Mobile Networks and Applications (2021) 26:649–656
https://doi.org/10.1007/s11036-019-01411-7

MAPS: Indoor Localization Algorithm Based on Multiple AP Selection

Pengyu Huang[1] · Haojie Zhao[1] · Wei Liu[1] · Dingde Jiang[2]

Published online: 1 February 2020

Abstract

In recent years, indoor fingerprint-based localization algorithm has been widely used by applications on smart phone. In these localization algorithms, it's very popular to use WiFi signal characteristics to represent the location fingerprint. With the fast popularization of WiFi, the WiFi access points (APs) could be seen everywhere. However, as the number of APs increases, the dimension of the fingerprint and the complexity of fingerprint-based localization algorithm subsequently increase. Responding to the above challenges, this paper proposes a novel indoor localization algorithm MAPS (indoor localization algorithm based on multiple access point selection). MAPS could effectively reduce the complexity of localization computation, and improve the performance of localization through AP selection method. With the first round AP selection, MAPS can obtain a stable subset of AP, thus reducing the dimension of fingerprint, and obtaining better discrimination. And with the second round of AP selection, AP subset could be further condensed to construct a decision tree in each location cluster. This step can further improve the localization performance. The experimental results shown, as compared with classical indoor localization algorithm, MAPS has better positioning accuracy, and could achieve the accuracy of over 90% within 2m location error.

Keywords Location fingerprint · Multiple AP selection · K-means · Location cluster · Decision tree

1 Introduction

In recent years, location-based services (LBS) are becoming more and more popular, and people have higher demands for localization and navigation. The GPS has high accuracy in outdoor environment. But in indoor environment, GPS signal can't pass through the wall. So it's difficult for GPS to provide location services in the indoor environment. Recently, WiFi is all around us, and we can connect to it even everywhere, such as supermarkets, campuses or airports. In addition, our smart phones can easily connect to WiFi, and get the RSSI (Received Signal Strength Indication) of the signal of WiFi. Therefore, lots of indoor localization algorithms based on WiFi has been proposed, and most of them are based on fingerprint.

The working process of indoor fingerprint-based localization algorithm could be divided into two phases: offline phase and online phase. In the offline phase, the localization area is divided into several square grids, and the center of the each grid is used as the reference location of this grid. Then, collecting WiFi information at these reference locations, such as RSSI or CSI (Channel State Information), and use them as the fingerprint of this location to generate the location fingerprint database. In the online phase, fingerprint-based localization algorithm matches real-time localization data with all reference location fingerprints in the fingerprint database. Choosing the reference location which has the highest similarity as the target location.

✉ Pengyu Huang
pyhuang@mail.xidian.edu.cn

Haojie Zhao
hjzhao@stu.xidian.edu.cn

Wei Liu
liuweixd@mail.xidian.edu.cn

Dingde Jiang
jiangdd99@sina.com

[1] School of Telecommunications Engineering, State Key Labs of ISN, Xidian University, 710071, Xi'an, Shaanxi, People's Republic of China

[2] School of Astronautics and Aeronautic, University of Electronic Science and Technology of China, 611731, Chengdu, People's Republic of China

For now, WiFi is ubiquitous. So when collecting WiFi data in the offline phase, at many reference locations, we can detect a large number of WiFi APs. If we use RSSI information of all detected APs to form a vector as the fingerprint of the reference location. This means the fingerprint vector of each reference location contains a huge dimension, which also results in an increase of the fingerprint database size. Furthermore, the study in [1] found that when the number of APs is too large, the increase of AP does not mean the progress of the positioning accuracy. Therefore, an intuitive idea is to select a suitable subset of APs to represent the fingerprint. Base of this selection mechanism, the dimension of fingerprint vector and the size of the fingerprint database can be reduced without affecting the positioning accuracy.

In this paper, we propose an indoor localization algorithm based on multiple AP selection. The main contributions of this paper are as follows:

(1) We design a novel multiple AP selection mechanism to effectively select stable and suitable APs. This method could effectively reduces the computing burden and storage requirements of the devices, and improve the positioning accuracy.
(2) The algorithm has strong adaptability to the environment and strong robustness. It can complete the AP selection process in any environment and build up a concise fingerprint database, so as to obtain high precise location at low computation cost.
(3) We implement the algorithm on a regular laptop and test it in a large-scale practical environment. In our experimental environment, there are more than 200 APs could be detected at some reference locations, and so it's more complex than the usual experimental environment.

