Scheduling-based QoS Balancing for Multi-service Cellular Networks

Francisco R. P. Cavalcanti Yuri C. B. Silva Tarcisio F. Maciel Elvis M. G. Stancanelli

***Abstract*— This paper proposes and evaluates methods for improving the capacity of queuing-limited cellular networks through the use of scheduling algorithms. Several scheduling disciplines are investigated, considering different combinations of multi-slot data services, such as streaming, WWW, and FTP.**

***Keywords*— Scheduling, QoS, balancing, streaming, GSM/EDGE**

I. Introduction

The support of multiple services will be one of the main features of the third generation of cellular systems and beyond, which will have web-browsing, e-mail, and audio/video streaming available, besides the traditional circuit-switched voice service. These services have different characteristics and requirements and the use of efficient radio resource management (RRM) techniques becomes essential to ensure their QoS levels and to optimize system capacity.

When multiple services share the same radio resources, they may also cause mutual interference. Also due to the different traffic patterns and QoS requirements, differences are often found with regard to the maximum interference and delay levels supported by each service, being the system capacity limited by the service which first violates its QoS demands. For this reason, QoS balancing schemes could be employed to improve capacity.

For interference-limited scenarios, the Service-Based Power Setting (SBPS) technique has already been shown as a rather effective interference balancing method for mixed-service GSM/EDGE networks [1]. However, when queuing has a significant impact over the QoS of packet-switched services, scheduling disciplines may also be employed in order to control the average queuing time verified by each service class, thus enhancing system capacity [2].

The purpose of this work is, therefore, to propose and evaluate scheduling algorithms for optimizing the performance of mixed-service cellular networks, considering mixes of data services and terminals with multi-slot capabilities.

This work is organized as follows: section II presents the studied RRM techniques; section III describes simulation aspects and models; section IV presents the simulation results; and section V presents some final remarks.

II. QoS and interference balancing

The performance of data services is closely related with the delay profile in the network. The delay may be a result of both interference as well as queuing effects. The former affects channel reliability, thus eventually incurring in retransmissions, while the latter is a consequence of channel unavailability. In order to optimize system capacity, both the interference and the queuing delays must be efficiently managed by RRM. Moreover, terminals with distinct multi-slot capabilities will be differently affected by these factors, since they occupy more than one channel.

In mixed-service scenarios, the QoS of all services must be simultaneously fulfilled. If capacity is interference-limited, the service which is most sensible against interference will reach its QoS limit first, while the other services will be verifying a surplus QoS. In this case, the interference caused by the services can be managed so as to reduce the interference verified by the most sensible one, thus increasing overall system capacity. If the interference level supported by all services in the mix becomes equal, interference balancing is achieved and the overall system capacity is maximized [1].

In case the system is not interference-limited, i.e., if the queuing delay has a more relevant impact over the system performance, a suitable scheduling algorithm must be used. If all services have the same priority, the most demanding service will reach its QoS limit first and thus limit system capacity, while the other services will probably be getting more than the necessary access time in the channel. In this case, an appropriate scheduling discipline can be used in order to transfer this excess from the less to the most demanding service(s).

A. Service-Based Power Setting (SBPS)

The SBPS technique consists of applying a simple per-service transmission power offset in order to control the interference caused by one service over the others. The reduction power offset may be estimated based on the interference profiles of the individual services, such as it is described in [1].

B. Scheduling algorithms

The choice of a scheduling algorithm to perform the QoS balancing task needs to take into account the following aspects: the algorithm has to be capable of efficiently diferentiating between the traffic flows in order to provide the contracted QoS levels, it could be dynamic in order to adapt to the different service mixes, and it should avoid excessive computational complexity.

