TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.3, March 2014, pp. 2079 ~ 2087
DOI: http://dx.doi.org/10.11591/telkomnika.v12i3.4512

Design and Application of Iterative Monte Carlo Localization for Mobile Wireless Sensor Networks Based on MCL

Wu Junsheng
School of Engineering College, Shanxi University,
Taiyuan, Shanxi, 030013, China
e-mail: wujsheng@163.com

Abstract

In recent years, wireless sensor network had been more and more widely used in our daily life, and with the proposed of Monte Carlo localization (MCL) algorithm, node localization of the mobile wireless sensor network had been solved effectively. But it needed to have a large number of samples if it used the Monte Carlo localization algorithm to obtain a high positioning accuracy. This paper proposed a new improved algorithm (iterative Monte Carlo localization algorithm) based on the Monte Carlo localization algorithm. In iterative Monte Carlo localization (IMCL) algorithm, information each of anchor node location was forwarded by its neighbor nodes only once and preserved by the receiving node in each period. Then the next period, merge it and the sent/forwarding information into a packet and forward. Make sure that points have more observations for estimating previous location sets. IMCL, meanwhile, also can make full use of observation to filter out some samples that were far from the real position of node, so as to improve the accuracy of node localization. We finally confirmed by experiment: IMCL algorithm had higher positioning accuracy compared with other algorithm.

Keywords: *IMCL, immune genetic algorithm, wireless sensor network, improve the accuracy of localization*

1. Introduction

Wireless Sensor network (WSNs) has characteristics of self-organization, low cost, low power consumption, and it has broad application prospects in the environmental monitoring, remote medical monitoring, military and other fields. Generally, it is meaningful for monitoring information that the location has been determined. GPS devices can be installed on each sensor node to determine the location information, but it is very expensive. Therefore, the research of wireless sensor network node location technology is of great significance.

In recent years, many scholars at home and abroad devote to the research of node location technology, and put forward many effective algorithm [1]. During the process of positioning, according to whether there is need to measure the distance between nodes, node localization algorithm can be divided into two categories:

(1) Positioning algorithm based on distance: these algorithms rely on the range information between the nodes of additional hardware measuring, such as the received signal strength (RSSI), the time of arrival (TOA) [2], arrival time difference of the different signal (TDOA) and the signal arrive Angle (AOA). The shortcomings of this method is that the extra range of hardware increases volume and the cost of node.

(2) Localization algorithm irrelevant with distance: this kind of algorithm achieves node localization mainly based on the connectivity between the nodes, and localization cost is low. Centroid algorithm is a simple localization algorithm which is irrelevant with distance, unpositioning node takes polygon centroid as its estimate which is composed by its neighbors of beacon nodes. DV - Hop algorithm is typical. In addition, many scholars also explore the positioning method based on the mobile beacon nodes [3-5].

Received August 22, 2013; Revised September 26, 2013; Accepted October 17, 2013

2. MCL Algorithm

2.1. The Introduction of MCL Algorithm

In MCL algorithm, time is divided into several discrete time, and suppose that the unpositioning node and beacon nodes do not know their own direction and speed, but their maximum velocity vmax is given [6]. Unpositioning node needs to relocation in each time period. $L_k=\{l_k^0, l_k^1, , , l_k^{n-1}\}$ means the location of the sample set of unpositioning node at k time[7]. At the beginning, unpositioning node has no any information about the location, so it randomly selects N points to form the initial location of sample set $L_0=\{l_0^0, l_0^1, , , , l_0^{n-1}\}$ from decorate area. In the following time period, unpositioning node repeats the period of forecast and filtering to implement the position. Take the positioning process at K time for example. In the prediction stage, according to the location of sample set at the k - 1 $L_{k-1}=\{l_{k-1}^0, l_{k-1}^1, , , l_{k-1}^{n-1}\}$ and its motion model, unpositioning node randomly collects N samples $L_k=\{(l_k^0)c, (l_k^1)c, , , , (l_k^{n-1})c\}$, namely $(l_k^i)c$(i=0, 1, 2, , , N-1) randomly selects from the circle which is centered in l_{k-1}^i and the radius is v_{max}. In updating stage, unpositioning node uses the the observed value $O_k=\{S_1, S_2, , , D_1, D_2, , , \}$ received at k moment to filter samples that do not satisfy the conditions. S_j and D_j are the position of the beacon nodes of unpositioning node. Suppose receive all the information instantly at k time and the define $d(l_1, l_2)$ is the Euclidean distance between location l_1 and l_2, r is the wireless communication distance of a node. Filter conditions of samples $(l_k^i)c$ are:

$$\text{filter}((l_k^i)c)=PS_j,\ d((l_k^i)c,\ S_j)[r$$

$$CPDj,\ r<d((l_k^i)c,\ D_j)[2r \tag{1}$$

If the value of the filter conditions is false, $(l_k^i)c$ will be filtered. In particular, if the distance between $(l_k^i)c$ and jump of the beacon node is greater than r or the distance between $(l_k^i)c$ and two jump beacon node is not between r and 2 r, then it means that $(l_k^i)c$ does not comply with the conditions, and should be filtered. On the other hand, n $(l_k^i)c$ should be stayed.

Due to the unfitted samples have been filtered, after filtering stage, there may be not enough samples, unpositioning node repeats prediction and filtering process until the samples are enough. Then, these samples form the location of the sample set $L_k=\{l_k^0, l_k^1, , , , l_k^{n-1}\}$ at k. And take all the sample average as position estimation of unpositioning node at k.

2.2. The Analysis of MCL Algorithm

According to the positioning process of MCL algorithm [8], we can deduce that the following two factors may affect the point positioning accuracy of unpositioning node at k .

(1) Observations O_k: The direct way to increase the observation is to improve the density of beacon node, but it will increase the cost of system, including hardware cost and communication consumption of beacon nodes. Another common method is to use location information of multiple hops beacon node, but it can lead nodes forwarding packets frequently in each time period to increase the node communication costs. In addition, multiple hops can also cause a large time delay of communication.

(2) The location of the sample set L_{k-1} at k-1: if at k - 1, the more closer to the actual location of unpositioning node l_{k-1}^{i} (i=0,1,2, , , N-1) is at k, the larger the probability that predicting samples (l_{k}^{i})c is close to the node's actual location is. And improve the positioning accuracy of unpositioning node at k.

3. IMCL Algorithm

3.1. The Basic Definitions and Symbols

For convenience, this section will illustrate the related definitions and symbols.

S_i means the beacon node i, i=0, 1, 2, , , ; N_i means the unpositioning node i, i=0, 1, 2, , , ; S_{k}^{i} is the position of beacon node S_i of unpositioning node at k; D_{k}^{i} is the position of beacon node S_i of unpositioning node at k; T_{k-1}^{i} is the position of beacon node S_i of unpositioning node at k-1. However unpositioning node can receive information at k, but not at k-1; M_{k-1}^{i} is the position of multiple jumps beacon node S_i of unpositioning node at k-1(l\2), similarly, the unpositioning node can receive information at k; O_{k-1}^{k} is the observation of unpositioning node at k which is used to calculate the position sample set at k-1, 0[l[k; l_{k-1}^{i} |k is a sample set during prediction stage,when the unpositioning node at k calculates the position sample set at k-1,0[l[k.

3.2 Design of Algorithm

Firstly, analyze how unpositioning node estimates observations of position sample sets in previous times. As shown in Figure 1, a circle means the moving unpositioning node, triangle means the moving beacon nodes. Linear between nodes means communication link. In order to simplify the graphic, Figure 1 only gives a few typical communication link. The line symbol p_j^t represents the position of the beacon node j at t, it is included in the node packets sent currently. Similar to the MCL algorithm, these packets which include position of beacon node can be forwarded at most once in each segment. However, the node which receives the packet will save these information, and merge them into momentum/forwarding packets to forward in the next period of time. At k - 2, for example, N2 receives the packets which contains its beacon node location (p_{2}^{k-2}), and forward the packet immediately; at k - 2, N_4 receives the packets which contains its beacon node location (p_{5}^{k-2}), but N_4 does not forward it immediately. At k -

1, N_4 receives the packets which contains its beacon node location (p_4^{k-1}), then forwards the packet, at this point, N_4 merges p_5^{k-2} into the packets and sends out. Obviously, IMCL algorithm does not increase the packet number. Of course, the bytes number of each packet will increase. However, through proper fusion technology, wen can minimize the number of bytes.

