Adaptive Admission Control in Mobile Cellular Networks with Streaming and Elastic Traffic

David Garcia-Roger, M.ª Jose Domenech-Benlloch, Jorge Martinez-Bauset and Vicent Pla

Departamento de Comunicaciones, Universidad Politecnica de Valencia
Camino de Vera s/n, 46022, Valencia, Spain
Phone: +34 963879733, fax: +34 963877309
{dagarro,mdoben}@doctor.upv.es,{jmartinez,vpla}@dcom.upv.es

Abstract. We propose a novel adaptive reservation scheme that handles, in an integrated way, streaming and elastic traffic. The scheme continuously adjusts the quality of service perceived by users, adapting to any mix of traffic and enforcing a differentiated treatment among services, both in fixed and variable capacity systems. The performance evaluation carried out verifies that the QoS objective is met with an excellent precision and that it converges rapidly to new operating conditions. Other key features of our scheme are its simplicity and its oscillation-free behavior.

1 Introduction

Applications expected to produce the bulk of traffic in the future multiservice Internet can be broadly categorized as streaming or elastic [1]. Streaming traffic requires a minimum transfer rate in order to work properly as well as some time related requirements such as bounded delay and jitter. Elastic traffic has loose time requirements and can adapt to the available resources. In the light of the above it seems natural to give priority to streaming traffic and leave elastic traffic use the remaining capacity (a small amount of resources might be reserved for the elastic traffic to prevent starvation in case of overload of the streaming traffic). Elastic flows are generally transported over TCP which takes care of rate adaptation and bandwidth sharing among the different flows. If the total traffic demand of elastic flows exceeds the available capacity some flows might be aborted due to impatience. Flow impatience due to a very low throughput can arise from human user impatience or because TCP or higher layer protocols interpret that the connection is broken. Abandonments are useful to cope with overload and serve to stabilize the system but, on the other hand, this phenomenon will negatively impact on the efficiency because capacity is wasted by non-completed flows [1]. This drop of efficiency led the authors of [1] to claim that session admission control (SAC) should be enforced for elastic traffic.

In this paper we propose an adaptive SAC scheme for mobile wireless cellular networks that handles in an integrated way both streaming and elastic traffic

and tries to maximize the carried traffic while meeting certain quality of service (QoS) objectives. The QoS objective for streaming traffic is expressed as upper bounds for the blocking probabilities of new and handover requests, while for elastic traffic it is defined as a bound for the abandonment probability. The proposed scheme is adaptive in the sense that if the offered load is above the system capacity, or the number of resource units decreases, or both simultaneously, the SAC system will react trying to meet the QoS objective for as many services as possible. Therefore the proposed scheme might be deployed in both fixed capacity systems (e.g. FDMA/TDMA) and systems limited by interference where capacity is variable (e.g. CDMA).

Our work is motivated in part by the fact that previous adaptive proposals like [2–6] deploy long measurement windows to estimate system parameters, which make the convergence period too long to cope with real operating conditions, or do not provide explicit indication of how the time window must be configured [7–9]. Another motivation is the fact that most of the studies devoted to adaptive schemes only consider the stationary regime and no evidence is provided about their behavior in the transient regime. Therefore, we consider that a fundamental characteristic of an adaptive scheme like its convergence speed to new operating conditions has not been sufficiently explored.

Our scheme does not rely on measurement intervals to estimate the value of system parameters. It generalizes the novel SAC adaptive strategy introduced in [10], which operates in coordination with two well known trunk reservation policies named Multiple Guard Channel (MGC) and Multiple Fractional Guard Channel, although only its operation with the MGC is described here. It has been shown that deploying trunk reservation instead of complete-sharing policies in mobile networks allows the operator to achieve higher system capacity, i.e. to carry more traffic while meeting certain QoS objectives [11].

Our new scheme has four key features that enhance the scheme in [10]. First it handles in an integrated way both streaming and elastic traffic. Second, it allows to enforce a differentiated treatment among different streaming services during under load and overload episodes. In the latter case, this differentiated treatment guarantees that higher priority services will be able to meet their QoS objective, possibly at the expense of lower priority ones. Third, the prioritization order of the streaming services can be fully specified by the operator. And fourth, the operator has the possibility of identifying one of the streaming services as best-effort, being it useful to concentrate on it the penalty that has to be unavoidably paid during overloads.

Adaptive SAC mechanisms have also been studied, for example in [4–6, 12], both in single service and multiservice scenarios, but in a context which is somewhat different to the one of this paper. There, the adjustment of the SAC policy configuration is based on estimates of both the mobility pattern and the handover arrival rates derived from the current number of ongoing calls in neighboring cells. It is expected that the performance of our scheme would improve when provided with such predictive information but this is left for further study.

