A Real-Time Traffic Scheduling Algorithm in CDMA Packet Networks

Lei Zan
School of Information and Computer Science and CAL-(IT)2
University of California at Irvine
Irvine, CA 92697, USA
lzan@ics.uci.edu

Geert Heijenk
Computer Science Department,
Twente University, The Netherlands
geert@heijenk.com

Magda El Zarki
School of Information and Computer Science and CAL-(IT)2
University of California at Irvine
Irvine, CA 92697, USA
elzarki@uci.edu

Abstract—**The demands for multimedia and packet data services over wireless devices have increased over the past few years. The direct impact on performance makes scheduling for real-time traffic important. This paper presents a novel scheduling algorithm called Fair Channel-Dependent Scheduling which schedules packet delivery to mobile stations in a fair manner and at the same time takes into consideration the channel conditions for power efficiency. With added delay information for real-time traffic, this approach aims at delivering real-time traffic in a timely manner, while maintaining a balance between power conservation and fairness. Through comparative simulations with two conventional scheduling algorithms, we show that our scheme indeed achieves better overall performance than comparable scheduling schemes.**

I. INTRODUCTION

Recently, there has been a significant increase in the amount of multimedia services provisioned over wireless networks. Wireless services include instant messaging, video conferencing, web browsing and email, which can be categorized into real-time (voice and video) and non-real-time (http data) traffic. Both types of traffic will be supported in the 3rd generation (3G) wireless systems, where code division multiple access (CDMA) is going to be widely deployed as the air interface [1]. Due to the stringent delay constraints of multimedia applications, certain quality of service (QoS) guarantees must be met. Since scheduling has a direct impact on the system capacity and delay as well as the throughput, it is therefore necessary to investigate scheduling algorithms suitable for multimedia traffic.

The distinguishing characteristic of real-time traffic is that it requires bounded delay while it can tolerate some packet losses. The delay can be bounded by associating a *deadline* with each packet. Once a packet misses its deadline, it will be dropped as it is no longer useful. Therefore the goal for any scheduling scheme for real-time traffic is to deliver packets in a timely manner. In wireless systems, physical factors such as differences in distance, signal propagation (e.g., shadowing), and multipath fading can all lead to varying channel conditions [2]. So a good scheduler should also be able to adapt to these changing channel conditions by mostly serving mobile stations at times when the channel conditions to those mobile stations are good. Such utilization of good channel conditions will result in an overall increase in system capacity. Meanwhile, the scheduler must be fair and not only favor the mobile stations with good channel conditions.

Previous work done in this area either focuses primarily on fairness without exploiting the wireless channel conditions [3][4][5][6], or on channel utilization without considering fairness [7]. A comprehensive survey regarding several popular scheduling algorithms such as earliest deadline first (EDF) and greatest degradation first (GDF) can be found in [3]. In this paper, we propose a novel scheduling technique called Fair Channel-Dependent Scheduling (FCDS), which ensures timely delivery of real-time packets as well as trying to provide fairness, while at the same time exploiting the changing channel conditions. Its purpose is to provide a fair service to all mobile stations, while minimizing the used transmitted power and consequently increasing the system capacity.

The rest of the paper is organized as follows. Section II introduces the proposed Fair Channel-Dependent Scheduling (FCDS) algorithm. The performance of our scheduling scheme is evaluated in Section III. Finally, we conclude this paper in Section IV.

II. FAIR CHANNEL-DEPENDENT SCHEDULING

In this scheme, delay information is associated with each packet for real-time traffic. If packets are considered to be urgent, the base station selects the most urgent packet to transmit immediately without considering the channel conditions. Otherwise, the base station makes scheduling decisions according to channel quality. We infer the channel quality from the transmitted power; high quality channels need less transmitted power than low quality ones to meet the same Signal-to-Interference-Ratio (SIR) requirement. A moving average of the transmitted power used for each mobile station is maintained by the base station. Power fluctuations that may inadvertently affect the moving average are dealt with by maintaining a moving variance for each mobile node. The base station uses these moving averages as well as the moving variances to make its scheduling decisions. Such decisions tend to be much fairer than the ones based solely on the absolute transmitted power because they do not favor nearby mobile stations with high quality channels. In the following, we present a detailed model to show how the scheduler works.

