Front. Electr. Electron. Eng. China 2007, 2(1): 13–22
DOI 10.1007/s11460-007-0003-1

RESEARCH ARTICLE

ZHAO Hai, SUN Peigang, ZHANG Wenbo, YIN Zhenyu, ZHAO Ming

Research on reference nodes placement and selection of ubiquitous computing locating service

Abstract Obtaining the location of an unknown node accurately is a key problem of a locating service under a ubiquitous computing environment. The paper proposes and proves three theorems of location reference node placement according to the analysis of the location error produced during location using a polygon location method and three important characteristics of chaos dynamics. Based on the three theorems, the location reference node selection (LRNS) algorithm is proposed by improving on the traditional polygon location algorithm. The simulation results indicate that the reference node placement theorems and the LRNS algorithm can meet the requirements of a ubiquitous terminal's real-time location and possess a preferable precision in location.

Keywords ubiquitous computing, locating service, location error, reference nodes placement theorems, location reference node selection algorithm

1 Introduction

In ubiquitous computing, context perception can provide users with relevant information of their environment, and nearly 80% of it is relevant to the location [1,2]. Therefore, location services have become a key service among ubiquitous computing services, which means that the location algorithms designed to obtain the user's position have become an important topic in ubiquitous computing fields [3].

The most common procedures of existing location algorithms can be divided as follows.

Translated from *Acta Electronica Sinica*, 2006, 34(8): 1456–1463
[译自: 电子学报]

ZHAO Hai (✉), SUN Peigang, ZHANG Wenbo, YIN Zhenyu, ZHAO Ming
School of Information Science and Engineering, Northeastern University, Shenyang 110004, China
E-mail: zhhai@neuera.com

SUN Peigang
Shenyang Artillery Academy, Shenyang 110162, China

1) Measure the distance between the reference nodes and the unknown mobile node;

2) Estimate the unknown node's position;

3) Optimize and update the unknown node's position through iteration [4−7].

These algorithms can obtain the node's position with a certain degree of accuracy. However, if ranging errors exist in the first step, an accumulation of errors will occur in the succeeding steps. How do we reduce the accumulated error and obtain the smallest possible location error? The related studies and algorithms for reducing location errors are few.

Some important conclusions are proved in mathematics in this paper. If the reference nodes are placed as equilateral triangles, the location errors will be reduced to the smallest number, and what's more, the reference nodes are placed as Sierpinski triangle [8] based on the polygon dynamic geometry, so the errors may be even smaller. Hence, the LRNS algorithm is presented. As simulation results indicate, the algorithm has several strong points such as high location accuracy, fewer calculations, and good real-time performance, which can be applied to the ubiquitous terminals' real-time locating and tracking, and meets ubiquitous computing location requirements.

2 Reference nodes placement theorems

2.1 The principle of polygons location

Definiens 1 The unknown nodes are termed as positioning nodes only if they need to be located, and they are either users with a ubiquitous terminal or wireless communication terminals. The reference nodes are nodes (sensors or base stations) that have data on their own position.

In two-dimensional space, if the reference node's two coordinates are known, the node's location can be calculated by the distances among the nodes, but the positioning node's coordinate may be uncertain. The reason is that two circles will bring two intersection points but will not be able to determine the location coordinate. Therefore, an unknown node needs to use at least three reference nodes to locate itself in two-dimensional space.

Assuming that there are three different reference nodes, $p_i = (x_i, y_i)$ $(i = 1, 2, 3)$, in order to get the location of positioning node's coordinate $p(x, y)$, the distance $r_i = d\,(p, p_i)$ between p and p_i needs to be measured by using time of arrival (TOA) method [9].

According to the polygon location algorithm, we know the following equations

$$\begin{cases}(x-x_1)^2+(y-y_1)^2=r_1^2\\(x-x_2)^2+(y-y_2)^2=r_2^2\\(x-x_3)^2+(y-y_3)^2=r_3^2\end{cases}$$

Get the equations' root

$$\begin{cases}x=\dfrac{1}{\Delta^2}(2T_1(y_1-y_3)-2T_2(y_1-y_2))\\y=\dfrac{1}{\Delta^2}(2T_2(x_1-x_2)-2T_1(x_1-x_3))\end{cases}$$

where $T_1 = r_2^2 - r_1^2 - x_2^2 + x_1^2 - y_2^2 + y_1^2$, $T_2 = r_3^2 - r_1^2 - x_3^2 + x_1^2 - y_3^2 + y_1^2$, $\Delta^2 = 4((x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2))$ and Δ^2 is called location operator.

If p_1, p_2, and p_3 are in the same line, location operator $\Delta^2 = 0$, at this moment, the root of the equations above is infinite, which means node p could not be located by those three reference nodes p_i $(i = 1, 2, 3)$. Therefore, in an actual location situation, the value of operator Δ^2 can be used to judge whether the node could be located or not. Furthermore, we should avoid making the three reference nodes in a line.

Based on the same principle, in three-dimensional space, the position of a node's coordinates can be obtained according to the distances between the positioning node and the four different reference nodes. But mostly, in ubiquitous computing, two-dimensional coordinate is commonly used. In this way, this paper is focused on the placement of the reference nodes in two-dimensional coordinate, aiming at a better positioning service for ubiquitous computing.

2.2 Reference nodes' placement theorems

The distance between the nodes is calculated under the assumption that the measurement is reliable; however, measurement errors are really inevitable. In the following statements, we will assume that the errors are in the range $(0, \pm\varepsilon)$, in which $\varepsilon > 0$. That is, if the actual distance between the two nodes is r, the actual measurement is between $(r, \pm\varepsilon)$. In the case of existing errors, the three circles will no longer intersect at a point, but will form a small area. As is shown in Fig. 1.

The small area enclosed by the three circles in Fig. 1 is noted as C_p. The size of C_p shows the degree of the location error. In order to identify the location node P's coordinate (x, y), we should first find the three reference nodes' coordinates (x_i, y_i), $i = 1, 2, 3$, then measure the distance between the positioning node and the reference node r_i, and get the

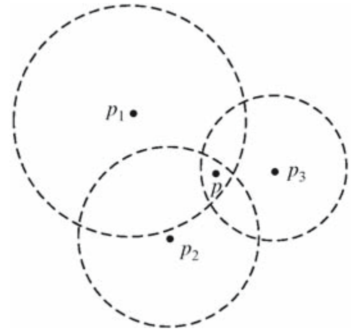

Fig. 1 The location error area with distance measurement errors

error of distance measurement ε_i. Finally, we can obtain the following equations

$$C_{p_i} = \{(x,y)\,|\,(x-x_i)^2+(y-y_i)^2 \leqslant (r_i+\varepsilon_i)^2,\ (x-x_i)^2+(y-y_i)^2 \geqslant (r_i-\varepsilon_i)^2\} \quad (1)$$

$$C_p = \left\{(x,y)\,|\,x \in \bigcap_{i=1}^{3} C_{p_i},\ y \in \bigcap_{i=1}^{3} C_{p_i}\right\} \quad (2)$$

$$S_{p_i} = \{(x,y)\,|\,x^2+y^2=\varepsilon_i^2, \varepsilon_i > 0\} \quad (3)$$

As each reference node is measured independently, we can simplify the program by assuming that the distance measurement error is equal, that is $\varepsilon_i = \varepsilon$, $i = 1, 2, 3$. In this way, if $\varepsilon \equiv 0$, points set C_p will become one point in Eq. (2), while if $\varepsilon > 0$, C_p will be a small protruding region and the size of C_p means value of location error.

