Paging Load Congestion Control in Cellular Mobile Networks

Ioannis Z. Koukoutsidis

School of Electrical and Computer Engineering,
National Technical University of Athens,
Zografou 157 73, Greece
koukou@telecom.ntua.gr

Abstract. This paper addresses the requirement for a congestion control mechanism, in order to efficiently handle paging traffic over cellular networks. The standard simultaneous paging approach can easily result in congestion, for relatively small values of the offered load. Instead of differentiating paging capacity based on the incoming load, this approach is more oriented towards a limited-resource system. Considering that a paging channel usually experiences high and low utilization periods, it is proposed to differentiate the paging mechanism according to the processed load. Given medium or even mild overload conditions, several forms of sequential paging substantially decrease blocking probabilities, while presenting good delay behavior. Based on a queueing analysis, we are able to quantitatively estimate potential improvements, and point out the basic points of such an adaptive scheme.

1 Introduction

In mobile communication networks, the need for expanded system capacity grows with the number of users and the amount of information required for a given service. However, the increase in demand always seems to be one step ahead of our capability to satisfy it. In view of this, congestion problems are often inevitable, especially in the wireless channels where capacity is even more scarce. Today, the increased user population densities and emerging bandwidth-consuming technologies make the balance even more unfavorable. Since bandwidth scarceness is a reality, new methods are essential to handle congestion problems.

Apart from the data traffic case, signaling congestion is a significant part of the overall problem. As communication technology becomes more complex, signaling plays a continuously augmenting role in establishing and maintaining connectivity. A large part of the signaling traffic in wireless networks is due to the location management operations [1]. These involve the *location update* and *paging* procedures, which are necessary to track user location in the presence of mobility. Location update is essentially a reporting mechanism, by which a moving subscriber informs network databases of its approximate position within a fixed or dynamic *location area* [2]. On the other hand, paging is a search procedure by which the exact cell where a user currently resides is retrieved. The

I. Niemegeers and S. Heemstra de Groot (Eds.): PWC 2004, LNCS 3260, pp. 39–53, 2004.

paging mechanism involves sending a *page request* (PR) message over a forward channel in all cells where the user is likely to be present, as indicated approximately by the location update procedure. Albeit location update and paging are antagonizing procedures, they work complimentarily to each other to provide a location management service. In absolute terms, location update is a more prolonged and costly procedure, however it is possible to view paging as the fundamental operation for call establishment [3], since, simply stated, the principal and foremost goal of location management is to retrieve the serving base station (BS) of the moving subscriber. Paging can also be considered more fundamental with respect to congestion problems since its largest part is associated with the wireless interface, which suffers from lack of resources.

Paging load control involves the critical issue of handling *mass* paging requests for different users. Large paging loads can cause severe congestion problems in system queues (switches, controllers, transceivers, etc.), which may lead to large delays or blocking of incoming call requests. In this paper, we study the congestion problem in paging channels within a cellular network, in presence of mobility. We focus mainly on the so-called 'control center'[1] of the network which distributes page requests, and the congestion problem that exists subsequently in BS channels. After studying the problem, we proceed to formulate a new paging load control method which dynamically adjusts the paging mechanism depending on the offered load in the system.

The rest of the paper is succinctly organized in two parts. The first part addresses the congestion problem and its associated parameters. The two basic mechanisms, *blanket* and *sequential* paging are discussed and a queueing analysis is presented with finite buffer capacity. Subsequent numerical results clarify the problem characteristics and lay the ground for the formulation of a paging load control method. The second part of the paper is devoted to devise the new method and its implementation characteristics. Finally, the paper ends with a discussion of the most important issues and a recapitulation of the major contribution.

2 Blanket and Sequential Paging: A Congestion Perspective

Simultaneous polling of cells in a location area is currently used to track user position. This procedure has been eloquently named *blanket paging* (BP), since it covers every possible cell where the user might be located at once. However, BP is also associated with high signaling cost and responsible for the majority of congestion problems. The flooding of downlink broadcast channels with paging messages and the apparent redundancy that inheres can lead to blocking of new calls and long delays at queues, especially at peak traffic periods. In all cases, it is possible to calculate the paging channel bandwidth in order to achieve certain quality constraints [4]. Still, it is equally challenging to reduce congestion problems in limited-capacity systems.

[1] In current systems, this is the Mobile Switching Center (MSC).

