A Combined Polling and ISMA-DS/CDMA Protocol to Provide QoS in Packet Mobile Communications Systems

Jordi Pérez-Romero, Ramón Agustí, Oriol Sallent
Department of Signal Theory and Communications, Polytechnical University of Catalonia,
c/ Jordi Girona, 1-3, Campus Nord, Edifici D4. 08034, Barcelona, Spain,
email: [jorperez,ramon,oriol]@tsc.upc.es

ABSTRACT

This paper presents a new mechanism that combines the flexibility of an access protocol such as ISMA-DS/CDMA with the ability of a polling mechanism to provide a specific bound for the access delay. This protocol is proposed for a packet transmission mobile communication system together with a scheduling algorithm that arranges the different transmissions depending on the quality of service required by the set of considered services.

I. INTRODUCTION

One of the most important goals of third generation mobile communication systems consists on the ability to provide different kinds of multimedia services to mobile users, while at the same time meeting some specific Quality of Service (QoS) requirements. These services usually require the ability to handle traffic sources of a very bursty nature that combine high activity periods with other periods in which no transmissions are needed. In this context, packet transmission techniques gain added interest in front of other circuit switching techniques thanks to the fact that they allow a more efficient use of the involved radio resources.

On the other hand, the future mobile communication systems will be mainly based on a DS/CDMA multiple access scheme, as it is stated in the different proposals such as UTRA (UMTS Terrestrial Radio Access) FDD and TDD or cdma2000. Since in DS/CDMA all the users share the same bandwidth during the same time, this scheme allows an inherent statistical multiplexing of the different transmissions, and consequently it appears to have a high flexibility when requiring the provision of multimedia services.

In a packet transmission scenario, different aspects should be taken into account when considering how resource management must be performed in order to ensure a specific quality of service to the different users. Particularly, the first point relays on how users can gain access into the considered resources, which is the responsibility for the *medium access protocol*. In this context, the random nature of the access associated with uplink transmissions entails an inherent difficulty when trying to ensure the QoS requirements so that other mechanisms that limit the randomness need to be considered.

On the other hand, once users have successfully acquired a resource that allows them starting their transmissions, a certain control needs to be applied that guarantees the specified QoS requirements. This is the responsibility for the *scheduling algorithm* that should be able to put transmissions in the most convenient order so that the QoS requirements can be met. Several scheduling algorithms have been proposed up to date in the literature, but most of them are oriented to a time division multiple access scheme in which only a single user transmits at any time. Some examples are GPS (General Processor Sharing), FFQ (Fluid Fair Queueing), WFQ (Weighted Fair Queueing), or Delay - EDD (Earliest Due Date), and others that try to consider the special problems of a radio channel, such as CIF-Q (Channel Condition Independent Fair Queueing) or SBFA (Server Based Fairness Approach) [1]-[4]. However, in a DS/CDMA multiple access scheme, it should be noted that the scheduling algorithm should operate in a slightly different manner, as more than a single user can transmit at the same time. Furthermore, the main limitation of such a system relays on the interference level, that can be regulated by allowing more or less transmissions. Consequently, a scheduling algorithm for a DS/CDMA scenario requires to manage the maximum number of simultaneous transmissions that can be allowed so that the maximum interference requirements are met. Some algorithms that operate on this basis are presented in [5][6].

In this paper, we present a new mechanism that ensures a given QoS when considering mixed services in the uplink of a packet transmission system. We work from the basis of an ISMA (Inhibit Sense Multiple Access) - DS/CDMA medium access protocol [7]-[9] and we combine it with a polling mechanism that allows users returning into the system in a bounded amount of time. Additionally, we present a scheduling algorithm that ensures the required QoS to the different transmissions. The rest of the paper is organized as follows. Section II presents the combined ISMA-DS/CDMA protocol with polling mechanism and Section III explains the proposed scheduling algorithm. Results are presented in Section IV and, finally, conclusions are summarized in Section V.