In [2], a strict scheduling priority algorithm is employed to perform the QoS balancing between WWW users with different QoS requirements. It has been shown that the

Francisco R. P. Cavalcanti, Yuri C. B. Silva, Tarcisio F. Maciel and Elvis M. G. Stancanelli, Wireless Telecommunications Research Group - GTEL, Federal University of Ceará, Fortaleza, Brazil, {rodrigo, yuri, maciel, emiguel}@gtel.ufc.br

algorithm is capable of providing capacity gains, and that it would be more efficient in channel-limited situations. An advantage of the algorithm is that it is rather simple to implement. However, depending on the evaluated scenario (e.g., services with different traffic patterns and QoS metrics), the determination of the correct priority adjustment may be a rather difficult task.

Another option to solve the problem would be to employ scheduling algorithms that try to track the current QoS conditions of each service in order to dynamically adjust the traffic handling priorities. Herein, a sort of fair throughput algorithm is proposed for performing the QoS balancing. The algorithm is referred to as TFQ (Target Fair Queuing), since it tries to differentiate among the services by adjusting different QoS targets.

The proposed algorithm tries to maximize the ratio between the throughput target and the mean perceived throughput, i.e., in each time slot, the user for which (R_{target}/R_{avg}) is maximal is served. Note that this algorithm is employed both for differentiating the services as well as for users of the same service.

Each service is attributed a specific throughput target, which can be adjusted based on its QoS requirements, e.g., WWW with a 10kbps target and streaming with 32kbps. It's worth to mention that even though the streaming QoS is determined based on the buffering delays, it is reasonable to assume that if the system is able to match the bitrate according to which the service is offered, then no rebufferings would occur and the user would be satisfied.

This algorithm is rather similar to the original PFQ and FT schemes (see table I), the difference being the use of the throughput targets instead of the instantaneous throughput (PFQ) or the unit constant (FT) [3].

TABLE I

FAIR SCHEDULING ALGORITHMS.

Algorithm	Criterion
Fair Throughput (FT)	$1/R_{avg}$
Proportional Fair Queuing (PFQ)	R_{inst}/R_{avg}
Target Fair Queuing (TFQ)	R_{target}/R_{avg}

III. SIMULATION MODELS

The simulations were performed in a dynamic discrete-time system-level simulation tool developed for the downlink evaluation of GSM/EDGE cellular systems. Some of the features of the simulator include the modeling of the radio link, propagation effects (path-loss and spatially correlated fading), mobility, frequency hopping, link adaptation, a detailed implementation of the RLC/MAC protocol, and a link-to-system level interface for mapping the measured SIR into block error rate (BLER) values. More details about the simulator may be found in [4].

Since this paper focuses on the simulation of packet data services, three service classes were selected for evaluation, which are namely: interactive (web-browsing), streaming (video streaming), and background (FTP). These traffic sources, each with their own peculiar characteristics, are evaluated in the upcoming sections both individually as well as within different service mixes.

The considered web-browsing service consists of a typical WWW session, with intermittent traffic and relatively large inactivity periods (reading-time). It is based on a measurement-based model described in [5]. The session consists of a number of page requests, each of which contains one or more objects. The adopted model assumes that all objects of a page are transmitted within a single IP packet. Each session is composed of a geometrically distributed number of packets with an average of 10 packets. When the download of a web page is successfully completed, the user spends some time reading it before making a new request. The time interval comprehended between the complete page retrieval by the user and the reception of the new page request by the network is modeled according to a Pareto distribution with average value of 10s and parameters $\alpha = 1.4$, $k = 3.45$ and cut-off value of 120s. The packet size is log-normally distributed with mean value of 4.1kbytes and standard deviation of 30kbytes. Moreover, 50 bytes are added to the packet size to account for protocol overhead (e.g., TCP/IP header) and packet sizes are limited to a maximum of 100kbytes.

The video streaming traffic is based on [6], [7]. It represents a video stream with an average bitrate of 32kbps and a frame rate of 7.5frames/s. The video frame sizes are generated according to a linear time series with state dependent parameters (10-state Markov chain), which are adjusted to correspond to an H.263 encoded video source. Video frames are segmented into RTP packets with small frames being put into one RTP packet while larger frames are split in two or more RTP frames. On the receiver side, the application stores each video frame in a buffer, whose length was set to 10s, until it is time to play the movie. The average duration of a streaming session is of 40s, during which there is a rather high channel activity. Due to this high activity and the demanding QoS requirements, more than one timeslot is often required to provide acceptable service conditions.

