Concurrently Searching for Mobile Users in Cellular Networks

Rung-Hung Gau, *Member, IEEE,* and Zygmunt J. Haas, *Senior Member, IEEE*

***Abstract*—In today's systems, upon arrival of calls to mobile users, the system attempts to locate the users sequentially (one by one) through a paging operation. In this letter, we propose to concurrently search for a number of mobile users in a mobile network, significantly reducing the cost of locating mobile users. The reduction in the paging costs due to such a concurrent search can be quite substantial, depending on the knowledge of the probabilities of the users' locations, the number of cells in the network, and the number of mobiles to be located. Additionally, we propose a low-complexity heuristic that reduces the average paging cost by 25%, in the case of no knowledge of probabilities of the mobiles' locations. With such knowledge, further reduction in the average paging costs of up to 90% can be achieved.**

***Index Terms*—Cellular networks, concurrent search, mobility management, paging.**

I. Introduction

In a cellular network, when a call to a mobile user arrives, a mobility management scheme is responsible for finding the current cell in which the mobile user resides. Typically, a mobility management scheme constitutes of a location update scheme and a paging scheme [1].

There is an intrinsic tradeoff between location update and paging. It is possible to see paging as a more fundamental operation than location update. However, as pointed out in [2], "the majority of the research on location management has actually focused on location update schemes, assuming some obvious version of the paging algorithm."

The concept of dividing a location area into paging zones was described in [3] and [4]. Lyberopoulos [5] proposed to page the cell that a mobile registered with most recently, and then page all other cells in the location area, if necessary. Akyildiz, Ho, and Lin [6] proposed a mobility tracking scheme that combines a movement-based location update policy with a selective paging scheme. Rose and Yates [7] proved that to minimize the average paging cost, the cells with the higher probabilities must be paged before the cells with the lower probabilities. Krishnamachari, Gau, Wicker, and Haas [8] proposed an efficient algorithm to solve the problem of minimizing the average paging cost under the worst-case paging delay constraint. Abutaleb and Lee [9] showed that the problem of minimizing the average paging cost under the mean paging delay constraint can be solved in $O(2^n)$ time.

Rezaiifar and Makowski [10] proposed a paging scheme based on the theory of optimal search. In their scheme, the "Smart Distributor" processes a number of paging requests in a time slot. However, as pointed out in their paper, it is possible that the "Smart Distributor" assigns too many paging requests to a base station and therefore some paging requests are blocked. In this paper, we propose a nonblocking paging scheme in which that once a paging request is assigned to a base station, the paging request will be successfully processed.

II. Concurrent Search

In this letter, we study the problem of concurrently locating a number of mobile users. Suppose there are n cells in the network and there are k mobile users to be located within k time slots. A straightforward sequential paging scheme is to page all cells at time slot i to locate mobile user i, where $1 \leq i \leq k$. This scheme requires $k \cdot n$ paging messages. Given the probabilistic information about the locations of mobile users, it is possible to reduce the average number of required paging messages. For simplicity, it is assumed that the locations of mobile users are statistically independent. Let $p(i,j)$ be the probability that mobile user i is in cell j, when a call to mobile user i arrives. Without loss of essential generality, it is assumed that $p(i,j) > 0, \forall i \in \{1,2,\ldots,k\}$, $j \in \{1,2,\ldots,n\}$. Let P be a $k \times n$ probability matrix such that $[P]_{i,j} = p(i,j)$. A $k \times n$ matrix M is said to be a probability matrix if $\forall 1 \leq i \leq k$, $1 \leq j \leq n$, $[M]_{i,j} \in [0,1]$ and $\forall 1 \leq i \leq k$, $\sum_{j=1}^{n} [M]_{i,j} = 1$. An information-theoretical approach to derive the probability matrix P could be found in [2].

Consider the following simple example to illustrate the efficiency of a concurrent search scheme. Suppose that there are only two cells in the system and two users are to be located. Assume that the system has no prior knowledge about the location of the users; i.e., each user can reside in each one of the two cells with probability 0.5. Using sequential paging, the system would page and locate user 1 in the first time slot. Then, the system would page and locate user 2 in the second slot. Thus, total of four pages would be required.

On the other hand, if the system were to page in the first slot for user 1 in cell 1 and for user 2 in cell 2, there is 50% probability that any of the two users would be found in the first time slot. If a user is not found in the first time slot, the system would page for the user in the "other" cell in the second time slot. Thus, on the average, a user will be located with probability 0.5 in one time slot and with probability 0.5 in two time slots;

Manuscript received December 4, 2002. The associate editor coordinating the review of this letter and approving it for publication was Prof. D. P. Taylor. This work was supported in part by the DoD Multidisciplinary University Research Initiative (MURI) program administered by the Office of Naval Research under Grant N00014-00-1-0564 and by the National Science Foundation under Grant ANI-9805094.

