Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Networks 46 (2004) 817–827
COMPUTER NETWORKS
www.elsevier.com/locate/comnet

Improving the accuracy of cell-based positioning for wireless networks ☆

Rong-Hong Jan *, Hung-Chi Chu, Yi-Fang Lee

Department of Computer and Information Science, National Chiao Tung University, 1001 Ta Hsueh Road, Hsinchu 30050, Taiwan

Received 15 October 2003; received in revised form 8 June 2004; accepted 11 June 2004
Available online 29 July 2004

Responsible Editor: N. Georgenas

Abstract

One fundamental issue for location-based services and applications is solving the location-sensing problem, i.e., determining where a given node is physically located in a network. In a previous paper, we have presented a location-sensing method, called the cell-based positioning method, and its positioning accuracy for the wireless networks with a hexagonal structure and mesh structure. Unfortunately, in real situations, a wireless network may not have a hexagonal or mesh structure. Thus, in this paper we consider networks with irregular structures and present an algorithm to determine the positioning accuracy of the cell-based method in such networks. In addition, we use the simulated annealing (SA) method to determine the locations and transmission ranges of base stations in order to achieve the best possible positioning accuracy. Simulation results show that the accuracy can be improved up to 30% by the SA method. The results are useful for deploying a wireless network for location-based applications.

Keywords: Location-sensing; Location determination; Location-based applications

1. Introduction

Wireless communication is a popular trend today because it lets users communicate easily with each other at almost any time and place. One of the most important applications for wireless networks, location-based services, allows mobile users to receive services based on their geographic locations. In recent years, more and more location-based services and applications have been

☆ This work was supported in part by the Lee and MTI Center for Networking Research, NCTU, Taiwan and the Ministry of Education and National Science Council, Taiwan, ROC, under grants 89-E-FA04-1-4 and NSC 92-2219-E-009-012, respectively.
* Corresponding author. Tel.: +886 3 5731637; fax: +886 3 5721490.
E-mail address: rhjan@cis.nctu.edu.tw (R.-H. Jan).

1389-1286/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2004.06.023

developed for mobile users. These services include emergency rescue, resource tracking and management, tour guide [1], location-sensitive billing, points of interest and so on.

One fundamental issue for location-based services and applications is the location-sensing problem, i.e., determining the physical location of a node. Many papers [2–13] have discussed the location-sensing methods. We classify these methods into two broad categories based on where the position coordinates of a handset are determined. If the handset collects signals from the network and determines the location, it is a *handset-based* method. In contrast, if some location equipment is installed at the base stations (BSs) to collect the signal direction or timing of the handset, and then a centralized server determines the handset's location. This approach is a *network-based* method.

1.1. Handset-based methods

Global Positioning System (GPS) [2,3] is a typical handset-based method. It calculates the locations at the handset by measuring the time and distance between a receiver and at least three satellites. GPS has a high position accuracy. However, times to first fix are generally longer, as GPS needs to measure distances to a minimum of three satellites and the processing time is thus much longer. Two modified GPS methods, assisted GPS [4] and differential GPS [5–7] are presented to improve the processing speed for location determining and reduce the power consumption of mobile station.

1.2. Network-based methods

Angle of Arrival (AOA) and Time Difference of Arrival (TDOA) are two of the most widely known network-based location-sensing methods [8,9]. AOA systems estimate the AOA of the handset signal at two or more BSs and apply simple triangulation to determine the handset's location. TDOA systems, on the other hand, use radio frequency (RF) receivers installed at multiple BSs to measure signal times of arrival data and estimate the handset's location. Variations of these approaches are discussed in [10].

Note that network-based methods need to install location equipment at the BSs. In contract, GPS solutions need to add the GPS receivers to handsets. Although GPS capabilities can be included in the handset chips with little extra cost, end users still have to upgrade their handsets. In [13], Chu and Jan present a simple, low-cost method for location-sensing, called the cell-based location-sensing method. In the cell-based method, the handset gathers all of the BS signals that it receives and transmits the BS identifications (IDs) to the location server. Based on these BS IDs, the server can then determine the location of the handset. Thus, the cell-based method only requires several lines of code on the Subscriber Identity Module (SIM) card to get the list of BSs within range. The code can be embedded on the SIM when the SIM is issued or delivered over-the-air to the SIM via Short Messaging Service (SMS) messaging. That is, the cell-based method requires no changes to either the existing wireless network architecture or the handset devices, and it can be readily applied to Global System for Mobile Communications (GSM).

However, only wireless networks with hexagonal or mesh structures are considered in [13]. In a real situation, it may not be feasible to place BSs in a hexagonal or mesh structure. This paper considers networks with irregular structures and presents an algorithm to determine the positioning accuracy of the cell-based method in irregular networks. The positioning accuracy of the cell-based method depends on the separation distance between two adjacent BSs and the transmission ranges of these BSs. Determining the optimal transmission range is a combinatorial optimization problem. Given a set of BS i, $i = 1,\ldots,n$, each with a fixed location and its transmission range r_i, $a \leqslant r_i \leqslant b$, we want to find a set of transmission ranges $(r_1^*, r_2^*, \ldots, r_n^*)$ such that the positioning accuracy is optimal. This is a difficult combinatorial problem. In this paper, we present a simulated annealing (SA) [14–19] algorithm to solve it because SA can provide an approximate solution for difficult optimization problems in reasonable time. SA was proposed by Kirkpatrick et al. [14], who reported promising results based on numerical experiments. Since then many papers have

reported on the topic. For detailed descriptions of SA, one can refer to a survey paper by Collins et al. [19].