II. ISMA-DS/CDMA PROTOCOL WITH POLLING

The ISMA-DS/CDMA protocol is based on the following behavior [8][9]:

1.- A base station has a number of spreading codes to be allocated to those users that work on a packet

This work was supported by CICYT project TIC 98-684.

0-7803-6465-5/00 $10.00 © 2000 IEEE

transmission basis. On a frame by frame basis, the base station broadcasts the status of the different codes just by making use of a single bit (1: busy, 0: free) per code.

2.- Users receive messages that need to be transmitted with certain QoS requirements. Specifically, we consider a delay limitation, which means that a given message needs to be transmitted in a maximum number of frames. Messages are divided into packets whose length depends on the number of bits that can be transmitted in each frame. This number of bits finally depends on the considered spreading factor, so that reduced spreading factors allow the transmission of a higher number of bits at the expense of generating a high interference, while high spreading factors allow the transmission of less bits, although a reduced interference is generated.

3.- In order to contend for resources, users apply a certain access probability that depends on the number of busy codes. The value of this probability is 1 whenever all the codes are free and decreases linearly when the number of busy codes increases up to a maximum *Kmax*. Once there are *Kmax* busy codes, the probability is set to 0 and no other users can enter the system until any code is released. This limitation allows a regulation in the overall interference level.

4.- If the previous random experiment allows the access, users randomly select an available code (i.e., a code whose status is 0, free) and transmit a preamble on it at the beginning of a frame. If the transmission is successful, which means that no other user has selected the same code and that multiuser interference has not corrupted the packet, the user keeps the code until the end of its message transmission. Then, the code is broadcast as busy until it is finally released.

5.- In the case the transmission is unsuccessful users will retry the access in the subsequent frames by repeating the procedure from step 3.

6.- Whenever a code has been acquired, the existence of a corresponding code in the downlink is assumed. This code allows to acknowledge the transmitted packets in the uplink and also to perform a closed loop power control. Several power control periods are assumed in each frame. Those packets that are not successfully transmitted are unacknowledged and retransmitted in subsequent frames. One frame delay in the acknowledgement process is assumed.

It should be noted that this mechanism does not guarantee a bound in the access delay, due to the random nature of the access. When considering bursty sources that alternate high activity periods with periods with no information to be transmitted, as it would be the case of a TCP data transfer, this limitation poses a problem, as the delay QoS requirements need to be guaranteed during all the transfer, which includes to have a bound for the access delay between activity periods.

The proposed solution to cope with this problem consists on combining the ISMA-DS/CDMA protocol with a polling strategy. The main point consists on reserving a spreading code with a certain periodicity for those users that have previously released a code. In this way, the access delay for those users will be bounded by the polling period. The behavior of the polling mechanism is explained as follows:

1.- After having acquired a code, users transmit their messages. Once their transmitting buffer is empty, they keep the code for a number of N_f frames. This allows possible retransmissions of the last transmitted packets to be performed without requiring a new code acquisition and also it allows to manage the case of a new message arriving just a little time after having transmitted the previous one. Then, after N_f frames with an empty buffer, the code is released.

2.- For those users that have released a code, a periodical reservation of a new spreading code is performed every P_p frames, and it is signaled through high layer messages in the downlink. If the polled user has new information in its buffer, it transmits in the reserved resource. It should be noted that users do not need to wait for the polling in order to access the system. Instead, they can try the access even if they have no reservation according to the rules of ISMA-DS/CDMA protocol. The polling should be regarded only as a last chance that allows a bound in the access delay.

3.- After a user has received N_p consecutive pollings and none of them has been answered, no more reservations will be performed to this user.

A flow diagram explaining the overall behavior is presented in Figure 1.

III. SCHEDULING ALGORITHM

The previously described protocol is able to provide a bound in the access delay. However, some additional mechanisms are needed in order to guarantee the delay bound when the packet transmissions after the initial access are considered. With the aim of solving this problem, we propose the following scheduling algorithm, in which the mobile terminals send requests to the base station which is responsible for ordering them and accepting those ones which are more suitable to meet the specific delay requirements.