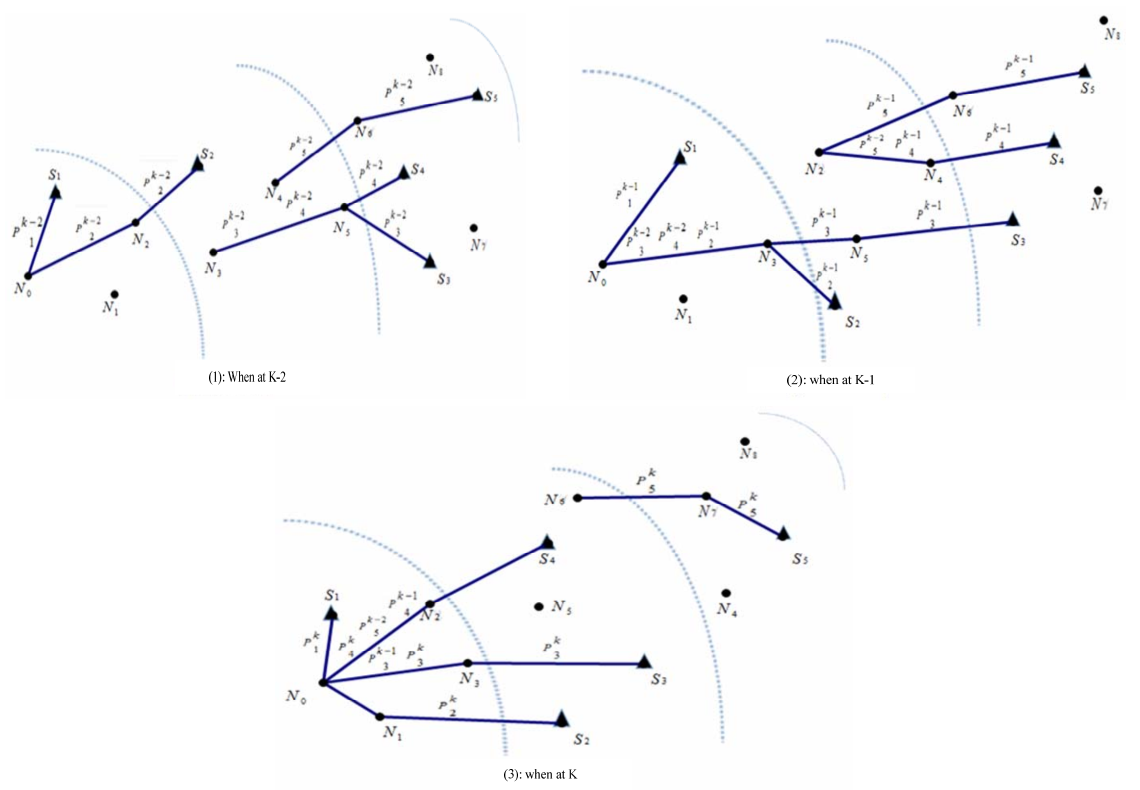

Figure 1. The Communication Link of Nodes between k-2 and k, where N^0 is Static

According to the analysis of Figure 1, from k - 2 to k, the observations of position sample set that the unpositioning node N_0 saves to estimate at k - 2 are:

$$O_{k-2}^{k-2}=\{S_{k-2}^{1}, D_{k-2}^{2}\}, O_{k-2}^{k-1}=\{S_{k-2}^{1}, D_{k-2}^{2}, T_{k-2}^{3}, T_{k-2}^{4}\},$$
$$O_{k-2}^{k}=\{S_{k-2}^{1}, D_{k-2}^{2}, T3_{k-2}^{3}, T_{k-2}^{4}, M_{k-2}^{5}\} \quad (2)$$

Obviously, O_{k-2}^{k-2} A O_{k-2}^{k-1} A O_{k-2}^{k}.