In addition to the two previously mentioned services, a simplified FTP-like traffic source has been implemented. It consists of a session with a single file transfer, which is assumed to have a constant size of 512kbytes. The reason for including this model was mainly to have a traffic source with high activity and not so demanding QoS requirements, i.e., a sort of combination of properties of both WWW and streaming services.

A summary of the main parameters of each traffic model is presented in table II. The μ and σ variables correspond to the mean and standard deviation, respectively. The Pareto distribution has the specific parameters α and k, which determine the shape of the curve. The m variable defines the cut-off value of the truncated distributions. The Markov chain initial state is represented by parameters: m_0 (expected frame size), b_0 (feedback parameter taking into account cumulative deviation from expected frame sizes), and c_0 (white noise standard deviation).

The quality requirements for the users of each traffic model are presented in table III. The QoS requirement of the WWW

TABLE II
TRAFFIC MODEL PARAMETERS

Parameter	Distribution	Value
Web browsing		
Number of packets per session	Geometric	$\mu = 10$ packets
Reading time between packets	Truncated pareto	$\mu = 10s$ $\alpha = 1.4$ $k = 3.45$ $m = 120s$
Packet size	Truncated lognormal	$\mu = 4.1$ kbytes $\sigma = 30$ kbytes $m = 100$ kbytes
Streaming (32kbps)		
Movie length	Exponential	$\mu = 40$s
Video frame size	Markov chain	$m_0 = 533$ $c_0 = 132.6$ $b_0 = 1.07$
Frame rate	-	7.5 frames/s
Buffer length	-	10s
Simplified FTP		
Number of packets per session	-	1 packet
Packet size	-	512 kbytes

service is expressed in terms of the average packet throughput (R_{avg}), while the streaming service has its requirement defined in terms of buffering delays. D_{buff} is the total buffering delay, D_{init} is the initial buffering delay, T_{clip} is the length of the video clip, and Θ is the fraction of the video length that the user accepts as re-buffering delay (set to 0.167) [8], [6]. In terms of global QoS requirements, it is assumed that the system capacity is reached when 90% of the users are still satisfied.

The evaluation scenarios include: frequency reuse patterns ranging from 1/1 to 4/12, so that different interference profiles can be contemplated, low mobility users (3km/h) in a macro cellular environment, 4-multi-slot terminals, random frequency hopping, and ideal link adaptation. The following mixed-service scenarios will be considered in the simulations: streaming combined with WWW and streaming with FTP. For more simulation details refer to table IV.

TABLE III
QOS REQUIREMENTS OF THE TRAFFIC SOURCES

Traffic	QoS requirement
Web-browsing	$R_{avg} \geq 10kbps$
FTP	$R_{avg} \geq 20kbps$
Streaming	$D_{buff} \leq D_{init} + \Theta \cdot T_{clip}$

TABLE IV
SIMULATION PARAMETERS

Parameter	1/3 reuse	4/12 reuse
Simulation timestep	20ms	20ms
Number of frequencies	12	36
Frequency hopping	random	random
Power control	disabled	disabled
Multislot capability	4	4
Traffi c models	STR+WWW	STR+FTP
Scheduling algorithms	RR, TFQ	RR, PFQ, TFQ
TFQ WWW target	10kbps	10kbps
TFQ FTP target	20kbps	20kbps
TFQ STR target	32kbps	32kbps

IV. ANALYSIS OF RESULTS

This section presents some results concerning the performance of scheduling algorithms for improving the capacity of mixed-service data networks. The mixed scenarios are evaluated for different frequency reuse patterns (1/3 and 4/12), which characterize distinct interference profiles.

