Distributed cell-centric neighborhood-related location area planning for PCS networks

Shih-Lin Wu, Jui-Te Chang, Jen-Jee Chen

Abstract

An intelligent location area planning (LAP) scheme should consider the frequent replannings of location areas (LAs) due to changes in user distribution and mobility patterns along with optimization of location management costs, including location updating and paging costs. Most schemes proposed in the literature are designed through centralized techniques, thus requiring more computing time to plan the LAs. Frequent replannings to accommodate environmental changes make the situation worse. As to the optimization of location management costs, most proposed schemes consider the inter-cell crossing rate as one of the key factors in determining the optimal partitions. In some cases, the inter-cell crossing rate may lead to an unsatisfactory outcome. Another problem is the ping-pong effect which is caused by the fixed borders between any two of LAs. In this paper, we propose a distributed cell-centric neighborhood-related LAP scheme in which each cell acts as the center of an LA and in which highly correlative neighboring cells are bundled into the LA if mobile terminals (MTs) remain within the LA long enough to reduce costs. Moreover, the ping-pong effect will be alleviated because MTs always locate at the center cell of an LA whenever a new location update is performed. Finally, the scheme can be implemented in a distributed manner so the computing time incurred by frequent replannings can be reduced. Simulation results show that our scheme indeed exhibits excellent results.

1. Introduction

In recent years, Personal Communication Service (PCS) networks have been broadly deployed with great success. The planning of location areas (LAs) is a key function in the location management of PCS networks, for which the whole service area is divided into several LAs with each LA consisting of one or more cells. Setting up a call requires the tracking of the location of a mobile terminal (MT). This tracking process is called location management and includes two procedures: location update and paging. Whenever an MT leaves the current LA, the action of location update is performed by registering the MT's new LA to the network's location server. Then, upon receiving an incoming call for the MT, the member cells of the registered LA are paged to determine the particular cell in which the MT is located. Thus, the location management cost includes the signaling traffic from the location update and paging signal.

Minimizing costs from location updating and paging is one of the most important issues for location management. In fact, there is a cost trade-off between the two procedures. When an LA contains a large number of cells, the number of location updates decreases and the number of paging cells increases. On the other hand, when an LA contains a small number of cells, the number of location updates increases and the number of paging cells decreases. Thus, partitioning cells into LAs to minimize the location management cost is a critical issue for implementation.

Considerable research has been conducted on location area planning (LAP) which can be classified into two types: static and dynamic. Static LAP [28,31] permanently partitions cells into fixed size LAs for all MTs. These schemes are simple and easy to implement, but they are not efficient because they cannot adapt to changes in user population, mobility, and call arrival rate. Their efficiency is also reduced by the ping-pong effect because MTs located close to the boundaries of LAs may have more back and forth movements between the LAs, thus incurring unnecessary location updates [12,19,23,29]. Dynamic LAPs are proposed to allow an individual MT to overcome the shortcomings in the static schemes and minimize location management costs. Basically, dynamic schemes plan the LAs and determine their size dynamically according to the MT's mobility and call arrival rate. Thus, the location management cost for each MT is optimized. Some well-known dynamic schemes are distance-based [7,22,34], movement-based [2,7,13,16,17], time-based [15,25], and profile-based [9,27,28,31]. Several studies [6,24,27,28,31] have indicated that, among
dynamic schemes, profile-based schemes incur the lowest location management cost. However, many recent researches [6,20,27,33] have pointed out that the existing PCS systems make it difficult to plan LAs for individual MTs because the huge number of MTs results in many additional costs. As a result, the dynamic schemes cannot be easily implemented in currently-operating PCS networks.

Bejerano et al. [6] point out that people usually stay in specific zones for long periods until their work or activity in the zone is complete. For example, students and teachers stay in their campus during the day; tourists visit scenic spots for a given duration; doctors and patients usually spend most of their time in the hospital and the surrounding area. Location management would be more efficient if LAs could be intelligently partitioned and mapped into specific zones. To exploit Bejerano’s observation, recent works, [6,8] propose new LAP schemes to efficiently divide cells into LAs. Most of the proposed schemes treat the inter-cell crossing rate as one of the key factors for determining the optimal cell set. As for the inter-cell crossing rate, merely counting the number of crossing MTs between cells is too simple to reflect real conditions and may lead to an unsatisfactory outcome in some cases. In the following section, we illustrate this paradox by example.