Almost all research on improved paging techniques has focused on some form of *sequential paging* (SP) [5]. Sequential paging attempts to reduce the existing redundancy by polling locations separately, instead of simultaneously. To minimize the associated paging cost, cells should be queried in order of decreasing location probability [6]. Sequential paging also alleviates the problem of congestion by reducing the mean number of page requests to be handled by a base station. On the downside, the major drawback is the introduced delay in establishing a call connection by performing successive steps, at each step waiting for a certain period until the system perceives if the user resides in the selected cell or not. Bounds on delay lead to a constrained optimization problem, which has been solved efficiently in many essays [5],[6],[7],[8]. The essence of tackling delays lies in forming sets of cells which are queried simultaneously at each step, otherwise called *sequential group paging* (SGP). It can be understood that SGP is an intermediate approach between BP and SP, yielding mean values of total cost C and delay D, so that

$$C_{SP} < C_{SGP} < C_{BP},$$
$$D_{SP} > D_{SGP} > D_{BP}.$$

Efforts to view and understand the queueing aspects of paging were made in [9],[10]. In similar approaches there, base station channels were modelled as $M/M/1$ queues in a system where PRs are distributed by a central control to the appropriate base stations. It was shown that sequential paging distributes the load more evenly to BS queues, which under very high loads can also reduce the overall delay in call establishment. The reduction in delay is due to the fact that PRs experience less time waiting to be served, which benefit outweighs the increase in delay caused by sequential paging. In other words, paging is performed *simultaneously* for *different* subscribers in the *same* time slot in *different* base stations, where of course there is a probability of locating a user. The authors conclude that when paging channels are not heavily loaded, flooding results in the shortest delay; however, when paging channels face congestion, flooding places a high volume of messages on these channels, which results in very high queueing delays. From our perspective, since congestion also entails the notion of delay, there is no clear distinction between the loss of performance attributed to a specific paging mechanism. The objective should be to increase the effective rate of incoming messages, while eliminating symptoms of congestion that lead to a degradation in system performance.

The analytical approaches in [9],[10] are similar and focus on upbringing the issue of delay, assuming infinite capacity queues. The reduction in congestion is only suggested indirectly by the reduction in the effective paging load to be accommodated in each queue. In an effort to quantitatively show the relationship, we adopt in the following a more realistic model with finite capacity queues and blocking. The acquired results are valuable in designing an efficient paging load control method.

3 Queueing Analysis

We assume that page requests are distributed to BSs by a central control, as shown in Fig. 1. Base stations have finite buffers in which to store messages. Without significant loss of generality, a single broadcast channel can be dedicated to each BS. Thus, an $M/M/1/K$ queue is proposed as a representation of the real-life system.

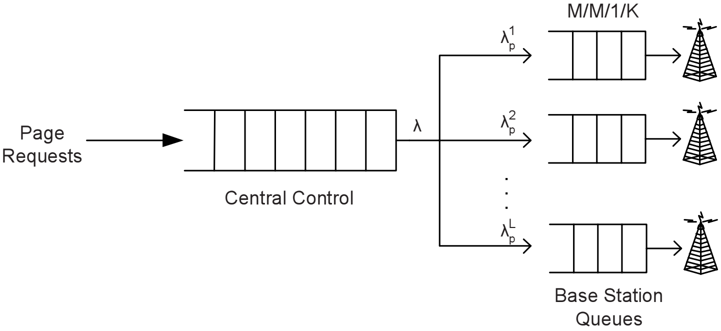

Fig. 1. Queueing system representation of page requests

The following basic notation is used in the analysis:
λ: aggregate PR arrival rate
λ_p: individual PR arrival rate at each BS
μ: service rate at each BS
K: total storage capacity (queue+server)
L: number of BSs in system
S: mean number of searches to locate a mobile
N: mean number of PRs in a system queue
T: mean waiting time in system (average response time)

We progressively complicate the analysis by studying each paging mechanism separately (BP, SP, SGP) and initially admitting a uniform distribution of user location. We aim to derive analytical results for the blocking probability and average response time of a PR. Incoming message requests are assumed to be served in chronological order (FIFO queues).

3.1 Blanket Paging

This is the simplest case; since all BS cells in a system area are polled simultaneously, the aggregate mean call arrival rate is transferred to all BS queues. As the behavior of all queues is identical, it suffices to study the single queue where the user will be found. The load in a single queue is given by $\rho = \lambda/\mu$. From

