QoS evaluation in a UMTS cell

- TECHNICAL REPORT RCL050601 -
(UNIVERSITY OF FIRENZE, DIP. SISTEMI E INFORMATICA)
(JUNE, 2005)

Paolo Lollini[1] Andrea Bondavalli[1] Felicita Di Giandomenico[2]

[1]University of Florence, Dip. Sistemi e Informatica, viale Morgagni 65, I-50134, Italy
{lollini, a.bondavalli}@dsi.unifi.it
[2]Italian National Research Council, ISTI Dept., via Moruzzi 1, I-56124, Italy
digiandomenico@isti.cnr.it

Abstract

The objective of this work is to evaluate the Quality of Service (QoS) of a Universal Mobile Telecommunication System (UMTS) cell. Among UMTS characteristics, we focus the attention on two aspects strictly related with the identified QoS measures: the random-access procedure and the admission control strategy. Using a model-based approach, we model and evaluate a UMTS cell under several service classes, thus providing useful insights on the perception of the QoS both from the users'perspective and from the mobile telephone operators'one.

1 Introduction

The telecommunication industry is continuously investing on innovative and competitive wireless services. In this paper, we concentrate on the Universal Mobile Telecommunications Systems (UMTS), which is the third generation mobile communication system standardized by 3GPP. High QoS is an always increasing requirement which absorbs significant effort by system suppliers. Quantitative assessment of the degree of QoS offered by a UMTS network (as well as by other networked systems) is therefore of primarily importance.

Though the evaluation of the QoS of a UMTS network in a single service scenario is a widely discussed topic in the literature, only few works aim at analyzing the UMTS system in presence of different traffic scenarios. [1] evaluates the performance of Code-Division Multiple-Access (CDMA) schemes with integrated services, namely real-time voice services and non real-time data services. [2] focuses the attention on multimedia services (real-time and non real-time) and it proposes an adaptive resource allocation system to provide appropriate QoS according to service requests from end users. In [3] the performance of a mixed traffic scenario on the downlink of a CDMA is evaluated, taking into account speech traffic and web-browsing traffic sharing the same frequency carrier.

This paper contributes to the analysis on the service accomplishment level perceived by UMTS users. The focus is on those UMTS characteristics which mainly impact on the analyzed QoS measures, namely the random-access procedure and the admission control strategy, and it accounts for different classes of users behavior, mainly distinguished by their throughput, the workload they add on the cell, the duration of the requested service, the duration of the idle period and the set of sub-channels available for preamble transmission. Following a modular approach, we build the UMTS cell model as composition of more simple submodels representing different system characteristics, and we solve it using a simulation method.

The rest of this paper is organized as follows. Section 2 presents the system under analysis and the identified QoS indicators. Section 3 details the modeling technique, the list of assumptions used in the modeling definition and the description of the submodels that compose the UMTS cell model. In Section 4 the numerical results of the simulation studies are presented and discussed, while conclusions are drawn in Section 5.

2 The investigated system and the QoS indicators

In this paper we focus the attention on two UMTS characteristics having important effects on the QoS, both from the users'perspective and from the mobile telephone operators'one: the random-access procedure and the admission control strategy. When a user needs a service from the network, its User Equipment (UE) sends a channel request to the network through the Physical Random Access CHannel (PRACH), a specific channel dedicated to the uplink transmission of channel request. Since the network does not control the PRACH usage, the access method, based on a random-access procedure, may cause collisions among requests by different UEs, thus worsening the expected QoS. Once the network receives the channel request, it performs the admission control procedure to decide if a traffic channel can be allocated to this new request. The goal is, in general, to ensure that the interference created after adding a new call does not exceed a pre-specified threshold, thus preventing the QoS to degrade below a certain level. In this section we give an overview of these two mechanisms detailing only those aspects that will be used in the modeling phase.