In long term evolution (LTE) networks, the central concept of mobility management is automatic reconfiguration [1]. An intelligent LAP scheme should adapt to changeable user distributions and mobility patterns and perform the replanning of LAs dynamically and frequently. Most proposed schemes are designed according to centralized techniques requiring more computing time for LA planning. The situation would be worse as frequent replannings are needed to accommodate the changeable environment.

In this paper, we propose a distributed cell-centric neighborhood-related LAP scheme to minimize location management costs while alleviating the ping-pong effect. The basic idea behind our scheme is that each cell acts as the center of an LA and bundles highly correlating neighbor cells into the LA. Cells with highly correlated relative relations have the lower location management costs. As stated above, we will use the LA residence time (LART) instead of the inter-cell crossing rate. In other words, we want to find a neighboring cell to be included in the LA if the average residence time of MTs dwelled within the area between the neighboring cell and the LA is the longest. Moreover, the ping-pong effect will be alleviated because an MT always locates at the center of a new LA whenever a new location update is performed. Finally, the scheme can be easily implemented by a distributed method because each cell plans its own LA and only needs to know its neighbors' information. As a result, we can significantly reduce the computing time incurred by frequent replannings. Through simulations, we will demonstrate the efficiency of our LAP scheme.

The rest of this paper is organized as follows. Related works are introduced in Section 2. In Section 3, we present the system model and the proposed cell-centric neighborhood-related LAP scheme. Simulation results with different cell topologies are shown in Section 4. Finally, conclusions are given in Section 5.

2. Related works

The major impact of LAs with nonoverlapped areas is the ping-pong effect [12]. The benefits of overlapped LAs have been illustrated in [5,21,32]. This is an important issue for determining the size of the overlapped LAs. The smaller the overlapped area is, the greater the ping-pong effects. However, the greater overlap requires a greater number of LAs, which increases the network management burden. To find the optimal overlap, Ma [19] developed a theoretical model to compute the location management cost under different LA sizes. However, this work cannot be applied to the real world because Ma’s MT mobility is assumed as a random walk model. Other ways to alleviate the ping-pong effect let each cell belong to the multiple LAs, such as in the Two Location Area (TLA) [18] and Three Location Area (TrLA) [12] schemes. This kind of schemes increases the paging cost because multiple LAs means more cells that need to be paged.

The main goal of LAP is the partition of the whole service area into LAs to minimize the total location management cost. Because LAP is NP-hard [6], many heuristic algorithms have been proposed using various techniques, such as simulated annealing methods [10,11], matrix decomposition schemes [3], and genetic algorithms [14,30]. For covering highways and railroads, Saraydar et al. [26] proposed a one-dimensional LAP which can significantly reduce the signaling overhead of the entire system. Bejerano et al. [6] developed a clustering algorithm to minimize the location management cost by growing LAs around an initial cell. They then try to further reduce the cost through a series of greedy cell exchanges around the boundary of the neighboring LAs with a polynomial time complexity. Similar to Bejerano’s approach, Chew et al. [8] derived a different cost function for location management using the non-linear binary integer programming model. The cost function can be divided into two parts: the constant cost which is fixed no matter what kinds of partitions are made, and the variable cost which is determined by what kind of partitions is made. The variable cost is

\[ r_g = C_p(l_i + l_j) - C_w(\mu_i + \mu_j), \]  

