Wireless Networks 10, 121–131, 2004
© 2004 Kluwer Academic Publishers. Manufactured in The Netherlands.

Optimal Sequential Paging in Cellular Wireless Networks

BHASKAR KRISHNAMACHARI *
Department of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA

RUNG-HUNG GAU
Institute of Communications Engineering, National Sun Yat-Sen University, Kaoshiung, Taiwan, ROC

STEPHEN B. WICKER and ZYGMUNT J. HAAS
School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853, USA

Abstract. In a high-capacity cellular network with limited spectral resources, it is desirable to minimize the radio bandwidth costs associated with paging when locating mobile users. Sequential paging, in which cells in the coverage area are partitioned into groups and paged in a non-increasing order of user location probabilities, permits a reduction in the average radio costs of paging at the expense of greater delay in locating the users. We present a polynomial time algorithm for minimizing paging cost under the average delay constraint, a problem that has previously been considered intractable. We show the conditions under which cluster paging, a simple heuristic technique proposed for use with dynamic location update schemes, is optimal. We also present analytical results on the average delay and paging cost obtained with sequential paging, including tight bounds.

Keywords: cellular wireless networks, mobility management, sequential paging

1. Introduction

When a call arrives for a mobile user in a cellular network, it is necessary to determine the location of this user in order to route the call appropriately. In the earliest cellular systems, this was accomplished by *paging* all the cells in the network [5]. Such an approach incurs a significant cost in radio bandwidth utilization and is cost-effective only in small networks. The second generation networks introduced the notion of *location updates*, whereby the system is divided into a number of location areas (LA's), and the mobile unit notifies the network when it moves from one location area to another [18]. Upon arrival of a call, all the cells within the user's current location area are paged.

A number of location updating schemes that have been proposed are based on improved LA partitioning techniques [16,19,25,33]. Schemes based on the selection of designated reporting cells can be found in [6,7,13]. The use of implicit location registration information obtained during transactions between the network and the mobile unit is discussed in [30]. Research has also been directed to dynamic location update schemes such as distance-based [2,3,8,17,24], movement-based [4,8], velocity-based [21,32], and time-based [8,29] strategies. A predictive distance-based mobile tracking scheme is presented in [22], and a novel information theoretic approach to user location, based on the Lempel–Ziv family of source compression algorithms, is proposed in [9].

There is a tradeoff between the frequency of location updates and paging costs: if location updates are frequent, there is less uncertainty about the user's position and fewer cells need to be paged; on the other hand, if the location updates are infrequent, the cost of paging increases. Still, since paging is necessary if the update scheme leaves any uncertainty at all in the mobile's exact location, it is possible to view paging as a fundamental operation. As pointed out in [9], however, "the majority of the research on location management has actually focused on update schemes, assuming some obvious version of paging algorithm."

A simple two-stage strategy for improving paging costs in the LA-based approaches is described in [23]. Paging is initially performed in cells where information about recent interactions with the mobile indicate that the user is most likely to be present. If the user is not found in these cells, then the remaining cells in the location area are paged. Selective or cluster paging strategies, such as those used in [3,4,11,17,26], assume that the mobile's last known position and its surroundings constitute the most probable current location. The direction of the mobile's motion is taken into account in [21] and [10], with the cells in the direction of the mobile's motion yielding a higher probability of user location. The use of user profiles, through probabilistic information gathered either from the user or the billing database, is considered in [27] as a means for reducing paging costs. A similar use of collected user mobility statistics in the form of multi-dimensional histograms is described in [31].

The notion that the policy of "paging more probable cells first" reduces the cost of paging is made rigorous in [28]. It is proved that sequential paging schemes, in which groups of cells are paged in non-increasing order of user location probabilities, are optimal in that they minimize the average number of cells paged per call arrival. There is an implicit tradeoff in

* Corresponding author.
E-mail: bkrishna@usc.edu

these schemes, however, between the delay in finding a mobile user and the corresponding paging cost. At one extreme, when a call arrives for the user, all base stations in that LA send out paging messages and the mobile will respond in one round no matter where it is. If the LA is large and a great number of cells are paged, the radio costs are high. At the other extreme, only one base station sends a paging message in each round. Although this minimizes the paging cost, in the worst case this would require as many rounds as the number of cells in the LA – a potentially unacceptable delay.

It is shown in [28] that if the user location probabilities are known, dynamic programming may be used to solve (a) the problem of minimizing the average cost of paging under maximum delay constraint, and (b) the problem of minimizing the weighted mean of paging cost and of average delay. We extend those results by specifying algorithms for these problems and analyzing their worst case running time and space requirements. Also discussed in [28] is the problem of minimizing the paging cost subject to a constraint on the average delay. The authors state that this problem is *not* amenable to solution via dynamic programming, since the total cost is not additive. They then proceed to provide an approximate solution via Lagrange multipliers using a continuous formulation. This problem of minimizing the average paging cost subject to the average delay constraint is also discussed in [1], where again the authors suggest that the problem may be intractable. We present an algorithm which solves this problem *exactly* in polynomial time with respect to the total number of cells.

We also present results on the performance of sequential paging schemes under two sets of assumptions: (a) uniform user location probabilities, and (b) cluster paging with geometric zone location probabilities. The first case, when the user is equally likely to be in any cell in the location area, is useful because it provides tight bounds on the performance of sequential paging with an arbitrary location distribution. Cluster paging is a special form of sequential paging that arises in the context of dynamic location updating schemes. In this case, we show necessary conditions for the optimality of cluster paging, and provide expressions for the average paging cost and paging delay when the probability of user location decreases geometrically with distance from the center cell, such as the case when a mobile user provides location updates relatively frequently.

The rest of the paper is organized as follows: section 2 introduces some of the notation and preliminary definitions. Section 3 presents two important problem formulations, discusses their structure, and presents algorithms to solve them. The time and space requirements for these algorithms are also provided. The performance of sequential paging schemes under specific user location probability distributions is discussed in section 4. Section 5 presents concluding comments. All proofs have been deferred to the appendix.

2. Definitions and notation

The *location area* is the set of n cells $C = \{1, 2, \dots, n\}$ such that the mobile user is guaranteed to be in one of these cells at the time of a call arrival[1]. We assume that for each user, the probability of the user being present in a cell can be estimated for each of these n cells at the time of a call arrival. Let π_i be the probability that the user is located in the ith cell. We assume, without loss of generality, that the cells are numbered in non-increasing order of user location probabilities, i.e. $i < j \Rightarrow \pi_i \geqslant \pi_j$.

A *sequential paging* scheme is one where the cells in a location area are partitioned into indexed groups referred to as *paging zones* on the basis of the cell-wise user location probabilities. Let $Z_1, Z_2, \dots, Z_w$ be the w-partition of the set C (i.e. a partition of C into w groups), where each Z_i is non-empty and corresponds to a distinct paging zone. When a call arrives for a user, the cells in paging zone Z_1 are paged simultaneously in the first round, then if the user is not found in Z_1, all the cells in paging zone Z_2 are paged, and so on. Let the number of cells in the ith paging zone be denoted by $n_i = |Z_i|$, and let p_i be the corresponding *zone location probability* of the user:

$$p_i = \sum_{j \in Z_i} \pi_j. \tag{1}$$

We say that $Z_1, Z_2, \dots, Z_w$ is a *non-increasingly ordered partition* of C if for all $i \in Z_k$ and $j \in Z_l$ such that $k \leqslant l$, it is true that $\pi_i \geqslant \pi_j$. Note that if we have a non-increasingly ordered partition, then each paging zone will consist of contiguously numbered cells. Thus, $Z_1 = \{1, 2, \dots, n_1\}$, $Z_2 = \{n_1 + 1, \dots, n_1 + n_2\}$, and so on.

