QoS-guaranteed packet scheduling in wireless networks

ZHANG Deng-yin¹² (✉), JIANG Juan¹, ANANI Adi², LI Hai-bo²

1. Institute of Signal Processing and Transmission, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
2. Department of Applied Physics and Electronics, Umeå University, Umeå 90187, Sweden

Abstract

To guarantee the quality of service (QoS) of a wireless network, a new packet scheduling algorithm using cross-layer design technique is proposed in this article. First, the demand of packet scheduling for multimedia transmission in wireless networks and the deficiency of the existing packet scheduling algorithms are analyzed. Then the model of the QoS-guaranteed packet scheduling (QPS) algorithm of high speed downlink packet access (HSDPA) and the cost function of packet transmission are designed. The calculation method of packet delay time for wireless channels is expounded in detail, and complete steps to realize the QPS algorithm are also given. The simulation results show that the QPS algorithm that provides the scheduling sequence of packets with calculated values can effectively improve the performance of delay and throughput.

Keywords packet scheduling, QoS, wireless network, cross-layer design

1 Introduction

The multipath fading and shadow effects in a wireless channel, as well as high dynamic characteristics of wireless network topology, make a wireless link with high bit error rate and packet loss rate. Thus, it faces plenty of challenges to provide reliable high-quality service in the mobile communications network. Moreover, based on dynamic changes in the network topology and different types of heterogeneous access networks, next-generation wireless networks must be able to support the multimedia communications of multiple QoS requirements, and simultaneously ensure high system throughput and low transmission delay. These require a scheduling technology of wireless networks with very high specific performance. In the design of modern communication systems, a packet scheduler has been transferred from the side of the radio network controller (RNC) to the side of node B (base station) [1], and thus it can easily obtain air interface measurement parameters, which are more likely to match the data speed of a different wireless channel and meet different QoS requirements of high-speed packet data services. However, the most crucial of all is that scheduling algorithms can maximize system throughput and reduce packet transmission delay.

Currently, there are many well-known packet scheduling algorithms (PSA), and their typical examples are the maximum carrier to interference algorithm (Max C/I) [2], the round robin algorithm (RR) [3], and the proportional fair algorithm (PF) [4]. The Max C/I [2] sorts predictive values with all pending mobile services based on its received signal C/I being sent in a descending order, and maximizes system capacity. In this way, the mobile station near the base station receives services all along for its good channel conditions, and the users in the marginal area gain scarce service opportunities because of lower C/I. Therefore, Max C/I is the most unfair one in all PSAs. In contrast, the RR [3] communicates by recurrently occupying wireless resources of equal time in accordance with a certain determined order, thus assuring its users of long-term equity and having an advantage of low complexity in its realization. Unfortunately, the RR results in very low system throughput, because it does not distinguish the specific wireless channel requirement among different users. As for the PF [4], it assigns each user a corresponding priority in a small area, and the user with the maximum priority gets the service first. Since the (C/I)(t) of different users is an independent and identical distribution, users can occupy the same length of time to communicate in this area, hence it is a fair scheduling algorithm. Moreover, as a result of the selection of a service moment, users only obtain service in relatively good conditions with fast fading, thus, the system throughput can be improved. As the algorithm does not consider the packet delay of different QoS
requirements, it only applies to non-real-time services.

For many data services with higher requirements of both flow and delay, such as video, streaming media and download services, it is necessary for a system to provide higher transmission rates and shorter latency [5]. To develop better data services, the third-generation partnership project (3GPP) has made an improvement in the air interface in these two areas, and has introduced the HSDPA technology [6] in the R5 version. The HSDPA significantly increases network capacity, and enables the supports to minimize input costs. It is known as one of the main solutions beyond 3G times, and provides a stable method for a universal mobile telecommunications system (UMTS) because of a higher data transfer rate and higher capacity evolution. To maximize the throughput of a wireless network and minimize packet transfer delay, a new QPS algorithm of the HSDPA is proposed in this article. Applying the thought of cross-layer design, the QPS focuses on both the channel quality at the physical layer and the packet delay at the MAC layer.