Suppose l_{pp_i} is a line passing through point p and point p_i, intersects the circle S_p in Eq. (3) at two points $q_{ij}, j = 1, 2$. Let S_p's tangent $\tilde{l}q_{ij}$ cross q_{ij}, then the region $\widetilde{C}_{p_i}$ will be a hexagon between line $\tilde{l}_{qi1}$ and $\tilde{l}_{qi2}$. Set $\widetilde{C}_p = \widetilde{C}_{p1} \cap \widetilde{C}_{p2} \cap \widetilde{C}_{p3}$ as Fig. 2 shows. If the measurement error ε is tiny, C_p's edging region can be linearized, and can be estimated as $\widetilde{C}_p$. Thus, the location problem of positioning mode in two-dimensional space could be translated into how to place three reference nodes to make the area size of C_p, interpreted as $S(\widetilde{C}_p)$, minimized.

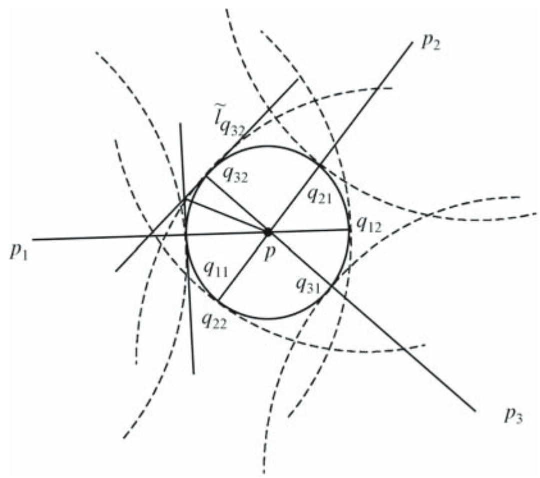

Fig. 2 An analysis of the location errors

Definiens 2 Suppose subset $S \subset \mathbb{R}^n$, $\forall \chi_1 \in S, \chi_2 \in S, \chi_1 \neq \chi_2$, where $\forall\ \lambda \in [0, 1]$, we can approve that $\lambda\chi_1 + (1-\lambda)\chi_2 \in S$ is true and S is a convex set.

Definiens 3 Assume S is a convex set, $\forall \chi_1 \in S$, $\chi_2 \in S$, $\chi_1 \neq \chi_2$, $\forall \lambda \in [0, 1]$, and $f(\lambda\chi_1 + (1-\lambda)\chi_2) \leqslant f(\chi_1) + (1-\lambda) f(\chi_2)$ is true, we define $f(\chi)$ as a convex function in convex set S.

Lemma 1 Suppose S is a convex set, $f(\chi)$ is a smooth function based upon S, $\forall \chi \in S, \exists f''(\chi) \geqslant 0$, then $f(\chi)$ is called as a convex function in convex set S.

Lemma 2 For the convex function $f(\chi)$, defined in subset S, $\forall m$ points $\chi_1, \chi_2, \chi_3, \ldots, \chi_m$, $\exists\ f\{(1/m)(\chi_1+\chi_2+\chi_3+\ldots+\chi_m)\} \leqslant (1/m)\ \{f(\chi_1)+f(\chi_2)+f(\chi_3)+\ldots+f(\chi_m)\}$, the inequalities become equality only if $\chi_1 = \chi_2 = \ldots = \chi_m$.

Theorem 1 Suppose α_{ij} is the angle (acute angle) between vector $\boldsymbol{pp}_i$ and $\boldsymbol{p}_j\boldsymbol{p}$, if $\alpha_{12} = \alpha_{23} = \alpha_{13} = \pi/3$, $S(\widetilde{C}_p)$ achieves the minimum value. That is, if the three reference nodes form an equilateral triangle, the positioning node's location error will be the smallest. At the same time, the three reference nodes shape a smallest location unit in the two-dimensional space.

Proof Based on Fig. 2, we can draw all six tangents for circle S_p and gain Fig. 3, for convenience in the arrangement; we make some minor adjustments in alphabetical labeling.

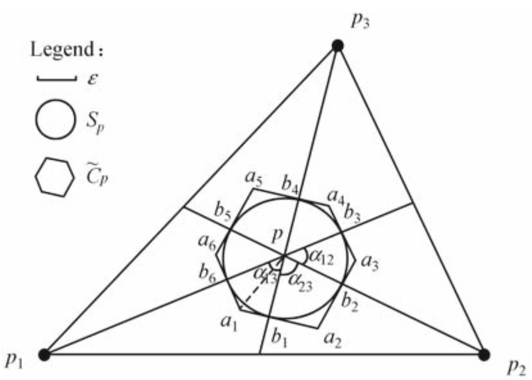

Fig. 3 The error area in location

Since the region constructed by $\widetilde{C}_p$ is a circumscribed hexagon S_p, we can know that the following polygons are congruent using simple primary geometry

$$pb_6a_1b_1 \cong pb_3a_4b_4;\ pb_1a_2b_2 \cong pb_4a_5b_5;\ pb_2a_3b_3 \cong pb_5a_6b_6$$

thus

$$S(\widetilde{C}_p) = 2S(b_6a_1a_2a_3b_3)$$

As $\Delta pb_6a_1 \cong \Delta pa_1b_1$, the length of line pb_1 is exactly ranging error ε; therefore, the following equations are established

$$|a_1b_1| = \varepsilon \tan\frac{\alpha_{13}}{2};\ S(\Delta pa_1b_1) = \frac{1}{2}|pb_1||a_1b_1| = \frac{1}{2}\varepsilon^2 \tan\frac{\alpha_{13}}{2}$$

$$S(pb_6a_1b_1) = 2S(\Delta pa_1b_1) = \varepsilon^2 \tan\frac{\alpha_{13}}{2}$$

Similarly

$$S(pb_1a_2b_2) = \varepsilon^2 \tan\frac{\alpha_{23}}{2};\ S(pb_2a_3b_3) = \varepsilon^2 \tan\frac{\alpha_{12}}{2}$$