We now have the following observations:

- The total number of paging zones into which the location area is partitioned, w, represents the worst case delay in locating the mobile user.
- The average paging delay (number of paging rounds) in locating a mobile, $\overline{D}$, can be expressed as follows:

$$\overline{D} = \sum_{i=1}^{w} i p_i. \tag{2}$$

- The average cost of paging (number of cells paged per call arrival), $\overline{L}$, can be expressed as follows:

$$\overline{L} = \sum_{i=1}^{w} \left(\sum_{j=1}^{i} n_j \right) p_i. \tag{3}$$

An example illustrating the notation and definitions

(1) $n = 5$ cells in location area, $C = \{1, 2, 3, 4, 5\}$.

(2) Cell-wise user location probabilities:

i	1	2	3	4	5
π_i	0.3	0.3	0.25	0.1	0.05

(3) Number of paging zones $w = 2$.

[1] This is a somewhat general definition of location area. It is applicable even for dynamic location updating schemes, since we can always provide an upper bound on how far the user could have traveled since the last location update.

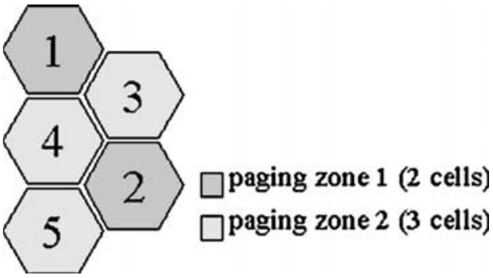

Figure 1. Example location area and paging zones.

(4) Paging zones: $Z_1 = \{1, 2\}$, $Z_2 = \{3, 4, 5\}$.

(5) Number of cells in each paging zone: $n_1 = |Z_1| = 2$, $n_2 = |Z_2| = 3$.

(6) Zone location probabilities: $p_1 = 0.3 + 0.3 = 0.6$, $p_2 = 0.25 + 0.1 + 0.05 = 0.4$.

(7) Average paging delay: $\overline{D} = 1p_1 + 2p_2 = 1 \times 0.6 + 2 \times 0.4 = 1.4$.

(8) Average paging cost: $\overline{L} = n_1 p_1 + (n_1 + n_2)p_2 = 2 \times 0.6 + 5 \times 0.4 = 3.2$.

These definitions and notation are reflected in figure 1. Note that cells that belong to the same paging zone need not be adjacent to one another geographically. As per our assumptions, the cells are numbered in non-decreasing order of user location probabilities. The first paging zone consists of the two most probable cells, and the second paging zone consists of the remaining three cells. When a call arrives, cells 1 and 2 are paged first, and if the user is not located in those two cells, then cells 3, 4, and 5 are paged. For this partitioning of the location area, the worst case delay is 2 rounds, the average paging delay is 1.4 rounds, and the average paging cost is 3.2 cells paged/call arrival.

3. Optimization of sequential paging schemes

It is possible to formulate the problem of optimal sequential paging in many different ways. There are three chief measures of performance that are of interest – average paging cost, average delay, and worst case delay. It has been shown in [28] that, to minimize $\overline{D}$ or $\overline{L}$, the paging zones must be partitioned in such a way that more probable cell locations must be paged at an earlier round than less probable ones. Theorem 1 presents this formally.

Theorem 1. The partition $Z_1, Z_2, \ldots, Z_w$ of the set of all cells in the location area, C, that minimizes $\overline{D}$ or $\overline{L}$ must be a non-increasingly ordered partition.

Proof. See theorem 1 in [28]. □

In the following discussions, we will restrict ourselves to non-increasingly ordered partitions only.

As mentioned earlier, there is a tradeoff between the average paging cost and delay. We can reduce the average paging cost if we are allowed to increase the worst case delay.

Theorem 2. Let $\overline{L}_1, \overline{L}_2$ be the minimum average paging cost that can be achieved with w_1, w_2 paging zones, respectively. If $w_1 < w_2$ then $\overline{L}_2 \leqslant \overline{L}_1$.

Proof. See appendix. □

Theorems 1 and 2 imply that if one wishes to minimize the average paging cost without any constraints on delay, the optimal sequential scheme is to page each cell one by one in non-increasing order of user location probabilities. In other words, the region is partitioned into n paging zones, each consisting of exactly one cell.

3.1. Minimizing average paging cost under worst case delay constraint

Problem A (Minimizing average cost of paging under worst case delay constraint).

$$\min \overline{L},$$

subject to:

$$w \text{ is a fixed natural number.}$$

We first note that theorem 1 holds irrespective of whether the delay w is constrained. An immediate consequence of this for the problem of minimizing $\overline{L}$ under the worst case delay constraint is that the total number of non-increasingly ordered partitions is equal to $\binom{n-1}{w-1} = O(n^{w-1})$. We can, however, do better in terms of running time. The following theorem establishes the optimal substructure inherent in the problem.

Theorem 3 (Optimal substructure for problem A). If $Z_1, Z_2, \ldots, Z_w$ is the partition of cells in the location area that minimizes average cost of paging under a worst case delay constraint of w, then $Z_1, Z_2, \ldots, Z_k$ is a partition of $\bigcup_{i=1}^{k} Z_i$ (the set of all the cells in the first k paging zones, $1 \leqslant k \leqslant w$) that minimizes the average cost of paging under a worst case delay constraint of k.

Proof. See appendix. □

Definition. $h[k, e]$ is the minimum average paging cost that can be achieved in paging the first e cells using $k \geqslant 2$ paging zones. We seek $h[w, n]$.

It can be seen that

$$h[2, n] = \min_{n_1=1}^{n} (n_1 p_1 + n p_2). \tag{4}$$

Thus, when only two paging zones are used, to calculate $h[2, n]$ we need only vary the number of cells that should be paged in the first round (n_1). In this case, the average paging cost can therefore be minimized in linear time with respect to the number of cells in the location area, i.e. in $O(n)$.

A quadratic-time algorithm

Now let us consider the general case, for arbitrary w and n. The following recursive relation holds $\forall k \geqslant 2$:

$$h[k+1, e] = \min_{j=k}^{e}\big(h[k, j] + ep_{k+1}\big), \tag{5}$$

where $p_{k+1} = \sum_{i=j+1}^{e} \pi_i$.

We have the following initial conditions for $h[\cdot, \cdot]$:

$$h[1, j] = j \sum_{i=1}^{j} \pi_i. \tag{6}$$

We can think of $h[\cdot, \cdot]$ as a two-dimensional table of size $w \times n$, and using the recursive equation (5) and the initial conditions in (6), build its entry from the bottom up or top-down using *memoization* [12]. It takes $O(n)$ time to find the minimum of $h[k, j]$, $\forall j$ and hence to calculate each entry of the two-dimensional table. Thus the solution to this optimization problem, $h[w, n]$ can be solved in $O(wn^2)$ time. As in simple dynamic programming, the partition of cells into w paging zones that corresponds to this minimum value of average paging cost can be found by tracing back through the table or by keeping track of decisions at each step as the algorithm proceeds. The memory requirement is constrained by the size of the table and is hence $O(wn)$.

3.2. Minimizing average paging cost under constraint on average delay

Problem B (Minimizing the average cost of paging under average delay constraint).

$$\min \overline{L}$$

subject to:

$$\overline{D} \leqslant D^*,\ \text{given a positive real number } D^*.$$

We note that while problem B is the one discussed in [1], an equivalent equality constraint is used in [28]. The inequality constraint is more meaningful, since we would generally be interested in keeping the average paging delay below some threshold, rather than at some exact value.