We assume that once a terminal has gained access into the system, it sends its transmission request for the next frame. This information is sent either at the initial access preamble or together with each allowed packet transmission. Particularly, this transmission request contains the following information:

- the spreading factor (or equivalently, the bit rate) that the terminal would like to use depending on the number of bits that should be transmitted. We assume a *packet* to be the number of bits transmitted in a frame at the highest spreading factor. Consequently, the bit rate can be specified as the number of *packets* that the user would like to transmit.

- the timeout or maximum delay that the *packets* involved in the request can allow, which depends on the specified QoS for the considered service. It should be noted that not all the *packets* in the request may have the same timeout, so that the most restrictive timeout is to be specified. Additionally, the number of *packets* affected by this timeout in the request is also given so that the bit rate can be changed by the scheduler if needed.

The selection of the appropriate spreading factor is done depending on an adaptive transmission bit rate algorithm, such as the ones proposed in [9][10]. Particularly, we assume that the initial bit rate is selected depending on the number of busy codes and some specific thresholds (*th1* < *th2* < ...< *thN*) so that the bit rate is the maximum one when the number of busy codes ranges from 0 to *th1*, half the maximum one when the number of busy codes ranges from *th1* to *th2*, and so on, thus decreasing the bit rate as the number of busy codes increases. Furthermore, as long as packet transmissions are performed, the preferred bit rate is increased or decreased depending on the number of consecutive successful or erroneous transmissions, respectively. Thus, after *min_suc* consecutive successful transmissions the bit rate is increased to twice its previous value, while after *max_tr* consecutive erroneous transmissions the bit rate is reduced to half its previous value.

On the other hand, the timeout of the packets in the buffer is also taken into account, so that some violations to the previous algorithm are allowed in order to be able to transmit those packets whose timeout is about to expire.

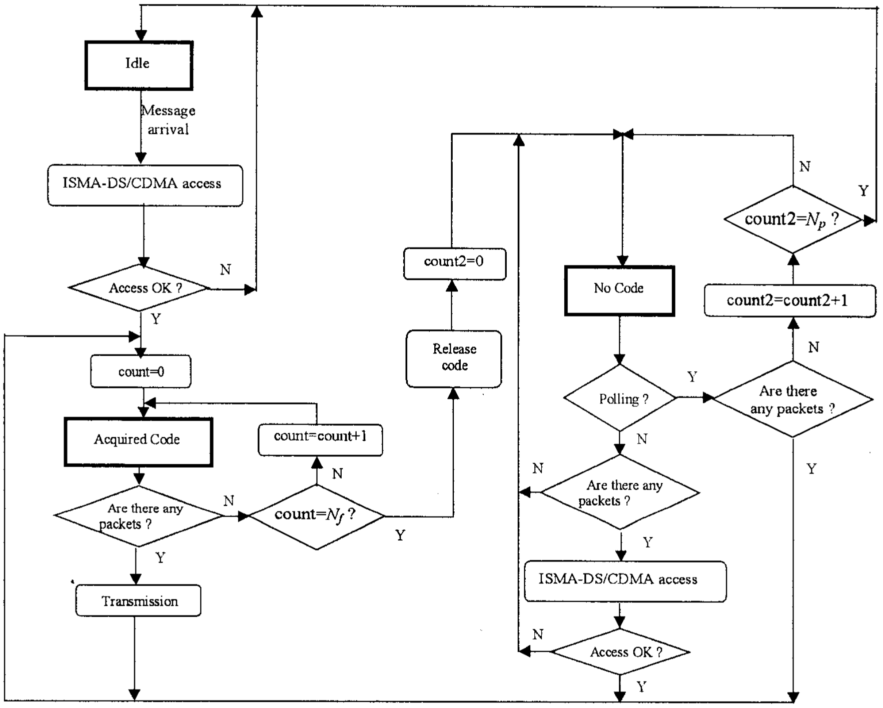

Figure 1 Flow diagram that describes the combined ISMA-DS/CDMA and polling protocol