where \( l_i \) is the call arrival rate of cell \( i \), \( \mu_i \) is the inter-cell crossing rate of MTs moving from cell \( i \) to cell \( j \), \( C_p \) is the unit paging cost, and \( C_w \) is the unit location update cost. The physical meaning of (1) is that once cell \( i \) and \( j \) are in the same LA, the paging cost increases. Since non-linear binary integer programming is NP-hard, the work provides a heuristic algorithm to find an optimal LA partition. The proposed algorithm, called non-linear binary integer programming (NBIP), performs LAP in the following steps. First, choose the pair of cells with the most negative value among all \( r_g \) to form a new LA, named as LA1. In the second step, for each of the neighboring cells, calculate the summation of all the cells in LA1 to the neighbor. Then include the neighboring cell with most negative value \( r_g \) into LA1. Repeat the second step until all of the neighboring cells have a positive value, thus completing LA1. The algorithm then goes back to the first step to construct other LAs until all cells have been partitioned into LAs. Our comments on these LAPs are as follows. First, these LAPs incur the ping-pong effect due to the presence of nonoverlapped areas. Second, when a cell is added to an LA, it would not be considered again in next selection even if it is also highly correlated with other cells or LAs. Third, these LAPs belong to the centralized approach meaning considerable computing time is incurred when the number of cells in the network becomes tremendously large. Finally, it is paradoxical that most LAPs take the rate of inter-cell traffic between two cells as the inter-cell crossing rate and give high preference to bind the two cells in an attempt to reduce the update rate. Many transportation systems, highways, railways, and rapid transit systems present two-way directional traffic. Usually, it is very heavy about the inter-cell traffic between two neighboring cells located at the line of transportation systems. In other words, the more inter-cell traffic two cells have, the stronger the attraction with which they should be bound into an LA. This may result in the formation of a large LA in the area of transportation systems. However, high inter-cell traffic also happens while people quickly pass through those cells. For example, passengers on an urban rapid transit system pass many stations without getting off the train before arriving at their destinations, resulting in a larger location management cost because a large LA is easily
formed but passengers may stay for only a short time. The relevance of two cells should take into account whether the same group of people has stayed in these cells for long enough to derive a benefit from the reduced update cost for covering the increased paging cost. We will demonstrate this paradox through simulations.

3. Distributed cell-centric neighborhood-related LAP scheme

In this section, our distributed cell-centric neighborhood-related LAP scheme is presented in detail. Section 3.1 introduces the basic concept of the scheme. Section 3.2 presents the system model in detail and derives the expected location management cost of an LA. Section 3.3 presents our proposed scheme.

3.1. Design concept

The design concept of our proposed LAP scheme is that each cell acts as the center of an LA and groups highly correlated neighboring cells into the LA if the cells and the center cell cover the same specific zone. Cells in a specific zone will be highly correlated to each other in the sense that once MTs move into the specific zone they will stay long enough time to reduce the location update cost, thus reducing location management costs. The proposed scheme can be implemented by a distributed method because each cell plans its own LA and only needs to know its neighbors’ information.

Towards this end, each cell works as the first member of its own LA and evaluates the benefit of each neighbor cell. The benefit between the LA and a neighbor cell is the reduced location update cost subtracts the increased paging cost if the LA includes the neighboring cell. Location update cost is reduced because a correlated cell is added into the LA but the paging cost increases because the cell paging number is raised. The former attracts a cell to be added into its LA and the latter confines the LA expansion. Below, we model the function of location management cost, show how a cell evaluates the benefit and demonstrate how our LAP scheme works.

3.2. Modeling location management cost

The parameters of the model are as follows:

- $C_p$: cost for paging a single cell.
- $C_u$: location update cost.
- $|LA_i|$: number of the cells in LA.
- $n$: number of the cells in the system.
- $\lambda_i$: incoming call arrival rate of an MT in cell $i$.
- $t_i$: the expected time of an MT stayed in LA.
- $r_{ij}$: the probability of the MT found in cell $j$ being within LA.
- $P_{ij}$: the transition probability from cell $i$ to cell $j$.
- $C_{LA_i}$: location management cost for an MT residing at $LA_i$.
- $|LA|$: number of the cells in LA.