It shows that unpositioning node can obtain more observations of position sample set of k-i at k-j than at k-s. According to the analysis of the influence factors 1 in the section 2.2, the result of position sample set of k-i at k-j is better than at k-s. At k, therefore, it is necessary for unpositioning node to recalculate the periods location of the sample set to get a better L_{k-1}. At the same time, IMCL refers to MCB algorithm in order to collect better samples during forecast period. The definition $L_{t'}^{t}$ is the position sample set of t_c that is calculated by unpositioning set at t. Figure 2 is the flow diagram of IMCL algorithm. Consider unpositioning node reiterate location of the sample set of previous n-1 times at k, thenL_{k}^{k} can be got from the process of n MCB iterative calculation, namely:

$$L_{k-n+1}^{k} = \text{MCB}\{L_{k-n}^{k-1},\ O_{k-n+1}^{k}\},\ L_{k-n+2}^{k} = \text{MCB}\{L_{k-n+1}^{k},\ O_{k-n+2}^{k}\},$$
$$L_{k-n+3}^{k} = \text{MCB}\{L_{k-n+2}^{k},\ O_{k-n+3}^{k}\},\ L_{k-1}^{k} = \text{MCB}\{L_{k}^{k}\text{-2},\ O_{k-1}^{k}\},$$
$$L_{k}^{k} = \text{MCB}\{{}_{k-1}^{k},\ O_{k}^{k}\} \quad (3)$$

The average of all samples in L_{k}^{k} are the estimated position of the unpositioning node at k.

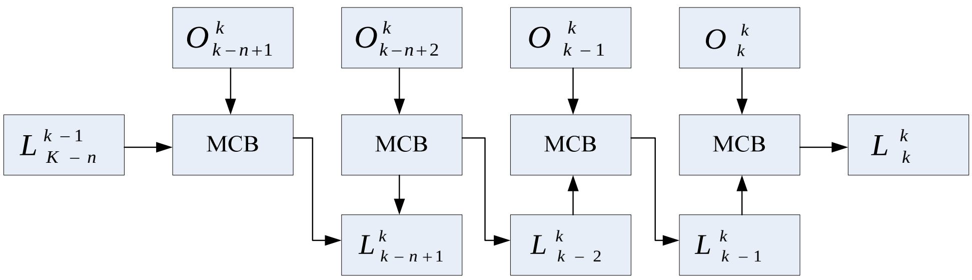

Figure 2. The Diagram of IMCL Algorithm

3.3. The Filtering Conditions

In order to make full use of the observed value, we need to complete the filtering conditions of MCL and MCB algorithm. In IMCL, observation Okk-l contains four types of beacon node location: Sik -l, Dik -l, Tik - l and Mik - l. Therefore, in the process of each MCB, prediction sample li'k-l|k has four corresponding filter conditions:

$$\text{filter1}({}_{k-1}^{i'}|k)=PS_{k-1}^{i}IO_{k-1}^{k},\ d(l_{k-1}^{i'}|k,\ S_{k-1}^{i})[r \quad (4)$$

$$\text{filter2}(l_{k-1}^{i'}|k)=PD_{k-1}^{i}IO_{k-1}^{k},\ r<d(l_{k-1}^{i'}|k,\ D_{k-1}^{i})[2r \quad (5)$$

$$\text{filter3}(l_{k-1}^{i'}|k)=PT_{k-1}^{i}IO_{k-1}^{k},\ 2r<d(l_{k-1}^{i'}|k,\ T_{k-1}^{i})[3r+2v_{max} \quad (6)$$

$$\text{filter4}({}_{k-1}^{i'}|k)=PM_{k-1}^{i}IO_{k-1}^{k},\ 2r<d(l_{k-1}^{i'}|k,\ M_{k-1}^{i}l) \quad (7)$$