The random-access transmission in the 3GPP specifications [4, 5] is based on a slotted ALOHA approach. It consists of several consecutive preamble transmissions with increasing power at well-defined time-offsets, called access slots. The preambles are sent through the PRACH depicted in Figure 1. It consists of 15 access slots divided in two frames (which are further divided into 12 subchannels). The first frame is synchronized with the even frames of the Primary Common Control Physical CHannel (SFN mod 2 = 0, where SFN is the System Frame

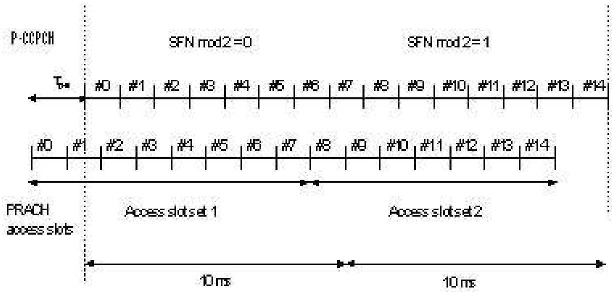

Figure 1: PRACH structure

Number) and it consists of the slots from 0 to 7; the second frame is synchronized with the odd frames of the PC-CPCH (SFN mod 2 = 1) and it consists of the slots from 8 to 14. Before starting the random-access procedure, the UE receives some information from the Broadcast CHannel (BCH) like available signatures, available slots in the available sub-channels, maximum allowed preamble retransmissions (M_{max}), ramping factor, persistency value and initial preamble power. Then the UE randomly selects one of the available sub-channels for transmission ($SelSubCh$), an access slot in that sub-channel, a preamble spreading code from the set of available codes and it starts a preamble transmission using the selected uplink slot, preamble signature and preamble transmission power. When the network receives the preamble, it replies with a positive acquisition indicator (Ack) if the admission control procedure has been passed, otherwise with a negative acquisition indicator (Nack). The network does not send any reply (NoAck) if it does not receive the preamble (e.g. in case of not enough preamble transmission power) or in case of preamble collisions (more preamble transmissions over the same time slot). In case of Nack or NoAck signals, the UE selects the next available access slot in the set of available PRACH subchannels, it randomly selects a new signature, it increases the preamble transmission power by the ramping factor obtained from BCH and, if the number of transmission attempts is still less than the maximum allowed, it starts a new preamble transmission. In case no more transmission attempts are available, the random-access procedure is terminated (*random-access procedure failure*). If the UE receives an Ack signal, it can finally start the transmission of the message using the allocated traffic channel.

Call Admission Control (CAC) is one of the important means to guarantee Quality of Service (QoS) in the telecommunication systems. When a UE asks for a service, the CAC algorithm decides whether to accept or reject this new request. There are several types of admission control algorithms studied in the literature, each one having different properties and aiming at optimizing different network parameters [6, 7, 8]. Here we consider an admission control algorithm based on the workload of the UMTS cell: a new call is accepted if the workload level reached after adding the call does not exceed a pre-specified threshold, both in uplink and in downlink. Equivalently:

$$\eta_{ul} + \Delta L_{ul} < \eta_{ul\_threshold} \quad , \qquad (1)$$

$$\eta_{dl} + \Delta L_{dl} < \eta_{dl\_threshold} \quad , \qquad (2)$$

where η_{ul}, ΔL_{ul} and $\eta_{ul\_threshold}$ (or η_{dl}, ΔL_{dl} and $\eta_{dl\_threshold}$) are, respectively, the cell workload before the admission of the new call, the workload increment due to the admission of the new call and the pre-specified threshold level in uplink (or downlink).

The QoS indicators that we analyze are the following:

1. The call blocking probability (P_{block}), that is the probability that a call is blocked due to a random-access procedure failure.
2. The mean access time (T_{access}), that is the mean duration of the random-access procedure from the time a user sends a channel request to the time it receives a traffic channel or the random-access procedure fails.
3. The cell workload in uplink (η_{ul}) and downlink (η_{dl}).
4. The number of traffic channels allocated, that corresponds to the average number of served users.

The first two measures concern the QoS from the users' perspective, while the others are more mobile operators' oriented.

3 The UMTS cell model

All the developed models are derived using the Stochastic Activity Networks [9] (SAN) formalism. The UMTS cell model is built through the composition of three types of submodels, each one capturing a different aspect of the system behavior (see Figure 2). Suppose to have N different service classes (e.g. conversational, interactive, background, etc.) distinguished by the throughput, the workload they add to the cell (ΔL_{ul} and ΔL_{dl}), the duration of the service requested ($ServTime$), the duration of the idle period ($IdleTime$), the duration of the idle period in case of random-access procedure failure ($BlockTime$) and the set of sub-channels available for preamble transmission ($SelSubCh$). For each service class *i* (with *i*=1,...,N), we build one model capturing the user's behavior belonging to this class ("User_*i*" submodel) and one model representing the random-access procedure associated to the same class ("Random-Access Procedure_*i*" submodel). On the contrary, the traffic channel allocation, the admission control strategy and the collisions control on preamble transmissions are shared by all the service classes and are modeled in the "Resource-Control" submodel.