$M/M/1/K$ queueing analysis [11], the equilibrium distribution for the number of paging messages in the system is

$$\pi_n = \begin{cases} \frac{1-\rho}{1-\rho^{K+1}} \cdot \rho^n, & \text{if } \rho \neq 1 \\ \frac{1}{K+1}, & \text{if } \rho = 1 \end{cases} \tag{1}$$

where $n = 0, 1, \ldots, K$. Substituting $n = K$, we have the blocking probability of a page request, $P_B = \pi_K$. The mean number of PRs in the system is

$$N = \sum_{n=0}^{K} n \cdot \pi_n = \sum_{n=0}^{K} n \cdot \frac{1-\rho}{1-\rho^{K+1}} \cdot \rho^n = \frac{1-\rho}{1-\rho^{K+1}} \cdot \sum_{n=0}^{K} n \cdot \rho^n = \\ = \frac{\rho[1-(K+1)\rho^K + K \cdot \rho^{K+1}]}{(1-\rho) \cdot (1-\rho^{K+1})} \tag{2}$$

if $\rho \neq 1$ and

$$N = \sum_{n=0}^{K} n \cdot \frac{1}{K+1} = \cdots = \frac{K}{2} \tag{3}$$

if $\rho = 1$. To find the mean waiting time in system, which is also the average response time of a PR, we may apply Little's law, considering the actual rate of PRs that are admitted into the system. The fraction of arrivals who are served is

$$\lambda_s = \lambda(1 - \pi_K) \tag{4}$$

Hence, the mean response time for each served PR is

$$T = \frac{N}{\lambda(1 - \pi_K)} \tag{5}$$

3.2 Sequential Paging

Uniform Case. We first consider the uniform case where a user has an equal probability to be located at each cell at the time of a call arrival, denoted as $p_i = 1/L$ $(i = 1, 2, \ldots, L)$. A uniform distribution produces the highest average number of attempts and hence provides a lower bound in the algorithm performance [6]. Assuming sequential steps are selected at random by the central control, each BS will initially receive exactly $1/L_{th}$ of the total load. This must be multiplied by the mean number of searches to locate a mobile; hence, the individual arrival rate at each BS channel is

$$\lambda_p = \lambda \cdot \frac{S}{L} \tag{6}$$

Given that $S < L$, we have that $\lambda_p < \lambda$. Hence as anticipated, the individual arrival rate of incoming PRs is always less in the SP case. For a uniform distribution, the mean number of paging attempts equals $S = \frac{L+1}{2}$.

Substituting $\rho = \frac{\lambda_p}{\mu}$ in Eqs. (1) and (2), we get the blocking probability and mean number of PRs at a single queue. However, considering that the system will make on average S attempts until the requested user is found, the total blocking probability is

$$P_B = S \cdot \pi_K \tag{7}$$

The average response time of a PR is the sum of the individual times at each queue. Hence, in an analogous manner, we have that

$$T = S \cdot \frac{N}{\lambda_p(1-\pi_K)} \tag{8}$$

Conditioning on the event that a mobile user will eventually be found, the mean number of searches is unaffected and equals $\frac{L+1}{2}$. However, due to blocking, the mean number of searches may be reduced to:

$$S = (1-\pi_K) + (1-p_1)(1-\pi_K)^2 + \cdots + (1-p_1-\cdots-p_{L-1})(1-\pi_K)^L \tag{9}$$

where p_i, $i = 1, \ldots, L$ is the location distribution and each of the summands corresponds to the probability of making each successive paging step. In the case of a uniform distribution, we have

$$S = \sum_{i=1}^{L} \frac{L-(i-1)}{L}(1-\pi_K)^i \tag{10}$$

Non-Uniform Case. Assume now that each BS has a different load, as a result of a non-uniform location distribution. Then the arrival rate at each paging channel can be represented as

$$\lambda_p^i = \phi_i \cdot \lambda$$

($i = 1, \ldots, L$), where ϕ_i is the fraction of the load distributed at each queue, based on a sequential polling mechanism and the underlying location distribution, $\{p_i\}$. Then we should calculate the blocking probability and delay separately at each queue. Mean values for the parameters discussed can then be produced by taking the average, weighted by the probability of a PR being processed at each queue. The mean number of paging attempts is in general given by $S = \sum_{i=1}^{L} i \cdot p_i$ and the mean number of searches w.r.t. blocking by (9).

The detailed analysis of how the underlying location distribution affects the individual incoming rates λ_p^i is an arduous task and is bypassed here. With much less complication, the behavior of the system in the non-uniform case can be shown by admitting an identical mean number of searches for all users, for which holds $S < \frac{L+1}{2}$. We also assume that polled locations are selected at random, so that the exact same load is delivered at each queue. In so doing, the previous sequential paging analysis can be applied, with parameters modified by S, $\{p_i\}$.