Our simulation results show that (1) the accuracy can be improved up to 30% by SA; (2) the accuracy increases if the number of BSs increases; however, after a threshold the accuracy improvement is not noticeable; and (3) if we can place the BSs in an appropriate place, better accuracy can be achieved. After allocating BSs to optimal places, adjusting transmission ranges of BSs gives little contribution to improving accuracy.

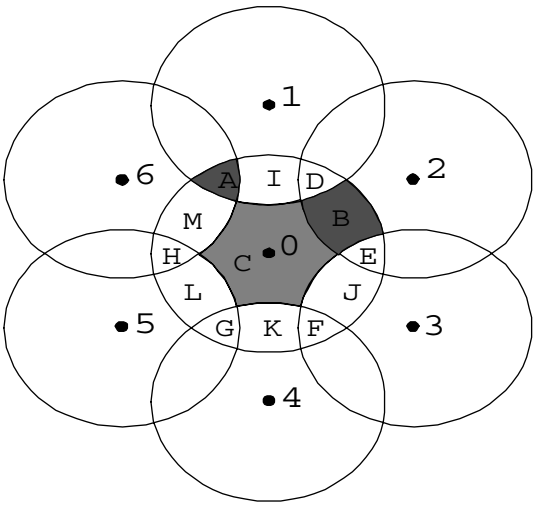
The remainder of this paper is organized as follows: Section 2 presents the cell-based positioning method and its application. Section 3 gives the positioning accuracy of networks. The improved accuracy with power adjustment and simulation results are shown in Section 4. Finally, the conclusions are given in Section 5.

2. Cell-based positioning method and its application

Consider a physical layout of a wireless network as shown in Fig. 1. The area covered by the BS is called a *cell* and each cell is circle-shaped. That is, one assumes a perfectly spherical radio propagation for this idealized model. The signal coverage of the BSs may overlap. The mobile station (MS) can receive radio signals containing the BSs' IDs, if the identifier is within the signal coverage of that BS. For example, as shown in Fig. 1, an MS in region A can listen to signals from BSs 0, 1 and 6; in region B, from BSs 0 and 2; and in region C, from BS 0. We define a localization region as one in which every MS in the region receives a unique set of BSs' signals. As shown in Fig. 1, the coverage of BS 0 has 13 localization regions, i.e., all the regions from A to M.

Suppose one has a location server in the network maintaining a table in which a set of BS ID is bounded to certain localization regions. When an MS reports to the location server that it can receive the signals from BSs 0, 1 and 6, the location server looks up the binding table and determines that the MS is in region A. One can thus determine the MS's location. This method is known as the cell-based positioning method, the topic of this paper.

In the following, we illustrate how to deliver location-based services by applying cell-based positioning method. Fig. 2 shows an architecture of the location-based service system. This system includes a central delivery platform and three servers, content server, location server, and Geographical Information System (GIS) server. The delivery platform integrates the servers to deliver mobile location-based services. The content server



Regions	BSs Id
A	0, 1, 6
B	0, 2
C	0
D	0, 1, 2
E	0, 2, 3
F	0, 3, 4
G	0, 4, 5
H	0, 5, 6
I	0, 1
J	0, 3
K	0, 4
L	0, 5
M	0, 6

Fig. 1. A layout of the wireless network.

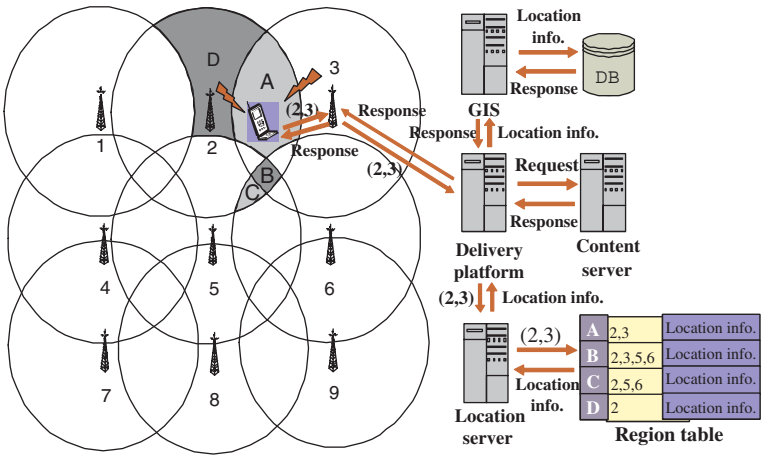


Fig. 2. An architecture of the location-based service system.

provides the relevant content, services and applications. The location server maintains the signal coverage of BSs and determines the location of mobile user. The GIS server provides the map and geographical information. As shown in Fig. 2, an MS in region A can listen to the signals from BSs 2 and 3. Then, the MS sends the location-based service request with cell ID $(2,3)$ to the delivery platform via BS 3. The delivery platform queries the MS's location by sending cell ID $(2,3)$ to location server. The location server looks up the region table and returns the MS's location to the delivery platform. Then, based on the MS's location, the delivery platform obtains the local information from GIS server and the relevant content from content server. Finally, the delivery platform prepares the requested service and replies it to the MS.

Note that the accuracy of the cell-based positioning method can be defined as the size of the localization region. The maximum of all localization regions is the location-sensing accuracy of the given network. As the localization regions become smaller, then the location-sensing accuracy improves.

