i.e., packet dropping rate due to buffer overflow) versus the average transmitted SNR of one subcarrier of the first session $SNR_1$. For comparison, the performance of three widely accepted wireless service disciplines is also plotted. They are CSD-RR (Channel State Dependent Round Robin) [12], PO (Pure Opportunistic) and PF (Proportional Fairness) [11]. Adaptive modulation is adopted at PHY layer for the three schemes.

Fig. 2. Buffer overflow probability of the first session when $SNR_1 = SNR_2$.

In Fig. 2, we set the average SNR of the second user $SNR_2$ equal to that of the first user, i.e., $SNR_1 = SNR_2$. From the figure, we can see that when the channel condition is poor, i.e., the average SNR is low, the buffer overflow probability is high for all four packet scheduling schemes. This is because at low SNR region, the system service rate is lower than the packet arrival rate, which leads to frequent buffer overflow. As the average SNR increases, this overflow probability decreases. Moreover, the proposed BATD algorithm significantly outperforms other schemes since it enjoys the most rapid decreasing rate. For instance, a 3~4dB power gain over the PO, PF and CSD-RR schedulers is observed at buffer overflow probability $10^{-2}$.

While Fig. 2 describes the scenario that both users undergo the same channel conditions, i.e., $SNR_1 = SNR_2$, in Fig. 3, we investigate another case. In this case, we fix $SNR_2$ (at 8dB) and change $SNR_1$. From Fig. 3, it can be seen that the observations mentioned above still hold. What is more, by comparing the two figures, we can see that the difference of the user’s SNR affects the performance of CSD-RR, PO and PF scheduling system, while leaves the BATD algorithm uninfluenced. From this point, it may be said that the proposed algorithm is stable. In addition, for both scenarios, the proposed BATD scheme enjoys the lowest buffer overflow probability. One of the main reasons lies in that we integrate the buffer status into our scheduling decision.

Fig. 3. Buffer overflow probability of the first session when $SNR_2 = 8dB$, $SNR_1 = 2 : 1 : 14$

In Fig. 4, the total system throughput, which is defined to be the total number of successfully transmitted packets in the simulation time, is plotted as a function of the SNR of the first user, when $SNR_2$ is fixed. The total system throughput is closely related to the packet loss rate due to buffer overflow. From the figure we can see that at the low SNR region the system throughput is low for all four scheduling schemes due to the corresponding high buffer overflow probability. Then
this throughput increases with the SNR. At the high SNR region when the buffer overflow probability is low enough to be negligible, the system throughput is mainly determined by the amount of input traffic. That is, the curves in Fig. 4 will converge to a constant (assume the total amount of input traffic is constant). In Fig. 3, we see that the proposed BATD scheme possesses the most rapid decreasing rate in buffer overflow probability. As a result, correspondingly in Fig. 4, we can see that the proposed BATD algorithm converges faster than any other schemes. Therefore, our scheme significantly outperforms CSD-RR and PF schedulers, and is slightly better than PO algorithm. For instance, the system throughput is increased by around 23% compared to CSD-RR and PF schedulers when $\frac{SNR_1}{SNR_2} = 9dB$. We believe the improvement is due to the appropriate packet selection criterion as well as the successfully exploitation of multiuser diversity and channel variation in the time and frequency domains.

![Fig. 5. Transmission efficiency when $SNR_2 = 6dB$, $SNR_1 = 2 : 1 : 16$](image)

While Fig. 4 describes the system performance from the point of total system throughput, Fig. 5 shows the transmission efficiency of the multiuser OFDM system for different scheduling schemes. Defined as the number of successfully transmitted bytes per OFDM symbol, the transmission efficiency highly depends on the packet dropping rate and the channel condition. From Fig. 5, it can be seen that at low SNR region, the transmission efficiency of the proposed BATD scheme is lower than any other schemes. The reason lies in that at this region, the packet loss rate is relatively high for the system. To avoid too much packet dropping and maintain lower packet loss rate, the BATD scheduler transmits packets even when the channel condition is worse, which leads to the degradation of the transmission efficiency. Therefore, it is a trade-off between the packet loss rate and the transmission efficiency at low SNR region. On the other hand, when the two users have similar channel condition (e.g., $SNR_2 = 6dB$ and $SNR_1 \in [6, 11]$), the transmission efficiency of the BATD exceeds that of other schemes greatly. For instance, about 4 dB is observed when the transmission efficiency is 6.5 bytes per symbol. The main contribution comes from the more transmitted packets (thus more bytes) that the BATD scheduler can guarantee. Finally, as the SNR further increases, all the scheduling schemes have almost the same performance. However, the performance of the BATD algorithm is still slightly better than other schemes.

V. CONCLUSION

In this paper, we develop a cross-layer BATD packet scheduling algorithm for wireless packet-switched OFDM systems. Under the constraint of finite buffer space, this proposed scheme endeavors to decrease the packet dropping rate due to buffer overflow, while keeping the total system throughput as large as possible and being able to guarantee certain fairness among users. The BATD scheduling algorithm is composed of two interactive parts. At PHY layer, AMC technology is adopted for each subcarrier of the OFDM system to maximize per-link throughput. Then at MAC layer, the decision of packet selection is made based on an integrated consideration of buffer status, traffic characteristics, the achievable throughput and the history service record of each user. Numerical results show that comparing to traditional scheduling schemes, the packet loss rate due to buffer overflow can be reduced greatly and the total system throughput is improved significantly by the proposed BATD algorithm. Meanwhile, the transmission efficiency is better than other traditional scheduling schemes under certain conditions.

REFERENCES