Suppose a base station is serving n ($n \geq 2$) mobile stations on a downlink channel using a hybrid CDMA/TDMA transmission scheme [8][9]. Furthermore, assume that at a certain time slot t, packets are queued and wait for transmission to m ($0 \leq m \leq n$) active mobile stations. For simplicity, we assume only one packet can be sent in each

0-7803-7822-9/03/$17.00 ©2003 IEEE.

time slot. It can be easily generalized to multiple transmissions per slot duration. We use the transmitted power to represent the channel quality because power control is essential in CDMA systems. The base station can get fast feedback information from the receiver side, where the transmitting node dynamically adapts its transmitted power to the current channel conditions so that the received power or SIR at the receiver is constant. As a result, the required transmitted power is a good estimate of channel conditions.

A scheduler can therefore use the transmitted power to decide which mobile station to send packets to next. One way to do this is to compare the transmitted powers for each of the m mobile stations with packets queued for transmission and select the mobile station which has the lowest transmitted power. This approach is referred to as Best-Channel-First (BCF) and is often used as the baseline for comparison. BCF is unfair since it generally favors mobile stations that are physically located nearby the base station since they require less transmitted power. In order to compensate for this, our scheduler keeps track of the moving average of the transmitted power for each mobile node. This moving average reflects the transmitted power used for each mobile station in the recent past. This information, along with the current transmitted power can be used by the base station to make scheduling decisions.

The exponentially weighted moving average $\hat{\mu}_{i,t}$, representing a weighted value of previous transmitted powers along with the current transmitted power, is given as:

$$\hat{\mu}_{i,t} = (1-\alpha_1)\hat{\mu}_{i,t-\Delta t} + \alpha_1 \bar{p}_{i,t} \quad (1)$$

where $\bar{p}_{i,t}$ is the transmitted power used to transmit to node i at time t. Δt is the interval with which the average is updated, we assume it is one time slot duration here and α_1 ($0 < \alpha_1 < 1$) is the parameter determining the weight of the current power compared to the previous power.

The scheduler uses $\bar{p}_{i,t}$ and $\hat{\mu}_{i,t}$ to make its scheduling decisions. A possible approach is to schedule a mobile node that requires the least transmitted power relative to its moving average, that is, $\bar{p}_{i,t} - \hat{\mu}_{i,t}$. This approach is fair in the sense that mobile stations far from the base station are treated equally to those nearby. However, it does not compensate for fluctuations in power. It favors mobile nodes with less power fluctuations. To compensate for this, we keep track of the degree of power fluctuations experienced in the past by maintaining a moving variance for each mobile node.

The moving variance $\hat{\sigma}_{i,t}^2$ for the transmitted power is given as:

$$\hat{\sigma}_{i,t}^2 = (1-\alpha_2)\hat{\sigma}_{i,t-\Delta t}^2 + \alpha_2(\bar{p}_{i,t} - \hat{\mu}_{i,t})^2 \quad (2)$$

where α_2 ($0 < \alpha_2 < 1$) is a weighting parameter.

Finally, the scheduler combines the current transmitted power $\bar{p}_{i,t}$, the moving average $\hat{\mu}_{i,t}$, and the moving variance $\hat{\sigma}_{i,t}^2$ to compute the normalized transmitted power $Z_{i,t}$. The mobile station with the smallest normalized transmitted power H_t (as show in Equation 3) will be scheduled to transmit in the time slot t.

$$H_t = \min_{1 \le i \le m} \{Z_{i,t} = (\bar{P}_{i,t} - \hat{\mu}_{i,t})/\hat{\sigma}_{i,t}\} \quad (3)$$

For real-time applications, it is imperative to consider the delay constraints when making scheduling decisions. Thus a *deadline* is assigned to each packet. If a packet misses its deadline before being transmitted, the packet is dropped rather than being delivered after its deadline. The scheduling algorithm must therefore be aware of both the delay requirement and the power constraints. In other words, the objective of a scheduling algorithm for real-time traffic is to deliver as many packets as possible before their deadline while minimizing the used resources such as the transmitted power.

We call a packet's remaining time before its deadline, the lifetime of a packet. In each time slot, we identify the most urgent packet, i.e., the packet with the shortest lifetime LT_{min}. The parameter, denoted by *urgent_threshold* is used to determine the urgency of the most urgent packet. If LT_{min} is below the *urgent_threshold*, the packet is delivered immediately regardless of its current channel conditions; otherwise, the scheduling is based on the normalized transmitted power as we introduced previously. The goal is to keep the packet drop rate below an acceptable level while minimizing power consumption. The *urgent_threshold* is the determining factor between packet drop rate and power consumption/fairness.