The location management cost for an MT residing at LA can be obtained from

$$C_{LA} = C_p|LA|\sum_{j\in LA} r_{ij} + C_u\Phi_i,$$

where $\Phi_i = 1/t_i$ is the location update rate of the MT for $LA_i$. Note that the call arrival rate of an MT in $LA_i$ is

$$\sum_{j\in LA_i} r_{ij}.$$

In the following, we show how to derive $r_{ij}$ and $\Phi_i$ for an MT in $LA_i$. This can be modeled as an absorbing Markov chain [19,33,35].

In Fig. 1, each transit state represents a cell in $LA_i$ and the absorbing states are the cells which do not belong to $LA_i$. Let the set $LA_i$ contain all of the absorbing state cells. We want to calculate the expected time an MT will stay in $LA_i$ before entering the absorbing states. For simplicity, let cells 1, 2, ..., and $|LA_i|$ be in $LA_i$. So, cells $|LA_i|+1$, ..., and $n$ are in absorbing states. Then the transition matrix $P$ can be defined as

$$P = \begin{pmatrix}
    p_{11} & \cdots & p_{1|LA_i|} & p_{1(|LA_i|+1)} & \cdots & p_{1n} \\
    \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
    p_{|LA_i|1} & \cdots & p_{|LA_i||LA_i|} & p_{|LA_i|(|LA_i|+1)} & \cdots & p_{|LA_i|n} \\
    p_{(|LA_i|+1)1} & \cdots & p_{(|LA_i|+1)|LA_i|} & p_{(|LA_i|+1)(|LA_i|+1)} & \cdots & p_{(|LA_i|+1)n} \\
    \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
    p_{n1} & \cdots & p_{n|LA_i|} & p_{n|LA_i|+1} & \cdots & p_{nn}
\end{pmatrix},$$

where each element $p_{ij}$ in $P$ is the transition probability from cell $i$ to cell $j$. In fact, the matrix $P$ can be further divided into 4 parts as below,

$$P = \begin{pmatrix}
    Q & A \\
    B & C
\end{pmatrix},$$

where $Q$ is the transition matrix between any two cells in the transit states, $A$ is the transition matrix between any cell in the transit states to any cell in the absorbing states, $B$ is the transition matrix between any cell in the absorbing states to any cell in the transit states, and $C$ is the transition matrix between any two cells in the absorbing states. To solve the residence time of an MT starting from each cell in $LA_i$, we need to calculate the fundamental matrix $N$,

$$N = (I - Q)^{-1},$$

where each element $n_{ij}$ in $N$ is the expected number of times an MT visited from cell $i$ to $j$ before leaving $LA_i$. The detailed derivation of (6) can be found in [35]. Then, the time to reach absorbing states $t_i$ can be derived as

$$t_i = \sum_{j\in LA_i} n_{ij}e,$$

where $e$ is the duration between two state transitions. The probability $r_j$ can be calculated by

$$r_j = \frac{n_{ij}}{\sum_{j\in LA_i} n_{ij}}.$$

In our scheme, each cell works as the first member of its own LA and evaluates the benefit of neighboring cells to decide which neighbor presents the greatest benefit for inclusion in the LA. Below, we show how to calculate the benefit between $LA_i$ and a neighbor cell $k$. Let $LA_i' = LA_i \cup \{k\}$. The mobility management cost of $LA_i'$

![Fig. 1. States of the Markov chain.](image-url)
can be calculated by using (1). For \( \text{LA}_i^j \), the mobility management cost is calculated by

\[
\text{C}_{\text{LA}_i^j} = \text{C}_p(|\text{LA}_i^j| + 1) \left( \sum_{j \in \text{LA}_i^j} r_j \lambda_j \right) + \text{C}_u \Phi_{\text{LA}_i^j}.
\]

As a result, the benefit of including cell \( k \) into \( \text{LA}_i \) can be obtained

\[
\Delta C_k = \text{C}_{\text{LA}_i^j} - \text{C}_{\text{LA}_i} = \text{C}_p \left( |\text{LA}_i^j| + 1 \right) \left( \sum_{j \in \text{LA}_i^j} r_j \lambda_j \right) - \left( |\text{LA}_i| \right) \left( \sum_{j \in \text{LA}_i} r_j \lambda_j \right) + \text{C}_u \left( \Phi_{\text{LA}_i^j} - \Phi_{\text{LA}_i} \right).
\]

\( (10) \)

So, when \( \Delta C_k < 0 \), it is beneficial to include cell \( k \) into \( \text{LA}_i^j \); otherwise, it is not worth adding cell \( k \) into \( \text{LA}_i \).