3.3 Sequential Group Paging

Uniform Case. Let us consider N_G polling groups, each group containing $\frac{L}{N_G} = \theta$ cells. Assume for simplicity that θ is an integer number[2] and each group is chosen randomly, so that every BS receives the exact same load. Similarly to the previous analysis, the arrival rate at each paging channel is

$$\lambda_p = \lambda \cdot \frac{S}{N_G} \tag{11}$$

where $S = \frac{N_G+1}{2}$. To find the mean response time of a PR, we have to consider the ensemble of all BS queues. Let $N_1, N_2, \ldots, N_\theta$ denote the number of PRs at the cell BSs of any given group. For $\rho \neq 1$, we have that

$$Pr\{N_i = n\} = \frac{1-\rho}{1-\rho^{K+1}}\rho^n$$

where $i = 1, 2, \ldots, \theta,\ n = 0, 1, \ldots, K$. Hence the cumulative distribution function (cdf) is

$$Pr\{N_i \leq n\} = \frac{1-\rho^{n+1}}{1-\rho^{K+1}} \tag{12}$$

Similarly, for $\rho = 1$ the cdf becomes

$$Pr\{N_i \leq n\} = \frac{n+1}{K+1} \tag{13}$$

At each paging stage j $(j = 1, 2, \ldots, N_G)$, all BSs in the j_{th} group are paged to find the mobile. Therefore, the delay in the j_{th} stage is directly related to the maximum queue length amongst the θ BSs that page in that group. Let N_m denote the maximum number of messages in any of the θ queues at any time instance. The cdf of N_m occurs as follows

$$Pr\{N_m \leq n\} = [Pr\{N_i \leq n\}]^\theta = \begin{cases} \left[\frac{1-\rho^{n+1}}{1-\rho^{K+1}}\right]^\theta, & \text{if } \rho \neq 1 \\ \left[\frac{n+1}{K+1}\right]^\theta, & \text{if } \rho = 1 \end{cases} \tag{14}$$

To find the mean value of N_m, we have that

$$E\left[N_m\right] = \sum_{n=0}^{K} Pr\{N_m > n\} = \begin{cases} \sum_{n=0}^{K}\left[1 - (\frac{1-\rho^{n+1}}{1-\rho^{K+1}})^\theta\right], & \text{if } \rho \neq 1 \\ \sum_{n=0}^{K}\left[1 - (\frac{n+1}{K+1})^\theta\right], & \text{if } \rho = 1 \end{cases} \tag{15}$$

The total blocking probability is calculated by $P_B = S \cdot \pi_K$, as the blocking within other BSs in the same group is irrelevant when the mobile is not found

[2] In the case where L is not an integer multiple of N_G, $\theta_0 = \lfloor \frac{L}{N_G} \rfloor$ cells should be assigned to the first $N_G - (L - \theta_0 \cdot N_G)$ groups and $(\theta_0 + 1)$ to the remaining $(L - \theta_0 \cdot N_G)$, in an optimal partition [8]. The analysis must be differentiated, as in the non-uniform case below.

there. A problem is encountered when calculating the mean response time; here, the average delay in an unsuccessful step is generally different from a successful one, since in the first case we have to consider the longest delay of all θ queues. If the user is eventually found after S steps, the total average delay can be calculated as

$$T = \frac{N}{\mu} + (S-1)(\frac{E[N_m]}{\mu}) \tag{16}$$

If not, the mean response time should be given by

$$T = S \cdot \frac{E[N_m]}{\mu} \tag{17}$$

The mean number of searches w.r.t blocking can be calculated similar to (9):

$$S = (1-\pi_K^\theta) + (1-p_1)(1-\pi_K^\theta)^2 + \cdots + (1-p_1-\cdots-p_{N_G-1})(1-\pi_K^\theta)^{N_G} \tag{18}$$

where π_K^θ is the combined probability of blocking in the whole group and $\{p_j\}$, $j=1,\ldots,N_G$ is the total location probability in a group of cells. However, after S searches, there is no way of knowing whether a user has been found or not. Due to the non-independence of successive paging steps, we note that there generally exists no closed-form solution for T.

Non-Uniform Case. This case is very complicated and outside the limited scope of this paper. We outline the solution as following. The delay and blocking probability are computed based on the BSs in each group. Here, the number and identity of BSs in a group are of primary importance. These should be calculated based on a system partition, as close to the optimal as possible. The total parameters must be calculated as weighted averages on all groups. The random selection hypothesis could also apply here, in order to simplify the solution.

Remark. It should be added that in real-life systems, there exists a timeout interval W, during which the system awaits the mobile unit's response [9]. For simplicity, we have incorporated the timeout into the service time, assuming the time for the system to perceive user presence is the same, whether he responds or not. For more accurateness the extra period $(W - \frac{1}{\mu})$ should be added to the SP and SGP schemes, for all $(S-1)$ unsuccessful attempts.

4 Numerical Results

In this section, numerical results are presented that show the comparative behavior of blanket and various forms of sequential paging. It is appropriate to consider a 'system scale', whereupon a larger system also has a higher storage capacity. In the case of a small system we have $L = 20$ locations and buffer size $K = 30$, whereas in the larger system we specify that $L = 40$ and $K = 100$.