Thus

$$\begin{aligned} S(\widetilde{C}_P) &= 2S(b_6a_1a_2a_3b_3) \\ &= 2(S(pb_6a_1b_1) + S(pb_1a_2b_2) + S(pb_2a_3b_3)) \\ &= 2\left(\varepsilon^2 \tan\frac{\alpha_{13}}{2} + \varepsilon^2 \tan\frac{\alpha_{23}}{2} + \varepsilon^2 \tan\frac{\alpha_{12}}{2}\right) \end{aligned}$$

That is

$$S(\widetilde{C}_p) = 2\varepsilon^2\left(\tan\frac{\alpha_{12}}{2} + \tan\frac{\alpha_{23}}{2} + \tan\frac{\alpha_{13}}{2}\right) \quad (4)$$

Regarding α_{12}, α_{23}, α_{13}, the following relational formula exists

$$\alpha_{12} + \alpha_{23} + \alpha_{13} = \pi$$

When $0 \leqslant x \leqslant \pi/2$, $(\tan\ x)'' = 2\tan\ x(1 + \tan\ x) \geqslant 0$, the following formula can be derived by Lemma 1 and Lemma 2

$$S(\widetilde{C}_p) = 6\varepsilon^2 \frac{1}{3}\left(\tan\frac{\alpha_{12}}{2} + \tan\frac{\alpha_{23}}{2} + \tan\frac{\alpha_{13}}{2}\right) \geqslant$$

$$6\varepsilon^2 \tan\frac{\alpha_{12} + \alpha_{23} + \alpha_{13}}{6} = 6\varepsilon^2 \tan\frac{\pi}{6} \quad (5)$$

When $\alpha_{12} = \alpha_{23} = \alpha_{13} = \pi/3$, the equality happens in Eq. (5). When the three reference nodes' positions form an equilateral triangle, the error of the positioning node is minimized. Meanwhile, since the location needs at least three reference points in two-dimensional space, the reference nodes that form the equilateral triangle become the smallest location unit in the two-dimensional space.

Theorem 2 If the distance between the reference nodes and the positioning node is beyond the signal transmission scope, it needs to increase the number of reference nodes around the original reference nodes in order to form new equilateral triangles. The coordinates of the new-added reference nodes can be calculated with the original ones. It is obvious that the reference node's coordinates are topologically replicative.

Proof It is easy to calculate that the reference nodes can be topologically replicated as Fig. 4 shows. Assume that A, B, C are the original three reference nodes, and arrange them as an equilateral triangle whose edge length is equal to $2a$. Take ΔABC as the original triangle seamlessly duplicates in turn. Let ΔABC's gravity O be the origin point of coordinates, we could get the coordinate system as in Fig. 4

Dot A, B, C are reference nodes with known position and their coordinates are $A\,(-a/2,\ \sqrt{3}a/3)$, $B(a/2,\ \sqrt{3}a/3)$, $C(0,\ -2\sqrt{3}a/3)$ respectively. The points A_1, B_1, C_1 are the unknown points obtained by ΔABC's edges first topological duplication. The coordinates of A_1, B_1 and C_1 can be easily obtained by studying the relations between the equilateral

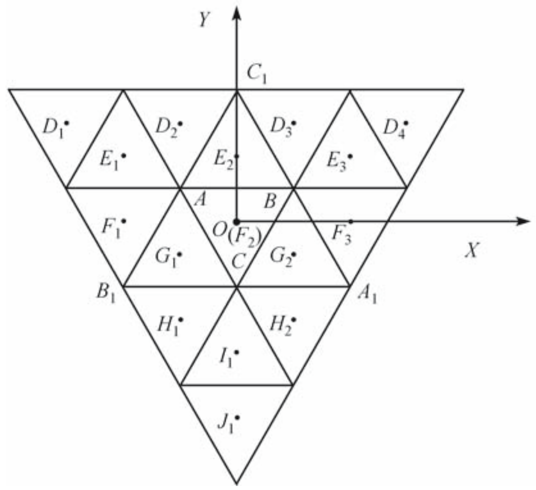

Fig. 4 Topological duplication of the reference nodes

triangle's edge and angle, and denoted as $A_1(a, -2\sqrt{3}a/3)$, $B_1(-a, -2\sqrt{3}a/3)$, $C_1(0, 4\sqrt{3}a/3)$ respectively. Analyzing this in sequence, other unknown nodes' positions could be gained. It can be seen that placing the reference nodes in the indoor environment, as long as we know the coordinates of an equilateral triangle, we could then determine the other reference nodes' coordinates by topological duplication. Fig. 4 only shows the partial duplication results. To satisfy the actual need, the reference nodes may be extended further.

Theorem 3 *n* (integer $n \geqslant 1$) equilateral triangles are used to compute the positioning nodes' position. As *n* increases, the unknown node location error will decrease. If *n* makes an unlimited increase which tends to be infinite, the location error tends to be a constant $\pi\varepsilon^2$ (ε for the ranging error). So the location error is convergent.

Proof

1) If $n = 1$, by Theorem 1, the reference nodes placed equilateral triangle constitutes the smallest unit of the plane's location, which means the smallest of the positioning errors. Let C_p be the possible region that unknown node exist, the positioning error is about $S(\widetilde{G}_{p_1}) = 2\sqrt{3}\varepsilon^2$.

2) If $n \geqslant 2$, by Theorem 2, we can see that the reference nodes' topological layout can be duplicated, and the rules of topological duplication is uniquely determined. When $n = 2$, around the positioning point, a new error region is formed and denoted as C_{p_2}. Therefore, the region that the unknown node may exist in can be denoted as $C_p = C_{p_1} \cap C_{p_2}$, here $C_p \subseteq C_{p_1}$ and $C_p \subseteq C_{p_2}$. If $n > 2$, followed by analogy, $C_p = C_{p_1} \cap C_{p_2} \cap \ldots \cap C_{p_i} \cap \ldots \cap C_{p_n}$, and $C_p \subseteq \{C_{p_1}, C_{p_2}, \ldots, C_{p_i}, \ldots, C_{p_n}\}$. So, seen from the area, the formula $S(C_p) \leqslant \min\{S(C_{p_1}), S(C_{p_2}), \ldots, S(C_{p_i}), \ldots, S(C_{p_n})\}$ is right. That is to say, the processing of an equilateral triangle topological duplication, the positioning node's error will decrease ceaselessly.