### 3.3. Proposed algorithm

In our scheme, each cell acts as the first member of its own LA, and each time the LA includes the neighbor cell with the best benefit among all neighbors, i.e., the one with the smallest value \( \Delta C_k \) as calculated by (10). The proposed scheme is performed until \( \Delta C_k \geq 0 \), for all neighbors \( k \). The formal steps are given as below. Note that the scheme will be executed by each cell to form its LA.

**Step 1**: Cell \( i \) includes itself as the first member of \( \text{LA}_i \), i.e., \( \text{LA}_i \leftarrow \{i\} \)

**Step 2**: For all \( k \) neighbors, calculate \( \Delta C_k \).

**Step 3**: Select \( k \) with the smallest \( \Delta C_k \) from all neighbors of \( \text{LA}_i \).

**Step 4**: If \( \Delta C_k < 0 \), \( \text{LA}_i \leftarrow \text{LA}_i \cup \{k\} \), go to Step 2; otherwise output \( \text{LA}_i \) and terminate the procedure.

In the following, an example is given to see how the above LAP algorithm works. Fig. 2 shows that six cells are located in four specific zones, i.e., campus (cells 4 and 5), hospital (cells 2 and 3), business area (cell 0), and highway (cell 1) connecting the hospital and the business area. The number attached to each arrow represents

![Fig. 2. Cells and their transition probabilities in the service area.](image)

<table>
<thead>
<tr>
<th>LAi</th>
<th>Step 1</th>
<th>Step 2</th>
<th>Step 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cells</td>
<td>( \Delta C_i )</td>
<td>Cells</td>
<td>( \Delta C_i )</td>
</tr>
<tr>
<td>( \text{LA}_0 )</td>
<td>0.1</td>
<td>-0.005</td>
<td>0, 1, 2</td>
</tr>
<tr>
<td>( \text{LA}_1 )</td>
<td>1, 2</td>
<td>-3.866</td>
<td>1, 2, 3</td>
</tr>
<tr>
<td>( \text{LA}_2 )</td>
<td>2, 3</td>
<td>-0.672</td>
<td>2, 3, 1</td>
</tr>
<tr>
<td>( \text{LA}_3 )</td>
<td>3, 2</td>
<td>-0.727</td>
<td>3, 2, 1</td>
</tr>
<tr>
<td>( \text{LA}_4 )</td>
<td>4, 5</td>
<td>-0.706</td>
<td>4, 5, 3</td>
</tr>
<tr>
<td>( \text{LA}_5 )</td>
<td>5, 4</td>
<td>-0.306</td>
<td>5, 4, 3</td>
</tr>
</tbody>
</table>

![Fig. 3. Ring topology.](image)
Fig. 4. Specific zone topology.

Fig. 5. Cost comparison in ring topology.

Fig. 6. Cost comparison in specific zone topology.
the transition probability of crossing the cell boundary per transition time period $e$. Assume that $C_p/C_u = 1.5$. Following the benefit function (10) and the LAP scheme, Table 1 shows each LAs step by step expansion while the final partitions for all LAs are underlined.

From Table 1, we can see that the ping-pong effect has been alleviated because our LAP includes natural overlays among the LAs. For example, an MT starts at the business area having $L_{A_0} = \{0,1\}$ and wants to go to the hospital zone. It then needs to register $L_{A_2} = \{1,2,3\}$ while it crosses the boundary between cells.