In addition, the movement distance of the nodes within a period of time is not more than v_{max}. This means that if N_j and S_i are not neighbor nodes ii k - l - 1, but are neighbor nodes in k - l, then the distance between them is between r-2v_{max} and r in k - l. On the contrary, the distance between them should be between r-v_{max} and r+v_{max}. This type of filtering is called Motion on Boundary, which can be written as the following form:

$$\text{filter5}(l_{k-1}^{i'}\text{-l}|k)=PS_{k-1}^{i}IO_{k-1}^{k}CS_{k-1}^{i}|O_{k-1}^{k},\ r\text{-}2v_{max}<d(l_{k-1}^{i'}|k,\ S_{k-1}^{i})[r \quad (8)$$

$$\text{filter6}(l_{k-1}^{i'}|k)=PS_{k-1-1}^{i}IO_{k-1-1}^{k}CS_{k-1}^{i}|O_{k-1}^{k},$$

$$r\text{-}v_{max}<d(l_{k-1}^{i'}|k,\ S_{k-1-1}^{i})[r+v_{max} \quad (9)$$

So the filtering conditions of prediction sample $l_{k-1}^{i'}|k$ are:

$$\text{filter}(_{k-1}^{i'}|k)=\text{filter1}(l_{k-1}^{i'}|k)C,\ \ C\text{filter6}(l_{k-1}^{i'}|k) \quad (10)$$

We find that T_{k-1}^{i} and M_{k-1}^{i} may be useless O_{k-1}^{k}. Because node is random in movement speed and direction will lead nodes cannot be connected in a certain period of time. In k - l, for example, the distance between S_i and N_i is between r and 2r, but they lack a rely node, so N_1 can't get the location information of Si at a time of k - l, but can receive this information at the time of k, namely T_{k-1}^{i} or M_{k-1}^{i}. The process according to the sub-conditions of filter 3 or 4 is clearly not appropriate. Although the probability is low, in order to make more precise positioning process, in IMCL algorithm, set a limit. If the number of filtering prediction sample which does not meet filter3 or filter4 are more than the limit value, then remove T_{k-1}^{k} or M_{k-1}^{k} from O_{k-1}^{k}.

4. The Simulated Analysis

According to IMCL algorithm, we extend the MCL simulator. This section uses extended simulator to test the performance of IMCL algorithm. All nodes randomly distributed in a 500m @ 500m square area, and the communication radius of beacon node and unpositioning node is 50 m, the density of unpositioning node (the average number of neighbored node) and number of samples are 10 and 50 respectively. We analyze the IMCL algorithm from several aspects:

(1) With the change of time, the positioning error of different MCL algorithm;

(2) Research the influence of beacon node density S_d to the positioning accuracy. In the rest tests, the density of beacon node is fixed, that is S_d=1;

(3) Research the influence of the node maxmum speed v_{max} to the positioning performance. In the rest tests,set v_{max}=0.2r;

(4) Analyze the influence of iterations n to the positioning accuracy and processigng time. In other tests, get n=6.

All the simulation results are got from the average of 20 times independent experiments. Positioning error is the ratio between absolute positioning error and the radius of wireless communication. The average position error is the average of positioning error of all unpositioning nodes. In particular, the average positioning error of k3.2 ~ k3.4 is measured in the stable stage of position.

4.1 Positioning Error

Figure 3 is the average positioning error curve of different MCL algorithm. During the positioning initialization phase, the average position error between IMCL and MCB is close, however, as time goes on, IMCL achieves better location performance. In the end, all MCL algorithm can achieve a stable state, i.e. the average positioning error fluctuates in a small scope. In stable phase, compared with the MCL and MCB, the average positioning error of IMCL algorithm has fallen about 40% and 40% respectively.

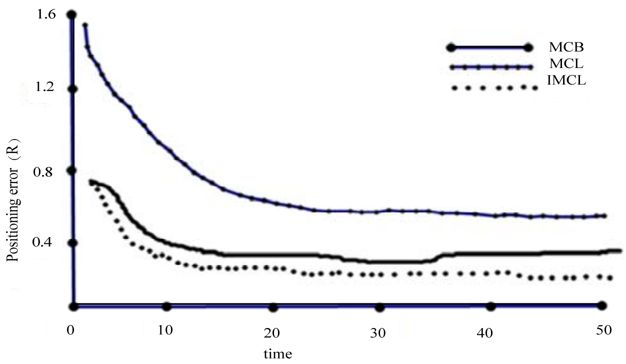

Figure 3. The Average Positioning Error of Different MCL $(S_d = 1, V_{\max} = 0.2x, n = 6)$