The operation of FCDS can be illustrated by the pseudo code in Figure 1.

FCDS Algorithm:

When packets arrive, they are queued and wait for transmission to m active mobile stations.
At each time slot t.
Identify the most urgent packet with LT_{min}
If the LT_{min} < *urgent_threshold*
Deliver the most urgent packet immediately
else
For each active mobile station i (i= 1,..., m):
Update $\hat{\mu}_{i,t}$ *according to (1).*
Update $\hat{\sigma}_{i,t}^2$ *according to (2).*
If there is a packet destined to mobile station i,
Compute the normalized transmitted power
$Z_{i,t} = (\bar{p}_{i,t} - \hat{\mu}_{i,t})/\hat{\sigma}_{i,t}$
The Base station scheduler selects the mobile station with the smallest $Z_{i,t}$ *and delivers the packet*

Figure 1. Pseudo code for FCDS scheduling algorithm

III. PERFORMANCE EVALUATION

As mentioned in Section II, the *urgent_threshold* divides the scheduling policy into two regions, one is referred to as the urgent region, where the packet is delivered based on its delay constraint regardless of the channel condition; the other region is the non-urgent region, where the goal is to conserve power and to insure fairness. It is not possible to achieve all the goals simultaneously since enhancement to one implies degradation to the other. As a result, this approach reaches a compromise between them. Here the performance of FCDS for real-time traffic is evaluated from three perspectives ---

timely packet delivery, power conservation and fairness. We prove through comparative simulations that our approach can deliver packets in a timely manner, and in addition be power efficient and fair.

Two common approaches are adopted for performance comparison. One is first-come-first-served (FCFS). The packet that arrives the earliest is chosen to be transmitted, with likely a higher transmitted power since it does not consider channel quality at all. The obvious advantage of FCFS is that it is fair if the packet arrivals to each mobile station are evenly distributed. The other scheme is BCF, it is a channel-dependent scheme. The base station selects the mobile station with the best channel condition in each time slot. Without considering delay information, BCF is expected to experience high packet drop rate.

A discrete event-driven simulator is used to study the characteristics of FCDS. The system architecture is as illustrated in Figure 2. The base station maintains one queue for each active user. When packets arrive, they will be put into one of the output queues based on their destinations. The *deadline* of each packet is assigned when the packet arrives at the base station. The *deadline* can be of a fixed value depending on the type of traffic or be variable depending on a delay measurement or delay estimation. In the simulation, only fast fading is considered. The weighting parameters α_1 and α_2 are set equal to 0.1 in our simulations.

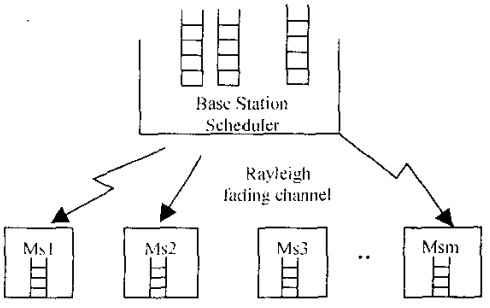

Figure 2. System architecture for CDMA downlink scheduling

Three metrics are used to evaluate the performance of our scheme. The main metric we measured is the packets' dropping rate due to missing their deadlines, denoted by late-packet-ratio. It reflects how timely the packets are delivered. Another metric we measure is the average transmitted power. In our performance evaluation, we regard the average transmitted power consumed by FCFS as unit one, and normalize the power of BCF and FCFS for comparison. Fairness is also a metric we are interested in. Here the late-packet-ratio for mobile station i is denoted as D_i. Then the coefficient of variation (*cov*) of the *late-packet-ratio* is used to reflect the fairness of the scheduling scheme.

$$\text{cov} = \frac{std}{mean} = \frac{\sqrt{\sum_{i=1}^{m}(D_i - \overline{D})^2}}{\sqrt{m}\overline{D}} \qquad (4)$$

where $\overline{D}$ is the mean of D_i for $i=1$ to m.

If the scheduling scheme is relatively fair, then each mobile station has a similar chance to transmit and equivalently the probability to be dropped is also similar when the packet misses its deadline, so the *cov* of the *late-packet-ratio* should be small. The smaller the *cov* is, the fairer the scheduling is.