A. Results for the 1/3 reuse scenario

Figure 1 displays the isolated performance of each service (WWW and streaming), from which the individual capacity limits can be taken (at 90% satisfaction). Figure 2 shows the capacity region for a combination of WWW and streaming users in a 1/3 reuse scenario. The extremes of the curve correspond to the individual service capacities, while the area underneath the curve corresponds to feasible service mixes, i.e., operating points for which both services still have their QoS requirements satisfied.

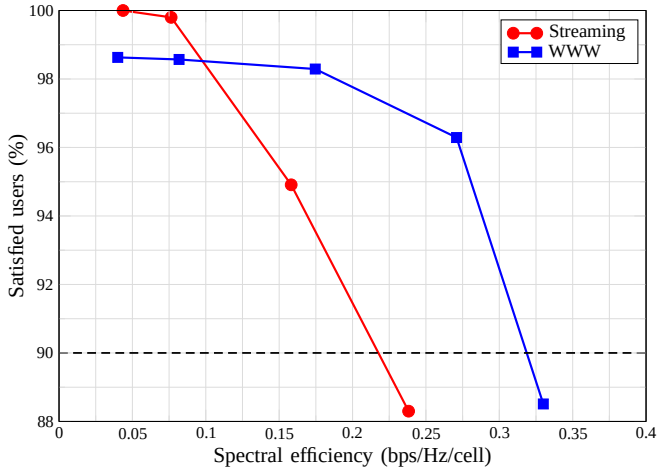

Fig. 1. Isolated performance of the WWW and streaming services in a 1/3 scenario.

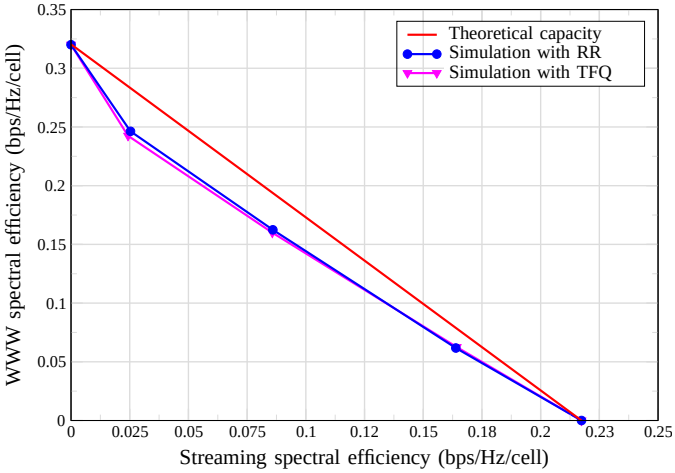

Fig. 2. Mixed capacity region for the WWW and streaming services in a 1/3 scenario.

It can be seen that the system capacity is below the expected theoretical curve. The streaming service is the one limiting capacity, since the WWW service still had excess quality at the combined capacity boundary. Also, the application of the

TFQ scheduling algorithm had practically no effect over the achieved results, due to the interference-limited nature of the evaluated scenario.

Such a situation indicates that interference balancing techniques, such as SBPS [1], would be more suitable to improve system capacity instead of the considered scheduling scheme.

B. Results for the 4/12 reuse scenario

In order to analyze a queuing-limited scenario, a 4/12 reuse pattern has been used instead of the 1/3 pattern. The individual capacity results can be seen in figure 3, while figure 4 shows the capacity region for a mix of streaming and FTP services. Similarly, there are curves representing the theoretical balanced performance, the Round Robin, the PFQ and TFQ algorithms.

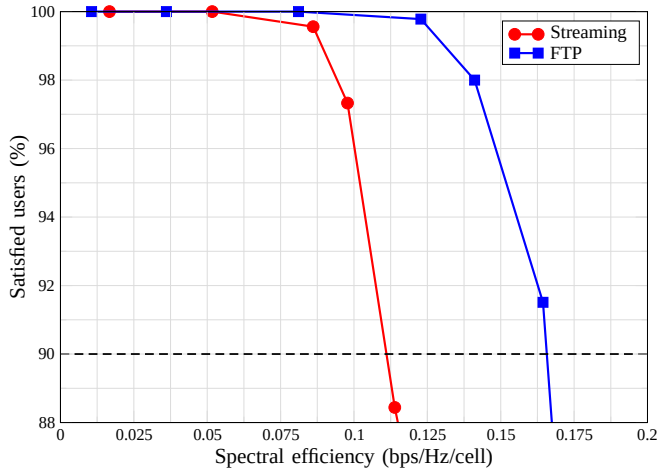

Fig. 3. Isolated performance of the FTP and streaming services in a 4/12 scenario.