4.2. The Density of Beacon Node

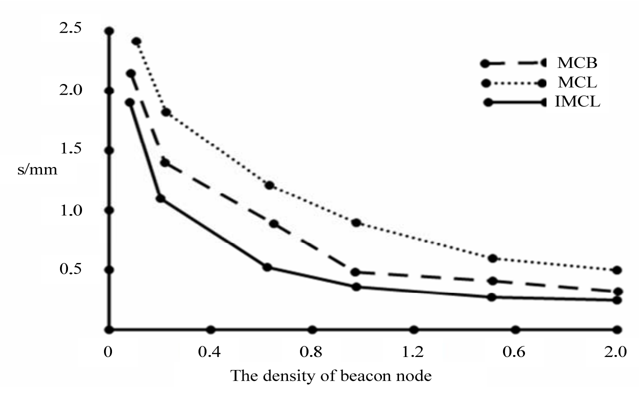

Figure 4. The Influence of Beacon Node Density to the Positioning Accuracy $(V_{\max} = 0.2r, n = 6)$

Increase the density of beacon node can improve positioning accuracy, but also increase the network cost and energy consumption of communication. As the change of S_d, the average position error of different MCL algorithm is shown in Figure 4. With the increase of S_d, positioning accuracy of all MCL algorithms is increased, because the unpositioning node can obtain more observations. But it is worth noting that the increase of S_d has large influence on positioning accuracy of the MCL and MCB. In particular, when S_d is greater than 1.5, the average positioning error of IMCL is close to MCB. For how to get more observations is one of the inspired factors of IMCL algorithm, so if the S_d is too high,there will exist redundant observation. This shows that under the condition of the low density of beacon node, IMCL algorithm has more advantages.

4.3. The Maximum Speed of Node

Figure 4 is the average position error curve of different MCL algorithm with the change of v_{max}. Pending a node and beacon node's actual speed is uniformly distributed in the interval [0, v_{max}], so the node's average speed is proportional to the v_{max}. From the diagram, we can see that along with the increase of v_{max}, the difference of the average position error between IMCL with MCB (MCL) decreases. Because v_{max} affects the positioning process from two aspects. First of all, in the prediction stage, the sample is selected from the circle which is centered on l_{t-1}^{i} and the radius is v_{max}, so the increase of v_{max} will cause a decline in the accuracy of the sample collection. The negative effect on all MCL algorithm is same. In each time period, on the other hand, the greater the average speed of nodes is, the more observed values received are, which can filter out more samples do not comply with the conditions. However, one of the key technologies of IMCL algorithm is to make full use of the observed value to filter out worse samples, so the positive effect is not obvious.

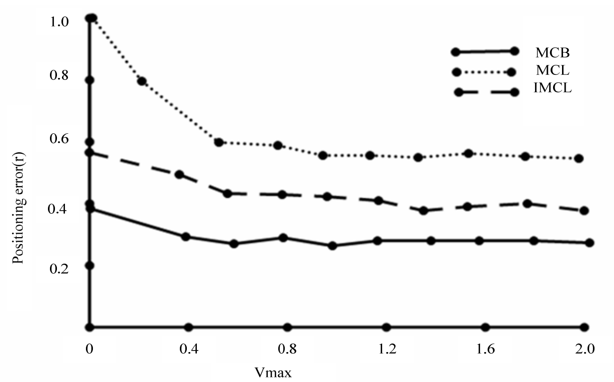

Figure 5. The Influence of Node Maximum Speed on the Positioning Accuracy

4.4. Iterations

Change the number of iterations n, the average position error and the average processing time of IMCL (in each time period of the positioning process, the time of performing prediction and update phase) are shown in Figure 6 and 7 respectively. With the increase of the number of iterations, the positioning accuracy has improved, the processing time also increases. However, when the number of iterations reaches to a certain value, the number of iterations has less influence on positioning accuracy. Therefore, by adjusting the numbers of iterations can the keep the relationship between the positioning accuracy and the processing time required balance. In addition, we can find an interesting result from Figure 7: the required processing time when the number of iterations is 1 is longer than that when the number of iterations is 2. Because when n = 1, IMCL algorithm degenerates into MCB algorithm. In MCB algorithm, the accuracy of position of sample set in predict phase is low, so after filtering stage, reserved samples is less, leading that we have to constantly perform re-sampling/update process in place to ensure that there is enough samples.