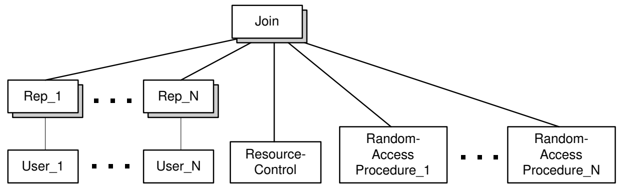

Figure 2: Composed model

As depicted in Figure 2, the models are then composed together using proper operators made available by the SAN formalism (namely, *Join* and *Rep* [1] [10]).

The UMTS cell model is based on these assumptions:

⋄– The number of users camped in the cell is constant, and no user can pass from a service class to another.

⋄– The admission control strategy is based on the workload, according to the previous relations (1) and (2).

⋄– If the cell accepts the service request, a Dedicated CHannel (DCH) is assigned to the user until the service is completed.

[1] The Join operator takes as input a) a set of submodels and b) some shared places owning to different submodels of the former set. Its output is a new model the comprehends all the joined submodels' elements (places, arcs, activities) but with the shared places merged in a unique one. The Rep operator allows the composition of replicas of one SAN, with a subset of the places shared among the SANs or all replicas.

◇– When the service is completed (or the random-access procedure fails) a user sends a new service request to the network.
◇– The cell uses two Frequency Division Duplex carriers.

3.1 User_*i* model

The "User_*i*" model depicted in Figure 3 represents the behavior of one user belonging to the service class *i*, with $i \in \{1,...,N\}$, where N is the total number of service classes. This model is replicated (through the Rep operator) K_i times if the service class *i* consists of K_i users. In this subsection we outline the meaning of its main elements.

The "User_*i*" model shares the following places:

- `prach_sent_i`, `ready_i` and `fail_i`, shared with the corresponding "Random-Access Procedure_*i*" model;
- `ok_ack_i`, `conn_fail_i`, `service_channel_i`, `channels_not_available_i` and `channels_i`, shared with the "Resource-Control" model.

A token in place `idle` represents a user whose service request has been satisfied and completed. After an idle period, defined by the deterministic transition `t_idle` that fires in $IdleTime$ seconds, the token is moved in place `ready_i` (the user sends a new channel request to the network) and a new random-access procedure begins. At this point the "Random-Access Procedure_*i*" model removes the token from place `ready_i` and, together with the "Resource-Control" model, determines the outcome of each channel request by putting a token in a specific place. The possible outcomes of a channel request are:

- channel request not sent. In this case the token in place `ready_i` is moved to place `fail_i` that represents a UE that cannot send the preamble transmission because the persistency check has not been passed. The input gate `c_fail` enables the instantaneous transition `t_fail` and the token is removed from place `fail_i` and added to place `wait`. Subsequently, after a time of 10 ms defined by the deterministic transition `t_wait`, a new channel request is sent.
- channel request sent to the network. In this case the token in place `ready_i` is moved to place `prach_sent_i` that represents a UE that has sent the preamble transmission through the PRACH. The input gate `check_prach_sent` enables the instantaneous transition `decide_ack` that fires. This transition has two cases that model the probability that the network receives the preamble from the UE. Since the transmission power increases with the number of transmission attempts, the probability that the network receives the preamble transmission increases as well. When the `decide_ack` transition fires, the token is removed from place `prach_sent_i` and it is put:
  - in place `ok_ack_i` (and in place `flag_1`), if the network receives the preamble transmission;
  - in place `conn_fail_i` (and in place `flag_2`), if the network does not correctly receive the preamble (not enough transmission power). In this case the input gate `check_attempts` enables the transition `t_check_attempts` that fires following a uniform distribution between 10 and 30 ms; then a token is removed from place `conn_fail_i` and, based on the marking of place `Mmax` representing the number of remaining preamble transmissions, it checks if the UE can send further preamble transmissions or not. In the first case a token is removed from place `Mmax` (one attempt less) and a token is put in place `wait`, otherwise a token is put in place `wait_block_1` and then, after a time of 10 ms defined by the deterministic transition `t_wait_block_1`, the token is moved to place `block` that represents the case of a random-access procedure failure. The exponential transition `t_block` fires with a mean of $BlockTime$ seconds and then the token is moved to place `ready_i` (the user sends a new channel request).
- channel request correctly received by the network but not satisfied due to the admission control policy. In this case the token in place `ready_i` is moved to place