An extension of our scheme to operate with rate-adaptive multimedia applications [9] is also left for further study.

Given that the operation of the SAC scheme when handling streaming traffic is independent of the elastic traffic because the former has higher priority than the latter, we describe first the operation of the SAC scheme and evaluate its performance with streaming traffic. Section 2 describes the model of the system and defines the relevant SAC policies for our study. Section 3 describes the fundamentals of the adaptive scheme, introducing the policy adjustment strategy and how multiple streaming services are handled. In Section 4 we present the performance evaluation of the proposed adaptive scheme when handling streaming traffic in different scenarios, both under stationary and non-stationary traffic conditions. Section 5 describes the operation of the scheme when handling elastic traffic and evaluates its performance. Finally, Section 6 concludes the paper.

2 System model and SAC policies

We consider the homogeneous case where all cells are statistically identical and independent. Consequently the global performance of the system can be analyzed focusing on a single cell. Nevertheless, given that the proposed scheme is adaptive it could also be deployed in non-homogeneous scenarios.

In each cell a set of R different streaming services contend for C resource units, where the meaning of a unit of resource depends on the specific implementation of the radio interface. For each streaming service, new and handover arrival requests are distinguished, which defines $2R$ arrival classes. For convenience, we denote by s_i the arrival class i, $1 \leq i \leq 2R$. Additionally we denote by s_r^n (s_r^h), the arrival class associated to new (handover) requests of streaming service r, $1 \leq r \leq R$, being $s_r^n = s_r$ and $s_r^h = s_{r+R}$, $1 \leq r \leq R$. For brevity, when we refer to a service or to a class, we mean a streaming service or a streaming arrival class respectively. Elastic traffic is discussed in Section 5.

For any service r, new (handover) requests arrive according to an inhomogeneous Poisson process with time-varying rate $\lambda_r^n(t)$ ($\lambda_r^h(t)$). For mathematical tractability we make the common assumption to model the inter-arrival time of handover requests as an exponential distribution, which is considered a good approximation [13]. Besides, although our scheme does not require any relationship between $\lambda_r^h(t)$ and $\lambda_r^n(t)$, for simplicity we suppose that $\lambda_r^h(t)$ is a constant fraction of $\lambda_r^n(t)$ [14, 15]. Service r requests require d_r resource units per session. As each service has two associated arrival classes, if we denote by c_i the amount of resource units that an arrival class requires for each session, then $d_r = c_r = c_{r+R}$, $1 \leq r \leq R$. For variable bit rate sources d_r resource units denotes the effective bandwidth of the session [15, 16].

For a service r session, both its duration and its cell residence (dwell) time are also assumed to be exponentially distributed with rates μ_r^s and μ_r^d. Hence, the resource holding time for a service r session in a cell is also exponentially distributed with rate $\mu_r = \mu_r^s + \mu_r^d$. Note that the proposed scheme can easily take into account terminals moving at different speeds by defining additional

arrival classes for any service. Note also that the exponential assumption also represents a good approximation for the cell dwell time (essentially, only its average matters), when the performance of the system is evaluated by computing blocking probabilities [17]. It should be highlighted that the operation of our scheme is based on the simple balance equations described in Section 3, which hold for any arrival process and holding time distribution. Hence the basis of the adaptive scheme holds beyond the assumptions made for modeling purposes.

We denote by P_i, $1 \leq i \leq 2R$, the blocking probability perceived by class i requests and by $P_r^n = P_r$ ($P_r^h = P_{R+r}$) the blocking probability perceived by new (handover) requests of service r. The QoS objective is expressed as upper bounds for the blocking probabilities of each arrival class. Thus, we denote by B_r^n (B_r^h) the bound for new (handover) blocking probabilities. Let the ongoing sessions vector be $\boldsymbol{n} := (n_1, \ldots, n_R)$, where n_r is the number of sessions in progress of service r in the cell initiated as new or handover requests. We denote by $c(\boldsymbol{n}) = \sum_{r=1}^{R} n_r d_r$ the number of busy resource units in state $\boldsymbol{n}$.

Finally, we denote by λ_{max} the system capacity, i.e. the maximum λ that can be offered to the system while meeting the QoS objectives, where λ is the aggregated arrival rate of new requests $\lambda = \sum_{r=1}^{R} \lambda_r^n$, $\lambda_r^n = f_r \lambda$ and $\sum_{r=1}^{R} f_r = 1$. Defining service penetrations (f_r) is a common approach when studying these systems [15].