R.-H. Gau is with the Institute of Communications Engineering, National Sun Yat-Sen University, Kaoshiung, Taiwan, R.O.C. (e-mail: runghung@mail.nsysu.edu.tw).

Z. J. Haas is with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14853 USA (e-mail: haas@ece.cornell.edu).

Digital Object Identifier 10.1109/LCOMM.2003.813995

i.e., on the average, 1.5 pages would be require per user, or 3 pages for the two users—a saving of 25% over the sequential paging.

If there are n cells in the system and n users to be located, the above *concurrent* search scheme would require $(n(n+1))/(2)$ total pages, while the sequential paging scheme requires $n \cdot n = n^2$ pages; i.e., a saving of 50% for large n! Of course, with some probabilistic knowledge of the mobiles' locations, this saving can be significantly larger.

III. The Conditional Probability Heuristic Algorithm

In this work, the conditional probability heuristic algorithm is proposed to concurrently search for and locate k mobile users in a network composed of n cells within k time slots. As in [5]–[7], [9], it is assumed that the mobile users will not move to other cells during k time slots. We first define some terms that we use in our conditional probability heuristic algorithm. The variable $s(i,t)$ represents the state of the ith mobile user at the beginning of time slot t. If mobile user i has not been located at the beginning of time slot t, then $s(i,t)=0$. Otherwise, $s(i,t)=1$. Since none of the mobile users are located at the beginning of time slot 1, $s(i,1)=0$, where $1 \leq i \leq k$. The matrix $A(t)$ stores the up-to-date probabilistic information about the locations of mobile users from the viewpoint of the system. Initially, $A(1)=P$, where P is the probability matrix defined earlier. At the end of time slot t, $A(t+1)$ is calculated based on $A(t)$ and the results of paging mobile users in time slot t. The variable $\phi_j(t)$ is the index of the mobile user that is paged by the base station of cell j in time slot t.

The conditional probability heuristic algorithm is presented as follows.

Step 0) Initially, $t=1$, $A(1)=P$ and $s(i,1)=0$, $\forall 1 \leq i \leq k$.

Step 1) At the beginning of time slot t, for each j, where $1 \leq j \leq n$, choose $\phi_j(t)$ such that $[A(t)]_{\phi_j(t),j} = \max_{1 \leq i \leq k}[A(t)]_{i,j}$. Then, for each j, where $1 \leq j \leq n$, if $\max_{1 \leq i \leq k}[A(t)]_{i,j} > 0$, the base station of the jth cell pages mobile user $\phi_j(t)$.

Step 2) At the end of time slot t, calculate $s(i,t+1)$'s as follows:

If mobile user i is located in time slot t, then $s(i,t+1)=1$.

Otherwise, $s(i,t+1)=s(i,t)$.

Step 3) Step 3: Update $A(t+1)$ as follows:

1) If $s(i,t+1)=1$, then $[A(t+1)]_{i,j}=0$, $\forall 1 \leq j \leq n$.
2) If $s(i,t+1)=0$, then
   1) $[A(t+1)]_{i,j}=0, \forall j$, such that $\phi_j(t)=i$.
   2) $[A(t+1)]_{i,j} = (([A(t)]_{i,j})/(1-\sum_{\alpha:\phi_\alpha(t)=i}[A(t)]_{i,\alpha}))$, $\forall j$, such that $\phi_j(t) \neq i$.

Step 4) Step 4: If $t=k$ or $A(t+1)=0$, stop. Otherwise, increase the value of t by one and then go to step 1.

In what follows, we provide some remarks related to the above algorithm. It should be noted that the algorithm is valid irrespective of the total number of cells in the network and the total number of mobile users to be located. In step 0, the initial states of all mobile users are zeros, since none of the mobile users have been located. In step 1, for every cell, one mobile user is selected to be paged. The selection is based on the probabilities to find mobile users in the cell. In every time slot, a base station pages the mobile user with the highest conditional probability to be located.

In step 2, mobile users with indexes $\phi_j(t)$'s, where $1 \leq j \leq n$, are paged. If mobile user i is located in time slot t, $s(i,t+1)=1$. Otherwise, the state of mobile user i remains unchanged.