In [28], the authors claim that this problem is *not* amenable to solution via dynamic programming, because the constraint renders the cost function non-additive. They provide an approximate solution using a continuous formulation and Lagrange multipliers to perform the optimization. The authors in [28] then show that these approximate solutions are very close to optimal. We note that this in itself suggests that the discrete problem is not computationally hard, and that it probably has an exact solution that can be determined in polynomial time. In [1], however, there is a proof that the problem is NP-complete. Unless P $=$ NP, this would seem to indicate that the problem does not have a polynomial time solution. Indeed, the authors in [1] indicate the only way to solve it exactly would take $O(2^n)$ time. What is the right answer? It turns out that the proof in [1] showing the similarity of this problem to the Knapsack problem does not take into account bounds on the average paging delay constraint. It is well known that while the general Knapsack problem is NP complete, there exists a polynomial solution that uses dynamic programming for the special case of the problem when the constraint is a polynomially bounded function of the instance size [14]. For our problem, since the average paging delay can never exceed the number of paging zones, we need only concern ourselves with values of D^* that are less than or equal to n. This considerably simplifies the problem and makes it tractable. We present below a dynamic programming approach that solves this minimization problem in polynomial time.

The first assumption we have to make regarding the problem is that D^* is a positive number that can be represented discretely as one of A discrete values. This is an entirely reasonable assumption for our purposes, since any practical system cannot estimate, calculate or store probabilities or delays with infinite precision. Also, as we mentioned above, it is assumed without loss of generality that $D^* \leqslant n$.

Definition. $h^\dagger[k, e, \alpha]$ is the minimum average paging cost that can be achieved when paging the first e cells using $k \geqslant 2$ paging zones, with a maximum average paging delay of $\alpha \leqslant n$. We seek $h^\dagger[w, n, D^*]$.

The result from theorem 1 still holds here, so that we only concern ourselves with non-increasingly ordered partitions of the set Q.

Theorem 4 (Optimal substructure for problem B). For the problem of minimizing the average cost of paging under average delay constraint, the following recursive relation holds:

$$h^\dagger[k+1, e, \alpha] = \min_{j=k}^{(e)} h^\dagger\left[k, j, \alpha - (k+1)\sum_{i=j+1}^{e} \pi_i\right] + e\sum_{i=j+1}^{e} \pi_i. \tag{7}$$

Proof. See appendix. □

The corresponding initial condition here is that

$$h^\dagger[1, j, \alpha] = \begin{cases} j\sum_{i=1}^{j} \pi_i, & \text{if } \sum_{i=1}^{j} \pi_i \leqslant \alpha, \\ \infty, & \text{otherwise.} \end{cases} \tag{8}$$

Equation (7) tells us that the optimal solution for the problem that involves $(k+1)$ paging zones and an average delay of α can be written as the sum of the optimal solution for k paging zones with a smaller delay and the average paging cost due to the $(k+1)$th zone. Specifically the delay for the optimal sub-problem is reduced by an amount that depends upon the "excess" probabilities due to the cells in the $(k+1)$th paging zone.

This recursive relation results in a dynamic programming solution. We can think of $h^\dagger[\cdot, \cdot, \cdot]$ as a 3 dimensional table of size $(w \times n \times A)$, where A is the number of discrete levels used to represent α. The recursive relation (7), together with the initial conditions (8) suffice to fill in all the elements of this table. The element corresponding to $h[w, n, D^*]$ gives us the optimal solution. To determine which cells are placed in which paging zone, one can trace back through the table, keeping track of the decisions made at each step. A value of ∞ (which can be replaced by some arbitrarily large number (any number greater than n will suffice) indicates that there is no solution that satisfies the given average delay constraint. The space-complexity of this dynamic programming algorithm is the size of the table $O(wnA)$. It takes $O(n)$ time to fill each element of the table, giving the algorithm a time-complexity of $O(wn^2A)$. Thus we have a polynomial time algorithm for this problem formulation.

4. Performance of sequential paging

This section presents results on the performance of sequential paging under assumptions regarding the user location probabilities. The first case we consider is the uniform distribution, where the cell location probabilities are $\pi_j = 1/n$ for all n cells. In the second case we consider, the zone location probabilities are distributed geometrically, i.e. $p_i = r^i/(\sum_{i=1}^{w} r^i)$.

4.1. Uniformly distributed user location probabilities

It is shown in [28] that sequential paging schemes have the worst case average paging cost and paging delay when the mobile users are equally likely to be in any cell.

Let Π_n be any arbitrary distribution of the cell-wise user location probabilities, for which the minimum possible average paging cost is $\overline{L}_w^{\Pi_n}$ and the minimum possible average delay is $\overline{D}_w^{\Pi_n}$ when w paging zones are used. Let U_n be the uniform distribution of probabilities $= \{\frac{1}{n}, \frac{1}{n}, \frac{1}{n}, \ldots\}$, for which the the minimum average paging cost is $\overline{L}_w^{U_n}$ and the minimum average delay is $\overline{D}_w^{U_n}$ when w paging zones are used. Then, we have the following:

Theorem 5. The uniform distribution case corresponds to the upper bounds on the minimum average paging cost and minimum average paging delay when n cells are partitioned into w paging zones: $\overline{L}_w^{\Pi_n} \leqslant \overline{L}_w^{U_n}$ and $\overline{D}_w^{\Pi_n} \leqslant \overline{D}_w^{U_n}$.

Proof. See corollary 2 of theorem 2 in [28]. □

The first result we have for the uniformly distributed location probabilities is that if we wish to minimize the average cost of paging, we do not need to run the dynamic programming algorithm described in section 3.3. This is because the optimal solution has a specific structure, described in the following theorem.

Theorem 6. If each cell has equal probability of user location then the w-partition of C which minimizes the average cost of paging is *balanced* such that the difference in the number of cells between any two paging zones is no more than one.

Proof. See appendix. □

Corollary 6.1. For the balanced partition in theorem 6 that minimizes the average cost of paging,

$$\forall i \in \{1, 2, \ldots w\}, \quad \left\lfloor \frac{n}{w} \right\rfloor \leqslant n_i \leqslant \left\lceil \frac{n}{w} \right\rceil.$$

We can normalize the average paging cost with respect to the average paging cost when using only one partition. Let the *normalized reduction in average paging cost* be defined as follows:

$$\Lambda_{\Pi_n}(w) = \frac{\overline{L}_1^{\Pi_n} - \overline{L}_w^{\Pi_n}}{\overline{L}_1^{\Pi_n}}. \tag{9}$$

We would like $\Lambda_{\Pi_n}(w)$, which represents the reduction in paging cost that is gained by using multiple paging zones, to be as close to 1 as possible. If $\Lambda_{\Pi_n}(w)$ is 0.5, for example, then we can get a reduction of 50% in the paging cost by using w paging zones instead of 1.

Because of the known structure of the paging zone partitions under the uniform distribution, analytical expressions can be derived for the performance of paging under a worst case delay constraint.

Theorem 7. The following is true:

$$\text{(i)} \qquad \lim_{n\to\infty} \overline{D}_w^{U_n} = \frac{w+1}{2},$$

$$\text{(ii)} \quad \lim_{n\to\infty} \Lambda_{U_n}(w) = \frac{1}{2}\left(1 - \frac{1}{w}\right).$$

Proof. See appendix. □

Theorem 7 and 9 together imply that for any arbitrary distribution of user locations Π_n:

$$\lim_{n\to\infty} \overline{D}_w^{\Pi_n} \leqslant \frac{w+1}{2}, \tag{10}$$

$$\lim_{n\to\infty} \Lambda_{\Pi_n}(w) \geqslant \frac{1}{2}\left(1 - \frac{1}{w}\right). \tag{11}$$

Equation (11) is a particularly appealing result. It tells us, for example, that when there are a large number of cells in the region, we can obtain asymptotically at least a 25% reduction in paging costs by using 2 paging zones, and at least a 40% reduction in paging costs by using up to 5 paging zones. At least for the uniformly distributed user location distribution, there are diminishing returns after this point, with no more than a further 10% gain possible if we increase the number of paging zones any further. Note that although we have presented asymptotic results in theorem 7, we can actually calculate the exact results for the uniform distribution easily as shown in the appendix.