The definition of the MGC SAC policy is as follows: one configuration parameter is associated with each arrival class i, $l_i \in \mathbb{N}$. An arrival of class i in state $\boldsymbol{n}$ is accepted if $c(\boldsymbol{n}) + c_i \leq l_i$ and blocked otherwise. Therefore, l_i is the amount of resources that class i has access to and increasing (decreasing) it reduces (augments) P_i. Number based SAC, that is a common technique in systems whose capacity is limited by blocking, has also been considered a good approach for those systems whose capacity is limited by interference, see for example [18] and references therein.

3 Operation of the SAC adaptive scheme

Most of the adaptive schemes proposed for single service scenarios deploy a reservation strategy based on *guard channels*, increasing its number when the QoS objective of the handover arrival class is not met. The extension of this heuristic to a scenario with multiple services is much more difficult to manage because the adjustment of the configuration parameter l_i has an impact not only on the QoS perceived by class i but also on the QoS perceived by the rest of classes. Our scheme has been designed to handle this difficulty.

As a first step to handle this difficulty, we classify the different arrival classes into two generic categories: i) several *protected* classes, for which specific QoS objectives must be met; ii) one *Best-Effort Class* (BEC), with no specific QoS objective. Additionally, in a multiservice scenario the operator can define priorities for the protected classes at its convenience in order to give greater protection to the most important classes. Note that BEC arrival requests perceive an unpre-

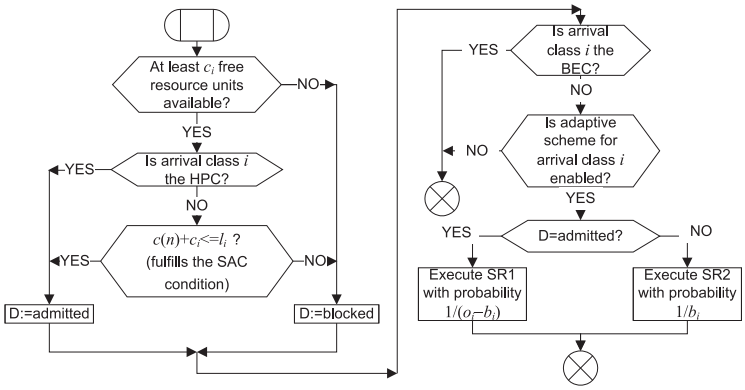

Fig. 1. Operation of the SAC adaptive scheme.

dictable blocking probability but those sessions accepted are allocated a constant amount of resources during its lifetime.

Let $\mathbf{s} = (s_1, \ldots, s_{2R})$ be the set of arrival classes, $\mathit{\Pi} := \{(\pi_1, \ldots, \pi_{2R}) : \pi_i \in \mathbb{N}, 1 \leq \pi_i \leq 2R, \pi_i \neq \pi_j \text{ if } i \neq j\}$ the set of all possible permutations of $\{1, 2, \ldots, 2R\}$ and $\pi^* \in \mathit{\Pi}$ the order defined by the operator, then $\mathbf{s}^* = (s_{\pi_1}, \ldots, s_{\pi_{2R}})$ is called the *prioritization order*, being s_{π_1} the *Highest-Priority Class* (HPC) and $s_{\pi_{2R}}$ the *Lowest-Priority Class* (LPC). If there is a BEC, this class will be the LPC in the prioritization order. We study two implementations, one in which the LPC is treated as a protected class and one in which the LPC is the BEC.

For the sake of clarity, the operation of our scheme is described assuming that the arrival processes are stationary and the system is in steady state. In practice, we can assume without loss of generality that the QoS objective for s_i can be expressed as $B_i = b_i/o_i$, where $b_i, o_i \in \mathbb{N}$. Then it is expected that if $P_i = B_i$ the class i will experience, in average, b_i rejected requests and $o_i - b_i$ admitted requests, out of o_i offered requests. For example, a QoS objective for s_i of $B_i = 1/100$ implies that $b_i = 1$ and $o_i = 100$. It seems intuitive to think that the adaptive scheme should not change the configuration parameters of those arrival classes meeting their QoS objective and, on the contrary, adjust them on the required direction if the perceived QoS is different from its target. Therefore, assuming integer values for the configuration parameters, like those of the MGC policy, we propose to perform a probabilistic adjustment each time a request is processed in the following way: i) if accepted, do $\{l_i \leftarrow (l_i - 1)\}$ with probability $1/(o_i - b_i)$; ii) if rejected, do $\{l_i \leftarrow (l_i + 1)\}$ with probability $1/b_i$. Therefore, when deploying this adjustment scheme under stationary traffic, if $P_i = B_i$, then, in average, l_i is increased by 1 and decreased by 1 every o_i offered new requests, i.e. its mean value is kept constant. Finally, note that when the traffic is non-stationary, the adaptive scheme will continuously adjust the QoS perceived by each class in order to meet its objective if possible. Our approach,

although simple, is innovative because to the best of our knowledge, something based on the same or a similar idea has not been proposed before.