Two scenarios are studied in the following section. The first is equal delay constraints are set for all the packets. The second is packets have different delay requirements. Performance is evaluated using the three metrics discussed above.

Equal deadline case. An equal deadline is assigned for each packet upon arrival. In this case, FCFS is the same as EDF. Performance regarding the *late-packet-ratio*, transmitted power and fairness are illustrated in Figures 3, 4 and 5 respectively.

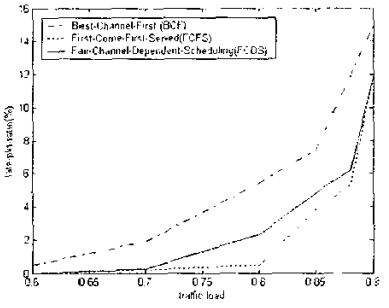

Figure 3. Late-packet-ratio for BCF, FCFS and FCDS

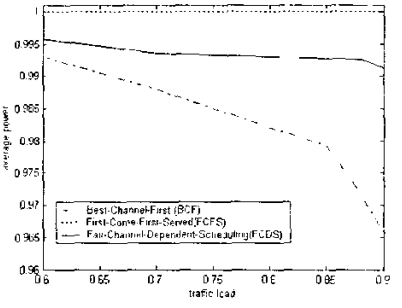

Figure 4. Normalized transmitted power for BCF, FCFS and FCDS

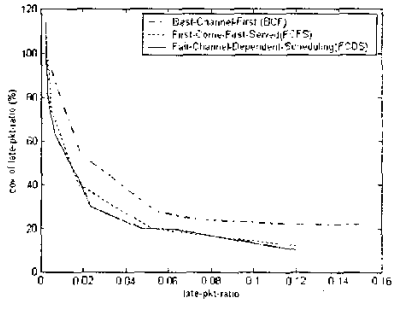

Figure 5. *cov* of late-packet-ratio for BCF, FCFS and FCDS

It can be seen that FCFS exhibits the smallest *late-packet-ratio* but consumes the most power to transmit packets as expected, since FCFS only cares the delay information without taking advantage of the channel conditions. On the other hand, BCF consumes the least power, but it has the largest *late-packet-ratio* since BCF is independent of delay information. FCDS however falls right in the middle and displays similar fairness to FCFS.

Different deadline case. In packet-switched networks, packets from different flows may experience dramatically different amounts of delay, for example, the queueing delay is a variable and the propagation delay is a function of distance between the transmitter and receiver. Therefore, instead of using the same deadline, different delay requirements are assigned to packets in this case.

By utilizing the diversity of deadline information, significant improvement can be achieved for FCDS as illustrated in Figure 6-8. From Figure 6, we observe that FCDS outperforms FCFS in terms of late-packet-ratio. It is due to the fact that FCFS can only make use of the packets' arrival time information. If the deadline is constant for all packets, then the lifetime for packets, which is deadline minus arrival time, is only related to arrival time. However, once the deadline is different for packets, the lifetime of packets is related to both deadline and arrival time. FCFS cannot make good decisions now, while FCDS can perform much better than FCFS. We also observe that FCDS can deliver real-time packets by the deadline with less resource consumption (see Figure 7) than that in equal deadline case as shown in Figure 4.