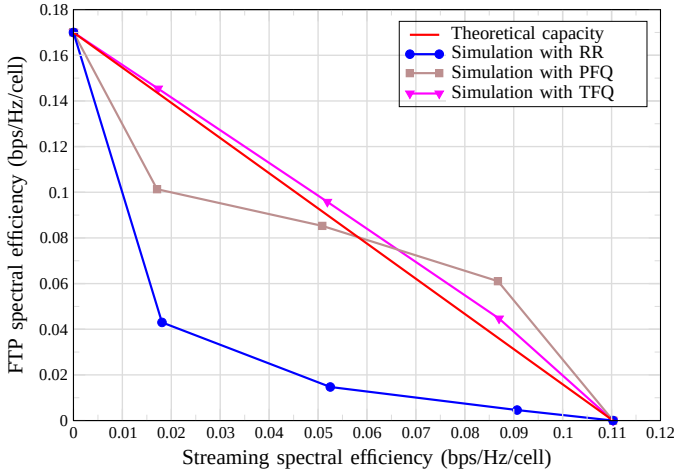

Fig. 4. Mixed capacity region for the FTP and streaming services in a 4/12 scenario.

The unbalanced performance with Round Robin was severely limited by the QoS of the streaming service. In this scenario the channel queues are larger, and the streaming service does not tolerate extensive delays, as opposed to the FTP service.

In such a scenario, throughput fair algorithms such as PFQ or TFQ should provide better results, which in fact can be verified from figure 4. Both algorithms present a much larger capacity region when compared to round robin.

The TFQ algorithm achieved an overall better performance than PFQ, mainly due to the fact that the latter does not take into account that streaming is a more demanding service. Only for higher loads of the streaming service that PFQ provided a slightly better performance, since there are less FTP users competing for the resources, and therefore it is more benefitial for the system to prioritize flows with higher instantaneous throughput (R_{inst}). Another advantage of TFQ over PFQ is that it is simpler to implement, since it does not depend on channel quality estimation measures for determining R_{inst} (note that the PFQ performance here presented considers ideal channel quality estimation).

When compared to the interference-limited scenario, the application of the TFQ algorithm this time provided significant capacity gains, approaching rather closely the theoretical balanced curve. This demonstrates that the algorithm was capable of efficiently tracking the QoS requirements of each service.

Figure 5 better illustrates the QoS balancing effect of the TFQ algorithm. It shows the QoS of both streaming and FTP services for a fixed streaming load and an increasing FTP load. With Round Robin the streaming service quickly limited system capacity, while for the TFQ scheme the QoS limits of both services were reached at roughly the same point, thus maximizing capacity.

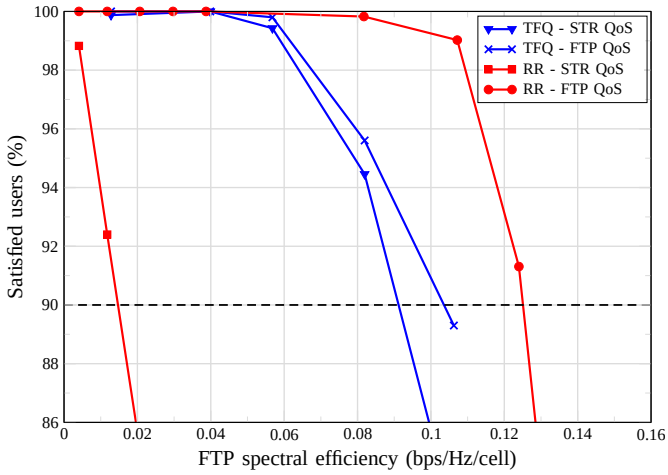

Fig. 5. QoS balancing for the 4/12 scenario (fixed streaming load: 0.052bps/Hz/cell)

Table V presents a summary of the FTP spectral efficiency gains of each scheduling algorithm with regard to the round robin algorithm. The gains are shown for low, medium, and high streaming loads.