3) Due to the ranging error ε, no matter how *n* increases, the area error of the positioning node would not decrease continuously, but in a trend towards a constant value. The constant will be given as follows. First, the topological duplications have been done respectively according to the three edges of the equilateral triangle. Changes of error area are shown in Fig. 5. Among them, Fig. 5(a) shows that three new equilateral triangles are added to locate the unknown node, and also three new reference nodes A_1, B_1, C_1 added through first topological duplication. The different location error regions of the two reference nodes placement methods are shown in Fig. 5(b), which is in Fig. 5(a) ΔABC internal map partial enlarged to show the error regional intersection. The respective error regions and its intersection geometry are shown in Fig. 5(c). The regions' formal descriptions are given, too.

For the initial equilateral triangle ΔABC, error region of positioning *p* is marked as C_{p_0}, the area of which can be obtained by Eq. (5) in Theorem 1's proof

$$S(\widetilde{C}_{p_0}) = 2\sqrt{3}\varepsilon^2$$

By topological duplication, three new triangles ΔA_1BC, ΔAB_1C, ΔABC_1 are added to the location. Since the original reference nodes *A*, *B*, *C* make no new contributions to locate the positioning node, we add a new A_1, B_1, C_1 to make contributions, and the newly added A_1, B_1, C_1 constitute a new equilateral triangle $\Delta A_1B_1C_1$. Import a new error region C_{p_1}, and intersecting with C_{p_0}. The Circle S_p is a common part of both. Outside of the S_p, seen from Fig. 5(b), the six small regions of C_{p_0} separated by S_p are re-intersected by C_{p_1}. Suppose that outside the round, the common region obtained

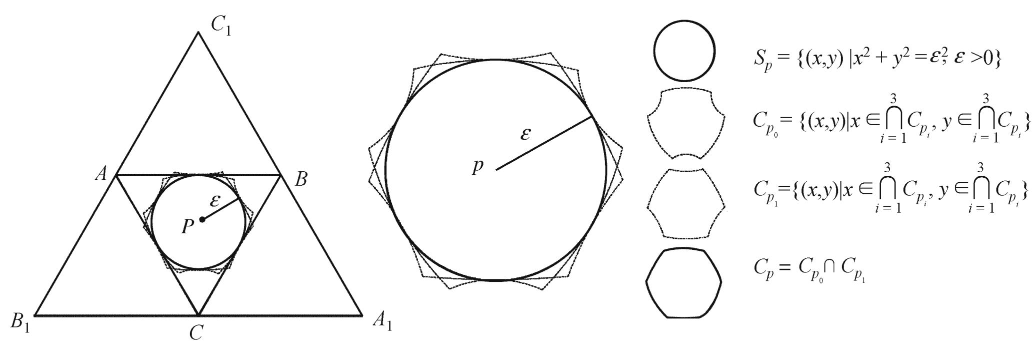

Fig. 5 Location error area decreases after topological duplication

through each small region intersection is $1/c$ of the original C_{p_0}'s small region, where $c>1$ and c is a constant. By deduction, the area of $C_p=C_{p_0}\cap C_{p_1}$, named as $S(\widetilde{C}_{p_1})$, can be obtained as follows:

$$S(\widetilde{C}_{p_1})=\frac{2\sqrt{3}\varepsilon^2-\pi\varepsilon^2}{c}+\pi\varepsilon^2=\frac{2\sqrt{3}\varepsilon^2}{c}+\frac{c-1}{c}\pi\varepsilon^2 =\frac{1}{c}S(\widetilde{C}_{p_0})+\frac{c-1}{c}\pi\varepsilon^2 \tag{6}$$

Regard $\Delta A_1B_1C_1$ as an equilateral triangle again, we topologically duplicate the three edges respectively, three new vertices A_2, B_2, C_2 are then introduced, and what's more, we introduce a new error region C_{p_2}, so that the area of $C_p=C_{p_0}\cap C_{p_1}\cap C_{p_2}$, named as $S(\widetilde{C}_{p_2})$ can be obtained as follows

$$S(\widetilde{C}_{p_2})=\frac{1}{c}\left[\frac{2\sqrt{3}\varepsilon^2+(c-1)\pi\varepsilon^2}{c}-\pi\varepsilon^2\right]+\pi\varepsilon^2 =\frac{2\sqrt{3}\varepsilon^2}{c^2}+\frac{(c-1)\pi\varepsilon^2}{c^2}+\frac{c-1}{c}\pi\varepsilon^2 =\frac{1}{c}S(\widetilde{C}_{p_1})+\frac{c-1}{c}\pi\varepsilon^2 \tag{7}$$

By iterative topological duplication, we can deduce the next iterative relationship according to Eqs. (6) and (7)

$$\begin{cases} S(\widetilde{C}_{p_0})=2\sqrt{3}\varepsilon^2 \\ S(\widetilde{C}_{p_n})=\frac{1}{c}S(\widetilde{C}_{p_{n-1}})+\frac{c-1}{c}\pi\varepsilon^2 \end{cases} \tag{8}$$

Solve the difference equations

$$S(\widetilde{C}_{p_n})=\frac{2\sqrt{3}\varepsilon^2}{c^n}+\frac{c-1}{c^n}\pi\varepsilon^2+\frac{c-1}{c^{n-1}}\pi\varepsilon^2+\ldots +\frac{c-1}{c^2}\pi\varepsilon^2+\frac{c-1}{c}\pi\varepsilon^2 \tag{9}$$

If n tends to infinity, limit Eq. (9), we will get

$$\lim_{n\to+\infty}S(\widetilde{C}_{p_n})=\lim_{n\to+\infty}\left(\frac{2\sqrt{3}\varepsilon^2}{c^n}+\frac{c-1}{c^n}\pi\varepsilon^2+\frac{c-1}{c^{n-1}}\pi\varepsilon^2+\ldots +\frac{c-1}{c^2}\pi\varepsilon^2+\frac{c-1}{c}\pi\varepsilon^2\right) =\frac{\frac{c-1}{c}}{1-\frac{1}{c}}\pi\varepsilon^2=\pi\varepsilon^2 \tag{10}$$

The physical meaning of Eq. (10) is that after the introduction of countless equilateral triangles, contributions to the error region are concentrated in the exterior circle S_p until six small error regions are all eliminated due to iteration. But owing to the ranging error ε, no matter how many reference nodes are added, there would be no contribution to the interior of the circle S_p. Therefore, after repeated topological duplication, the error tends to be a constant ε^2. That is to say, the location error converges.

An iteration of topological duplication is presented theoretically in Theorem 3. In fact, due to the influence of the size of the indoor space, the medium of communication between nodes and other factors, the number of new added reference nodes is limited. In this instance, during each iteration, the initial equilateral triangle is termed as the basic unit for topological duplication, which constitutes the seamless tiled structure around the initial equilateral triangle. Then, the new reference nodes grow and occupy small space, which are more suitable for actual applications.