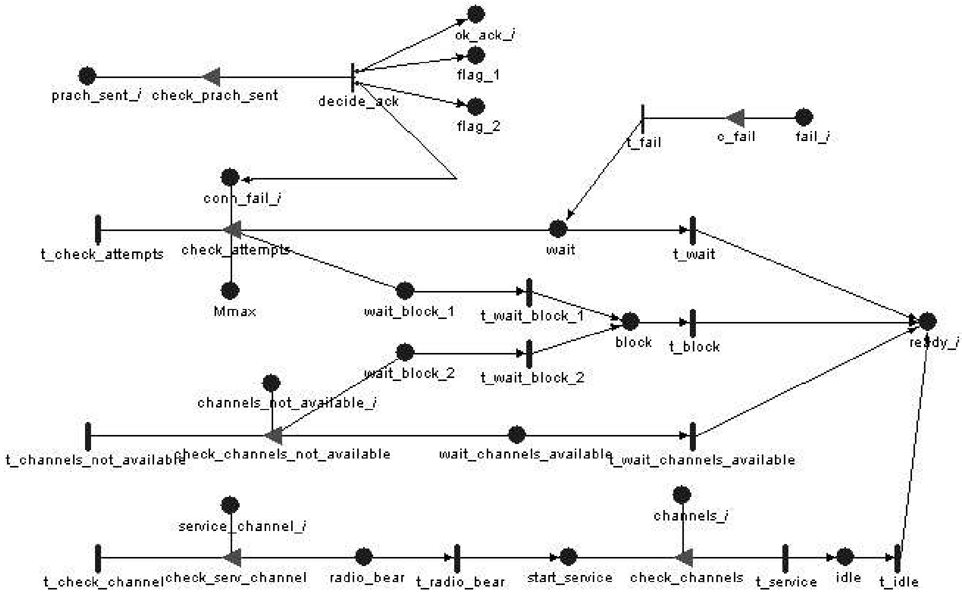

Figure 3: User_*i* model

channels_not_available_*i*. The input gate check_channels_not_available enables the transition t_channels_not_available that fires following a uniform distribution between 10 and 30 ms; then a token is removed from place channels_not_available and, if there are other preamble transmission attempts (the marking of place Mmax is greater than zero), a token is put in place wait_channels_available, otherwise a token is put in place wait_block_2. In the first case the UE sends a new channel request after a time defined by the transition t_wait_channels_available that fires following a uniform distribution between 10 and 1000 ms, and then the token is moved to place ready_*i*. In the other case, the user has no more transmission attempts and then, after a time defined by the transition t_wait_block_2 that fires following a uniform distribution between 10 and 1000 ms, the token is moved to place block.

- channel request not correctly received by the network due to preamble collisions. In this case the token in place ready_*i* is moved to place conn_fail_*i* (but not in place flag_2). Therefore a token in place conn_fail_*i* can represent a non correctly reception of the preamble transmission due to not enough transmission power or due to preamble collisions (distinguished through the number of tokens in place flag_2).

- channel request correctly received and satisfied by the network. In this case the token in place ready_*i* is removed and one token is added to place service_channel_*i* and channels_*i*. A token in place service_channel_*i* represents a dedicated channel allocated to satisfy the service request, while the number of tokens in place channels_i corresponds to the total number of traffic channels allocated for the service class *i*. The input gate check_serv_channel enables the transition t_check_channel that fires following a uniform distribution between 10 and 30 ms; then a token is removed from place service_channel and one token is put in place radio_bear. The deterministic transition t_radio_bear represents the setup time for the radio bear creation and, when it fires, the token is moved to place start_service. Then the input gate check_channels enables the deterministic transition t_service that fires in $ServTime$ seconds (service time); then a token is removed from place start_service and from place channels, and a token is added to place idle.