In step 3, probabilistic information about the locations of mobile users is updated based on the paging results in the previous time slot. If mobile user i has been located in time slot t, it is unnecessary to search for the mobile user in subsequent time slots and therefore $[A(t+1)]_{i,j}=0$, $\forall 1 \leq j \leq n$. On the other hand, if mobile user i has not yet been found in time slot t, the probabilistic information about the location of the mobile user should be updated. If $\phi_j(t)=i$, the system has searched cell j at time slot t for mobile user i and should not search cell j again. Therefore, $[A(t+1)]_{i,j}=0, \forall j$ such that $\phi_j(t)=i$. If $\phi_j(t) \neq i$, given that mobile user i is not in any one of the cells that are paged in the first t time slots, the probability that mobile user i is in cell j is equivalent to $(([A(t)]_{i,j})/(1-\sum_{\alpha:\phi_\alpha(t)=i}[A(t)]_{i,\alpha}))$, based on the definition of condition probability, $Prob(A|B) = ((Prob(A \cap B))/(Prob(B)))$.

In step 4, the algorithm checks if it should stop. Since a base station does not page any mobile more than once during the first k time slots and there are only k mobiles to be located, a base station is always able to page all k mobiles in the first k time slots. Therefore, all k mobiles will be located within k time slots and the algorithm is able to stop when it has run for k iterations.

The conditional probability heuristic algorithm is illustrated by the following example. Suppose

$$P = \begin{bmatrix} 0.4 & 0.3 & 0.2 & 0.1 \\ 0.3 & 0.25 & 0.25 & 0.2 \\ 0.1 & 0.2 & 0.3 & 0.4 \end{bmatrix}.$$

Then, $A(1)=P$ and $s(1,1)=s(2,1)=s(3,1)=0$. Since $[A(1)]_{1,1} > [A(1)]_{2,1} > [A(1)]_{3,1}$, $\phi_1(1)=1$. Similarly, $\phi_2(1)=1$, $\phi_3(1)=3$ and $\phi_4(1)=3$. Therefore, in time slot 1, the base station in cell 1 and the base station in cell 2 page mobile user 1, while the base station in cell 3 and the base station in cell 4 page mobile user 3. With probability $[A(1)]_{1,1}+[A(1)]_{1,2}=0.4+0.3=0.7$, mobile user 1 is located in time slot 1. Similarly, with probability $[A(1)]_{3,3}+[A(1)]_{3,4}=0.3+0.4=0.7$, mobile user 3 is located in time slot 1. For illustration purposes, it is assumed that in the first time slot, mobile user 1 is located, while mobile user 3 is not located. Therefore, $s(1,2)=1$, $s(2,2)=0$, and $s(3,2)=0$.

$A(2)$ is derived based on step 3 of the algorithm. Since $s(1,2)=1$, elements in the first row of $A(2)$ are all zeros. Since $\sum_{\alpha:\phi_\alpha(1)=2}[A(1)]_{2,\alpha}=0$, the second row of $A(2)$ is identical to the second row of $A(1)$. Furthermore, $\sum_{\alpha:\phi_\alpha(1)=3}[A(1)]_{3,\alpha}=[A(1)]_{3,3}+[A(1)]_{3,4}=0.3+0.4=0.7$ and therefore $[A(2)]_{3,1}=(([A(1)]_{3,1})/(1-0.7))=(1/3)$.

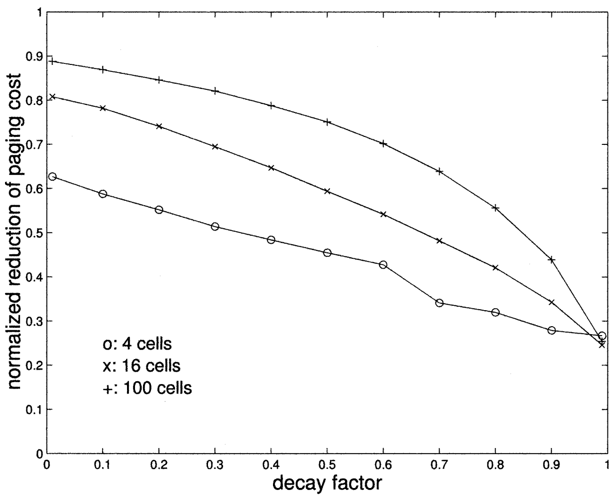

Fig. 1. Performance of a concurrent search algorithm.

Similarly, $[A(2)]_{3,2} = (0.2/0.3) = (2/3)$. In addition, $[A(2)]_{3,3} = [A(2)]_{3,4} = 0$. Then,

$$A(2) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.3 & 0.25 & 0.25 & 0.2 \\ \frac{1}{3} & \frac{2}{3} & 0 & 0 \end{bmatrix}.$$

The above procedure is repeated until all the mobile users have been located. The computational complexity of the algorithm is $O(k^2 \cdot n)$.