2.3 Chaos Dynamics implied the theorems

In ubiquitous computing, a key problem of location is how to place the reference nodes. Theorem 1 shows that an equilateral triangle constitutes the basic location unit in a two-dimensional space, that is to say location "initial sensitivity". Theorem 2 indicates that the mobile positioning nodes can be tracked efficiently. And the "replicable topology" can be used to add new reference nodes. Theorem 3 proves that the location error of positioning nodes will not reduce infinitely, but to a constant, which is interpreted as a "period astringency". These are exactly in agreement with the three key characteristics of a chaotic dynamical system.

Equation (8) gives the initial condition and the iterative difference equations according to the location error, which are regarded as chaos dynamics equations. The long status of the dynamic equations enables the location error to be a constant $\pi\varepsilon^2$. The evolution process of the dynamic system is directly expressed in Fig. 4 in view of geometry; it is shaped as a Sierpinski triangle, which means self-similarity, see Ref. [10], we can get the fractal dimension of the topology shown in Fig. 4, which is $D=\ln 3/\ln 2=1.584\,96$.

In actual application, the reference nodes of the initial equilateral triangle should be placed subject to the size of the indoor space and the wireless transmission medium among the nodes. Considering the location of the positioning node, it is impossible to make all the reference nodes added by the topological duplication participate in location at one time. The selection of reference nodes is concerned with the following factors.

1) The threshold value of the location error. The more iteration, the slower the error area decreases. Therefore, we can set a threshold of the location error, beyond which the topological duplication stops;

2) Convergence delay. As the number of reference nodes increases, the location costs increase, and the real-time performance weakens;

3) Complexity of the calculation. Limited to ubiquitous terminals' capacities; the amount of the location computing should be as small as possible. In general, we can look for the reference nodes around the positioning node as Fig. 6 shows.

Six reference nodes can then be obtained by topological duplication based on the initial triangle, which constitutes the optimal computing unit of LRNS algorithm.

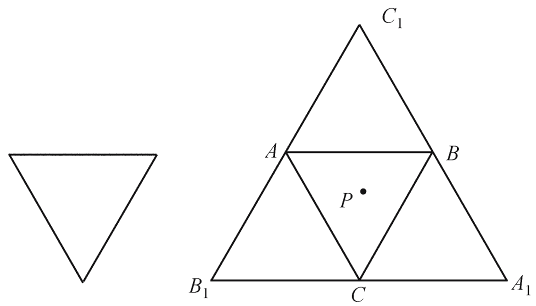

Fig. 6 The optimal computing unit of the LRNS algorithm

3 The location reference selection algorithm

In ubiquitous computing, the location algorithms aim at how to use the existing reference nodes to the position, to track a mobile terminal and obtain its movement track [11,12]. Ordinary ubiquitous terminals work under a resource-constrained environment, which means limited computing power, storage capacity and communications capability. However, the traditional polygon location algorithm requires much time and space costs, so it is extremely difficult to apply to a resource-constrained ubiquitous terminal [13]. Based on the reference nodes' theorems, the LRNS algorithm is presented to replace traditional algorithms.

3.1 The traditional polygon location algorithm

To compare with the LRNS algorithm presented in this paper, we will describe a traditional polygon location algorithm as follows.

1) Each reference node sends broadcasting packets, including location information denoted as {ID, T_{send}, (a, b)}. ID is identification of the reference node. T_{send} is time when the message is sent and (a, b) is its coordinate;

2) In virtue of the received multiple broadcasting packets, the parameters T_{send} and T_{rec} can be updated. Then, the distance between the two nodes can be computed according to $T_{\text{diff}} = T_{\text{rec}} - T_{\text{send}}$. Consequently, the unknown node's position can be approximated with the first three packets;

3) For all N received packets, take 3 packets out of N, we will get C_N^3 combinations. Calculate the distance between the two nodes respectively and decide whether they are in a line or not;

4) For each group's nodes which are not in a line, computing the unknown node's position respectively, we will get estimated positions $\{L_1, L_2 \ldots\}$;

5) Average elements of the estimated positions set, get estimated position of the unknown node L_{avg}. Then, the algorithm comes to an end.

Obviously, the traditional polygon location algorithm roughly calculates the unknown node's position at first, and then improves the locating precision through multiple similar iterative processes. With the interests of the reference nodes' amount, the algorithm will exponentially increase. In conclusion, the traditional algorithm is unable to meet the requirements of real-time positioning.

3.2 The location reference nodes selection algorithm

Based on theorems presented above, it is essential to place reference nodes as multiple equilateral triangles, which spread out seamlessly in the possible region of positioning node. The LRNS algorithm is described as follows.

1) Each reference node sends broadcasting packets, including location information denoted as {ID, T_{send}, (a, b)}, and ID is identification of the reference node. T_{send} is time when the message is sent, and (a, b) are its coordinates;

2) The unknown node receives multiple broadcasting packets. For N received packets, combine C_N^3 times. During each combination, compute the distance between two nodes respectively using $T_{\text{diff}} = T_{\text{rec}} - T_{\text{send}}$, and then judge whether the three reference nodes can form an equilateral triangle;

3) Roughly compute the unknown node's position using the initial equilateral triangle;

4) As for each equilateral triangle, meeting requirements, we compute the unknown node's position respectively and get a location set $\{L_1, L_2 \ldots\}$;

5) Average elements in the estimation positions set, termed as L_{avg}, then L_{avg} is the estimated position of the unknown node. The algorithm comes to an end.

The LRNS algorithm has several advantages, such as fewer calculations, high real-time performance, and small location errors. In this way, it can be used to carry out real-time tracking to mobile users and meet location needs under the ubiquitous computing environment perfectly.

4 Location error analyses

Mainly, factors that affect the location error are: 1) measurement error of the distance between the nodes; 2) the number of reference nodes that are included in the location; 3) the relative geometric position of the reference nodes.

Compare the location error introduced by the reference nodes placement according to the three theorems with error brought in by random placement, we know performances of the placement theorems and LRNS algorithm.

First, compute the location errors introduced by placing the reference nodes based on the theorems. Seen from Theorem 1, the location error will be the smallest if the location reference nodes are placed as an equilateral triangle. By computing Eq. (5) in Sect. 2.2, minimum value of location error can be obtained as follows

$$S(\widetilde{C}_{p_0}) = 2\sqrt{3}\varepsilon^2 \tag{11}$$

According to Theorem 2 and Theorem 3, use the topology duplication to introduce new reference nodes. Then, we can further reduce the location error, which tends to be $\pi\varepsilon^2$ as the reference nodes increase.