Figure 1 shows the operation of the SAC policy and the adaptive scheme in more detail. As shown, to admit a class i request it is first checked that at least c_i free resource units are available. Note that once this is verified, HPC requests are always admitted, while the rest of classes must also fulfill the admission condition imposed by the SAC policy. Note also that the $l_{\pi 1}$ is always updated to detect when the HPS becomes congested. Due to paper length limitations, the subroutines SR1 and SR2 mentioned in Fig. 1 are not explained in detail. In general, the adaptive scheme associated to each class is always operating (except for the BEC), but to be able to guarantee that the QoS objective is met for as many classes as possible, particularly during overload episodes or changes in the traffic mix, the adjustment algorithm described before requires additional mechanisms which might include the disabling of the adaptive scheme associated to other low priority classes.

When the QoS objective for class i is not met, the MGC policy configuration will be adjusted using two different mechanisms. The *direct* way is to increase the configuration parameter l_i, but its maximum value is C, i.e. when $l_i = C$, full access to resources is provided to class i and setting $l_i > C$ does not provide additional benefits. In these cases, an *indirect* way to help class i is to limit the access to resources of lower priority classes by reducing their associated configuration parameters. It is clear that when a higher priority class s_i needs to adjust the configuration parameter of a lower priority class s_j, the adaptive scheme must adjust l_j only when arrivals from s_i occur, while no adjustments must be carried out when arrivals from s_j occur. To operate in this way the adaptive scheme associated to s_j is disabled.

4 Performance Evaluation

We evaluate the performance of the proposed adaptive SAC scheme by solving the continuous-time Markov chain (CTMC) that describes its operation, both in the stationary and transient regimes. In both regimes P_i is determined as the percentage of time an arrival request from s_i would be rejected.

In general, we have a multidimensional CTMC which state space is given by $(n_1, \ldots, n_R, l_1, \ldots, l_{2R})$. We allow l_i to take positive and negative values as a means to remember past adjustments and to identify the adjustment type the scheme uses (direct or indirect). Given that the general multidimensional diagram is difficult to draw, we show a bidimensional CTMC in Fig. 2 as an example. This system has only one service and therefore two classes, s^h and s^n, with $d = 1$ and C resource units. It is assumed that s^h is the HPC and therefore their requests are always accepted (if free resources are available), while s^n is a BEC. The system state vector is defined as (n, l^h), where n is the number of resource units occupied. In this system, l^h is adjusted following the probabilistic adjustment rule described previously and $l^n = C - \max\{0, (l^h - C)\}$. Note that during under load episodes $l^n = C$, but during overload episodes s^h might have

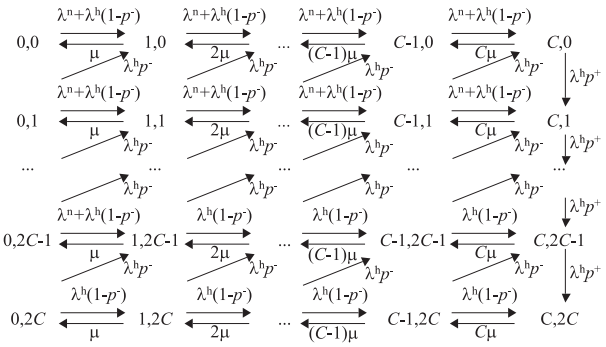

Fig. 2. State diagram of the CTMC in a scenario with two classes.

Table 1. Definition of the scenarios under study

	d_1	d_2	f_1	f_2	$B_1^n(\%)$	$B_2^n(\%)$	$B_r^h(\%)$	λ_r^n	λ_r^h	μ_1	μ_2
A	1	2	0.8	0.2	5	1					
B	1	4	0.8	0.2	5	1					
C	1	2	0.2	0.8	5	1	$0.1B_r^n$	$f_r\lambda$	$0.5\lambda_r^n$	1	3
D	1	2	0.8	0.2	1	2					
E	1	2	0.8	0.2	1	1					

to resort to the indirect adjustment in which case l^n is decreased accordingly. If the QoS objective for s^h is expressed as $B = b/o$, then $p^- = 1/(o-b)$ and $p^+ = 1/b$.