2 QPS algorithm design

2.1 QPS algorithm model

The aim of the wireless packet scheduling is to maximize the network throughput and to reduce transmission time to meet the users’ QoS requirements [7]. To improve the throughput, the method of wireless signal transmission must be optimized, considering the real-time change of network conditions and the current status of wireless channel quality. Real-time monitoring of the dynamic changes of network conditions needs sensing of a more fine-grained delay and packet loss rate. In wireless communications, the packet loss of the transmission control protocol (TCP) layer is mainly caused by the high error rate in the wireless link because of its poor transmission characteristics. It is known that the packet delay and the packet loss rate are the characteristics of the network layer and the physical layer respectively, and the packet scheduling module operates at the media access control (MAC) layer [8]. Here, the aim is to work out coordinated packet scheduling through sharing the delay information among the physical layer, the MAC layer and the network layer. As packet delay directly depends on packet arrival rate, packet size and modulation mode, it is feasible to perceive the characteristics of packet transmission in advance based on the delay. In this way, a packet scheduling scheme at the MAC layer can be obtained using cross-layer design technique.

Based on the above consideration, the QPS packet scheduling algorithm is proposed with its model shown in Fig. 1. First, according to the packet arrival rate \( \lambda \) and packet size, the channel environment can be simulated using the Markov Chain to forecast possible packet delay. Then the associated cost can be computed based on a designed cost function, and the packets are queued in an ascending order. The packet with the lowest cost will be sent first. The packet scheduler only needs to process the first packet in the queue, and then one by one each time. The complexity of packet scheduling, for simplicity, can be said to consist in achieving real-time scheduling and meeting the requirements of high-speed data transmission in next-generation wireless networks. The QPS algorithm continuously evaluates real-time channel quality of the wireless network, dynamically monitors network changes, and then does real-time granular adjustment, to effectively regulate the throughput. Moreover, it balances the QoS parameters and ensures the fairness of each user.

![Fig. 1 QPS algorithm model](image)

2.2 Cost function design

Packet delay and throughput are two important indicators for evaluating the performance of a packet scheduling algorithm, and are closely related to the performance of the service the network provides. Because of the difference in the service type (e.g., real-time voice and video services and non-real-time data service) and the associated largest allowable packet delay, the design of the cost function must consider the requirement of the delay parameter.

The delay assessment of a wireless network includes two metrics, the normalized average delay rate \( T \) and the number of excessively delayed packets \( C \) given by Eqs. (1) and (2) respectively.

\[
T = \frac{1}{N} \sum_{k=1}^{N} \frac{t_k}{r_k} \tag{1}
\]

\[
C = \sum_{k=1}^{N} \max \left\{ 0, \frac{t_k}{r_k} - 1 \right\} \tag{2}
\]

where \( t_k \) is the actual end-to-end transmission delay of the packet \( k \), \( r_k \) is the required delay value based on a user’s QoS, and \( N \) is the number of sampling packets. When \( T \) approaches 1, it means that the packet is transmitted within the allowed band of delay requirement.

Some classical scheduling algorithms have been discussed
earlier. For example, the RR algorithm transmits the packet by cycling wireless resources in a sequence, thus it can ensure fairness. However, the RR does not take into account specific conditions of different users in the wireless channel; hence it results in a very low throughput. The PF algorithm considers the user’s channel environment and the average transmission rate, therefore, it achieves a good balance between throughput and fairness. Nevertheless, the PF does not consider packet delay, thus it cannot meet the QoS requirement of multiple users. To design a scheduling algorithm with higher performance both in delay and throughput, based on the evaluation parameters T and C, as well as the weight parameter β, the authors defined a new cost function F that has a good balance between delay and a channel quality of the wireless network, as shown below:

\[ F = \sum_{i=1}^{n} \left(1 - \beta \right) \frac{1}{t_i} + \beta \max \left( \frac{1}{t_i} - 1, 0 \right) \]  

(3)

In Eq. (3), β is the parameter initialized by the system, and it represents the rate between the actual delay rate T and the packet number C, whose actual transmission time exceeds the delay requirements.

### 2.3 Link delay estimation

It is difficult to acquire the actual end-to-end transmission delay of a packet in real-time. Here, it is replaced by a link delay estimation of the packet to compute the cost function. Discrete-time Markov Chain (DTMC) is a classical model used to simulate the wireless channel, as shown in Fig. 2. The authors used two-state DTMC to simulate wireless-channel environment [9]. Assume that a service data unit (SDU) encapsulates \( n \) pieces of a protocol data unit (PDU) at the MAC layer, and the states \( S_0 \) and \( S_1 \) are used to represent successful and unsuccessful PDU transmissions respectively. Suppose further that the channel feedback is error-free. Then, the transition probability matrix of the Markov model is:

\[
T_{CM} = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix}
\]

Its steady-state probability distributions are \( \pi_0 = p_{00}/(p_{00} + p_{01}) \) and \( \pi_1 = p_{10}/(p_{00} + p_{10}) \) respectively.