Secondly, compute the location error introduced by the random reference nodes placement. This error is related to the triangle composed reference nodes. Now compute the mathematical expectation of error $E[S(\widetilde{C}_{p_0})]$, namely the average error area. Integrated with and substitutes Eq. (4) in Sect. 2.2, $E[S(\widetilde{C}_{p_0})]$ is expressed as

$$E[S(\widetilde{C}_{p_0})] = 2\varepsilon^2 E\left[\tan\frac{\alpha_{12}}{2} + \tan\frac{\alpha_{23}}{2} + \tan\frac{\alpha_{13}}{2}\right]$$

For convenience of the calculation and deduction, we replace α_{12} , α_{23} , α_{13} by x, y, z, then x, y, z and $x+y+z=\pi$. Eliminate the unknown variable z, we have

$$E[S(\widetilde{C}_{p_0})] = 2\varepsilon^2 E\left[\tan\frac{x}{2} + \tan\frac{y}{2} + \cot\frac{x+y}{2}\right] \tag{12}$$

Since the variable x, y follows the uniform distribution in region D, $D=\{(x, y)|x>0, y>0, x+y<\pi\}$, the joint probability density function of x, y is

$$f(x,y) = \frac{2}{\pi^2}$$

Substitute it into Eq. (12), we get

$$\begin{aligned} E[S(\widetilde{C}_{p_0})] &= 2\varepsilon^2 E\left[\tan\frac{x}{2} + \tan\frac{y}{2} + \cot\frac{x+y}{2}\right] \\ &= \frac{4\varepsilon^2}{\pi^2}\iint_D \left(\tan\frac{x}{2} + \tan\frac{y}{2} + \cot\frac{x+y}{2}\right)\mathrm{d}x\mathrm{d}y \\ &= \frac{4\varepsilon^2}{\pi^2}\left(\int_0^\pi\int_0^{\pi-x}\left(\tan\frac{x}{2} + \tan\frac{y}{2} + \cot\frac{x+y}{2}\right)\mathrm{d}y\mathrm{d}x\right) \\ &= \frac{4\varepsilon^2}{\pi^2}\left(\int_0^\pi(\pi-x)\tan\frac{x}{2}\,\mathrm{d}x - 4\int_0^\pi \ln\sin\frac{x}{2}\,\mathrm{d}x\right) \end{aligned} \tag{13}$$

Through the previous deduction of Eq. (13) we have

$$\begin{aligned} \int_0^\pi(\pi-x)\tan\frac{x}{2}\,\mathrm{d}x &= -2\left((\pi-x)\ln\cos\frac{x}{2}\Big|_0^\pi \right. \\ &\left. + \int_0^\pi \ln\cos\frac{x}{2}\,\mathrm{d}x\right) = -2\int_0^\pi \ln\cos\frac{x}{2}\,\mathrm{d}x \end{aligned} \tag{14}$$

Substitute the result of Eq. (14) into Eq. (13), and do suitable variable replacement and the computation, we can know

$$\begin{aligned} E[S(\widetilde{C}_{p_0})] &= \frac{4\varepsilon^2}{\pi^2}\left(-2\int_0^\pi \ln\cos\frac{x}{2}\,\mathrm{d}x - 4\int_0^\pi \ln\sin\frac{x}{2}\,\mathrm{d}x\right) \\ &= \left(\frac{24}{\pi}\ln 2\right)\varepsilon^2 \end{aligned} \tag{15}$$

Introducing new reference nodes can further improve location error. At the same time this error will tend to be $\pi\varepsilon^2$ as the reference nodes grow. Undoubtedly, it requires more calculation and consumes more processing costs.

Finally, compare Eq. (11) with Eq. (15), the result indicates that placing reference nodes according to the reference nodes placement theorems can reduce the location error and improve the location precision. The improvement in location precision is termed β, and computed as

$$\beta = \frac{\left(\frac{24}{\pi}\ln 2\right)\varepsilon^2 - 2\sqrt{3}\varepsilon^2}{\left(\frac{24}{\pi}\ln 2\right)\varepsilon^2} = 34.9\%$$

After computing the location error in an improved situation, we have noted that the location accuracy can be improved by 34.9% based on an equilateral triangle according to the three theorems. The errors will be further reduced through equilateral triangles topology duplication. By the way, the above analysis is on the assumption that the distance measurement error is ε. However, in the actual location, there are all kinds of uncertain factors, which means that the ideal value for location accuracy is unattainable. What's more, placing reference nodes with a purpose can improve the location accuracy of the unknown node effectively.

5 Simulation experiments and the algorithm evaluation

5.1 Simulation environment and test methods

Three methods have been used in the test of reference nodes placement theorems and the LRNS algorithm.

1) Using MATLAB to compute for the location error, and inspect the changing tendency as of reference increase;

2) Test the real-time performance on the platform of Webit5.0;

3) Simulate the wireless environment of a polygon location algorithm by ns-2, and validate the performance of the LRNS algorithm proposed in this paper by importing noises or other interference factors such as distance measurement error.

Webit5.0 is a ubiquitous terminal. It is developed independently by the embedded technology laboratory of Liaoning Province. Its master control chip is 8-bits micro controller AVR Atmega128. WebitOS5.0 is an embedded real-time operating system based on priority preempted. WebitOS5.0 has a light TCP/IP protocol stack, and supports hard real-time application and communications between equipment. The reason that we chose this platform for testing is to argue whether the location algorithm can run in the equipment with limited resources and whether it can satisfy the mobile users' real-time tracking and location services.

5.2 Analysis and validation of the three theorems

The location error of the positioning node is affected by factors such as distance measurement error ε and the number