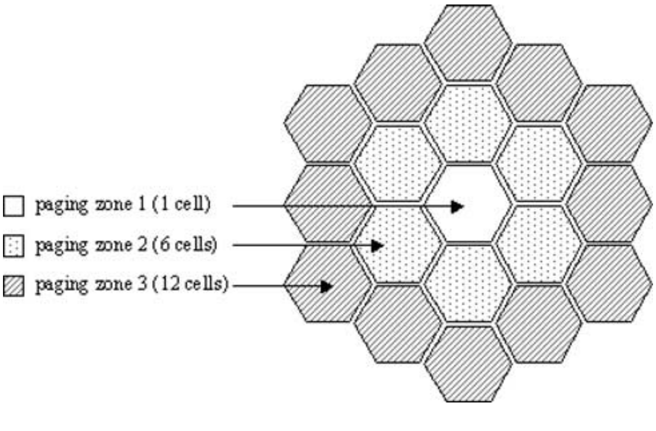

Figure 2. An example of cluster paging for hexagonal cells, $w = 3$ paging zones.

4.2. Cluster paging with geometric distribution

A cluster paging scheme was introduced in [26]. In this special case of sequential paging, successively concentric rings are paged from the last known location outwards. An example with a maximum of 3 rounds of paging can be seen in figure 2. Paging zone 1 consists of the center cell, paging zone 2 consists of the 6 cells in the ring surrounding it, and paging zone 3 is the ring of 12 cells on the outside. Cluster paging finds applications particularly in the context of dynamic location update schemes such as distance, movement and timer-based techniques.

Under what conditions is cluster paging optimal? The following theorem suggests the answer.

Theorem 8. Assume we have a location area consisting of w groups of cells, such that in the ith group, each cell has equal user location probability λ_i. Let k_i the number of cells in the ith group, and $p_i = k_i \cdot \lambda_i$ be the probability of user location in the whole group. Further, let the following conditions hold: If $i < j$, then $k_i < k_j$ and $p_i > p_j$. Under these conditions, the w-partition of the $n = \sum_{i=1}^{w} k_i$ cells in this location area that minimizes the average paging cost is the w-partition in which the cells of the ith group form the ith paging zone.

Proof. See appendix. □

The key assumption in this theorem, which makes it non-trivial, is the requirement that if $i < j$, then $k_i < k_j$ and $p_i > p_j$. This need not be true simply because the cells have monotonically decreasing user location probabilities. As a counter-example, consider a scenario with four cells and two cell groups, with location probabilities as follows: $\{\frac{1}{3}, \frac{2}{9}, \frac{2}{9}, \frac{2}{9}\}$. In this case, $p_2 = \frac{2}{3} > p_1 = \frac{1}{3}$, and the partition which minimizes the average paging cost consists of $Z_1 = \{1, 2\}$ and $Z_2 = \{3, 4\}$. Note that in this optimal partition the cells of the second group are not all paged at the second paging round.

Therefore cluster paging is an optimal sequential paging scheme when the following conditions hold:

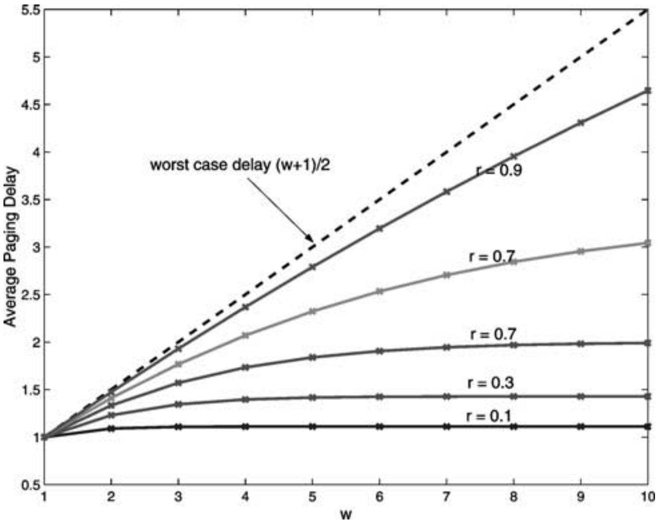

Figure 3. The average paging delay for geometric distribution of zone location probabilities in cluster paging.

- There are $w - 1$ distinct rings outside the center cell in the location area, where w is the number of paging rounds.
- Each cell in the ith ring has equal probability of user location, for $i = 1, 2, \ldots, w - 1$.
- As we move outwards from the center cell, each consecutive ring has a lower probability of user location.
- The number of cells in each successively outward ring is increasing.

It is argued in [26] based on empirical studies that the probability of user location decreases geometrically with distance from the center cell in a location area. Thus for some $0 < r \leqslant 1$, we have for cluster paging that

$$p_i = \sum_{j=1+k_2+\cdots+k_{i-1}}^{1+k_2+\cdots+k_i} \pi_j = \frac{r^i}{\sum_{i=1}^{w} r^i}. \tag{12}$$

The number of cells in each ring/paging zone can be characterized as follows: say the the ith paging zone consists of k_i cells. We let $k_1 = 1$ and $k_i = M \cdot (i - 1)$ for $i \geqslant 2$, where M is a positive integer constant that depends on the cell shape (for example, $M = 6$ for hexagonal cells, and 8 for rectangular cells). Thus the number of cells in each successive paging zone increases linearly.

Under these assumption on the geometric distribution of zone location probabilities and the cellular topology, it is possible to evaluate the average delay and paging costs for the optimal sequential paging scheme:

$$\begin{aligned}\overline{D}_w &= \frac{\sum_{i=1}^{w} i r^i}{\sum_{i=1}^{w} r^i} = \frac{1-r}{r - r^{w+1}} \sum_{i=1}^{w} i r^i \\ &= \frac{-w r^{w+2} + (2w+1) r^{w+1} - (w+1) r^w + r + 1}{1 - r - r^{w-1} + r^w},\end{aligned} \tag{13}$$

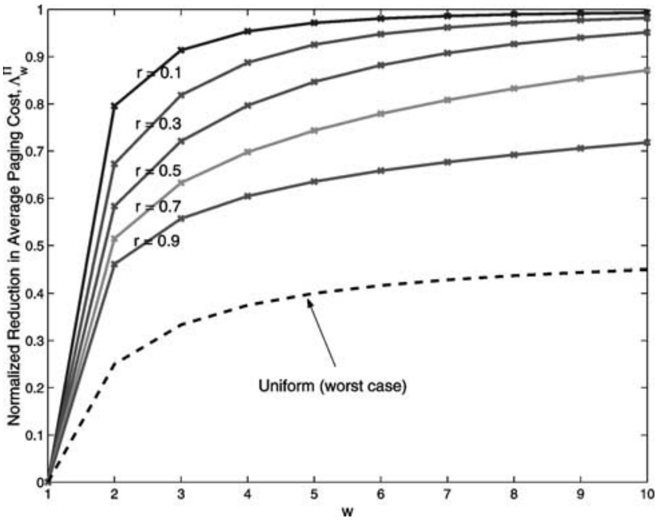

Figure 4. The normalized reduction in average paging cost for geometric distribution of zone location probabilities in cluster paging.

$$\begin{aligned}\overline{L}_w &= \frac{\sum_{i=1}^{w}(r^i \sum_{j=1}^{i} k_j)}{\sum_{i=1}^{w} r^i}\\ &= \frac{1-r}{r-r^{w+1}} \cdot \sum_{i=1}^{w}\left(1+\sum_{j=1}^{i-1} jM\right) r^i. \qquad (14)\end{aligned}$$

Figures 3 and 4 plot the average delay $\overline{D}_w$ and the normalized reduction in average paging cost $\overline{\Lambda}_w^{\Pi_n}$ with respect to w for cluster paging when the zone location probabilities decrease geometrically, along with the worst case bounds that arise when $r = 1$ (uniform distribution). Note that in these figures the total number of cells n in the location area is also increasing with w, since $n = 1 + 0.5Mw(w-1)$. These results indicate that the best performance, both in terms of low average paging delay as well as high normalized reduction in average paging cost, is obtained when the parameter r of the geometric distribution is small. This corresponds to a situation where the user location probabilities are "concentrated" in a relatively small number of cells. The results are intuitive as we should expect to make significant savings on the average paging cost as well as on paging delay when the user's movements are restricted to a few cells in a large location area.