V. CONCLUSIONS

This paper presented an investigation of the application of scheduling algorithms as QoS balancing tecniques in GSM/EDGE cellular networks. Their applicability was shown to be highly dependant on queuing-limitation. For a 1/3

TABLE V
FTP SPECTRAL EFFICIENCY GAINS WITH REGARD TO THE RR ALGORITHM FOR DIFFERENT STREAMING LOADS.

Algorithm	Low load	Medium load	High load
Absolute gains (bps/Hz/cell)			
PFQ	0.06	0.07	0.06
TFQ	0.10	0.08	0.04
Relative gains (%)			
PFQ	134	482	1275
TFQ	238	551	886

reuse pattern, it was verified that the TFQ algorithm had no impact over system capacity whatsoever, due to the strong interference-limitation. Nevertheless, for a 4/12 reuse scenario, the TFQ algorithm was capable of providing rather significant capacity gains, thus balancing the system QoS, i.e., both services reached their QoS requirements at roughly the same capacity point.

It was also shown that TFQ presented an overall better performance than PFQ, achieving a larger mixed capacity region. The main idea behind TFQ was to combine the fairness characteristic of the FT algorithm with a QoS differentiation scheme, which was done by taking into account the throughput targets of each service. A perspective for further studies could be, for instance, to enhance TFQ by adding the channel quality estimation characteristics of PFQ.

ACKNOWLEDGMENT

This work is supported by a grant from Ericsson of Brazil - Research Branch under ERBB/UFC.08 technical cooperation contract.

REFERENCES

[1] A. Furuskär, "Radio resource sharing and bearer service allocation for multi-bearer service, multi-access wireless networks - methods to improve capacity," Ph.D. dissertation, Royal Institute of Technology, Stockholm, Sweden, May 2003.

[2] A. Furuskär, P. de Bruin, C. Johansson, and A. Simonsson, "Controlling QoS for mixed voice and data services in GERAN - the GSM/EDGE Radio Access Network," *IEEE 3G Wireless Conference*, May 2001.

[3] W. Perndl, "Scheduling algorithms for UMTS FDD downlink," Master's thesis, Technische Universität Wien, Vienna, Austria, 2001.

[4] F. R. P. Cavalcanti, W. M. de Sousa Jr., Y. C. B. Silva, and T. F. Maciel, "Combined performance of packet scheduling and smart antennas for data transmission in EGPRS," *IEEE Vehicular Technology Conference*, vol. 2, pp. 797–801, May 2002.

[5] C. Johansson, L. D. Verdier, and F. Khan, "Performance of different scheduling strategies in a packet radio system," *IEEE International Conference on Universal Personal Communications*, vol. 1, pp. 267–271, October 1998.

[6] C. Johansson, H. Nyberg, and P. de Bruin, "Streaming services in GSM/EDGE-radio resource management concepts and system performance," *IEEE VTS 54th Vehicular Technology Conference VTC Fall*, pp. 1765–1769, 2001.

[7] H. Nyberg, C. Johansson, and B. Olin, "A streaming video traffi c model for the mobile access network," *IEEE VTS 54th Vehicular Technology Conference VTC Fall*, pp. 423–427, 2001.

[8] A. Schieder, H. Ekstrom, H. Nyberg, C. Johansson, P. de Bruin, and H. Nordstrom, "Resource effi cient streaming bearer concept for GERAN," *The 5th International Symposium on Wireless Personal Multimedia Communications*, pp. 858–862, 2002.