3. Positioning accuracy of networks

This section considers a wireless network with n BSs in which the location of BS_i, (x_i, y_i) and its transmission range, r_i, are given where $i = 1,2,\ldots,n$. The radio coverage of BS_i is denoted as circle C_i. By using simple geometry, we can find all the intersections of all the circles. Then, we can formulate the positioning accuracy problem into a *geometry graph model* $G = (V, E)$. Each vertex in set V stands for intersection points of C_i and C_j, and an arc (u, v) is in E if (u, v) is a simple segment of the circle where a simple segment means there is no intersection point between u and v. We called a vertex v as a border vertex if v is not inside another circle. An arc (u, v) is called a border arc if both u and v are border vertices. For example, a wireless network as shown in Fig. 3 can be represented

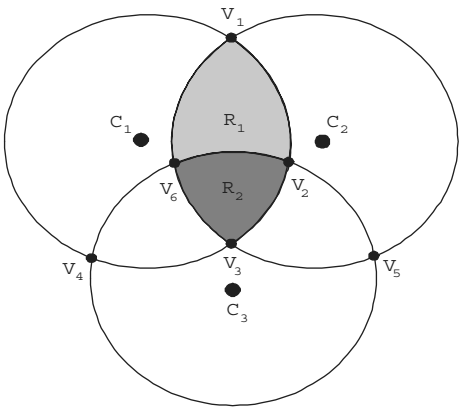


Fig. 3. A geometry graph G.

by the graph $G = (V, E)$, where $V = \{v_1, v_2, \ldots, v_6\}$ and $E = \{(v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_1), \ldots, (v_5, v_4), \ldots, (v_2, v_5)\}$. Vertices v_1, v_4 and v_5 are border vertices and arcs (v_1, v_4), (v_4, v_5) and (v_5, v_1) are border arcs. Note that the localization region R_1 is bounded by a set of arcs (v_1, v_2), (v_2, v_6), (v_6, v_1). Thus, the problem of finding the accuracy for the cell-based positioning method is equivalent to finding the maximum size of a localization region in the graph $G = (V, E)$. An algorithm is presented below for finding all localization regions of G and calculating their areas. Therefore, the maximum size of a localization region can be determined.

3.1. Localization region-finding algorithm

Let Adj[u] denote the adjacency list of u. That is, Adj[u] consists of all the vertices adjacent to u in G. For two vertices $v_1, v_2 \in \text{Adj}[u]$, we define angle $\angle v_1 u v_2$ to be the angle from arc (v_1, u) to arc (v_2, u) in a counter-clockwise direction. A cycle $P_i = (v_{i_1}, v_{i_2}), \ldots, (v_{i_k}, v_{i_1})$ is called a *simple cycle* if it forms a localization region. A simple cycle can be found by the following search procedure. Start to search from any arc (u, v) and set $P_i = (u, v)$. The following arc $(v, v_{j_1^*})$, $v_{j_1^*} \in \text{Adj}[v] \setminus \{u\}$, can be chosen into P_i is the arc with minimum angle $\angle u v v_{j_1^*}$. (i.e., $\angle u v v_{j_1^*} = \min_{v_{j1} \in \text{Adj}[v] \setminus \{u\}} \angle u v v_{j_1}$). Now, $P_i = (u, v)$, $(v, v_{j_1^*})$. Next, consider arc $(v, v_{j_1^*})$ and find the following arc $(v_{j_1^*}, v_{j_2^*})$ for P_i such that $\angle v v_{j_1^*} v_{j_2^*} = \min_{v_{j_2} \in \text{Adj}[v] \setminus \{u\}} \angle v v_{j_1^*} v_{j_2}$. Repeat the same procedure until the vertex u is reached. Then, a simple cycle $P_i = (u, v), (v, v_{j_1^*}), \ldots, (v_k^*, u)$ is found. For example, as shown in Fig. 4, start from (v_1, v_2). The next arc selected is (v_2, v_3). This is because $\angle v_1 v_2 v_3 < \angle v_1 v_2 v_5 < \angle v_1 v_2 v_6$. Similarly, choose (v_3, v_4) as the next arc of (v_2, v_3). By this way, a simple cycle (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_1) is found.

Note that such a simple cycle found is in a clockwise direction relative to arc (u, v). Similarly, we can also find another simple cycle for (u, v) in a counter-clockwise direction. This can be done by selecting the next arc $(v, v_{i_1^*})$ of (u, v) such that $\angle u v v_{i_1^*} = \max_{v_{i_1} \in \text{Adj}[v] \setminus \{u\}} \angle u v v_{i_1}$. Repeat this procedure until the vertex u is reached. For example, simple cycle (v_1, v_2), (v_2, v_6), (v_6, v_1) was found by a counter-clockwise search.

Suppose that for a given graph G, it has ℓ simple cycles. Let $P_i = (v_{i_1}, v_{i_2}), \ldots, (v_{i_k}, v_{i_1})$, $i = 1, \ldots, \ell$ denote all simple cycles in G. Observing these cycles, we found that for any arcs (v_i, v_j) if they are not border arcs, there are two cycles P_s and P_t containing it; otherwise only one cycle P_u contains it. Thus, we can maintain data structures *counter*[(u, v)] and *region*[(u, v)] for each arc (u, v). If (u, v) is not a border arc, *counter*[(u, v)] is set to 2; otherwise, *counter*[(u, v)] = 1. The *region*[(u, v)] is a set consists of the localization regions separated by (u, v). For example, if (u, v) separates regions i and j, then *region*[(u, v)] = $\{i, j\}$.