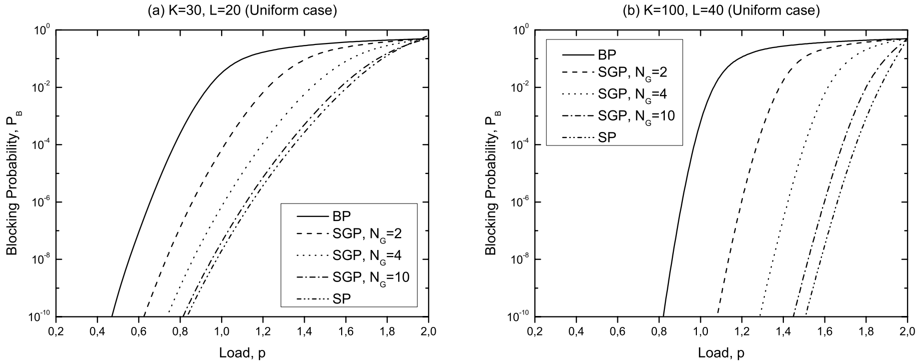

Fig. 2. Blocking probability vs. aggregate load for blanket and sequential paging cases (a) $K = 30$, $L = 20$, (b) $K = 100$, $L = 40$

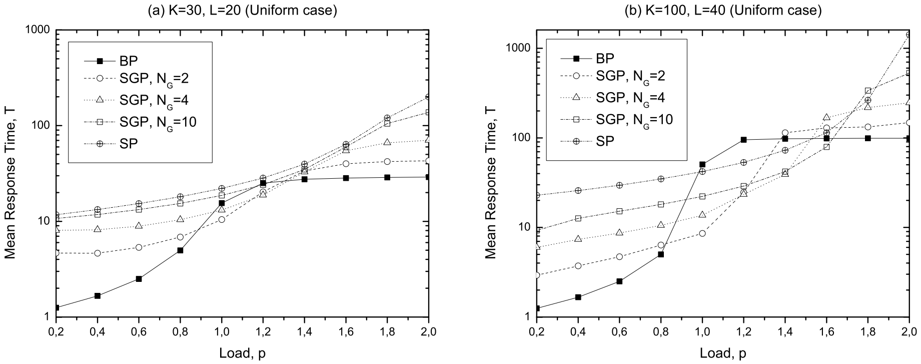

Fig. 3. Mean response time of a served PR vs. aggregate load for blanket and sequential paging cases (a) $K = 30$, $L = 20$, (b) $K = 100$, $L = 40$

Apart from the load parameter ρ, the aggregate incoming call rate determines the waiting time in system. For simplification, the service service rate at BS channels is taken equal to unity, so that $\lambda = \rho$ in all test cases.

Figures 2(a),(b) present the blocking probabilities under a uniform location distribution, for various values of the load parameter ρ. The y-axis' are drawn on a logarithmic scale to efficiently represent blocking behavior. We note that generally in communications systems, the fraction of blocked calls should be kept less than 10^{-2}. The blocking probability is always less in the sequential cases, despite the fact that the system might make numerous attempts to locate the mobile. This implies much less congestion in system queues, as a result of the distribution of the aggregate load. Indeed, blocking decreases when increasing the number of sequential steps, with the exception of very high loads, where the combination of increased congestion and multiple paging attempts can increase the percentage of rejected calls. Hence, for normal and increased loads, sequential paging in its various forms can achieve a notable decrease in blocking probability over the flooding scenario. The same behavior is observed for a larger system, where due to increased capacity congestion appears at higher loads. Also the higher number of locations in case (b) expands the search in a larger area, which yields a better relative improvement for sequential paging. Finally, it is worth adding that a more abrupt increase of blocking probabilities always occurs in the case of increased queueing capacity.

Clearly, if we had been indifferent with respect to the delay introduced by successive paging steps, SP would have been the preferred strategy, since almost always it achieves better blocking performance. However, numerous paging stages in such a transparent system would lead to a result identical to that of congestion, since delay is also excessive before establishing a call. What's more, in the event of excessive delay the calling party would normally back off, and thus paging should be cancelled, resulting –similarly to blocking– in uncompleted calls.

Fig. 3 shows respective results in terms of the mean response time of a PR. The y-axis' are again logarithmic to better discern the growth of the curves. In order to avoid the pitfall of producing smaller response times for more congested channels, results show the mean response time of a *served* PR, i.e. we assume that eventually the requested user is found. As anticipated, the introduction of more paging steps generally increases delays. However, when congestion starts building up, the waiting time in a queue might have the adverse effect on total response times. In fact, for a specific range of load values, a decrease in the total response time can be achieved if we adopt a sequential paging strategy. This is more evident in the case of a larger system, where for ρ values approximately in the range $(1 \leq \rho \leq 1.2)$, BP performs even worse than the extreme case of sequentially polling each cell. However, for very high load values, the situation is 'back to normal', with sequential paging suffering both from congestion and very large delays. This characteristic behavior is encountered in all test cases.