### Table 2
Numerical results in ring topology.

<table>
<thead>
<tr>
<th>Population</th>
<th>Algorithm</th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
<td>NBIP</td>
</tr>
<tr>
<td>Location update times</td>
<td>26247</td>
<td>19224</td>
<td>52548</td>
<td>38377</td>
<td>78688</td>
<td>57515</td>
</tr>
<tr>
<td>Paging times</td>
<td>149628</td>
<td>324401</td>
<td>296508</td>
<td>644219</td>
<td>447528</td>
<td>971888</td>
</tr>
<tr>
<td>Total signaling cost</td>
<td>280863</td>
<td>420521</td>
<td>559248</td>
<td>836104</td>
<td>840968</td>
<td>1259460</td>
</tr>
<tr>
<td>Reducing ratio</td>
<td>33.21%</td>
<td>33.11%</td>
<td>33.23%</td>
<td>33.10%</td>
<td>33.14%</td>
<td>33.14%</td>
</tr>
</tbody>
</table>

### Table 3
Numerical results in specific zone topology.

<table>
<thead>
<tr>
<th>Population</th>
<th>Algorithm</th>
<th>1000</th>
<th>2000</th>
<th>3000</th>
<th>4000</th>
<th>5000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
<td>NBIP</td>
</tr>
<tr>
<td>Location update times</td>
<td>12539</td>
<td>4928</td>
<td>25210</td>
<td>10048</td>
<td>38046</td>
<td>15033</td>
</tr>
<tr>
<td>Paging times</td>
<td>99491</td>
<td>185181</td>
<td>199085</td>
<td>369212</td>
<td>300571</td>
<td>557536</td>
</tr>
<tr>
<td>Total signaling cost</td>
<td>162186</td>
<td>209821</td>
<td>325135</td>
<td>419452</td>
<td>490801</td>
<td>632701</td>
</tr>
<tr>
<td>Reducing ratio</td>
<td>22.70%</td>
<td>22.49%</td>
<td>22.43%</td>
<td>22.72%</td>
<td>22.36%</td>
<td>22.36%</td>
</tr>
</tbody>
</table>

Fig. 7. Cost comparison with different $C_p/C_u$ ratios in ring topology.

Fig. 8. Cost comparison with different $C_p/C_u$ ratios in specific zone topology.
and age of 10 times and each time runs 24 h. The different populations with mean time 0.83 calls/hour. Each simulation result is the averaged incoming call arrival rate of an MT follows the Poisson distribution between two state transitions, is set to 5 min. Assume that the parameter of LA2 includes cell 1 and 2, the ping-pong effect will not occur if the MT is located at the boundary between cells 1 and 2 and moves back and forth across that boundary.

4. Simulation results

We have developed a simulator to evaluate the performance of the proposed scheme. NBIP [8] are used as a reference for comparison. Two different topologies of the network are employed in simulations and their transition probabilities between cells are shown in Figs. 3 and 4, respectively.

Fig. 3 uses the ring topology which can be thought as a public transit system while Fig. 4 uses the zone topology which is modeled as an area with some specific zones, such as school, hospital, and shopping mall. The parameter used in (7), i.e., the time period between two state transitions, is set to 5 min. Assume that the incoming call arrival rate of an MT follows the Poisson distribution with mean time 0.83 calls/hour. Each simulation result is the average of 10 times and each time runs 24 h. The different populations and \( \frac{C_u}{C_p} \) ratios are examined by the following two simulations.

4.1. Population effect

In the first simulation, the population effect is investigated. The cost ratio of \( \frac{C_u}{C_p} \) is fixed at 5. The relation between population and total location management cost about the two topologies are shown in Figs. 5 and 6. The detailed numerical results are shown in Tables 2 and 3. We can see that the times to perform location update and paging increase as more MTs present in the system. Although the more MTs incur the higher total location management cost, our proposed scheme still presents less cost than NBIP about 20% ~ 30% no matter what population and topology are used. The first reason is that the cell-centric LAP alleviates the ping-pong effect. Before going to explain the second reason, we give an interesting phenomenon that the reducing ratios in ring topology are always better than in specific topology as shown in Tables 2 and 3. The ring topology is to simulate that people always move in one direction and shows the shortcoming in NBIP algorithm. As stated in Section 2, NBIP considers the inter-cell crossing rate between cells as the location update rate such that the more inter-cell crossing rate the two cells have, the more attraction they should be bound into an LA. But the idea is not entirely correct in some cases. For example, the highly inter-cell crossing rate may also happen while people just pass through those cells quickly with small residence time. This is very similar to the scenario that passengers take a train of the rapid transit system in a city and pass many stations without getting off the train before arriving at their destinations. This case results in a higher location management cost because NBIP forms a large LA easily but passengers may stay for a short time period. The relevance of two cells should take into account the total time spent in the cells.