![Fig. 2 Finite state Markov channel model](image)

Assume that each PDU is independent of the others, the under state is \( i \) (including \( p_0 \) and \( p_1 \)), and the probability of \( m \) pieces of a PDU in an SDU transmitted unsuccessfully is \( P_m = (n/m) \pi^m (1 - p_i)^{n-m} \). The transition matrix of the SDU transmission is:

\[
T_{CS} = \begin{bmatrix} 1 & 0 & \ldots & 0 \\ p_{00} & p_{10} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ p_{0N} & p_{1N} & \ldots & p_{NN} \end{bmatrix}
\]

Then it follows that:

\[
Q = \begin{bmatrix} T_0 & 0 \\ 0 & T_1 \end{bmatrix}, \quad T_{CM} = \begin{bmatrix} p_{00}T_0 & p_{01}T_0^* \\ p_{10}T_1 & p_{11}T_1^* \end{bmatrix}.
\]

Here, the \( Q \) matrix means that if the system is in state \( S_0 \) at time \( t \), the probabilities of it remaining in state \( S_0 \) and transferring to state \( S_1 \) at the time of \((t + \Delta t)\) are \( p_{00}T_0 \) and \( p_{01}T_0^* \) respectively. Similarly, if the system is in state \( S_1 \) at time \( t \), the probabilities of it remaining in state \( S_1 \) and transferring to state \( S_0 \) at the time of \((t + \Delta t)\) are \( p_{10}T_1 \) and \( p_{11}T_1^* \) respectively.

It is hypothesized that the largest re-established number of PDU is \( L(L \geq 0) \), thus, the loss probability of the SDU is \( p_{L0}(L) = \pi^e \) where \( e = [1 \ 1 \ \ldots \ 1]^T \) and \( \pi = [0 \ 0 \ \ldots \ \pi_0 \ 0 \ \ldots \ \pi_a] \). According to M/G/1, the unlimited source queuing theory, the average service time for correctly transferring an SDU is:

\[
E[S] = \frac{\sum_{n=1}^{L} n P(S = n)}{\sum_{n=1}^{L} n \pi^e \alpha}
\]

where \( \alpha = [P_0 \ P_{20} \ \ldots \ P_{ka}] \). According to the Pollaczek-Khinchine formula [10], the average queuing time is:

\[
E[W] = \frac{\lambda E[S^2]}{2(1 - \lambda E[S])}
\]

where \( \lambda \) is the packet arrival rate. Therefore, the delay is obtained when the wireless link transfers an SDU correctly:

\[
t = E[S] + \frac{\lambda \sum_{n=1}^{L} n \pi^e \alpha \lambda}{2 - 2\lambda \sum_{n=1}^{L} n \pi^e \alpha}
\]

### 2.4 QPS algorithm summarization

The QPS algorithm can be summarized as follows:

1) First, the probable delay, \( t \), of the packet transmission at the current transmission time interval (TTI) is estimated using the Markov Chain to simulate the channel environment.

2) Next, the cost required by the packet to execute the channel transmission is computed according to the cost function \( F \), as well as the delay rate \( T \), the packet number \( C \) whose transmission time estimation exceeds the delay requirements, and the required delay value \( r \).

3) For the packets that cannot obtain service for a long period because of poor channel environment, if the queue’s waiting time exceeds the QoS requirements, the packet is discarded directly. Otherwise, the cost function is recalculated.
in accord with the characteristics of the current channel.

4) Subsequently, sorting the packets of the queue elongated by the cost of packet transmission, the packet that costs the least becomes first to be sent. Therefore the packet scheduler only needs to take the first packet from the queue each time. In this way the complexity of packet scheduling can be simplified, thus making real-time scheduling available, and accommodating the requirements of high-speed data transmission in the next-generation wireless networks.