It is worth noting that a different number of paging groups will be optimal for different load values. For example, in Fig. 3(b) choosing $N_G = 2$ is optimal for $\rho = 1$, yet it performs worst when $\rho = 1.4$. Finally, it is noted that if we had a higher queueing capacity, together with a small number of locations, the delay performance of SP would be improved accordingly to the confinement of the search space.

The effects of a non-uniform location distribution are depicted in Fig. 4. Here, no partitioning is assumed and cells are polled sequentially. System parameters are set to $K = 30$, $L = 20$. Non-uniform cases (SP-NU) are portrayed by reducing the mean number of paging attempts to locate a mobile, according to an underlying location distribution. Results are also compared against the BP scheme.

For illustrative purposes, let us define a parameter $a = \frac{S}{L}$, where $\frac{1}{L} \leq a \leq \frac{L+1}{2L}$, depending on the mean number of searches. This can be called the 'reciprocal of search concentration', as for small values of a there exists a large concentration of location probabilities and vice-versa. Then in general, SP can sustain loads $\frac{1}{a}$ times greater than the BP case, which is extremely important in congestion situations. The effects of this are transferred to the blocking probability curves of Fig. 4(a). For more concentrated distributions the delay performance

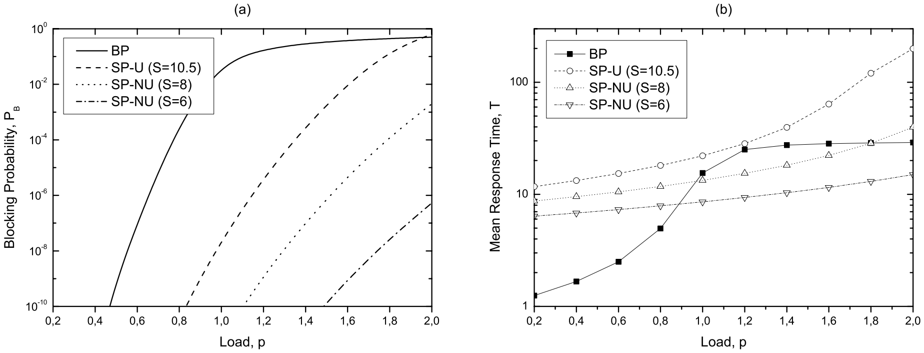

Fig. 4. Blocking probabilities (a) and mean response times (b) vs. increasing values of the aggregate load, in a general case with uniform and non-uniform location distributions. Results refer to the BP and SP cases in a system where $K = 30$, $L = 20$

is improved as well, as shown in Fig. 4(b). Despite the fact that all cells were polled sequentially in this example, the concentration of search can give smaller response times than the simultaneous paging approach, for a specific range of load values.

5 Paging Load Control

Having seen the basic dimensions of the problem, it is concluded that neither form of paging can efficiently handle the flow of messages in a wireless network. Each strategy performs better at a specific range of load values. Therefore a control mechanism should be envisaged, especially in view of a limited resource system. The goal is to increase the rate of *served* paging messages, while withholding the delay beneath acceptable levels. From an analytical view, this is a difficult optimization problem, since blocking and delay may be contradictory constraints. Also, the behavior of the system can change with unpredictable load variances.

Here, an efficient control method is proposed for handling the overall problem. Our proposal is outlined as follows. In a paging system queue, there are usually alternating intervals of low and high activity periods. In order to reduce congestion, the system (i.e. the MSC) can monitor, or receive as feedback, the state of its BS paging channels. The state depends on the offered paging load and can be represented in multiple scales. Depending on the queue load state, the system differentiates its paging policy: under low load conditions, the system does not face congestion problems; so it can increase throughput by simultaneously polling all cells. However, in high load situations, it is best that polling is done sequentially, so as to distribute PRs among BSs and reduce the average load. The case of very high loads is an extreme situation, since neither sequential paging performs well. Therefore, it should be treated differently. Essential methods are discussed in [12], for example by applying admission control policies or allocating more available channels for paging, which might be necessary in such an utmost case. Fig. 5 shows a schematic representation of different paging load regions and their associated paging mechanism.