Table 4

<table>
<thead>
<tr>
<th>Ratio</th>
<th>( \frac{C_u}{C_p} = 1:1 )</th>
<th>( \frac{C_u}{C_p} = 1:5 )</th>
<th>( \frac{C_u}{C_p} = 1:10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
</tr>
<tr>
<td>Location update times</td>
<td>66637</td>
<td>48056</td>
<td>26247</td>
</tr>
<tr>
<td>Paging times</td>
<td>59740</td>
<td>85789</td>
<td>148585</td>
</tr>
<tr>
<td>Total signaling cost</td>
<td>126384</td>
<td>133845</td>
<td>279820</td>
</tr>
<tr>
<td>Reducing ratio</td>
<td>5.57%</td>
<td>33.12%</td>
<td>5.53%</td>
</tr>
</tbody>
</table>

Table 5

<table>
<thead>
<tr>
<th>Ratio</th>
<th>( \frac{C_u}{C_p} = 1:1 )</th>
<th>( \frac{C_u}{C_p} = 1:5 )</th>
<th>( \frac{C_u}{C_p} = 1:10 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OURS</td>
<td>NBIP</td>
<td>OURS</td>
</tr>
<tr>
<td>Location update times</td>
<td>38434</td>
<td>51196</td>
<td>12591</td>
</tr>
<tr>
<td>Paging times</td>
<td>69384</td>
<td>64454</td>
<td>99312</td>
</tr>
<tr>
<td>Total signaling cost</td>
<td>107818</td>
<td>115650</td>
<td>162267</td>
</tr>
<tr>
<td>Reducing ratio</td>
<td>6.77%</td>
<td>22.66%</td>
<td>8.61%</td>
</tr>
</tbody>
</table>
Thus, we have better performance in the ring topology. In our algorithm, we calculate the cost for covering the increasing paging cost. The cells can be assigned independently and concurrently at every cell, thus significantly reducing time complexity and cost. We show how the incoming arrival call rate, location update rate, and the ratio of 

\[ \frac{C_u}{C_p} \]  

effect the performance in location management. Simulation results indicate the proposed algorithm outperforms NBIP for various populations and ratios of \( \frac{C_u}{C_p} \).

### 4.2. Ratio \( \frac{C_u}{C_p} \) Effect

In the second simulation, we adjust the ratio of \( \frac{C_u}{C_p} \) to investigate the effect of the two algorithms. The population for this simulation is fixed at 1000 MTs. As shown in Figs. 7 and 8 as well as Tables 6 and 7 show the LA planning results of various populations and ratios of \( \frac{C_u}{C_p} \). The reason is the same as the previous simulation. This is more obvious in the ring topology. Thus, our algorithm always presents lower cost than NBIP.

### 5. Conclusions

In this paper, we propose a new location management scheme to plan the LA through cell-centric LAP which can alleviate the ping-pong effect and minimize the total location management cost. To the best of our knowledge, the idea of cell-centric location area planning, which is a tradeoff between static and dynamic LAPS, has not previously appeared in the literature. The proposed cost function used in our scheme is a distributed method which can be executed independently and concurrently at every cell, thus significantly reducing time complexity and cost. We show how the incoming arrival call rate, location update rate, and the ratio of \( \frac{C_u}{C_p} \) affect the performance in location management. Simulation results indicate the proposed algorithm outperforms NBIP for various populations and ratios of \( \frac{C_u}{C_p} \).

### References