The main idea behind the region finding algorithm is that we search each cycle i, $P_i = (v_{i_1}, v_{i_2}), \ldots, (v_{i_k}, v_{i_1})$, to see which forms localization region i. Then, for each arc (u, v) in P_i, set *counter*[(u, v)] = *counter*[(u, v)] $- 1$ and *region*[(u, v)] = *region*[(u, v)] $\cup \{i\}$. Note that if all counters of the arcs become zero, then all localization regions are found.

A brief pseudocode for the region finding is given as follows:

```
0  Procedure LRSEARCH(G = (V, E))
1    RegionNumber = 0
2    for all (u, v)∈E do
3      region[(u, v)] = ∅
4      if (u, v) is not a border arc then counter[(u, v)] = 2
5        else counter[(u, v)] = 1
6    for all (u, v)∈E do
7      while counter[(u, v)] ≠ 0 do
8      begin
9        if counter[(u, v)] = 1 and region[(v, u)] ⊆ region[(u, w)]
10         then find a simple cycle in counter-clockwise
           direction, say P_i
11         else find a simple cycle in clockwise direction,
           say P_i
12         (comment: w∈Adj[u] with
           ∠uvw = min_{v_j∈Adj[v]\{u}} ∠uvv_j)
13       RegionNumber = RegionNumber + 1
14       for all (u_i, v_i)∈P_i do
15         counter[(u_i, v_i)] = counter[(u_i, v_i)] − 1
16         region[(u_i, v_i)] = region[(u_i, v_i)] ∪ {RegionNumber}
17     end
18   return(region[(u, v)])
```

An example to illustrate the above procedure is given in Fig. 5. After applying lines 7–17 of Procedure LRSEARCH to arcs, (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_5), (v_5, v_6) and (v_6, v_1), the 11 localization

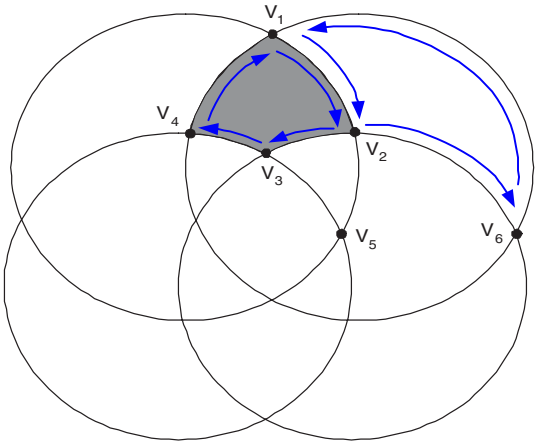


Fig. 4. A simple cycle search in clockwise direction.

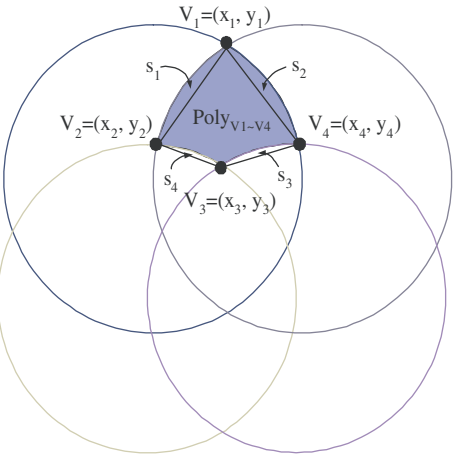


Fig. 6. The component of a circle area.

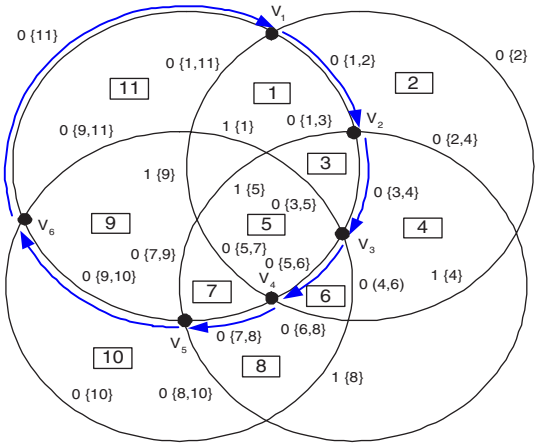


Fig. 5. An example of region finding.

regions are found as shown in Fig. 5. (Note that 13 localization regions will be found after all arcs are considered.) Values in the square represent region number and values along the arcs represent *counter*$[u, v]$ and *region*$[(u, v)]$.

3.2. Area of the localization region

Consider a localization region R_1 which is formed by simple cycle (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_1) in Fig. 6. Note that region R_1 includes polygon $v_1v_2v_3v_4$ and bow segments S_1 and S_2, and excludes bow segments S_3 and S_4. Let $g_{v_1\cdots v_k}$ denote polygon $v_1\cdots v_k$ and $s(X)$ denote area of X. Then, the area of localization region (v_1, v_2), (v_2, v_3), (v_3, v_4), (v_4, v_1) is equal to $s(g_{v_1v_2v_3v_4}) + s(S_1) + s(S_2) - s(S_3) - s(S_4)$. In general, if localization region R_i includes polygon $g_{v_1,v_2\cdots v_k}$ and bow segments $S_1,\ldots,S_a$, and excludes $S_{a+1},\ldots,S_k$, then the area of the region is

$$s(R_i) = s(g_{v_1\cdots v_k}) + \left(\sum_{i=1}^{a} s(S_i)\right) - \left(\sum_{i=a+1}^{k} s(S_i)\right).$$

The areas of a polygon and bow segments can be found as follows.