3.2 The random-access procedure_*i* model

The "Random-Access Procedure_*i*" model for the service class *i* is depicted in Figure 4 and it represents the Physical RACH structure sketched in Section 2. In this subsection we outline the meaning of its main elements.

The "Random-Access Procedure_*i*" model shares the following places:

- prach_sent_*i*, ready_*i* and fail_*i*, shared with the corresponding "User_*i*" model;
- num_slot and count_sfr, shared with all the other "Random-Access Procedure_*j*" model, $\forall j \in \{1, ..., N\}$ with $j \neq i$.

A token in place ready_*i* represents a new channel request belonging to service class *i*. The input gate sync_prach is used to synchronize the channel requests with the beginning of one of the two frames of the PRACH. The number of tokens in num_slot identifies the current slot number in the current frame, while the current frame is identified by the number of tokens in place count_sfn. Based on the persistency value received through the BCH, the UE decides whether to send the preamble on the PRACH or not. The preamble transmission is not allowed with a probability $(1 - Asc)$: in this case the instantaneous transition access_attempt puts a token in place fail_*i* and a new persistency check is performed in the next transmission time interval (the token is moved again to place ready_*i*). Otherwise, with probability Asc, the preamble transmission is allowed and a token is put in place success. Depending on the marking of count_sfr and on the available PRACH sub-channels ($SelSubCh$), the instantaneous transition sel_slot selects the slot of the PRACH that the UE can use to send its channel request. Places slot1, ..., slot8 represent the slots that can be selected during the random-access procedure (slot8 is never selected if the current PRACH frame is the second). The input gates c_slot1, ..., c_slot8 synchronize the service requests with the selected slot and, when the selected access slot is available, they enable the instantaneous transitions ack1, ..., ack8 and the tokens are moved to place prach_sent_*i*, representing those users of service class *i* that have sent the preambles to the network. Tokens in place slot_not_available represent those UE that could send the preamble (they passed the persistency check) but they have to wait for the next frame because of slots unavailable in the current frame. The input gate check_next_frame and the instantaneous transition nextFrame synchronize the new channel requests with the next frame.

3.3 Resource-Control model

The resource control model is depicted in Figure 5. In this subsection we outline the meaning of its main elements.

The "Resource-Control" model shares the following places:

- num_slot and count_sfr, shared with the "Random-Access Procedure_*i*" model, $\forall i \in \{1, ..., N\}$;
- ok_ack_*i*, conn_fail_*i*, service_channel_*i*, channels_not_available_*i* and channels_*i*, shared with the corresponding "User_*i*" model, for each service class $i \in \{1, ..., N\}$.

The input gate manage_prach is used to synchronize the channel requests with the beginning of one of the two PRACH frames. The number of tokens in num_slot (initially set to 8) identifies the current slot number in the current frame, while the current frame is identified by the number of tokens in place count_sfn (initially set to 0): it is the first frame (with 8 access slots) if the mark of count_sfn is odd, otherwise it is the second frame (with 7 access slots). The deterministic transition t_slot fires in $\frac{20}{15}$ ms (slot length), as each frame has a length of 10 ms, and then a token is moved to place slot. When all the tokens have been removed from num_slot, the input gate manage_prach enables the instantaneous transition t_manage_prach that fires; consequently the marking of place num_slot is reset to 7, a token is added