The performance evaluation is carried out for five different scenarios (A, B, C, D and E) that are defined in Table 1, being the QoS parameters B_i expressed as percentage values. The parameters in Table 1 have been selected to explore possible trends in the numerical results, i.e., taking scenario A as a reference, scenario B represents the case where the ratio c_1/c_2 is smaller, scenario C where f_1/f_2 is smaller, scenario D where B_1/B_2 is smaller and scenario E where B_1 and B_2 are equal.

The system capacities when deploying the MGC policy without the adaptive scheme for the five scenarios defined in Table 1, $\{A, B, C, D, E\}$, with $C = 10$ are $\lambda_{max} = \{1.89, 0.40, 1.52, 1.97, 1.74\}$, respectively. Refer to [11] for details on how to determine the system capacity. For all scenarios defined in Table 1 we assume the following prioritization order $\mathbf{s}^* = (s_2^h, s_1^h, s_2^n, s_1^n)$. We evaluate two implementations that differ in the treatment of the LPC, (s_1^n), one in which it is a protected class and one in which it is the BEC.

4.1 Performance under Stationary Traffic

For the two implementations of the adaptive scheme, Table 2 shows the ratio P_i/B_i for the four arrival classes in the five scenarios considered. In all cases, an

Table 2. P_i/B_i when deploying the MGC policy and a stationary load equal to λ_{max}.

a) LPC is a protected class.

P_i/B_i	Scenario				
	A	B	C	D	E
Class 1N	1.004	1.030	1.036	1.841	1.223
Class 2N	0.998	0.992	1.001	0.998	1.007
Class 1H	1.006	0.992	1.002	1.007	0.999
Class 2H	0.848	0.899	0.803	0.988	0.985

b) LPC is the best-effort class.

P_i/B_i	Scenario				
	A	B	C	D	E
Class 1N	0.938	1.404	0.007	2.348	1.857
Class 2N	1.003	1.065	1.000	1.004	0.999
Class 1H	1.007	1.001	1.007	0.999	0.999
Class 2H	0.993	1.006	0.988	0.989	0.999

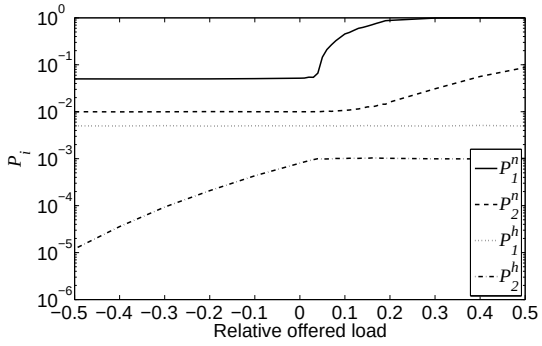

Fig. 3. Variation of P_i with the relative offered load in stationary conditions when the LPC is a protected class.

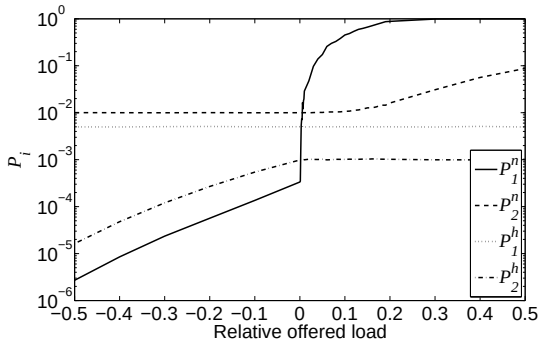

Fig. 4. Variation of P_i with the relative offered load in stationary conditions when the LPC is the BEC.

aggregated load equal to the system capacity (λ_{max}) is offered. Note that the adjustment is much more precise when the LPC is the BEC.

Figures 3 and 4 show the variation of P_i with the relative offered load ($(\lambda - \lambda_{max})/\lambda_{max}$) in scenario C with $C = 10$ resource units. Note that the

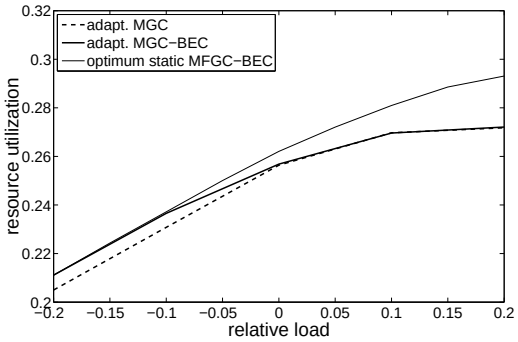

Fig. 5. Resource utilization factor.