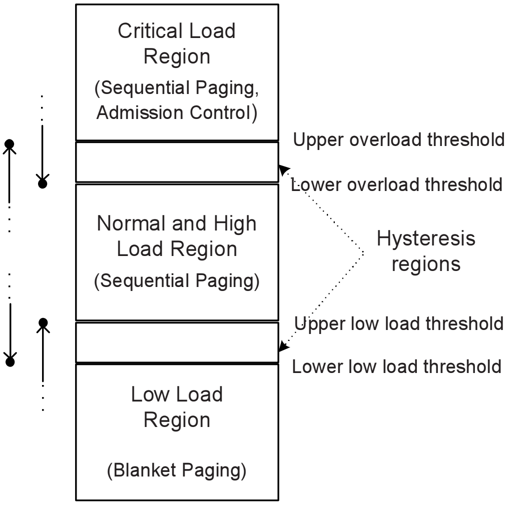

Fig. 5. Paging congestion control regions with upper and lower load thresholds

Control triggers must be generated periodically by the system in order to check congestion conditions. This could be done by sending polling control messages and waiting for the system's response, or by the BSs automatically reporting congestion conditions, in a feedback fashion [13]. Alternative ways of load sampling are counting the number of messages in the queue, the number of idle periods, or estimating blocking probabilities. The frequency of load sampling can also be programmed in advance. Generally, congestion detection should be performed at a smaller time scale for increasing paging traffic; besides this being the most important region, system behavior can change more abruptly for such values. In addition, overloading usually occurs in specific periods of a day, special events or occasions (e.g. New Year's eve, big athletic events, natural disasters, etc.). At those times and places, triggering should be performed more frequently to efficiently control congestion levels.

The choice of load thresholds is the most critical part of a congestion control algorithm. Systems should be designed so that even at peak throughput, the queueing lengths and delays are kept within predefined bounds. Attention should be drawn to the fact that we're dealing with a real-time system, with often rapid and unpredictable changes in behavior. The choice of the threshold level and the associated paging mechanism should depend on the number of channels and buffer capacities, as well as the location distribution. In the case of $M/M/1$ queues, the authors in [10] have defined a *crossover load*, above which sequential paging delivers better delay performance. However, in the case of finite capacities two such roots of a higher-degree polynomial equation are required, as indicated by the numerical results presented in Section 4. This is very difficult to solve analytically or numerically. But even if we can end up in approximate solutions, delay should not be the sole criterion to differentiate the paging mechanism, as we would ignore the large benefits acquired by reducing blocking probabilities, even with a sacrifice in delay. Hence, the choice of threshold values should be a compromise between blocking and delay. Based on the evaluated test cases, a rough estimate is as follows: BP should be applied when the offered load is $\rho < 0.8$. In the range $0.8 < \rho < 1.5$, some form of sequential group paging should be applied; it is noted that SP should generally be avoided, except in the case of very small number of locations or very concentrated distributions. Above $\rho = 1.5$ is generally a critical overload region.

In order to better cope with the real time environment, hysteresis regions are proposed, as shown in Fig. 5. Marked-end arrows show the points of changing polling behavior upon entering a new load region, depending on the previous queue state. In general, hysteresis is essential to improve stability and robustness of the system. A hysteresis region leaves an error margin and prevents frequent changes in the paging mechanism. This should also be designed to give the congestion control software some time to initiate action and bring the resource occupancy down. If the congestion trigger is delayed, the system might reach pretty close to 100% occupancy, thus leading to terrible service to all users in the system.

Based on the results in Section 4, the benefits of the outlined control algorithm are straightforward. A large scale simulation with realistic traffic conditions should be further conducted in order to study the statistics of load variance. This would enable us to explicitly define upper and lower thresholds for a specific system implementation.

6 Ending Notes

In the last section of the paper, several important issues are left for discussion. First, despite the fact that given specific parameters of the problem, sequential paging can decrease the mean response time of a PR, using SP or SGP as a means to tackle the delay is not a sensible approach. Sequential paging should be used primarily to relieve blocking, even with a small increase in delay. The definition of a *utility function* which compromises blocking and delay and would be an appropriate index of performance remains under investigation.

Also, sequential paging might not always solve congestion hardships. Instead, it might just transpose the problem; essentially, the execution of the paging procedure in steps postpones the query of certain cells at later time epochs. So we might just transfer the problem, if at later time epochs arrives a significant number of page requests for other users at the same base station. This is worsened by the fact that PRs for different subscribers can arrive irregularly, or in the case of large inhomogeneities in paging traffic load amongst different BSs. Such characteristics can lead to hysteresis-type congestion problems, as outlined in

[14]. In view of this, a realistic simulation study becomes more crucial to view the actual benefits of such a scheme.