5. Conclusion

In high-capacity cellular networks with limited radio resources, it is desirable to minimize the radio costs when locating mobile users during call arrival. Sequential paging schemes permit a reduction in paging costs at the expense of potentially greater delay.

We presented a polynomial-time algorithm to solve the problem of minimizing the average paging cost under the constraint on average delay. This problem had previously been considered computationally intractable. Thus in conjunction with [28], our results show that the task of determining the optimal sequential paging scheme is, in general, feasible.

We also presented some analytical results on the performance of sequential paging schemes under assumptions on the distribution of user location probabilities. The first case we discussed is that of the uniform user location probabilities which provides tight bonds on the performance of sequential paging for any arbitrary distribution. We showed that the normalized reduction in average paging costs increases as $\frac{1}{2}(1-\frac{1}{w})$ with respect to w, the total number of paging zones. This implies, for example, that using as few as 2 paging zones we are guaranteed to obtain at least a 25% reduction in paging costs on average compared to the policy of paging all cells in the location area. Similarly, the average paging delay was shown to be upper bounded by $(w+1)/2$.

The second case we considered was that of cluster paging with geometric zone location probabilities. Cluster paging is a special case of sequential paging that arises particularly in the context of dynamic location updating schemes. We showed conditions under which cluster paging is an optimal sequential paging scheme. Also, a common assumption in cluster paging schemes, justified by empirical observations in [26], is that user location probabilities decrease geometrically with distance. Under this assumption, we derived results for the performance of the optimal cluster paging scheme. These results confirm the intuition that the more "concentrated" the user location probabilities are in a portion of the location area, the better the performance of sequential paging.

It must be noted that sequential paging schemes may be somewhat complicated to implement because they are devised on a per-user basis and require the tracking of location statistics for each user. One extension of this work would consider simpler but sub-optimal aggregate paging schemes in which multiple users may be paged jointly [15]. In any case, the use of sequential paging schemes is predicated upon the ability to obtain good estimates of the cell-wise user location probabilities, which require the use of an appropriate location tracking scheme. Another related issue that arises is the question of the sensitivity of sequential paging to the estimates of user location probabilities. We have addressed some of these issues in [20], but these questions are still topics for future work.

Appendix. Proofs

Theorem 1. The partition $Z_1, Z_2, \ldots, Z_w$ of the set of all cells in the location area, C, that minimizes $\overline{D}$ or $\overline{L}$ must be a non-increasingly ordered partition.

Proof. See theorem 1 in [28]. □

Theorem 2. Let $\overline{L}_1, \overline{L}_2$ be the minimum average paging cost that can be achieved with w_1, w_2 paging zones respectively. If $w_1 < w_2$ then $\overline{L}_2 \leqslant \overline{L}_1$.

Proof. Let $w' = w_1 + 1$, and let $\overline{L}_{w'}$ be the minimum average paging cost that can be achieved using w' paging zones. It suffices to show that $\overline{L}_{w'} \leqslant \overline{L}_1$, since this would imply that $\overline{L}_2 \leqslant \cdots \leqslant \overline{L}_{w'} \leqslant \overline{L}_1$. This can be shown by constructing *a* w'-partition of C that has an average paging cost no greater than $\overline{L}_1$.

Let $Z_1, Z_2, \ldots, Z_{w_1}$ be the partition of C that yields the minimum average paging cost $\overline{L}_1$. If $w + 1 \leqslant n$, we can always choose a paging zone Z_m such that it has more than one cell: $n_m > 1$. Now partition this paging zone in any manner into two non-empty paging zones Z_{m_1} and Z_{m_2}. $Z_1, Z_2, \ldots, Z_{m_1}, Z_{m_2}, \ldots, Z_{w_1}$ is now a w'-partition of C. Let $\overline{L}^*$ be the average paging cost for this partition. The following holds:

$$\begin{aligned}\overline{L}_1 - \overline{L}^* &= \sum_{i=1}^{w_1} p_i \sum_{j=1}^{i} n_j - \sum_{i=1}^{w'} p_i \sum_{j=1}^{i} n_j \\ &= p_m \sum_{j=1}^{m} n_j - \left(p_{m_1} \sum_{j=1}^{m_1} n_j + p_{m_2} \sum_{j=1}^{m_2} n_j \right) \quad \text{(A.1)} \\ &= p_m \sum_{j=1}^{m} n_j - (p_{m_1} + p_{m_2}) \sum_{j=1}^{m} n_j + p_{m_1} n_{m_2} \\ &= p_{m_1} n_{m_2} \geqslant 0. \quad \text{(A.2)}\end{aligned}$$

We now have that $\overline{L}^* \leqslant \overline{L}_1$. Since $\overline{L}_{w'}$ is the optimal paging cost with w' paging zones, $\overline{L}_{w'} \leqslant \overline{L}^* \leqslant \overline{L}_1$. □

Theorem 3 (Optimal substructure for problem A). If $Z_1, Z_2, \ldots, Z_w$ is the partition of cells in the location area that minimizes average cost of paging under a worst case delay constraint of w, then $Z_1, Z_2, \ldots, Z_k$ is a partition of $\bigcup_{i=1}^{k} Z_i$ (the set of all the cells in the first k paging zones, $1 \leqslant k \leqslant w$) that minimizes the average cost of paging under a worst case delay constraint of k.

Proof. For the optimal w-partition $Z_1, Z_2, \ldots, Z_w$, the average paging cost $\overline{L}_w$ can be expressed as:

$$\overline{L}_w = \sum_{i=1}^{w} p_i \sum_{j=1}^{i} n_j = \sum_{i=1}^{w-1} p_i \sum_{j=1}^{i} n_j + p_w n. \quad \text{(A.3)}$$

From this it is obvious that $Z_1, Z_2, \ldots, Z_{w-1}$ is the optimal partition of $\bigcup_{i=1}^{w-1} Z_i$. For if not, then there exists some other partition of $\bigcup_{i=1}^{w-1} Z_i$ with a lower average paging cost, and using this other partition along with Z_w as a w-partition of C would result in an overall average paging cost lower than $\overline{L}_w$, which contradicts our assumption that $Z_1, Z_2, \ldots, Z_w$ is the optimal w-partition.

Using the above argument inductively, it can be seen that $\forall k \in \{1, 2, \ldots, w\}$, $Z_1, Z_2, \ldots, Z_k$ is an optimal partition of $\bigcup_{i=1}^{k} Z_i$. □

Theorem 4 (Optimal substructure for problem B). For the problem of minimizing the average cost of paging under average delay constraint, the following recursive relation holds:

$$\begin{aligned}&h^{\dagger}[k+1, e, \alpha] \\ &= \min_{j=1}^{e} \left(h^{\dagger} \left[k, j, \alpha - (k+1) \sum_{i=j+1}^{e} \pi_i \right] + e \sum_{i=j+1}^{e} \pi_i \right). \quad \text{(A.4)}\end{aligned}$$

Proof. We can justify equation (A.4) using the definition of $h^{\dagger}[\cdot, \cdot, \cdot]$ (see section 3.2).