Requests are received in the base station and the scheduling algorithm is applied in order to decide which ones can transmit. The scheduler acts in two steps:

1.- Prioritization: All the requests and also the pollings that are to be performed in the next frame are arranged in a table in increasing timeout order. For those requests with equal timeout, the ones with a higher number of packets are put first. An example of such a prioritization is shown in Table I. Those requests with timeout = 1 are called *critical requests*, as if they are not served in the considered frame they will be discarded.

2.- Allocation: In order to decide how many requests are accepted, an E_b/N_o criterion is considered. Particularly, each request has a minimum E_b/N_o that depends on its bit rate and the QoS for this specific type of service. The minimum E_b/N_o is only specified for the *critical requests*, as retransmissions can be allowed for those packets with timeout > 1. After all the requests have been ordered according to the prioritization process, the allocation process consists on accepting as many requests as possible so that the minimum E_b/N_o is satisfied for all the accepted requests. The process is described as follows:

Assuming that all the requests in the table from 1 to i-1 have been accepted, in order to see if the request in position i can also be accepted, the algorithm does the following:

Step 1.- For each request in positions 1 through i, it calculates the corresponding E_b/N_o if the considered request i was also accepted.

Step 2.- The request in position i is accepted only if the minimum E_b/N_o is still satisfied for all the requests from 1 through i. In this case, the algorithm proceeds with the request in position i+1 from step 1. On the contrary, if not all the minimum E_b/N_o are satisfied, the process stops and only the requests from 1 to i-1 are accepted.

If there still remain *critical requests* that have not been accepted, the algorithm is applied again but reducing the bit rates of the different requests depending on the number of packets affected by the timeout, so that interference is reduced.

All those requests that are not accepted are put off for the next frame, and their timeout is decreased by 1. The result of the scheduling is transmitted in the corresponding downlink of each user.

IV. SIMULATION RESULTS

In order to evaluate the behavior of the proposed mechanism, different simulations have been performed. The following simulation conditions have been assumed:

- Frame time = 10 ms
- 5 possible bit rates, corresponding to transmitting $160*2^k$ bits per frame, k=0..4. The respective spreading factors are assumed to be $256/2^k$.

The following generic services are assumed:

- Users 1: Voice users. An ON/OFF model is considered. During an ON period, a user generates an 80 bit packet in each frame, which is encoded by a half rate code thus obtaining a 160 bit packet. Consequently, this users only apply the maximum spreading factor, which is 256. We consider that each packet needs to be transmitted in a single frame.' If not, it is discarded. This means that all voice packets correspond to what has been named as *critical requests*. We assume that the maximum number of packets that can be dropped, either because they are not transmitted or because they are not successfully received, is 1%. As the access delay of these users needs to be minimum, we assume that they keep the code even in the OFF periods.

- Users 2: Long Delay Data users. An ON/OFF model is assumed in which message length has an exponential distribution with a mean 800 bits and interarrival time 30 ms (3 frames). We assume a maximum message length of 3200 bits. For these users we assume a maximum delay of 300 ms = 30 frames per message. As the delay allows for a number of retransmissions, a type II hybrid ARQ retransmission strategy is considered. In this case, the information is initially sent without correcting capability and, if a retransmission is required, the corresponding redundancy obtained after applying a half rate code is transmitted. This mechanism allows to send the redundancy only when it is necessary and thus the number of useful data bits can be higher for low interference situations [12].

- Users 3: Low Delay Data users. An ON/OFF model is also assumed. Message length is exponentially distributed with a mean 400 bits and interarrival time 30 ms (3 frames). A maximum message length of 1200 bits is assumed. For these users, the maximum delay is assumed to be 50 ms = 5 frames. Consequently, very few retransmissions can be allowed. Thus, a type I hybrid ARQ retransmission strategy is assumed in which the information is sent encoded by means of half rate channel codes.