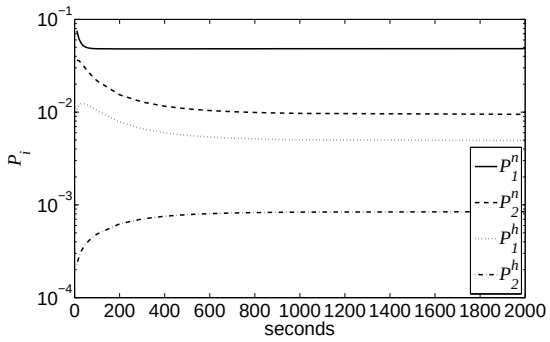

Fig. 6. Transient behavior of blocking probabilities.

adaptive scheme tries to enforce $P_i = B_i$ when possible for the protected classes, and therefore during under load episodes the system is rejecting more requests than strictly required. Nevertheless, some classes (BEC and/or HPC) benefit from this extra capacity. When the LPC is a protected class (Fig. 3), it does not benefit from the capacity surplus during under load episodes and it is the first to be penalized during overload episodes. On the other hand, when the LPC is the BEC (Fig. 4), it benefits during under load episodes and, as before, it is the first to be penalized during overload episodes. In both implementations, note that s_2^n is also penalized when keeping on reducing l_1^n (below zero) would be ineffective to meet the QoS objective of higher priority classes.

In Fig. 5 the resource utilization factor $E[c(\boldsymbol{n})]/C$ of the adaptive scheme in scenario A is compared to the one of an optimum static Multiple Fractional Guard Channel (MFGC) policy, which performance is close to the performance of an optimal policy [11]. The configuration parameters of the MFGC policy have been determined by formulating the problem as a non-linear programming algorithm in which for each λ we search for the values of the configuration parameters that maximize the carried traffic while still meeting the QoS objective. Therefore

we refer to this policy as the *optimum* MFGC policy. We also refer to it as *static* because for each arrival rate studied the optimum configuration parameters are determined and they are unique. On the other hand, the adaptive scheme does not know the arrival rates a priori and therefore it continuously changes the configuration parameters of the MGC policy to meet the QoS objective.

The term "adapt. MGC" refers to the adaptive scheme when the LPC is a protected class, while "adapt. MGC-BEC" refers to the adaptive scheme when the LPC is the BEC. Note that for $\lambda = \lambda_{max}$ the utilization achieved by the MFGC policy is only 2% higher than the utilization achieved by the adaptive scheme. The system capacity achieved by the MFGC policy is 8.6% higher than the one achieved by the MGC and therefore when $\lambda > \lambda_{max}$, it achieves a slight better the resource utilization. Our adaptive scheme can also operate with the MFGC instead of the MGC policy but its operation in not discussed in this paper. Note that both implementations of the adaptive scheme behave identically during overload ($\lambda > \lambda_{max}$).

4.2 Performance under Non-Stationary Traffic

In this section we study the transient regime after a step-type traffic increase from $0.66\lambda_{max}$ to λ_{max} is applied to the system in scenario A when the LPC is a protected class. Before the step increase is applied the system is in the steady state regime.

Figure 6 shows the transient behavior of the blocking probabilities. The convergence period is lower than 1000 s. A comparative performance evaluation of our scheme and the schemes proposed in [2, 3] in a single service scenario shows that the convergence period obtained is 10 to 100 times lower than in both previous proposals [19]. Besides, our scheme shows a non-oscillating behavior unlike [2, 3]. Note that the convergence period will be even shorter when the offered load is above the system capacity thanks to the increase of the rate of probabilistic-adjustment actions, which is an additional advantage of the scheme.

5 Adaptive scheme for elastic flows

Like in other studies of the same nature [1] we focus on the flow level and ignore the detailed mechanisms operating at the packet level. Since our focus is the radio interface at the access network, we assume that each elastic flow is rate limited either by terminal capabilities or because it is bottlenecked at the radio link, i.e. it will receive its fair share of the radio link bandwidth up to a maximum which has a common value for all terminals. For the sake of mathematical tractability we assume that the flow size (given in bytes) is exponentially distributed. While it is commonly accepted that the statistical distribution of Internet document sizes shows a greater variability than the exponential distribution, in the light of the results in [1] the numerical results obtained by using an exponential document size can be considered as a lower bound of performance.

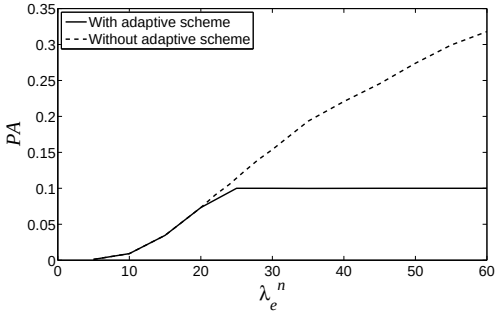

Fig. 7. Abandonment probability of elastic flows with and without adaptive scheme.