Looking at the right-hand side of equation (A.4), we see that we are separating the properties of the first k partitions from those of the $(k+1)$st partition. For a fixed value of j, the additive contribution of the $(k+1)$st partition to the average cost of paging is exactly the second term: $e \sum_{i=j+1}^{e} \pi_i = e p_{k+1}$.

Now, if $Z_1, Z_2, \ldots, Z_{k+1}$ is the $(k+1)$-partition of the set $\bigcup_{i=1}^{k+1} Z_i$ that yields the minimum average cost of paging, then the set $\bigcup_{i=1}^{k} Z_i$ must be partitioned to yield the lowest possible cost of paging (for the same reasons as in theorem 3 for problem A), subject to the constraint that the average delay due to the first k paging zones be no more than $\alpha - (k+1) p_{k+1}$. Note that this average delay constraint must be satisfied if the average delay for the $(k+1)$-partition of $\bigcup_{i=1}^{k+1} Z_i$ is constrained to be less than α. The minimization of the first term with respect to j, $h^{\dagger}[k, j, \alpha - (k+1) \sum_{i=j+1}^{e} \pi_i]$, represents this lowest possible cost of paging under the average delay constraint of $\alpha - (k+1) p_{k+1}$ for the first k paging zones. Adding these two terms on the right-hand side of equation (A.4) yields the minimum average paging cost for $k+1$ paging zones under the constraint that the average paging delay may not exceed α, which is the left hand side. □

Theorem 5. The uniform distribution case corresponds to the upper bounds on the minimum average paging cost and minimum average paging delay when n cells are partitioned into w paging zones: $\overline{L}_w^{\Pi_n} \leqslant \overline{L}_w^{U_n}$ and $\overline{D}_w^{\Pi_n} \leqslant \overline{D}_w^{U_n}$.

Proof. See corollary 2 of theorem 2 in [28]. □

Theorem 6. If each cell has equal probability of user location then the w-partition of C which minimizes the average cost of paging is *balanced* such that the difference in the number of cells between any two paging zones is no more than one.

Proof. This can be shown by the following argument. Consider two w-partitions of C: partition $Z_1', Z_2', \ldots, Z_w'$ with an average paging cost of $\overline{L}'$, and partition $Z_1'', Z_2'', \ldots, Z_w''$ with an average paging cost of $\overline{L}''$. Let these two partitions be identical in all but two paging zones l and m: $Z_l' \neq Z_l''$ and $Z_m' \neq Z_m''$. For the first w-partition, these two paging zones are unbalanced, i.e. $n_l' = n_m' + k$, where $k \geqslant 2$. For the second w-partition, they are balanced, i.e. $n_l'' = n_l - \lfloor k/2 \rfloor$, $n_m'' = n_m + \lfloor k/2 \rfloor$. Note that we can generate the fully balanced partition mentioned in the theorem by repeatedly applying these balancing steps two paging zones at a time, starting from any arbitrary w-partition of C. Hence, if we can show that $\overline{L}'' < \overline{L}'$, then it is true that the average paging cost of the fully balanced partition is the minimum that can be achieved. Since the two partitions only differ in paging zones l and m,

$$\overline{L}' - \overline{L}'' = \sum_{i=1}^{w} \frac{n'_i}{n} \sum_{j=1}^{i} n'_j - \sum_{i=1}^{w} \frac{n''_i}{n} \sum_{j=1}^{i} n''_j \tag{A.5}$$

$$= \frac{1}{n}\left[\left(n'_l \sum_{i=1}^{w} n'_i + n'_m \sum_{i=1}^{w} n'_i - n'_l n'_m\right) - \left(n''_l \sum_{i=1}^{w} n''_i + n''_m \sum_{i=1}^{w} n''_i - n''_l n''_m\right)\right] \tag{A.6}$$

$$= \frac{1}{n}\left[n''_l n''_m - n'_l n'_m\right]$$

$$= \frac{1}{n}\left[\left(n'_m + \left\lceil \frac{k}{2} \right\rceil\right)\left(n'_m + \left\lfloor \frac{k}{2} \right\rfloor\right) - n'_m\left(n'_m + k\right)\right] \tag{A.7}$$

$$= \frac{1}{n}\left(\left\lceil \frac{k}{2} \right\rceil \left\lfloor \frac{k}{2} \right\rfloor\right) > 0. \tag{A.8}$$

□

Corollary 6.1. For the balanced partition in theorem 8 that minimizes the average cost of paging,

$$\forall i \in \{1, 2, \ldots w\}, \quad \left\lfloor \frac{n}{w} \right\rfloor \leqslant n_i \leqslant \left\lceil \frac{n}{w} \right\rceil.$$

Theorem 7. The following are true:

$$\text{(i)} \qquad \lim_{n\to\infty} \overline{D}_w^{U_n} = \frac{w+1}{2},$$

$$\text{(ii)} \quad \lim_{n\to\infty} \Lambda_{U_n}(w) = \frac{1}{2}\left(1 - \frac{1}{w}\right)$$

Proof. Let us do these one by one.

(i) From corollary 8.1, it follows that

$$\forall i \in \{1, 2, \ldots, n\}, \quad \frac{1}{n}\left\lfloor \frac{n}{w} \right\rfloor \leqslant p_i \leqslant \frac{1}{n}\left\lceil \frac{n}{w} \right\rceil. \tag{A.9}$$

Therefore, since $\overline{D}_w^{U_n} = \sum_{i=1}^{w} i p_i$,

$$\sum_{i=1}^{w} i \frac{1}{n}\left\lfloor \frac{n}{w} \right\rfloor \leqslant \overline{D}_w^{U_n} \leqslant \sum_{i=1}^{w} i \frac{1}{n}\left\lceil \frac{n}{w} \right\rceil \tag{A.10}$$

$$\Rightarrow \lim_{n\to\infty} \overline{D}_w^{U_n} = \lim_{n\to\infty} \frac{1}{n} \cdot \frac{n}{2} \cdot \frac{w(w+1)}{2} = \frac{w+1}{2}. \tag{A.11}$$

(ii) From corollary 6.1, it follows that

$$\forall i \in \{1, 2, \ldots, n\}, \quad \frac{1}{n}\left\lfloor \frac{n}{w} \right\rfloor \leqslant p_i \leqslant \frac{1}{n}\left\lceil \frac{n}{w} \right\rceil \tag{A.12}$$

and

$$i\left\lfloor \frac{n}{w} \right\rfloor \leqslant \sum_{j=1}^{i} n_j \leqslant i\left\lceil \frac{n}{w} \right\rceil. \tag{A.13}$$

Therefore,

$$\frac{1}{n}\sum_{i=1}^{w}\left\lfloor \frac{n}{w} \right\rfloor\left(i\left\lfloor \frac{n}{w} \right\rfloor\right) \leqslant \overline{L}_w^{U_n} \leqslant \frac{1}{n}\sum_{i=1}^{w}\left\lceil \frac{n}{w} \right\rceil\left(i\left\lceil \frac{n}{w} \right\rceil\right) \tag{A.14}$$

$$\Rightarrow \frac{1}{n}\left(\left\lfloor \frac{n}{w} \right\rfloor\right)^2 \frac{w(w+1)}{2} \leqslant \overline{L}_w^{U_n} \leqslant \frac{1}{n}\left(\left\lceil \frac{n}{w} \right\rceil\right)^2 \frac{w(w+1)}{2}. \tag{A.15}$$

We also know that average cost of paging when only one paging zone is used is $\overline{L}_1^{U_n} = n$. Therefore,

$$\lim_{n\to\infty} \Lambda_{U_n}(w) = \lim_{n\to\infty} \frac{\overline{L}_1^{U_n} - \overline{L}_w^{U_n}}{\overline{L}_1^{U_n}} = \lim_{n\to\infty} \frac{n - \overline{L}_w^{U_n}}{n} \tag{A.16}$$

$$= 1 - \lim_{n\to\infty}\left(\frac{1}{n} \cdot \frac{1}{n} \cdot \frac{n^2}{w^2} \cdot \frac{w(w+1)}{2}\right) = \frac{1}{2}\left(1 - \frac{1}{w}\right). \tag{A.17}$$

□

Theorem 8. Assume we have a location area consisting of w groups of cells, such that in the ith group, each cell has equal user location probability λ_i. Let k_i the number of cells in the ith group, and $p_i = k_i \cdot \lambda_i$ be the probability of user location in the whole group. Further, let the following conditions hold: If $i < j$, then $k_i < k_j$ and $p_i > p_j$. Under these conditions, the w-partition of the $n = \sum_{i=1}^{w} k_i$ cells in this location area that minimizes the average paging cost is the w-partition in which the cells of the ith group form the ith paging zone.