Table I Prioritization of requests

Position	Requests	Timeout	E_b/N_o min
1	Critical requests	1	$(E_b/N_o \text{ min})_1$
...		...	...
...	Pollings	...	...
j		1	$(E_b/N_o \text{ min})_j$
...	Rest of requests	...	...
M		N	$(E_b/N_o \text{ min})_M$

By taking into account these considerations in a single cell scenario together with the gaussian approximation for calculating the bit error rate, and the Varshamov - Gilbert bound [11] for evaluating the correcting capability of the half rate channel codes the minimum E_b/N_o for the different bit rates is calculated. For voice users, by considering a maximum BLER (block error rate) of 10^{-2}, these calculations lead to an E_b/N_o min = 1.87. For the data users, calculations are performed in order to have a block error rate lower than 10^{-6}. The results are shown in Table II for both the type I and II hybrid ARQ cases. Note that this minimum E_b/N_o poses only a limitation when considering *critical requests*.

These considerations lead to define the thresholds for the bit rate determination as *Th1*=15, *Th2*=28, *Th3*=50, *Th4*=90, and *Kmax*=133. The maximum number of consecutive successful and erroneous transmissions are *min_suc*=5 and *max_tr*=1, respectively. We also consider N_f=5, P_p=10 and N_p=20 for users 2 and N_f=5, P_p=4 and N_p=20 for users 3. We also assume that there are a total of 130 spreading codes for each group of users.

In Figure 2 and Figure 3 we present the percentage of packets out of time for the different users as a function of the overall system throughput (number of correctly transmitted useful bits) in the cases where no scheduling is applied and when the proposed algorithm is used, respectively. The simulations have been performed by assuming a total of 20 voice users, 20 low delay users and a variable number of long delay users. For the case when no scheduling is considered, users only apply the ISMA-DS/CDMA protocol with the described bit rate adaptation mechanism. It can be observed how the low delay users are those who take more benefit from the scheduling algorithm. Particularly note that the maximum overall throughput that these users can allow without any scheduling mechanism is around 6000 bits / frame (see Figure 2) while in the case that the algorithm is considered, this maximum throughput raises up to approximately 11000 bits /frame (see Figure 3). For these users, the ISMA-DS/CDMA protocol is not enough to guarantee the delay bound due to the low delay that they can tolerate. Consequently, some other mechanism such as the scheduling algorithm is required.

Table II Minimum Eb/No depending on the bit rate

Spr. Factor	Eb/No min (ARQ-I)	Eb/No min (ARQ-II)
16	1.59	1.59
32	1.74	1.60
64	1.97	1.75
128	2.32	1.97
256	2.88	2.32

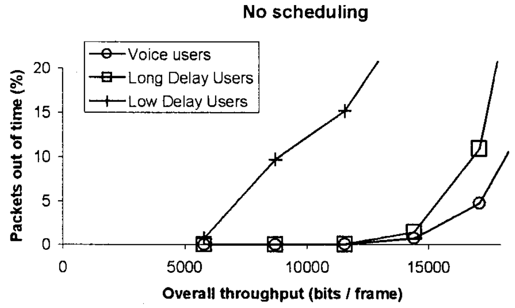

Figure 2 Packets out of time when no scheduling is applied

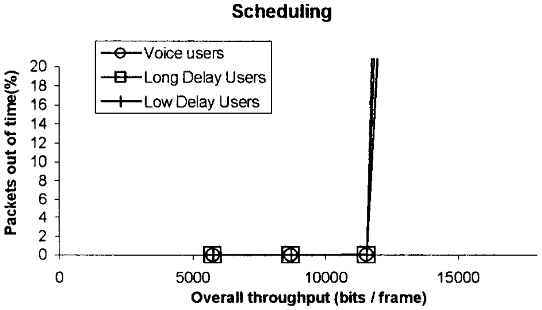

Figure 3 Packets out of time when the scheduling is applied

When the long delay users are considered, it should be stated that they perform quite well by applying only the ISMA-DS/CDMA protocol, thanks to their low stringent requirements in terms of delay. For this reason, they do not take a high benefit from the algorithm. On the other hand, when considering voice users, it should be pointed out that when no scheduling is applied, these users can always transmit without any restriction. However, when applying the algorithm, it can be possible for high loads that transmission is not allowed to these users in order to give priority to transmissions of other users that generate a high interference and are about to expire. Then, the scheduling algorithm tends to impose restrictions to these users compared to the case without scheduling.