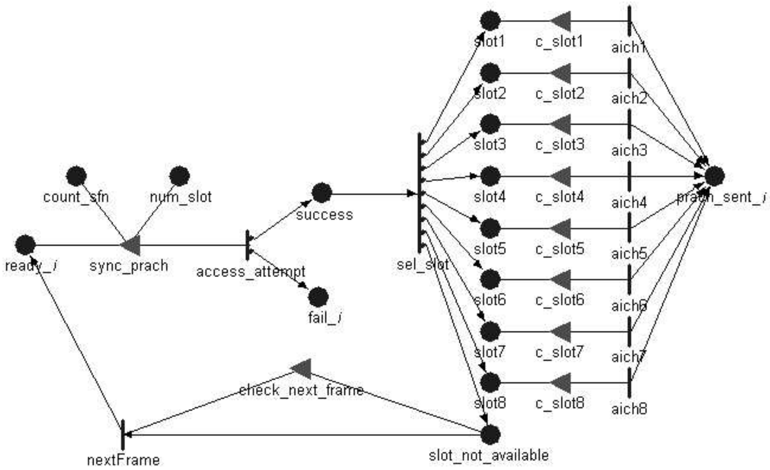

Figure 4: Random-access procedure_*i* model

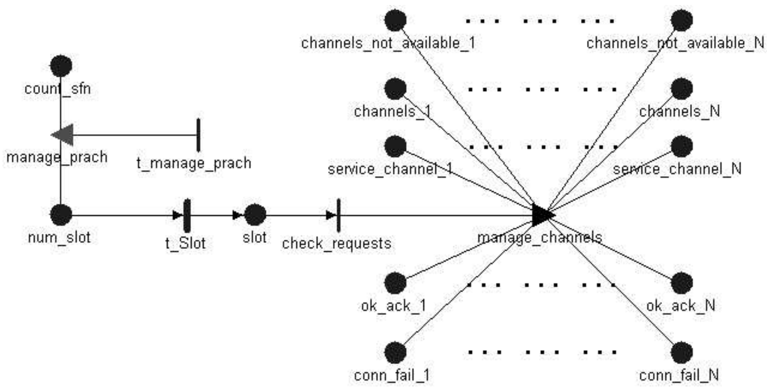

Figure 5: Resource-Control model

to count_sfn (now it is odd), and so on. Once a token arrives in place slot, the instantaneous transition check_requests fires and then the function defined in the output gate manage_channels is performed. This function is responsible to detect preamble collisions and to allocate traffic channels basing on the admission control policy. Here we briefly introduce the places involved in the function activation. For each $i \in \{1,...,N\}$,

- a token in place ok_ack_*i* represents a preamble transmission received by the network and issued by a user belonging to the service class *i*;
- a token in place conn_fail_*i* represents a preamble transmission issued by a user belonging to service class *i* that has not been received by the network because of an inadequate preamble transmission power or because of preamble collisions;
- a token in place service_channel_*i* represents a dedicated channel allocated to a user belonging to the service class *i*;
- the tokens in place channels_*i* represent the total number of traffic channels allocated for the service class *i*;
- a token in place channels_not_available_*i* represents a channel request issued by a user belonging to service class *i* that cannot be accepted due to the admission control strategy.

The function defined in the output gate manage_channels works as it follows. Once the instantaneous transition check_requests fires, it checks if then number of preambles arrived in the same time slot is greater than zero (marking of ok_aich_1 + ... + marking of ok_aich_N > 0). In this case a preamble collision has occurred and then the tokens are instantaneously moved to the corresponding places conn_fail_1, ..., conn_fail_N. Otherwise it checks if the service request can be accepted by the admission control policy. Basing on the type of requested service (service class *i*) and on the number of traffic channels already allocated (marking of the place channels_*i*, for *i*=1,...,N), it verifies if the workload level reached with the new call ($\eta_{ul} + \Delta L_{ul}$, and $\eta_{dl} + \Delta L_{dl}$) is still less than the maximum allowed ($\eta_{ul\_threshold}$, and $\eta_{dl\_threshold}$). If this is the case, a token is added to the corresponding places channels_*i* and service_channel_*i*. If the new call cannot be accepted, one token is added in the corresponding place channels_not_available_*i*.

4 QoS evaluation

We evaluated the identified QoS measures in a representative scenario with three service classes. A steady-state simulation has been performed, using the simulator provided by the Möbius tool, at varying of the number of users camped in the cell. Table 1 summarizes the settings

Parameters	**Serv.Class 1**	**Serv.Class 2**	**Serv.Class 3**
$Throughput$	12.2 kbps	32 kbps	64 kbps
ΔL_{ul}	0.016	0.037	0.050
ΔL_{dl}	0.013	0.046	0.083
$\eta_{ul\_threshold}$	1.3		
$\eta_{dl\_threshold}$	1.6		
Asc	60 %	50 %	
$SelSubCh$	0,1,2,3	4,5,6,7	8,9,10,11
M_{max}	8		
$ServTime$	300 sec.	266 sec.	131 sec.
$IdleTime$	150 sec.	300 sec.	
$BlockTime$	30 sec.		