3.2.1. The area of polygons

The area of a polygon can be calculated by a cross-product formula [20] in the counter-clockwise order of vertices. That is, if the coordinates of polygon $g_{v_1\cdots v_n}$ are $(x_1, y_1),\ldots,(x_n, y_n)$, then the area of polygon $g_{v_1\cdots v_n}$ can be determined by

$$s(g_{v_1\cdots v_n}) = \frac{1}{2}\sum_{k=1}^{n}\begin{vmatrix} x_k & x_{k+1} \\ y_k & y_{k+1} \end{vmatrix},$$

where $x_{n+1} = x_1$, $y_{n+1} = y_1$.

3.2.2. The area of bow segments

The area of bow segment S_i, as shown in Fig. 7, can be calculated by

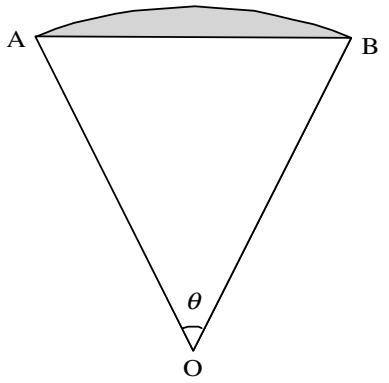


Fig. 7. The area of a bow segment.

$$s(S_i) = \tfrac{1}{2}\theta\, r^2 - s(\triangle_{\mathrm{AOB}}) = \tfrac{1}{2}\theta\, r^2 - \frac{1}{2}\left[2r^2 \sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right)\right],$$

where r is the transmission range of the cell and $\theta = \angle\mathrm{AOB}$.

Suppose that we found ℓ localization regions by using a region finding algorithm and evaluated each area of region R_i. Then, the accuracy $e(r_1, r_2, \ldots, r_n)$ of the cell-based positioning method can be obtained by

$$e(r_1, r_2, \ldots, r_n) = \max_{1 \leqslant i \leqslant \ell}\{R_i\}, \tag{1}$$

where r_i is transmission range of BS i, $1 \leqslant i \leqslant n$.

4. Improving accuracy with power adjustment

If the transmitting power of the BS can be adjusted, then the coverage of the BS varies. Consider this problem: given a set of BS i, $i = 1, \ldots, n$, with a fixed location and its transmission range $r_i, a \leqslant r_i \leqslant b$, we want to find a set of transmission ranges $(r_1^*, r_2^*, \ldots, r_n^*)$ such that the positioning accuracy is optimized. This problem is equivalent to finding a set of transmission ranges $(r_1^*, r_2^*, \ldots, r_n^*)$ such that $e(r_1, r_2, \ldots, r_n) = \max_{1 \leqslant i \leqslant \ell}\{R_i\}$ is minimized. That is,

$$z(r_1^*, r_2^*, \ldots, r_n^*) = \min_{(r_1, r_2, \ldots, r_n)} e(r_1, r_2, \ldots, r_n),$$

where $a \leqslant r_i \leqslant b$, $i = 1,2,\ldots,n$.

In this section, we applied SA to solve this optimization problem.

4.1. Simulated annealing (SA)

Recently, SA has become more popular for solving large-scale combinatorial optimization problems with approximate optimization solutions. The advantage of SA is that it provides a general purpose solution for a wide variety of combinatorial optimization problems. Thus, SA is used in many fields such as computer-aided design of integrated circuits, image processing, code-designed, neural network theory and so on.

In general, the SA algorithm is similar to metal-cooling. During slow cooling, a metal rearranges the atoms into regular crystalline structures with high density and low energy. The SA algorithm starts with an initial solution $s_0 = (r_1^0, r_2^0, \ldots, r_n^0)$, and finds the value of cost function $e(s_0) = e(r_1^0, r_2^0, \ldots, r_n^0)$, also known as *fitness function* (see Eq. (1)). Let s_i be the current solution with cost function $e(s_i)$. For each iteration j, generate a random neighbor s_j of s_i and evaluate its cost function $e(s_j)$. If $e(s_j) \leqslant e(s_i)$, then s_j is accepted (i.e., set $s_i = s_j$ and $e(s_i) = e(s_j)$). Otherwise, the s_j will be accepted with the probability $p = \min\{1, \exp((e(s_i) - e(s_j))/T)\}$. The parameter of T means the "temperature" which changed with parameter α for each iteration. This is known as the Metropolis criteria [15] and the pseudocode is shown below:

Step 1: Initialize the temperature T. Generate an initial solution s_0 and set current solution $s_i = s_0$.
Step 2: Generate a trial solution s_j, a random neighbor of s_i.
Step 3: Let $\Delta e = e(s_j) - e(s_i)$.
Step 4: If $\Delta e \leqslant 0$, then the trial solution s_j is accepted. Set current solution $s_i = s_j$ and $e(s_i) = e(s_j)$.
If $\Delta e > 0$, then the trial solution s_j is accepted with the probability $p = \exp(-\Delta e/T) > d$, where d is a random number in $[0,1]$. Set current solution $s_i = s_j$ and $e(s_i) = e(s_j)$.
Otherwise, go to Step 2.
Step 5: Repeat Steps 2–4 for I_t iterations.

Step 6: $T = T \times \alpha$.
Step 7: Repeat Steps 2–6, until $T < T_{\text{stable}}$.