As it has been shown, the overall behavior of the scheduling algorithm depends on the specific types of user and their QoS requirements. However, when we regard the system as a whole the more restrictive users are those that impose the throughput limitation. Taking this into account, the maximum allowable throughput in the system without scheduling is about 6000 bits / frame, while when making use of the scheduling this maximum raises up to 11000 bits / frame, which gives an idea of the obtained benefits. Note also in Figure 3 that when applying the algorithm all three kinds of users have a very similar performance (in fact, their three responses are overlapped), which gives an idea of the fairness of the algorithm.

V. CONCLUSIONS

In this paper a medium access protocol that combines a random access protocol such as ISMA-DS/CDMA and a polling mechanism has been presented together with a scheduling algorithm that arranges requests depending on the maximum allowed delay and interference. The polling mechanism is intended to bound the access delay, while the scheduling algorithm is responsible for guaranteeing the maximum delay. A good performance has been observed for the algorithm, particularly when considering users with stringent delay requirements.

REFERENCES

[1] H.Zhang, "Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks", Proc. of the IEEE, V.83, No.10, Oct.1995, pp.1374-1396

[2] V. Bharghavan, S.Lu, T.Nandagopal, "Fair Queuing in Wireless Networks: Issues and Approaches", IEEE Personal Communications, February 1999, pp. 44-53.

[3] T.S.Eugene Ng, I. Stoica, H. Zhang, "Packet Fair Queueing algorithms for Wireless Networks with Location-Dependent Errors", Infocom'98.

[4] P. Ramanathan, P. Agrawal, "Adapting Packet Fair Queueing Algorithms to Wireless Networks", Mobicom'98, Dallas, Texas, USA,1998

[5] I. F. Akyldiz, D. A. Levine, "A Slotted CDMA Protocol with BER Scheduling for Wireless Multimedia Networks", IEEE/ACM Trans. On Networking, Vol. 7, No2, April 1999, pp. 146-158.

[6] S. Ramakrishna, J. M. Holtzman, "A scheme for Throughput Maximization in a Dual-Class CDMA System", IEEE Journal on Selected Areas in Commun., Vol.16, No 6, August, 1998, pp. 830-844.

[7] R. Prasad, "Performance analysis of mobile packet radio networks in real channels with inhibit-sense multiple access", IEE Proceedings-I, Vol. 138, No. 5, October 1991, pp. 458-464.

[8] J. Pérez-Romero, R. Agustí, O. Sallent, "Performance of an ISMA CDMA packet data network", Proceedings VTC in Fall, Amsterdam, September, 1999.

[9] J. Pérez, R. Agustí, O. Sallent, "An adaptive ISMA-DS/CDMA MAC protocol for Third Generation Mobile Communication Systems", submitted to IEEE Trans. on Vehicular Technology, February 2000.

[10] O. Sallent, R. Agustí, "A Proposal for an Adaptive S-ALOHA Access System for a Mobile CDMA Environment", IEEE Trans. On Vehicular Technology, Vol. 47, No. 3, August 1998, pp. 977-986.

[11] S. Lin, D.J. Costello, *Error Control Coding: fundamentals and applications*, Prentice-Hall, 1983.

[12] J. Pérez, R. Agustí, O. Sallent, "Type-II Hybrid ARQ Scheme in a DS-CDMA Packet Transmission Network", PIMRC'99, Osaka, Japan, September 1999