Table 1: Main parameters'values

for the main system parameters, distinguished by service class. In all the simulations we are assuming that each service class has the same number of users ($\frac{1}{3}$ of the total number of users camped in the cell).

Figure 6(a) shows the uplink (η_{ul}) and downlink (η_{dl}) workload at varying of the number of users camped in the cell. It can be noted that both of them increase linearly until the maximum workload level in uplink is reached (with about 80 users in the cell). From this point, η_{ul} remains constant while η_{dl} decreases. This trend can be explained by analyzing the plots in Figure (6)(b) that describes the number of allocated channels for each service class. It can be noted that, once the system reaches the maximum uplink workload level, the number of channels allocated to the first service class increases, while decreases for the other two service classes. This is due to the chosen admis-

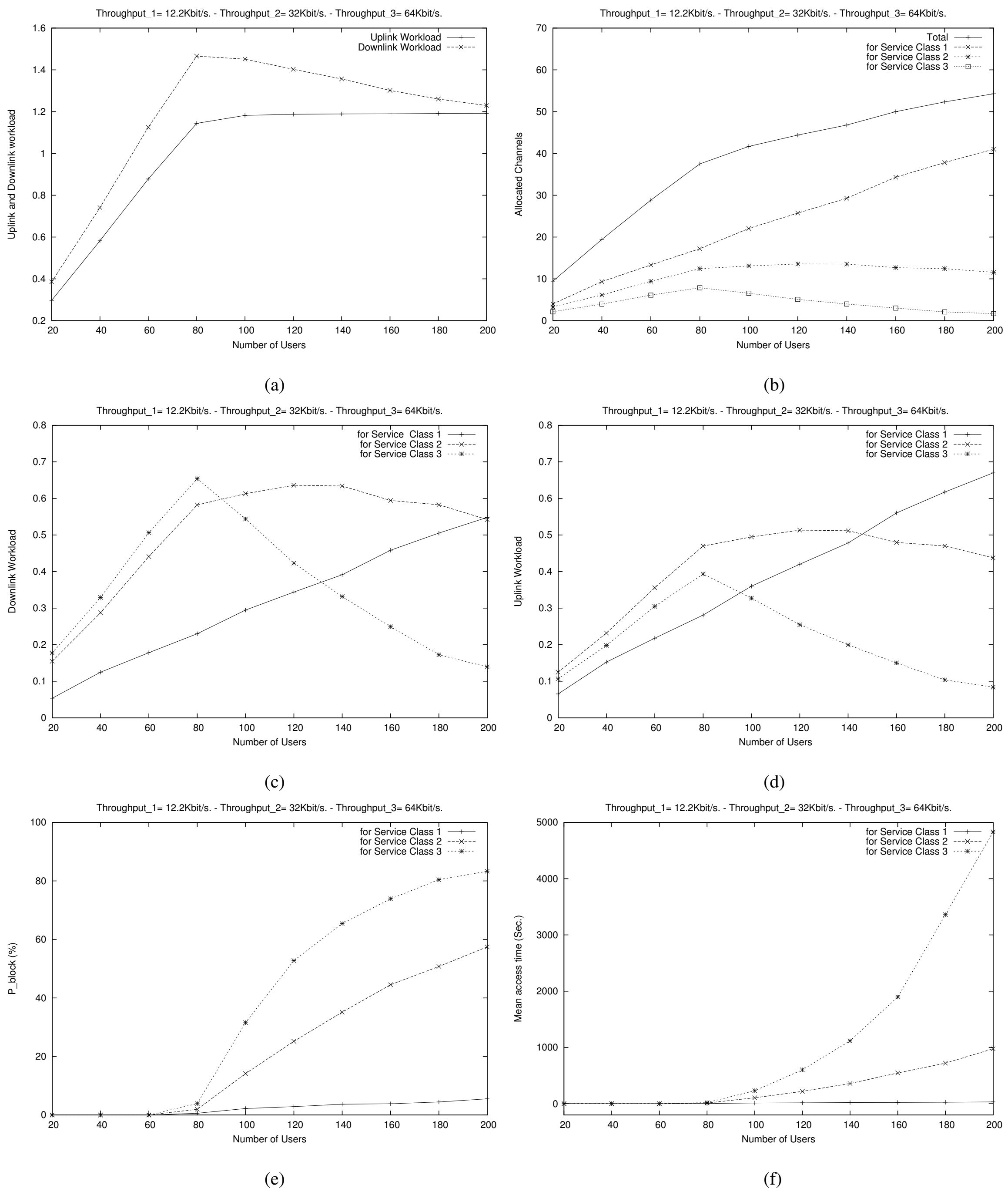

Figure 6: (a) Total workload, (b) Allocated channels, (c) Downlink workload, (d) Uplink workload, (e) Call blocking probability, and (f) Mean access time

sion control strategy, that accepts a new call if the workload level reached after adding the call does not exceed a pre-specified threshold, both in uplink and in downlink. Since the workload increment caused by the service class 1 is lower than the others, than it is more likely that the system accepts these types of calls when the workload of the cell is near its maximum level. Figures 6(c) and 6(d) show, respectively, the uplink (η_{ul}) and downlink (η_{dl}) workload variations for each of the three service classes. Again, the workload contribution of service class 3 rapidly becomes lower if the number of users in the cell increases beyond the 80 units. Finally, Figures 6(e) and 6(f) show, respectively, the call blocking probability and the mean access time for each service class. As expected, the users in the first service class have a good QoS perception as both measures are really low. On the contrary, when the cell reaches its maximum workload level the two measures rapidly increase for the users in service class 2 and 3, thus worsening the perceived QoS.

5 Conclusions