In this simulation, we set $I_t = 300$, $T = 1$, $T_{\text{stable}} = 0.05$ and $\alpha = 0.85$.

4.2. Simulation results

This simulation ran on networks of 25, 36, 49, ..., 225 BSs in a square service area of 500 units × 500 units, respectively. We divided the service area into n grids (see Fig. 8), where n is the number of BSs. As shown in Fig. 8, white points, called candidate points, represent possible locations to place a BS in the grid. The black points represent block points where one cannot place a BS. Assume that the signal of the BS must cover the entire grid.

Four types of adjustments are considered for improving the accuracy of the cell-based positioning method:

Type 1: Assume that the locations of BSs are given and the transmission ranges of the BSs are identical. We evaluate $e(r, r, \ldots, r)$ for $r = a, a + \delta, \ldots, a + k\delta$, where $k = \lfloor 0.6a/\delta \rfloor$ and a is the minimal transmission range such that the entire service area is covered. Then find $z(r^*, r^*, \ldots, r^*) = \min\{e(r, r, \ldots, r) | r = a, a + \delta, \ldots, a + \lfloor 0.6a/\delta \rfloor \delta\}$. (In our simulation, we set $\delta = 1$.)

Type 2: Assume that the locations of BSs are given and the transmission ranges $r_i, i = 1,2,\ldots,n$, are in $[a, 1.6a]$. The SA method is applied to the network to find a set of transmission ranges $(r_1^*, r_2^*, \ldots, r_n^*)$ such that $z(r_1^*, r_2^*, \ldots, r_n^*) \approx \min_{(r_1, r_2, \ldots, r_n)} e(r_1, r_2, \ldots, r_n)$.

Type 3: Assume that the transmission ranges r_i $(i = 1,2,\ldots,n)$ of the BSs are given. SA is applied to the network to allocate the locations (x_i, y_i) of BSs such that $z(r_1, r_2, \ldots, r_n) \approx \min_{(x_1,y_1),\ldots,(x_n,y_n)} e(r_1, r_2, \ldots, r_n)$, where (x_i, y_i) is the coordinate of BS i.

Type 4: This is a combination of Types 2 and 3. For a given $(r_1, r_2, \ldots, r_n)$, SA is applied to allocate the locations of BSs. Then, use SA again to adjust the transmission ranges r_i $(i = 1,2,\ldots,n)$ of the BSs.

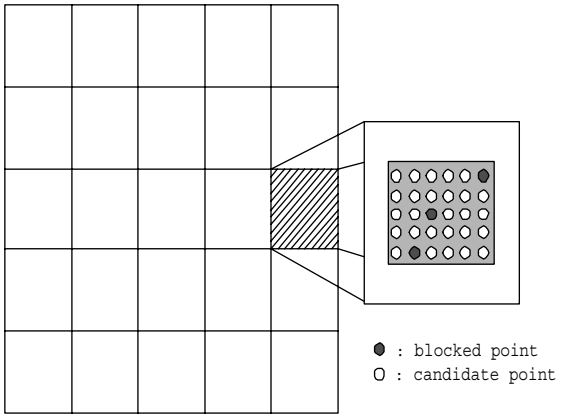


Fig. 8. Grid-based deployment.

Because the optimal value of $z(r_1^*, r_2^*, \ldots, r_n^*)$ is hard to find, we use a lower bound of $z(r_1^*, r_2^*, \ldots, r_n^*)$, denoted as LB, for comparison. A lower bound of $z(r_1^*, r_2^*, \ldots, r_n^*)$ is obtained as follows. By simulation, we can estimate the maximum number of localization regions for each network. Then, the whole service area divided by the maximum number of localization regions can be used as a lower bound of $z(r_1^*, r_2^*, \ldots, r_n^*)$. Table 1 summarizes the maximum number of localization regions and the lower bound of $z(r_1^*, r_2^*, \ldots, r_n^*)$ for each network.

In order to show the performance of the proposed methods, comparisons between the accuracy both before and after the adjustments are made.

Table 1
The maximum numbers of localization regions and LBs of $z(r_1^*, r_2^*, \ldots, r_n^*)$

Number of BSs in the network	25	36	49	64	81	100	121	144	169	196	255
Maximum number of regions	168	246	325	416	499	601	673	739	798	865	893
LB of $z(r_1^*, r_2^*, \ldots, r_n^*)$	1488	1016	769	601	501	416	371	338	313	289	280

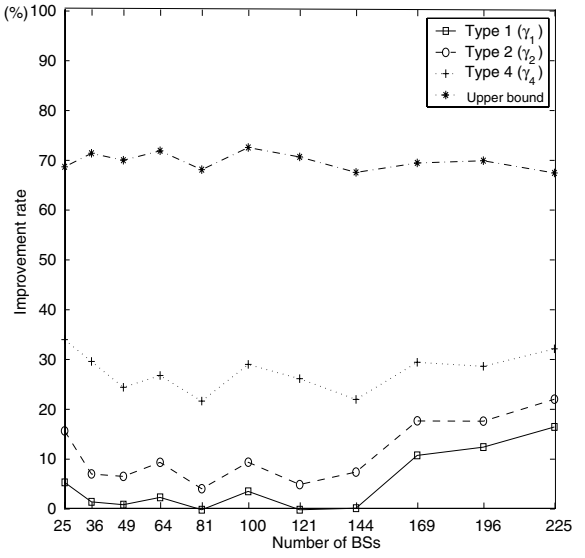


Fig. 9. The improvement rates of Types 1, 2, and 4, and the upper bound.