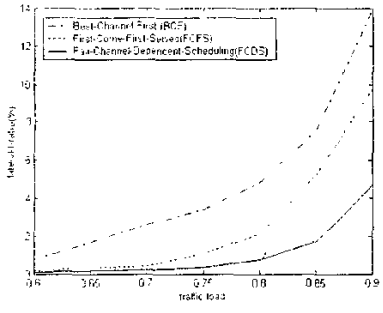

Figure 6. Late-packet-ratio for BCF, FCFS and FCDS

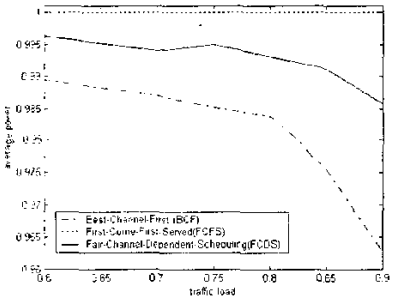

Figure 7. Normalized transmitted power for BCF, FCFS and FCDS

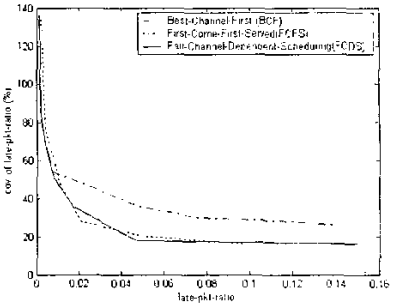

Figure 8. *cov* of late-packet-ratio for BCF, FCFS and FCDS

Sensitivity Analysis of urgent_threshold. As previously discussed, the *urgent_threshold* is the parameter determining the urgency of a packet such that certain packets are allowed to use more transmitted power when their deadline approaches. For small values of urgent-threshold, packets are more likely to be scheduled based on good channel conditions, since they can survive a relatively long time before they are considered to be urgent. In this case, less transmitted power is used.

On the other hand, for large *urgent_threshold* values, packets are quickly categorized as urgent packets and will therefore need higher levels of power to be transmitted to overcome bad channel conditions. Based on the above observations, we can see that the *urgent_threshould* determines the tradeoff between late-packet-ratio and required transmitted power. If there is not any specific requirement for each of the two metrics, we can locate an optimal range for *urgent_threshold* where the late-packet-ratio and transmitted power are both reasonably small. In the example demonstrated in Figure 9, the optimal range for *urgent_threshould* is about from 2 to 4 slots.

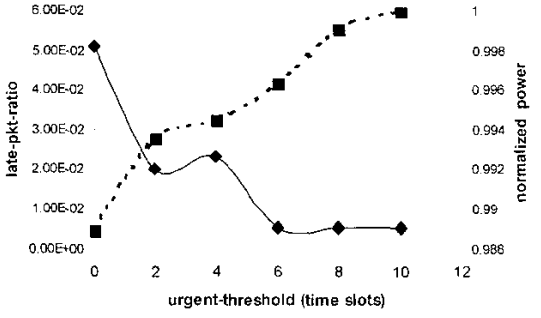

Figure 9. Sensitivity analysis of urgent-threshold

IV. CONCLUSION

We have presented a Fair Channel-Dependent scheduling algorithm for real-time traffic on the CDMA downlink in this paper. This algorithm not only achieves timely delivery of packets to minimize the packet drop rate, but also utilizes the changing channel conditions to conserve the transmitted power and provide fairness to each mobile station. We compared the performance of our approach to two

conventional scheduling schemes. Our algorithm consumes much less power than FCFS; on the other hand, it displays lower packet drop rate and exhibits more fairness than BCF. In summary it balances timely packet delivery and power efficiency/fairness.

REFERENCES

[1] "Global CDMA Business Opportunities," Telecom reports by Datacomm Research Company, published August 1, 2000.

[2] P. M. Shankar, Introduction to Wireless Systems, John Wiley & Sons, Inc. publishers, Chapter 2.

[3] M. Adamou, S. Khanna, I. Lee, I. Shin, S. Zhou, " Fair Real-time Traffic Scheduling over a Wireless LAN," Proc. Of IEEE Real-Time Systems Symposium, Dec. 2001, pp 279-287

[4] S. Lu, V. Bharghavan and R. Srikant, "Fair Scheduling in Wireless Packet networks," IEEE/ACM Transactions on Networking, Vol. 7, Issue 4, August 1999, pp. 473-489.

[5] S. K. Baruah, N. K. Cohen, C. G. Plaxton, and D. A. Varvel, "Propotionate Progress: A Notion of Fairness in Resource Allocation, Algorithmica," Vol. 15, No. 6, pp. 600-625, June 1996.

[6] R. West, K. Schwan, and C. Poellabauer, "Scalable Scheduling Support for Loss and Delay Constrained Media Streams," IEEE Real-Time Technology and Application Symposium, 1998.

[7] K. Lee and M. El Zarki, "Scheduling Real-Time Traffic in IP-Based Cellular Networks," Proc. PIMRC'2000, pp. 1202-1206, Sept. 2000.

[8] E. Brand and A. Hamid Aghvami, "Multidimensional PRMA with Prioritized Bayesian Broadcast – A MAC Strategy for Multiservice Traffic over UMTS," IEEE Transactions on Vehicular Technology, Vol. 47, November 1998, pp. 1148-61.

[9] T. Ojanperä, "FRAMES – Hybrid Multiple Access Technology," Proceedings of ISSSTA, Mainz, Germany, September 1996, pp. 320-4.