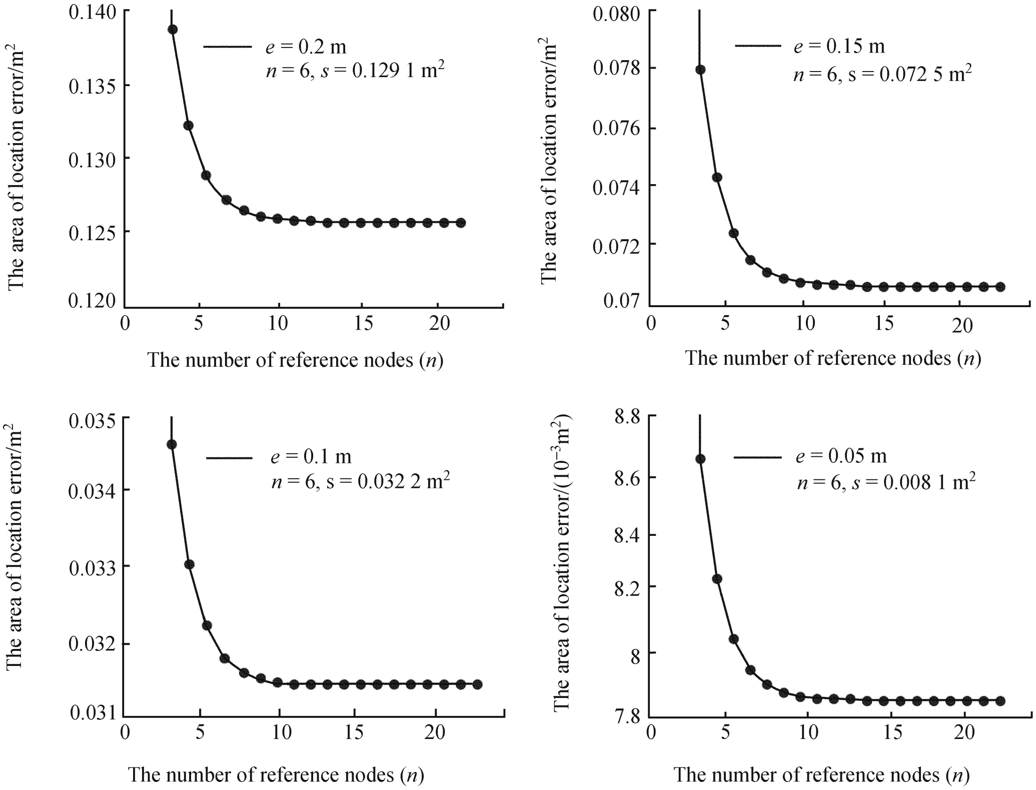

Fig. 7 The effects of ε and n to location errors

of reference nodes participating in location n. Consider that the location errors vary with ε and n, values respectively by MATLAB. As we know, indoor distance measurement error varies from several centimeters to dozens of centimeters. To discuss, we select four different measurement errors: 20 cm, 15 cm, 10 cm, and 5 cm. The test results are shown in Fig. 7.

Seen from Fig. 7, the distance measurement error greatly influences the location error area. In addition, they follow a square relationship. So during actual location computing, we should do our best to improve the accuracy of the distance. Also, the number of reference nodes has a great influence on the location error. When the reference nodes increases, the location error decreases exponentially. If the number of reference nodes is larger than 6, the error decreases slowly. It is worth noting that the location error tends to be a constant if the number of reference nodes increases unlimitedly, which validates Theorem 3.

In the instance of the same distance error, decreasing the location error can result in the increase of reference nodes, and then the growth in communication and computing between nodes. That is to say, the real-time performance and the success rate of locating will get worse. In Fig. 8, we chose 200 nodes, each node has been located 500 times, and the average location error is computed as well as the success rate of locating. To reflect the reality better, three kinds of simulation tests have been taken: 1) with location signal, but no noise and disturbance; 2) adding noise a bit; 3) Join a few noise and disturbance signal. The result is shown in Fig. 8.

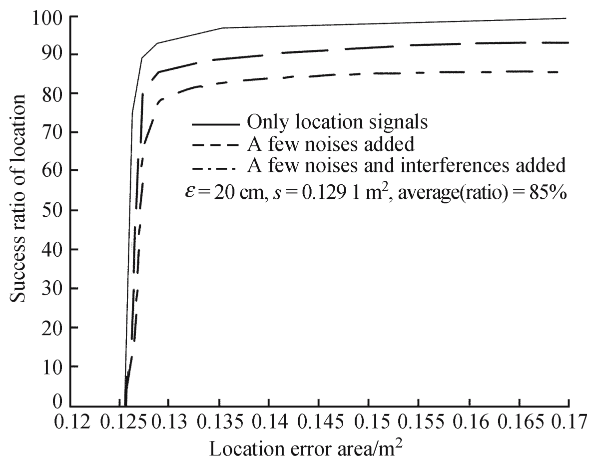

Fig. 8 Location error area vs. location success ratio

As seen from the left top figure of Fig. 7, when $\varepsilon = 20$ cm, the location error $s = 0.129\ 1$ m^2, the number of the reference nodes $n = 6$, as the abscissa indicates. In Fig. 8, set the area of location error 0.129 1 m^2, viz. choose 6 reference nodes to locate, the average success rate of location is about 85%, which seems really high. When the location error is smaller than this value, viz. more reference nodes participating in the iterative operation, the success rate of location descends rapidly since the computation becomes very complex or the delay in computation gets too large. This validates the optimal computation unit of LRNS algorithm shown in Fig. 6. Assume that the side length of the equilateral triangle composed of the reference nodes is 4.6 m; the distance of signal transmission between nodes is in the range of [5.3 m, 7.1 m]. 6.2 m in the simulation ensures the 6 reference nodes surrounding the location node can communicate with each other, but cannot communicate with other nodes randomly.

5.3 Comparison between the LRNS algorithm and the traditional algorithm

The LRNS algorithm proposed in this paper is an improvement of the traditional polygon location algorithm according to the theorems. Now compare the LRNS algorithm with the traditional algorithm in three different aspects: location error, location real-time performance and position tail track.

5.3.1 Location error

For simulation, 21 nodes have been built randomly by ns-2, select two placement modes, and their communication distance is 6.2 m. The placement of reference nodes has two solutions, one is the method according to the reference node placement theorems, and the other using random placement. The relationship between the location error and the number of reference nodes is shown in Fig. 9.

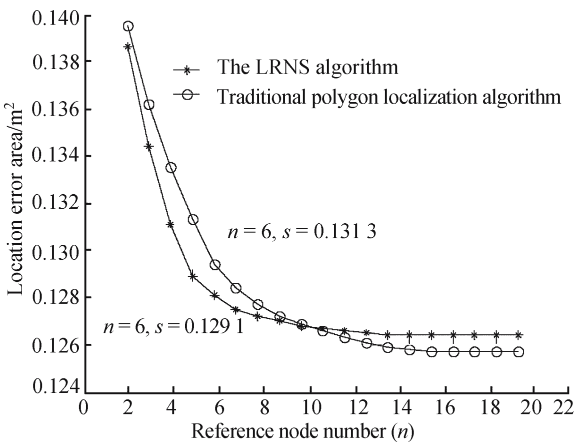

Fig. 9 Location error area compared between the both algorithms

As seen from Fig. 9, with the increase of reference nodes, the location error of an unknown node decreases. The decrease seems to be smart in the beginning and slows down later. When the number of reference node is about 6, there is a great difference of the location error area between the two algorithms, and the LRNS algorithm performs better. When the number of reference nodes number goes beyond 10, the traditional polygon location algorithm behaves better. Why? Because the number of equilateral triangle grows slower than the in-equilateral triangle with the increasing of reference nodes, and the number of the new reference nodes participating in the location is less. At the same time, the calculations of the traditional polygon location algorithm increases exponentially, which means getting a higher location precision by increasing computation.