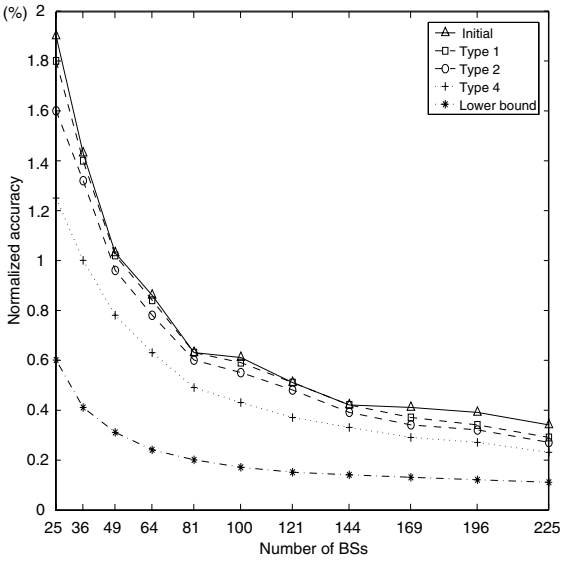


Fig. 10. The relationships between the accuracy and the number of BSs in the network.

Let Z_i denote the accuracy $z(r_1^*, r_2^*, \ldots, r_n^*)$ found by Type i and Z_0 denote the initial accuracy $e(r_1^0, r_2^0, \ldots, r_n^0)$ where $(r_1^0, r_2^0, \ldots, r_n^0)$ is an initial solution. Fig. 9 shows the improvement rate γ_i of Type i, $i = 1, 2, 4$, where the improvement rate $\gamma_i = |Z_i - Z_0|/Z_0 \times 100\%$. Besides, an upper bound of the improvement rate $\bar{\gamma}^* = |\mathrm{LB} - Z_0|/Z_0 \times 100\%$ is also shown in Fig. 9. From Fig. 9, note that the accuracy can be improved up to 30% by the Type 4 method.

Fig. 10 shows the relationships between the accuracy and the number of BSs in the network. The y-axis is the normalized accuracy defined as the percentage of the localization area compared to the entire service area. For example, the normalized accuracy of the lower bound is $\frac{338}{250{,}000} \times 100\% = 0.14\%$ for the network with 144 BSs. From Fig. 10, we learned that increasing the number of BSs from 25 to 144, the accuracy is significantly improved. However, when the number of BSs is more than 144, the improvement of accuracy becomes insignificant.

Finally, the improvement rates of the accuracy of Types 3 and 4 are compared in Fig. 11. The results of the two types are almost the same. This means that for accuracy, the factor of allocating the BSs' locations is more significant than the factor of adjusting transmission ranges. If we can place the BSs in appropriate places, the accuracy of the cell-based positioning method will be better.

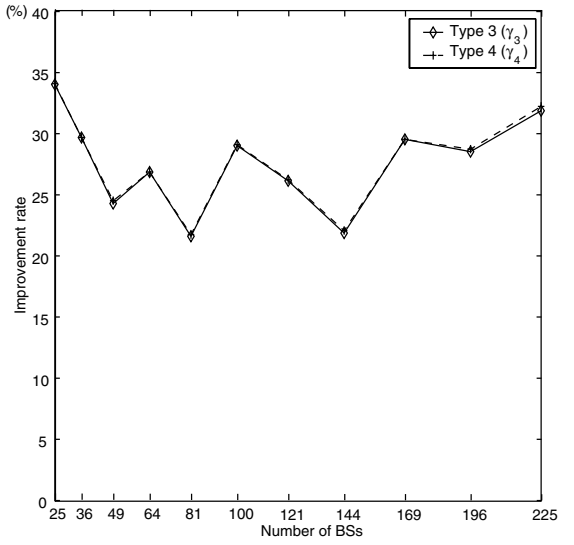


Fig. 11. The improvement rates of Types 3 and 4.

5. Conclusions

This paper presented an algorithm to determine the positioning accuracy of the cell-based method for networks with irregular structures. Because the positioning accuracy of the cell-based method is dependent on the separation distance between two adjacent BSs and the transmission ranges of these BSs, we proposed a SA method to determine the locations and transmission ranges of BSs in order to achieve a better positioning accuracy. Our simulation results showed that the accuracy can be improved up to 30% by the proposed method.

One of the main advantages of the cell-based method is that it requires no changes to the existing network architecture or to the handset. It only requires several lines of code on the SIM card to get the list of BSs. Thus, it does not incur substantial cost for either network operators or end users. The accuracy of cell-based positioning increases as the number of cells within range increases. Therefore, the cell-based method functions best in the urban area.

However, the proposed cell-based method is restricted in the idealized radio model, i.e., we assume the perfect spherical radio propagation in the idealized radio model. In fact, the coverage of a BS is not necessarily a circle. In most cases, it is location-dependent and probably irregular. The positioning problem with a more realistic radio model might be interesting for possible future work. The following cases are the most interesting:

1. *Non-circular coverage*: The coverage of a BS may not be circular. For example, in the real world, the BSs may have 1–3, or even 6 sectors. Besides, the terrain can produce irregular coverage areas. Our cell-based method might be extended to solve the positioning problem with irregular coverage, if each BS's coverage can be precisely defined and no two localization regions receive the same set of BS signals.
2. *Multiple power levels*: BSs can be of Macrocell, Microcell and Picocell types and have variable power. Or, BSs can transmit beacon signals with multiple power levels. In such cases, better positioning accuracy can be achieved. We are currently working on these extensions, and the results will be reported in our future papers.
3. *Noisy environments*: Noisy environments are characterized by severe multipath phenomenon, fading, obstructions, etc. Adapting the cell-based method to noisy environments is also our future work.