QPS algorithm is described in pseudocodes as follows:

```
begin
    EnClassifier()  // the packets enter the classifier
    ComputeDelayTime()  // estimate the delay value \( t \) according to the channel environment
    The cost function \( F() \rightarrow \) the delay value \( t \)
    Scanning queue, and waiting for the timer
    If (Timeout => \( t_{cut} \)) then
        Drop_packet()
        Return failing information "packet waiting for overtime, dropping packets"
    If (\( F(K) > F(K+1) \)) then
        Packet(K) <- Packet(K+1)  // sort the queue according to the value
    If (isQueueTail)
        sendPacket()  // send the packet to the physical layer
    else
        go Next Packet
end
```

3 QPS algorithm simulation

3.1 Simulation environment

To simplify the simulation and express the algorithm clearly, the authors set the system parameter \( \beta \) in the cost function to be 0.5, which means that both the normal delay rate, \( T \) and the value of the packet number \( C \), have the same weight. Hence the cost function is linear. We take network simulator NS-2 as the simulation environment to evaluate the QPS algorithm, and the access services consist of voice, video and data. The simulation parameters and delay requirements, \( r_k \), are shown in Tables 1 and 2 separately.

<table>
<thead>
<tr>
<th>Table 1 Simulation parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>Packet arrival rate ( \lambda )</td>
</tr>
<tr>
<td>Largest number of retransmission ( k )</td>
</tr>
<tr>
<td>Packet type</td>
</tr>
<tr>
<td>Packet size</td>
</tr>
<tr>
<td>Power control</td>
</tr>
<tr>
<td>Channel model</td>
</tr>
<tr>
<td>Modulation mode</td>
</tr>
<tr>
<td>Chip rate</td>
</tr>
<tr>
<td>Time slot</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2 Delay request of different packet types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Packet type</td>
</tr>
<tr>
<td>-----------------------------</td>
</tr>
<tr>
<td>Voice</td>
</tr>
<tr>
<td>Video</td>
</tr>
<tr>
<td>Data</td>
</tr>
</tbody>
</table>

3.2 Throughput calculation

The performance evaluation of the throughput in a wireless network is primarily displayed in the bit rate size of transmission. Whereas this method does not take into consideration the fact that the user’s packets overlap in the queuing time. In this article, the authors adopt a new indicator, that is, the average ratio of all the transferred bits to corresponding time consumption, which is defined as follows:

\[
H = \frac{1}{N} \sum_{k=1}^{N} \frac{b_k}{t_k}
\]

where \( b_k \) is the number of bits that packet \( k \) actually transmits, \( t_k \) is the consumption time, and \( N \) is the number of packets.

By sampling the throughput at various time points, the authors calculate the average value to represent the throughput performance. Such treatment is beneficial to the quantification of throughput.

3.3 Simulation results

The authors simulate the throughput and delay performance on the RR, PF, and QPS algorithms respectively. The results are shown in Figs. 3 and 4.

Figure 3 shows that the delay characteristics on the RR, the PF, and the QPS algorithms do not differ greatly when the network load is small. However, with the load increasing, the delay of the RR algorithm at the time of 11 s grows rapidly, mainly because of the waiting time for the polling packet. The delay of the PF algorithm increases gradually, whereas it is significantly higher than that of the QPS algorithm, mainly because the PF algorithm does not consider priority processing on the delay-sensitive real-time packets.
Fig. 4 shows that the throughput average value of the QPS algorithm is significantly larger than that of the RR and the PF algorithms. This shows that the QPS algorithm increases the efficiency of packet scheduling and optimizes the network throughput.

![Fig. 4 Average throughput performance](image)

4 Conclusions

This article presents a cross-layer scheduling algorithm (QPS) to improve the QoS performance of wireless networks. By analyzing the performance evaluation parameters of delay and throughput, a new cost function $F$ is designed. Considering the user’s QoS parameters, combined with the channel quality of the physical layer and the delay of the link layer, the scheduler analyzes each packet cost, finds the packet with the optimal cost, puts it in the front of the queue, and then sends it. Therefore, this algorithm can effectively use wireless network resources, guarantee fairness, and meanwhile balance the scheduling of the throughput. Future study will focus on improving the response rate of scheduling, and reducing the computation amount of node B to realize the real-time scheduling of high-speed packets.

Acknowledgements

This work was supported by the Joint European Project ICT-LEAP, the National Natural Science Foundation of China (60573141), the Hi-Tech Research and Development Program of China (2007AA701302), and the ‘Six Heights of Talent’ Project of Jiangsu Province.

References

1. Assaad M, Zeghlache D. Cross-layer design in HSDPA system to reduce the TCP effect. IEEE Journal on Selected Areas in Communications, 2006, 24(3): 614–625