5.3.2 Real-time performance

On the platform of the Webit5.0, we can compare the performances of the two algorithms with various numbers of reference nodes *n*, especially the real-time performance. In this experiment, only 12 reference nodes are used because of the capacity limitation of Webit5.0. The result is shown in Fig. 10.

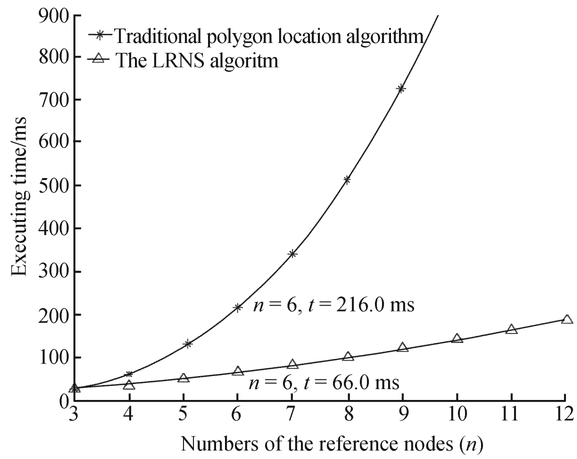

Fig. 10 Reference nodes vs. cost positioning time

From the experimental results, when the number of reference nodes is about 6, the time cost of LRNS algorithm is about 66.0 ms, and the traditional polygon location algorithm is about 216.0 ms. Especially, along with the increasing of reference nodes, the time cost of the latter will increase according to power law, while the time cost of LRNS algorithm tends to be linear. In other words, to guarantee the real-time positioning with a small location error in an indoor environment, we shouldn't take many into location computing. Once the optimal computing unit is formed, location costs would only be tens of milliseconds. Thus, the LRNS algorithm meets the requirements of real-time positioning.

5.3.3 Positioning track

To compare performances between the proposed LRNS algorithm and the traditional polygons location algorithm, we use ns-2 to simulate a $21.0\ \text{m} \times 15.0\ \text{m}$ indoor environment. We place 21 reference nodes and one mobile positioning node by random placement and by the above reference nodes placement theorems respectively. The results show that mobile speed of the positioning node is about 0.7 m/s, which is the average indoor walking speed. With the movement of the node, the number of reference nodes within its communication radius changes correspondingly, and also the reference nodes in the positioning. Therefore, we can do some comparisons between the positioning tracks by the two algorithms and the real mobile track. Error is on the level of centimeter. Easy to display, we only give part of the results in Fig. 11.

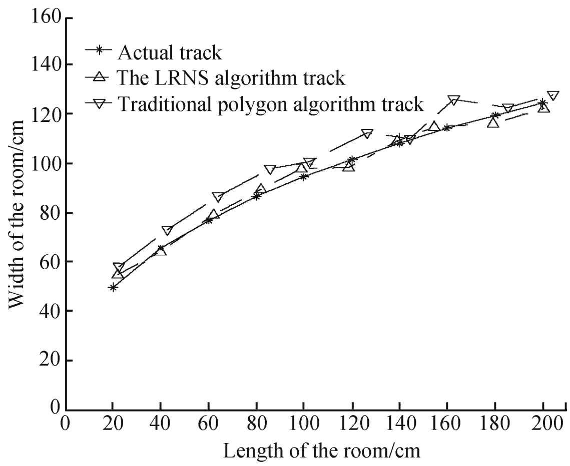

Fig. 11 Comparisons between positioning tracks and real mobile track

Clearly, the proposed LRNS algorithm is more accurate than the traditional polygon algorithm. Furthermore, the positioning track by LRNS algorithm is close to the actual walking track.

6 Conclusions

It is obvious that positioning of an unknown node accurately is a key problem of location services in ubiquitous computing. To sum up, the paper is organized in the following way. First, reducing the location error is analyzed. Also, three theorems about reference nodes placement are presented and proved. The three theorems accord with the chaotic dynamic system and provide a theoretical basis for reference nodes placement in two-dimensional space. Secondly, the reference nodes selection algorithm is presented on the basis of the improvement to the polygons location algorithm according to the three theorems. It is proved that it cannot only reduce location error but meet the real-time location requirements. Finally, the placement theorems of reference nodes and the location reference nodes selection algorithm are validated through simulation.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant No. 69873007), the Hi-Tech Research and Development Program of China (No. 2001AA415320).

References

1. Steer D, Fauconnier D. Location services architecture for future mobile networks. IEEE VTC’2000-Spring. Tokyo: IEEE Press, 2000, 1362−1366
2. Cabri G, Leonardi L, Mamei M, et al. Location-dependent services for mobile users. IEEE Transactions on Systems Man and Cybernetics, Part A, 2003, 33(6): 667−681
3. Hightower J, Borriello G. Location systems for ubiquitous computing. IEEE Computers, 2001, 34(8): 57−66
4. Niculescu D, Nath B Ad-hoc positioning system. IEEE GLOBE-COM 2001. New York: IEEE Press, 2001, 2926–2931
5. Savarese C, Langendoen K, Rabaey J. Robust positioning algorithms for distributed Ad-hoc wireless sensor networks. Proceedings of the USENIX Technical Annual Conference. Monterey: USENIX Association, 2002, 317−328
6. Savvides A, Park H, Srivastava M. The bits and flops of the *N*-hop multi-lateration primitive for node localization problems. Proceedings of the first ACM International Workshop on Wireless Sensor Networks and Application (WSNA’02). Atlanta: ACM Press, 2002, 112−121
7. Langendoen K, Reijers N. Distributed localization in wireless sensor networks: a quantitative comparison. Elsevier Computer Networks, 2003, 43(4): 499−518
8. Ding J, Hitt L R, Zhang X M. Sierpinski pedal triangles. Linear Algebra and its Application, 2005, 1−15
9. Choi W J, Tekinay S. Location-based service provisioning for next generation wireless networks. International Journal of Wireless Information Networks, 2003, 10(3): 127−139
10. Hitt L R, Zhang X M. Dynamics geometry of polygons. Elemente der Mathematik, 2001, 56(1): 21−37
11. Dearle A. Toward ubiquitous environments for mobile users. IEEE Internet Computing, 1998, 2(1): 22−32
12. Doherty L. Algorithms for position and data recovery in wireless sensor networks. Master’s Thesis. UC Berkeley, 2000, 1−54
13. Priyantha N B, Chakraborty A, Balakrishnan H. The cricket location-support system. In: Proceedings of ACM MOBICOM. Boston: ACM Press, 2000, 32−43