We consider the same system model described in Section 2, adding a service with elastic demands, as follows. We denote by s_e^n (s_e^t), the arrival class associated to new (handover) requests of elastic flows. Their requests arrive according to Poisson processes with time-varying rate $\lambda_e^n(t)$ ($\lambda_e^h(t)$). For an elastic session, its cell residence (dwell) time is exponentially distributed with rate μ_e^d. If we denote by d_e the maximum number of resource units an elastic flow uses, and by n_e then number of elastic flows in the system, then we define the flow service rate as μ_e^s when $n_e d_e \leq (C - c(\boldsymbol{n}))$ and $\mu_e^s(C - c(\boldsymbol{n}))/(n_e d_e)$ when $n_e d_e > (C - c(\boldsymbol{n}))$, where $c(\boldsymbol{n})$ is the number of resource units occupied by streaming sessions.

To model the behavior of users we consider the impatience time as an independent exponentially distributed random variable. The impatience rate μ_I is assumed to be inversely proportional to the share of resources allocated to each elastic flow, thus we define $\mu_I = 0$ when $n_e d_e \leq (C - c(\boldsymbol{n}))$ and $\mu_I = K(n_e d_e/(C - c(\boldsymbol{n})))$ when $n_e d_e > (C - c(\boldsymbol{n}))$, where K is a constant. We denote by BA the QoS objective expressed as an upper bound for the abandonment probability, i.e. the ratio between unsuccessfully completed flows and accepted flows, and by PA the actual perceived abandonment probability.

The SAC policy for elastic service is as follows: one configuration parameter is associated with s_e^n, $l_e \in \mathbb{N}$. When there are n_e ongoing elastic flows, a new request is accepted if $n_e + 1 \leq l_e$ and blocked otherwise. Handover requests of elastic flows are always accepted. The adaptive scheme for elastic flows follows a similar approach to that described in Section 3. When the QoS objective for elastic flows can be expressed as $BA = a/b$, where $a, b \in \mathbb{N}$, then we propose to perform a probabilistic adjustment in the following way: i) $\{l_e \leftarrow (l_e - 1)\}$ with probability $1/a$ each time an elastic flow abandons due to impatience; ii) $\{l_e \leftarrow (l_e + 1)\}$ with probability $1/(b - a)$ each time an elastic flows completes its service successfully, i.e. either it finishes or it hands over to another cell. A methodology to infer TCP flow interruption has been proposed in [20].

We evaluate by simulation the performance of the scheme with streaming traffic and elastic flows in scenario A with $C = 10$, considering the LPC as the best-effort class. The streaming traffic offers a constant load equal to the system

capacity ($\lambda = \lambda_{max} = 1.89$). To avoid starvation the system reserves 1 resource unit for elastic traffic. The rest of the parameters that model the elastic traffic are: $\mu_e^s = 2.0$, $\mu_e^d = 2.0$, $K = 0.4$, $\lambda_e^h = 0.5\lambda_e^n$, with a QoS objective of $BA = 0.1$. Figure 7 shows that the adaptive scheme assures the QoS objective. Without the adaptive scheme the abandonment probability increases as the elastic arrival rate increases. This is due to the fact that less resources are available per elastic flow as more elastic flows are accepted in the system, which consequently increases the abandonment rate. Finally, note that high abandonment probabilities bring as a consequence an inefficient use of system resources because resources assigned to flows that are not completed are totally wasted.

6 Conclusions

We developed a novel adaptive reservation scheme that can adapt to non-stationary traffic both in fixed and variable capacity systems. The operation of our scheme is based on simple balance equations which hold for any arrival process and holding time distribution.

Three relevant features of our proposal are: its capability to handle streaming and elastic traffic, its ability to continuously track and adjust the QoS perceived by users and the simplicity of its implementation.

We evaluated the performance of the scheme when handling multiple streaming services and showed that the QoS objective is met with an excellent precision while achieving an oscillation-free convergence period. This confirms that our scheme can handle satisfactorily the non-stationarity of a real traffic. We also evaluated the performance of the scheme when handling elastic flows in a scenario with streaming background traffic. We showed that the scheme guarantees an upper bound for the abandonment probability of elastic flows.

Acknowledgments

This work has been supported by the Spanish Ministry of Education and Science (30%) and by the European Union (FEDER 70%) under projects TSI2005-07520-C03-03, TEC2004-06437-C05-01 and under contract AP-2004-3332, and by the Generalitat of Valencia under contract CTB/PRB/2002/267.