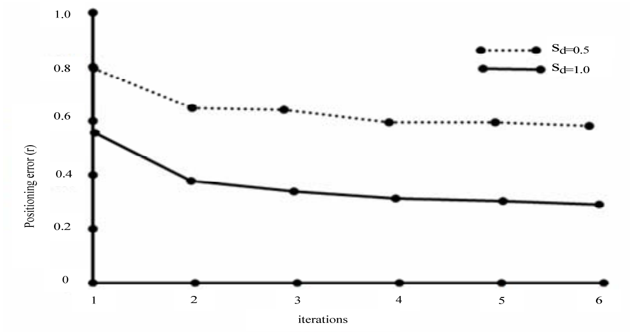

Figure 6. The Influence of Iterations on Positioning Accuracy $(V_{\max} = 0.2r)$

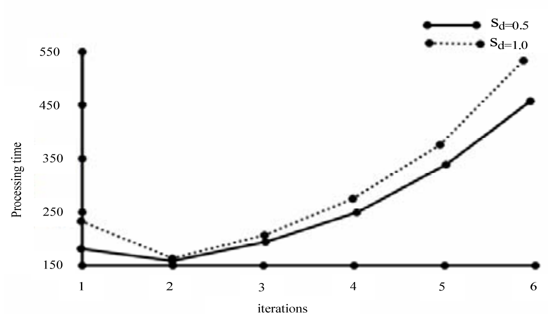

Figure 7. The Relationship between Iterations and Processing Time $(V_{\max} = 0.2r)$

5. Conclusion

Based on the MCL positioning technology of mobile wireless sensor network, this paper puts forward an iterative Monte Carlo localization algorithm. Use extension MCL simulator to process simulation experiments, and analyze algorithm performance from beacon node density, node maximum velocity, number of iterations, etc. The simulation results show that compared with the MCB and MCL, IMCL algorithm is better in performance. In particular, when beacon node density is lower, the superiority of IMCL algorithm can be better reflected. In addition, by adjusting the number of iterations can keep the relationship between the positioning accuracy and the processing time required balance.

References

[1] Jennifer Yick, Biswanath Mukherjee, Dipak Ghosa. Wireless sensor network research. *Computer network.* 2008; 52(12): 2292-2330.

[2] Kuang Xinghong, He Shaohui. Wireless sensor networks distributed node localization using mobile signal. *China university of post and telecommunications*. 2007; 14(4): 7-12.

[3] Pubudu N Pathirana, Nirupama Bulusu, Andrey V Savkin. Used node localization of mobile robot delay tolerance in the sensor network. *IEEE Portable computer*. 2005; 4(3): 285-296.

[4] Kuo-Feng Su, Chia-HoOu. Hewijin Christine Jiau. Mobile anchor point in the wireless sensor network localization. *IEEE Trading vehicle technology*. 2005; 54(3): 1187-1197.

[5] Luo ji, Zhang Qian. Mobile sensor network localization based on relative distance. *IEEE Application of telecommunications assembly*. 2007; 1076-1080.

[6] Dong Qifen, Yu Li, Chen Yourong, Hong Zhen. Iterative Monte Carlo localization algorithm research in the mobile wireless sensor network. *Chinese Journal of Sensors and Actuators.* 2010; (12): 1803-1809.

[7] Huang Meigen, Chang Xinfeng. A research of mobile wireless sensor network node localization algorithm based on Monte Carlo method. *Chinese Journal of Sensors and Actuators*. 2010; (4): 562-566.

[8] Xiao Tiantian, Yu Min. The research of mobile sensor network node localization algorithm based on Monte Carlo method. Jiangxi: Jiangxi Normal University. 2011.