IV. Performance Evaluation

In this section, simulations are used to study the performance of the conditional probability heuristic algorithm. The simulation model is described as follows. The network contains $n = w^2$ cells, which are indexed by (α, β), where $0 \le \alpha, \beta \le w-1$. The distance between two cells indexed by (α_1, β_1) and (α_2, β_2) is denoted by $d_1(\alpha_1, \beta_1, \alpha_2, \beta_2)$ and is defined to be $|\alpha_1 - \alpha_2| + |\beta_1 - \beta_2|$. To create the tth row of a $k \times n$ probability matrix P, a cell with index (α^*, β^*) is at first randomly selected such that α^* and β^* are two independent random variables, which are uniformly distributed in $\{0, 1, 2, \ldots, w-1\}$. Then, the probability that mobile user t is in a cell indexed by (α, β) is set to be

$$\frac{r^{d_1(\alpha,\beta,\alpha^*,\beta^*)}}{\sum_{i=0}^{w-1}\sum_{j=0}^{w-1} r^{d_1(i,j,\alpha^*,\beta^*)}}$$

where the decay factor, r, is a real number in [0, 1]. The decay factor is a parameter that determines the uniformity of the probability density function of the location of a mobile user. In our simulations, $w \in \{2, 4, 10\}$ and $k = 16$.

Let $C_{seq}(P) = k \cdot n$ be the paging cost when the sequential paging scheme is used to locate k mobile users in a cellular network composed of n cells. Let $C_{cph}(P)$ be the corresponding average paging cost when the conditional probability heuristic is used. The normalized reduction of paging cost is defined to be $((C_{seq}(P) - C_{cph}(P))/(C_{seq}(P)))$.

Fig. 1 shows the performance of the conditional probability heuristic algorithm when there are 16 mobile users to be located and the total number of cells in the network is 4, 16, or 100. Typically, as the total number of cells increases, the normalized reduction of the average paging cost increases. In addition, as the decay factor increases, the normalized reduction of the average paging cost decreases.

V. Conclusions

In this letter, we propose to concurrently search for and locate k mobile users in a cellular network within k time slots based on the probabilistic information about the locations of the mobile users. The reduction in the paging costs due to such a concurrent search can be quite substantial, depending on the knowledge of the probabilities of the users' locations, the total number of mobile users to be located, and the total number of cells in the network. Additionally, we propose a low computational complexity algorithm (the *Conditional Probability Heuristic*). Simulation results indicate that the Conditional Probability Heuristic can reduce the average paging cost by 25%, in the case of no knowledge of the probabilities of the mobiles' locations. If such information is available, further reduction in the average paging cost of up to 90% can be achieved. In practice, estimations of the location probabilities of mobile users could be used as the input of the proposed conditional probability heuristic algorithm.

The concurrent search approach could be extended to serve calls with different arrival times. However, due to the space limitation, extensive analytical and simulation results are omitted and will be included in [11].

References

[1] Y.-B. Lin and I. Chlamtac, *Wireless and Mobile Network Architectures*. New York: Wiley, 2001.
[2] A. Bhattacharya and S. K.. Das, "LeZi-update: An information-theoretic approach to track mobile users in PCS networks," in *ACM MobiCom Conf.*, 1999, pp. 1–12.
[3] D. Munoz-Podrguez, "Cluster paging for traveling subscribers," in IEEE Vehicular Technology Conf. 1990, May, 6–9, pp. 748–753, to be published.
[4] D. Plassmann, "Location management strategies for mobile cellular networks of 3rd generation," in *IEEE Vehicular Technology Conf.*, June 1994, pp. 649–653.
[5] 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 Trans. Veh. Technol.*, vol. 44, pp. 543–553, Aug. 1995.
[6] I. F. Akyildiz, J. S. M. Ho, and Y.-B. Lin, "Movement-based location update and selective paging for PCS networks," *IEEE/ACM Trans. Networking*, vol. 4, pp. 629–638, Aug. 1996.
[7] C. Rose and R. Yates, "Minimizing the average cost of paging under delay constraints," *ACM/Kluwer Wireless Networks*, vol. 1, no. 2, pp. 211–219, 1995.
[8] B. Krishnamachari, R.-H. Gau, S. B. Wicker, and Z. J. Haas, "Optimal sequential paging in cellular networks," in *IEEE Vehicular Technology Conf.*, Fall 2001, pp. 92–96.
[9] A. Abutaleb and V. O. K. Li, "Paging strategy optimization in personal communication systems," *ACM/Kluwer Wireless Networks*, vol. 3, no. 3, pp. 195–204, 1997.
[10] R. Rezaiifar and A. M. Makowski, "From optimal search theory to sequential paging in cellular networks," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1253–1264, Sept. 1997.
[11] R.-H. Gau and Z. J. Haas, "Concurrent search of mobile users in cellular networks," *IEEE/ACM Trans. Networking*, to be published.