References

1. Bonald, T., Roberts, J.: Congestion at flow level and the impact of user behaviour. Comp. Net. **42** (2003) 521–536
2. Zhang, Y., Liu, D.: An adaptive algorithm for call admission control in wireless networks. In: Proc. IEEE Glob. Comm. Conf. (GLOBECOM'01), San Antonio, TX, (USA) (2001) 3628–3632
3. Wang, X.P., Zheng, J.L., Zeng, W., Zhang, G.D.: A probability-based adaptive algorithm for call admission control in wireless network. In: Proc. Int. Conf. on Comp. Netw. & Mob. Comp. (ICCNMC'03), Shanghai, China (2003) 197–204

4. Yu, O.T.W., Leung, V.C.M.: Adaptive resource allocation for prioritized call admission over an ATM-based wireless PCN. IEEE Jour. on Sel. Areas in Comm. **15**(7) (1997) 1208–1225
5. Ramanathan, P., Sivalingam, K.M., Agrawal, P., Kishore, S.: Dynamic resource allocation schemes during handoff for mobile multimedia wireless networks. IEEE Jour. on Sel. Areas in Comm. **17**(7) (1999) 1270–1283
6. Yu, O., Khanvilkar, S.: Dynamic adaptive QoS provisioning over GPRS wireless mobile links. In: Proc. IEEE Int. Conf. on Comm. (ICC 2002). Volume 2., New York, NY, (USA) (2002) 1100–1104
7. Jeon, W.S., Jeong, D.G.: Call admission control for mobile multimedia communications with traffic asymmetry between uplink and downlink. IEEE Trans. Veh. Tech. **50**(1) (2001) 59–66
8. Wei, Y., Lina, C., Rena, F., Raad, R., Dutkiewiczb, E.: Dynamic handoff scheme in differentiated QoS wireless multimedia networks. Comp. Comm. **27**(10) (2004) 1001–1011
9. Huang, L., Kumar, S., Kuo, C.C.J.: Adaptive resource allocation for multimedia QoS management in wireless networks. IEEE Trans. Veh. Tech. **53**(2) (2004) 547–558
10. Garcia-Roger, D., Domenech-Benlloch, M.J., Martinez-Bauset, J., Pla, V.: Adaptive admission control scheme for multiservice mobile cellular networks. In: Proc. 1st EuroNGI Conf. on Next Gen. Internet Net. Traf. Eng. (NGI), Roma, Italy (2005) 288–295
11. García, D., Martínez, J., Pla, V.: Admission control policies in multiservice cellular networks: Optimum configuration and sensitivity. Lec. Notes in Comp. Sci. **3427** (2005) 121–135
12. Soh, W.S., Kim, H.S.: Dynamic bandwidth reservation in cellular networks using road topology based mobility prediction. In: Proc. 23rd Ann. Joint Conf. IEEE Comp. & Comm. Soc. (INFOCOM). Volume 4., Hong Kong, China (2004) 2766–2777
13. Orlik, P.V., Rappaport, S.S.: On the handoff arrival process in cellular communications. Wi. Net. Journal **7**(2) (2001) 147–157
14. Jabbari, B.: Teletraffic aspects of evolving and next-generation networks. IEEE Pers. Comm. **3**(6) (1996) 4–9
15. Biswas, S., Sengupta, B.: Call admissibility for multirate traffic in wireless ATM networks. In: Proc. 16th Ann. Joint Conf. IEEE Comp. & Comm. Soc. (INFOCOM). Volume 2. (1997) 649–657
16. Evans, J.S., Everitt, D.: Effective bandwidth-based admission control for multiservice CDMA cellular networks. IEEE Trans. Veh. Tech. **48**(1) (1999) 36–46
17. Khan, F., Zeghlache, D.: Effect of cell residence time distribution on the performance of cellular mobile networks. In: Proc. IEEE 47th Veh. Tech. Conf. (VTC'97-Spring), Phoenix, (USA) (1997) 949–953
18. Koo, I., Furuskar, A., Zander, J., Kim, K.: Erlang capacity of multiaccess systems with service-based access selection. IEEE Comm. Lett. **8**(11) (2004) 662–664
19. Garcia-Roger, D., Domenech-Benlloch, M.J., Martinez-Bauset, J., Pla, V.: Comparative evaluation of adaptive trunk reservation schemes for mobile cellular networks. In: Proc. 3rd Int. W. Conf. Perf. Mod. & Eval. Het. Net. (HET-NETs), Ilkley, U.K. (2005)
20. Rossi, D., Mellia, M., Casetti, C.: User patience and the web: a hands-on investigation. In: Proc. IEEE Glob. Tel. Conf. (GLOBECOM). Volume 7. (2003) 4163–4168