This paper has presented a study on modeling and analysis of a UMTS cell, accounting for different classes of services. QoS measures, relevant from both a user' s and an operator's perspective, have been evaluated, to better understand the underlying processes and get useful insights on proper configurations of UMTS cells. Actually, this analysis constitutes a first, necessary step towards the study of a more complex scenario, involving multiple cells and accounting for dependability critical conditions, such as cell outages. These constitute the main directions for future developments.

6 Acknowledgments

The authors want to acknowledge the contribution of Gabriele Petruzzi to the early phases of this work.

References

[1] Sebnem Zorlu Ozer and Symeon Papavassiliou. Performance Analysis of CDMA Systems With Integrated Services. In *IEEE Transactions On Vehicular Technology*, Vol. 52, No. 4, July 2003.

[2] Lei Huang, Sunil Kumar and C.-C. Jay Kuo. Adaptive Resource Allocation for Multimedia QoS Management in Wireless Networks. In *IEEE Transactions On Vehicular Technology*, Vol. 53, No. 2, March 2004.

[3] Matteo Cesana and Antonio Capone. Impact of mixed voice and data traffic on the UMTS-FDD performance. In Proc. of *IEEE Global Telecommunications Conference*, GLOBECOM '02, Vol. 1, Page(s):758 - 762, 17-21 Nov. 2002.

[4] ETSI TS 25.214 *Physical Layer Procedure (FDD)* (Release 1999) http://www.3GPP.org

[5] ETSI TS 25.321 *MAC protocol specification* (Release 2002) http://www.3GPP.org

[6] K. Al-Begain and A. Zreikat. Interference Based CAC for Up-link Traffic in UMTS Networks. In Proceedings of *World Wireless Congress*, 2002, 28-31 June 2002, pp. 298-303, San Francisco, USA.

[7] M. Andersin, Z. Rosberg, J. Zander. Soft and safe admission control in cellular networks. In *IEEE/ACM transaction on Networking*, Vol. 5, no. 2, April 1997, pp. 255-265.

[8] Y. Ishikawa and N. Umeda. Capacity design and performance of Call Admission Control in Cellular CDMA Systems. In *IEEE Journal on Selected Area in Communications*, Vol. 15, pp. 1627-1635, Oct. 1997.

[9] J. F. Meyer, A. Movaghar, and W. H. Sanders. Stochastic Activity Networks: Structure, Behavior, and Application. In Proceedings of *International Workshop on Timed Petri Nets*, Torino, Italy, pp. 106- 115, July 1985.

[10] W. H. Sanders and J. F. Meyer. Reduced Base Model Construction Methods for Stochastic Activity Networks. In *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 1, Jan. 1991, pp. 25-36.