In addition to relaxing the idealized radio model, other measures of accuracy are interesting for possible future work. For example, the worst case of accuracy error can be defined as the diameter of the region. Evaluating the worst-case accuracy should be interesting and useful.

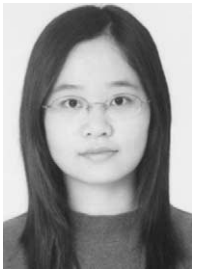
References

[1] N. Davies, K. Cheverst, K. Mitchell, A. Efrat, Using and determining location in a context-sensitive tour guide, IEEE Computer 34 (8) (2001) 35–41.
[2] K. Chadha, The global positioning system: challenges in bringing GPS to mainstream consumers, in: IEEE International Solid-State Circuits Conference, 1998, pp. 26–28.
[3] A. Marsh, M. May, M. Saarelainen, Pharos: coupling GSM and GPS-TALK technologies to provide orientation, navigation and location-based services for the blind, in: Proceedings of IEEE EMBS International Conference on Information Technology Applications in Biomedicine, 2000, pp. 38–43.
[4] G.M. Djuknic, R.E. Richton, Geolocation and assisted GPS, IEEE Computer 34 (2) (2001) 123–125.
[5] E. Kotsakis, A. Caignault, W. Woehler, M. Ketselidis, Integrating differential GPS data into an embedded GIS and its application to infomobility and navigation, in: Proceedings of the 7th EC–GI and GIS Workshop EGII-Managing the Mosaic, June 2001.
[6] G.J. Morgan-Owen, G.T. Johnston, Differential GPS positioning, Electronics and Communication Engineering Journal 7 (1) (1995) 11–21.
[7] J.C. Jubin, D.L. Shaver, Wide-area differential GPS reference-station placement, in: Position Location and Navigation Symposium, 1996, pp. 503–514.
[8] C. Drane, M. Macnaughtan, C. Scott, Positioning GSM telephones, IEEE Communications Magazine 36 (4) (1998) 46–54.
[9] J.M. Zagami, S.A. Parl, J.J. Bussgang, K.D. Melillo, Providing universal location services using a wireless E911 location network, IEEE Communications Magazine 36 (4) (1998) 66–71.
[10] T.S. Tappaport, J.H. Reed, B.H. Woerner, Position location using wireless communications on highways of the future, IEEE Communication Magazine 34 (1996) 33–34.
[11] L. Doherty, K.S.J. Pister, L.E. Ghaoui, Convex position estimation in wireless sensor networks, in: Proceedings of INFOCOM 3 (2001) 1655–1663.

[12] C. Drane, M. Macnaughtan, C. Scott, The accurate location of mobile telephones, in: 3rd World Conference on Intelligent Transport Systems, October 1996.
[13] H.-C. Chu, R.-H. Jan, A cell-based location-sensing method for wireless networks, Wireless Communication and Mobile Computing 3 (2003) 455–463.
[14] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 672–680.
[15] P.J.M. van Laarhoven, E.H.L. Aarts, Simulated Annealing: Theory and Applications, Reidel, Dordrecht, 1987.
[16] D.S. Johnson, C.R. Aragon, L.A. McGeoch, C. Schevon, Optimization by simulated annealing: an experimental evaluation, part II. Graph coloring and number partitioning, Operations Research 39 (1991) 378–406.
[17] A.H. Mantawy, Y.L. Abdel-Magid, S.Z. Selim, A simulated annealing algorithm for unit commitment, IEEE Transactions on Power Systems 13 (1) (1998) 197–204.
[18] A.Y. Zomaya, Natural and simulated annealing, Computing in Science and Engineering 3 (6) (2001) 97–99.
[19] N.E. Collins, R.W. Eglese, B.L. Golden, Simulated annealing: an annotated bibliography, American Journal of Mathematical and Management Sciences 8 (3–4) (1988) 209–307.
[20] G.B. Thomas, R.L. Finney, Calculus and Analytic Geometry, ninth ed., Addison-Wesley, Reading, MA, 1998.

Rong-Hong Jan received the B.S. and M.S. degrees in Industrial Engineering, and the Ph.D. degree in Computer Science from National Tsing Hua University, Taiwan, in 1979, 1983, and 1987, respectively. He joined the Department of Computer and Information Science, National Chiao Tung University, in 1987, where he is currently a Professor. During 1991–1992, he was a Visiting Associate Professor in the Department of Computer Science, University of Maryland, College Park, MD. His research interests include wireless networks, mobile computing, distributed systems, network reliability, and operations research.

Hung-Chi Chu received the B.S. and M.S. degrees in Computer Science and Engineering from Tatung University, Taiwan, in 1995, 1997. Science 2001, he has been working toward the Ph.D. degree in Computer and Information Science at National Chiao Tung University, Taiwan. His research interests include wireless networks and artificial intelligence.



Yi-Fang Lee received the B.S. degree in Computer Science from Tamkang University in 2001 and M.S. degree in Computer and Information Science from National Chiao Tung University, Taiwan, in 2003. Her research interests include wireless networks, mobile computing and wireless Internet.