Proof. For the purpose of this proof, it will be helpful to think of the partition of cells into paging zones as being equivalent to the non-increasingly ordered partitioning of the corresponding set of user location probabilities. Look at the following examples with $w = 3$ and $n = 6$ for an illustration:

$$\mathbf{P}_1 = \{\lambda_1|\lambda_2\lambda_2|\lambda_3\lambda_3\lambda_3\}, \quad \mathbf{P}_2 = \{\lambda_1\lambda_2|\lambda_2\lambda_3|\lambda_3\lambda_3\}.$$

Here the pipes indicate the partition boundaries. Thus $\mathbf{P}_1$ represents the partition we have claimed to be optimal: elements of group i are placed in the ith bin. Let's denote such w-partitions as $\mathbf{P}_w^*$. In partition $\mathbf{P}_2$, we have an element of group 2 in the 1st bin, and an element of group 3 in the 2nd bin.

Let $S_i = \sum_{j=1}^{i} k_j$ be the cumulative sum of cells in the first i groups. The average paging cost associated with the partition $\mathbf{P}_w^*$ is

$$\overline{L}_{\mathbf{P}_w^*} = \sum_{i=1}^{w} p_i \cdot S_i = \sum_{i=1}^{w} k_i \cdot \lambda_i \cdot S_i. \tag{A.18}$$

Now consider each element $e \in \{1, 2, \ldots, n\}$ of an arbitrary partition $\mathbf{P}$; say the element e belongs to group i, and is in bin j. We construct a credit/debit value for each such element as follows:

- If there are N elements in the same bin j belonging to "future" groups l, $l > i$, then element e has a positive *credit* $C_e = +(N \cdot \lambda_i)$. Otherwise $C_e = 0$.
- If there are M elements in "future" bins l, $l > j$, that are from the same group i, then the element e has a negative *debit* $D_e = -(M \cdot \lambda_i)$. Otherwise $D_e = 0$.

The following facts can be derived easily from the definition of this accounting scheme: (a) each element either has a non-zero credit or a non-zero debit value, but not both, and (b) the sum of all credits and debits for all elements e of $\mathbf{P}$ is the "excess" average paging cost resulting from this partition. In other words:

$$\overline{L}_{\mathbf{P}} - \overline{L}_{\mathbf{P}_w^*} = \sum_{e=1}^{n}(C_e + D_e). \tag{A.19}$$

To prove that $\mathbf{P}_w^*$ minimizes the average paging cost, it therefore suffices to show that $\forall \mathbf{P} \neq \mathbf{P}_w^*$, $\sum_{e=1}^{n}(C_e + D_e) > 0$. The partitions $\mathbf{P}_w^*$ and $\mathbf{P}$ differ from each other in that $\mathbf{P}$ has some bins which contain elements belonging to higher group number than that bin, and bins which contain elements belonging to a lower group number. One can imagine that this took place by a sequence of steps during which the boundaries of the partitions were moved from their original position in $\mathbf{P}_w^*$ to either their right or left by an arbitrary number of places.

In particular, we can think of getting from the partition $\mathbf{P}_w^*$ through a finite sequence of simple moves to $\mathbf{P}$ as follows:

1. First move the boundary of the bins in $\mathbf{P}_w^*$ (if any) that need to be moved to the right by moving the right boundary of the rightmost such bin first to its required position in $\mathbf{P}$, then the boundary of next such bin and so on.
2. Now move the boundary of the bins in $\mathbf{P}$ (if any) that need to be moved to the left, by moving the boundary of the leftmost such bin first to its required position in $\mathbf{P}$, then the boundary of the next such bin and so on.

At each step of a move of type 1, the elements of bins on either side of boundary that is being moved to the right looks as follows:

$$\{\ldots\lambda_i\ldots\lambda_i\ldots\lambda_{j-1}\ldots\lambda_{j-1}\lambda_j\ldots\lambda_j|\lambda_j,\lambda_j\ldots\}. \tag{A.20}$$

When the element with value λ_j is moved from the right bin to the left bin (correspondingly, when the boundary is moved one place to the right), the total debits of all elements in the partition can increase by no more than $-(\lambda_j \cdot k_j)$. This is because only the λ_j element being moved to the left will experience an increased debit after the move. And the total credits increase by at least $+(\lambda_{j-1} \cdot k_{j-1})$, since each element of group $(j-1)$ will be present in the bin to the left and will gain a credit when the "future" element λ_j is moved into the same bin. Since we know that $\lambda_j \cdot k_j < \lambda_{j-1} \cdot k_{j-1}$ from the hypothesis of the theorem, the sum of credits and debits actually increases at each such step.

Similarly we can show that the sum of credits and debits also increases at each step of a move of type 2. Since the sum of credits and debits is 0 for $\mathbf{P}_w^*$, and keeps increasing at each step as we construct the partition $\mathbf{P} \neq \mathbf{P}_w^*$, the sum of credits and debits is positive for $\mathbf{P} \neq \mathbf{P}_w^*$. □

Acknowledgements

This work was supported in part by the NSF grant number ANI-9704404 and the ONR MURI contract number N00014-00-1-0564.

References

[1] A. Abutaleb and V.O.K. Li, Paging strategy optimization in personal communication systems, Wireless Networks 3(3) (1997) 195–204.

[2] A. Abutaleb and V.O.K. Li, Location update optimization in personal communication systems, Wireless Networks 3(3) (1997) 205–216.

[3] I.F. Akyildiz and J.S.M. Ho, Dynamic mobile user-location update for wireless PCS networks, Wireless Networks 1(2) (1995) 187–196.

[4] I.F. Akyildiz and J.S.M. Ho, Movement-based location update and selective paging for PCS Networks, IEEE/ACM Transactions on Networking 4(4) (1995) 629–638.

[5] ANSI/EIA/TIA, Mobile station-land station compatibility specification, EIA/TIA Technical Report 553 (1989).

[6] A. Bar-Noy and I. Kessler, Tracking mobile users in wireless networks, IEEE Transactions on Information Theory 39 (1993) 1877–1886.

[7] A. Bar-Noy, I. Kessler and M. Naghshineh, Topology-based tracking strategies for personal communication netowks, Mobile Networks and Applications 1(1) (1996) 49–96.

[8] A. Bar-Noy, I. Kessler and M. Sidi, Mobile users: To update or not to update?, Wireless Networks 1(2) (1995) 175–185.

[9] A. Bhattacharya and S.K. Das, LeZi-Update: An information-theoretic approach to track mobile users in PCS networks, in: *Proc. of MOBICOM 1999* (1999) pp. 1–12.

[10] Y. Birk and Y. Nachman, Using direction and elapsed-time information to reduce the wireless cost of locating mobile units in cellular networks, Wireless Networks 1(4) (1995) 403–412.