In addition, a detailed view of the system provides more insight. As it has been noted, the reduction in delay produced by sequential paging is due to paging different subscribers at the same time slot, in different BSs. Therefore, in a microscopic look, the problem becomes one of concurrently serving different page requests in a slotted system. As it was shown in [15], the optimal solution to the concurrent search problem, when searching for n terminals, is to solve n independent sequential paging problems. However, practically this is not possible, if the system does not have multiple (theoretically $O(n)$) paging channels or if it does not have the capability to send more than one message in a single slot. If we were to increase the effective throughput with no concern for delays, the PR with the highest probability of locating a mobile should be processed at each slot of the paging channel. However, in order to combat delays, a priority metric should be introduced, both in terms of oldness in the system and location probability [15]. The issues of conflicts and synchronization are more critical at slot level here. On the other hand, our proposal aims at a higher level control which can be more applicable in a real-life system.

It is equally remarked that the sequence in which jobs are served once they are in the queue does not generally affect mean performance parameters [11]. Nevertheless, it does affect the mean response time of a *specific* page request. Here we have considered a FCFS strategy, which is also the most realistic case. It is not considered that other service disciplines are widely applicable in the context of study. However, the analysis can be extended to cover such issues, especially in the direction of prioritized page requests.

Finally, the analysis assumed an error-free environment. In practice, paging errors are a cause of re-propagation of messages and thus a cause of congestion. Paging strategies in the presence of transmission errors have been previously presented in [16],[17]. Adjusting the polling mechanism to an erroneous environment is necessary, however the essentials of the higher-level control algorithm remain unaffected.

Returning to the context of this work, a more precise statement regarding the expected gain of the proposed adaptive method is closely tied with the detailed association of load ranges with paging mechanisms and must be further researched. Of equivalent interest are further analytical performance evaluation results for inhomogeneous user location distributions and different service or storage capacities at each cell. In this respect, a more powerful modeling approach by means of a network of queues is deemed necessary.

In conclusion, despite the variety of open issues for analysis, as well as implementation, it is evident that if the control mechanism –as outlined above– is properly applied, an improved system performance can easily be attained.

References

1. Pollini, G.P., Meier-Hellstern, K.S., Goodman, D.J.: Signaling traffic volume generated by mobile and personal communications. IEEE Comm. Mag. **33** (1995) 60–65
2. Akyildiz, I.F., McNair, J., Ho, J.S.M., Uzunalioǧlu, H., Wang, W.: Mobility management in next-generation wireless systems. Proc. of the IEEE **87** (1999) 1347–1384
3. Bhattacharya, A., Das, S.K.: Lezi-update: An information-theoretic approach to track mobile users in PCS networks. In Proc. ACM/IEEE Mobicom '99 (1999) 1–12
4. Saraydar, C.U., Rose, C.: Minimizing the paging channel bandwidth for cellular traffic. In Proc. ICUPC '96 (1996)
5. Krishnamachari, B., Gau, R.-H., Wicker, S.B., Haas, Z.J.: Optimal sequential paging in cellular networks. Wireless Networks **10** (2004) 121–131
6. Rose, C., Yates, R.: Minimizing the average cost of paging under delay constraints. Wireless Networks **1** (1995) 211–219
7. Abutaleb, A., Li, V.O.K.: Paging strategy optimization in personal communication systems. Wireless Networks **3** (1997) 195–204
8. Wang, W., Akyildiz, I.F., Stüber, G.L.: An optimal paging scheme for minimizing signaling costs under delay bounds. IEEE Comm. Letters **5** (2001) 43–45
9. Goodman, D.J., Krishnan, P., Sugla, B.: Minimizing queueing delays and number of messages in mobile phone location. Mobile Networks and Applications **1** (1996) 39–48
10. Rose, C., Yates, R.: Ensemble polling strategies for increased paging capacity in mobile communication networks. Wireless Networks **3** (1997) 159–167
11. Bose, S.K.: An Introduction to Queueing Systems. Kluwer Academic/Plenum Publishers (2000)
12. Keshav, S.: Congestion Control in Computer Networks. PhD Thesis, UC Berkeley TR-654 (1991)
13. Shenker, S.: A theoretical analysis of feedback flow control. In Proc. ACM Sigcomm '90 (1990)
14. Ackerley, R.G.: Hysteresis-type behavior in networks with extensive overflow. BT Tech. Journal **5** (1987) 42–50
15. Gau, R.-H., Haas, Z.J.: Concurrent search of mobile users in cellular networks. IEEE/ACM Trans. on Networking **12** (2004) 117–130
16. Awduche, D.O., Ganz, A., Gaylord, A.: An optimal search strategy for mobile stations in wireless networks. In Proc. ICUPC '96 (1996)
17. Verkama, M.: Optimal Paging—A search theory approach. In Proc. ICUPC '96 (1996)