[11] T.X. Brown and S. Mohan, Mobility management for personal communication systems, IEEE Transactions on Vehicular Technology 46(2) (1997) 269–278.

[12] T.H. Cormen, C.E. Leiserson and R.L. Rivest, *Introduction to Algorithms* (MIT Press, Cambridge, MA, 1999).

[13] S.K. Das and S.K. Sen, A new location update strategy for cellular networks and its implementation using a genetic algorithm, in: *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 1997)* (September 1997) pp. 185–194.

[14] M.R. Garey and D.S. Johnson, *Computers and Intractability – A Guide to the Theory of NP-Completeness* (W.H. Freeman, New York, 1979).

[15] R.-H. Gau and Z.J. Haas, Concurrently searching for mobile users in cellular networks, IEEE Communication Letters (June 2003).

[16] A. Hac and Z. Zhou, Locating strategies for personal communication networks: A novel tracking strategy, IEEE Journal on Selected Areas in Communication 15(8) (1997) 1425–1436.

[17] J.S.M. Ho and I.F. Akyildiz, Mobile user location update and paging under delay constraints, Wireless Networks 1(4) (1995) 413–425.

[18] R. Jain, Y.-B. Lin and S. Mohan, Location strategies for personal communications service, in: *The Mobile Communications Handbook*, ed. J. Gibson (CRC Press, 1996).

[19] S.J. Kim and C.Y. Lee, Modeling and analysis of the dynamic location registration and paging in microcellular systems, IEEE Transactions on Vehicular Technology 45(1) (1996) 82–90.

[20] B. Krishnamachari, R.-H. Gau, S. Wicker and Z. Haas, On the performance of sequential paging for mobile user location, in: *Proc. of IEEE Vehicular Technology Conference (VTC Fall 2001)*, Atlantic City, NJ (October 2001).

[21] B.-K. Lee and C.-S. Hwang, A predictive paging scheme based on the movement direction of a mobile host, in: *Proc. of IEEE Vehicular Technology Conference (VTC '99)* (1999) pp. 2158–2162.

[22] B. Liang and Z.J. Haas, Predictive distance-based mobility management for PCS networks, in: *Proc. of the Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '99)*, Vol. 3 (1999) pp. 1377–1384.

[23] G.L. Lyberopoulos, J.G. Markoulidakis, D.V. Polymeros, D.F. Tsirkas and E.D. Sykas, Intelligent paging strategies for third generation mobile telecommunication systems, IEEE Transactions on Vehicular Technology 44(3) (1995) 543–553.

[24] U. Madhow, M.L. Honig and K. Steiglitz, Optimization of wireless resources for personal communications mobility tracking, IEEE/ACM Transactions on Networking 3(6) (1995) 698–707.

[25] J.G. Markoulidakis, G.L. Lyberopoulos, D.F. Tsirkas and E.D. Sykas, Evaluation of location area planning scenarios in future mobile telecommunication systems, Wireless Networks 1(1) (1995) 17–29.

[26] D. Munoz-Rodriguez, Cluster paging for traveling subscribers, in: *Proc. of IEEE Vehicular Technology Conference* (1990).

[27] G.P. Pollini and C.-L. I, A profile-based location strategy and its performance, IEEE Journal on Selected Areas in Communications 15(8) (1997).

[28] C. Rose and R. Yates, Minimizing the average cost of paging under delay constraints, Wireless Networks (1995).

[29] C. Rose and R. Yates, Minimizing the average cost of paging and registration: A timer-based method, Wireless Networks 2(2) (1996) 109–116.

[30] B. Suh, J.-S. Choi and J-K. Kim, Mobile location management strategy with implicity location registration, in: *Proc. of IEEE 49th Vehicular Technology Conference* (1999) pp. 2129–2133.

[31] J. Sun and H.C. Lee, Optimal mobile location tracking by multilayered model strategy, in: *Proc. of 3rd IEEE International Conference on Engineering of Complex Computer Systems* (1997) pp. 86–95.

[32] G. Wan and E. Lin, A dynamic paging scheme for wireless communication systems, in: *Proc. of ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 1997)* (1997) pp. 195–203.

[33] H. Xie, S. Tabbane and D. Goodman, Dynamic location area management and performance analysis, in: *Proc. of 43rd IEEE Vehicular Technology Conference* (May 1993) pp. 535–539.

Bhaskar Krishnamachari obtained his B.E. in electrical engineering from The Cooper Union for the Advancement of Science and Art, New York, in 1998. He obtained his M.S. (1999) and Ph.D. (2002) in electrical engineering from Cornell University, where he was the recipient of a three-year graduate fellowship and the prestigious Olin Presidential Fellowship. In fall 2002, Bhaskar Krishnamachari became an Assistant Professor in the Department of Electrical Engineering-Systems at University of Southern California. He currently heads the Autonomous Networks Research Group at USC. Dr. Krishnamachari's present research is focused on discovering fundamental principles and designing efficient mechanisms for self-configuration and information extraction in wireless ad-hoc and sensor networks. He is a member of the Tau Beta Pi and Eta Kappa Nu honor societies.
E-mail: bkrishna@usc.edu

Rung-Hung Gau received his B.S.E.E. from National Taiwan University, Taipei, Taiwan, ROC, in 1994 and his M.S.E.E. from University of California at Los Angeles, Los Angeles, CA, USA, in 1997. In 2001, he earned his Ph.D. degree from School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, USA. In 2002, he became an Assistant Professor at the Institute of Communications Engineering, National Sun Yat-Sen University, Kaoshiung, Taiwan, ROC. His research interests include mobility and resource management in PCS networks, multicast flow control, quality of service issues and application of optimization techniques in network problems.
E-mail: runghung@mail.nsysu.edu.tw

Stephen B. Wicker is Professor and Associate Director for Research of the School of Electrical and Computer Engineering. He is a co-author of *Codes, Graphs, and Iterative Decoding* (Kluwer, 2003), *Turbo Coding* (Kluwer, 1999), and *Reed–Solomon Codes and Their Applications* (IEEE Press, 1994). He is also the author of *Error Control Systems for Digital Communication and Storage* (Prentice Hall, 1995). Professor Wicker teaches and conducts research in wireless information networks, digital communication systems, and error control coding. His current work focuses on complex adaptive networks, with an emphasis on the interplay between game theory, graph theory, and the design of distributed control algorithms. Professor Wicker has served as a consultant in wireless telecommunication systems and error control coding for various companies and governments in North America, Europe, and Asia. He is also the recipient of several teaching awards.
E-mail: wicker@ece.cornell.edu

Zygmunt J. Haas received his B.Sc. in electrical engineering in 1979 and M.Sc. in electrical engineering in 1985. In 1988, he earned his Ph.D. from Stanford University and subsequently joined AT&T Bell Laboratories in the Network Research Department. From September 1994 till July 1995, Dr. Haas worked for the AT&T Wireless Center of Excellence, where he investigated various aspects of wireless and mobile networking, concentrating on TCP/IP networks. As of August 1995, he joined the faculty of the School of Electrical and Computer Engineering at Cornell University. Dr. Haas is an author of numerous technical papers and holds fifteen patents in the fields of high-speed networking, wireless networks, and optical switching. He has organized several Workshops, delivered tutorials at major IEEE and ACM conferences, and serves as editor of several journals and magazines, including the IEEE Transactions on Networking and the IEEE Transactions on Wireless Communications. He has been a guest editor of IEEE Journal on Selected Areas in Communications issues on Gigabit Networks, Mobile Computing Networks, and Ad-Hoc Networks. Dr. Haas is a Senior Member of IEEE, a voting member of ACM, and the Chair of the IEEE Technical Committee on Personal Communications. His interests include: mobile and wireless communication and protocols.
E-mail: haas@ece.cornell.edu