2 Related works

There are numerous AP selection algorithms proposed in recent years. These algorithms could be roughly divided into two categories, Highest Signal algorithm [2, 3] and Information Gain-based AP selection algorithm [4]. In [2, 3]. In the highest signal algorithm, the authors used the APs RSSI to represent the importance of AP. The higher the signal strength, the more important the AP is. This type of AP selection algorithm is very easy, but AP's RSSI changes frequently in WiFi environment, and it is very sensitive in the indoor environment. So signal strength is not suitable to represent the importance of the AP. In order to improve this problem, the information gain-based AP selection algorithm is proposed. In [4], the author proposed an intelligent AP selection algorithm InfoGain (Information Gain-based AP selection). InfoGain algorithm uses position information entropy and conditional entropy to indicate the localization capabilities of different APs. In [5], AP selection was based on the principle of minimizing redundancy, using the correlation of APs to define redundancy. The correlation is got by computing two APs divergence measure. Paper [6] proposed a real-time AP selection algorithm. Like [5], it was also focused on how to minimize redundancy between APs. Paper [6] proposed two algorithms to get the Ideal AP subset. In [7], the AP selection algorithm combines information gain with mutual information entropy, and uses mutual information entropy to express the similarity of APs. If the mutual information entropy of two APs is big, this paper just chooses the one with higher information gain. Paper [8] proposed a RBF-based localization algorithm, in which the covariance matrix of RSSI is used to select AP. This paper combines RSSI covariance matrix with weight matrix, and uses scaling parameters represents the importance of AP. Then, rank APs in terms of their scaling parameters, and pick out the APs with the highest scaling parameters to form an AP set. In recent years, a novel method of localization based on adaptive cluster splitting (ACS) and AP reselection is proposed [9]. In this algorithm, AP reselection is simply use the mature information gain algorithm. In order to solve the sample capacity of fingerprints and RSSI variation property, a cost-efficient indoor localization approach is proposed to optimize the sample capacity as well as AP number and locations [10]. In the offline phase of this algorithm, the APs are optimally selected based on the concept of information gain criterion.

In addition to the above two categories, some new categories have been proposed. For example, a heuristic AP selection algorithm based on the error analysis is proposed to efficiently select a subset of APs in localization [11].

The rest of the paper is organized as follows: Section 3 describes MAPS in detail. Section 4 is about the experimental results, and the analysis of the results. Section 5 is a conclusion of MAPS.

3 Algorithm description

As mentioned in the previous section, the statistical distribution of RSSI is always been required in the information gain-based AP selection algorithm [4–10]. This is because they select AP or make up the decision model based on statistical distribution of RSSI of AP. However, in the actual experimental environment, we can find that some APs only could be detected for a few time in the process of collecting AP data. This means that there is very little information available to these APs. Furthermore, due to the lack of information, using the information of these APs to build fingerprint database will affect the performance of the localization algorithm.

To further illustrate this problem, APs data were collected at a reference location in the actual experimental environment. During the collection of AP data, we use the network card of a laptop to scan the AP around us. An AP data collection period consists of 50 scan cycle, and one scan cycle is 6 seconds. If the signal strength of an AP exceeds a threshold during a scan cycle, the MAC address of this AP will be recorded, and this MAC address (AP) will also be recorded as appearing once. The result of the data collection is plotted as Fig. 1. Figure 1 is a histogram to show the number of times that different APs have been detected over the collection period at a reference location in our experimental environment. In Fig. 1, the horizontal axis is the interval range of times that the AP was detected during the collection period. And the vertical axis indicates the number of APs whose number of occurrences is within a certain range. For example, in Fig. 1, the first column shows that 51 APs that were detected one to five times during the collection period. As shown in Fig. 1, we conducted a total of 50 scans within the collection period. Figure 1 shows that more than 100 APs can be detected in the whole sampling period, but only 6 APs were detected more than 45 times. So, the number of APs that can be detected in total the collection period is only a small proportion.

So, from Fig. 1, we could find that nearly half of the APs were collected less than five times, and more than 70% of the APs were collected less than twenty-five times during the sampling period. If an AP appears too few times during the sampling period, we will get few data about this AP, and using these so little data can't correctly estimate the probability distribution of this AP's RSSI. However, these algorithms [4–10] all depend on some form of the probability distribution of AP's RSSI, so these algorithms are not always available in real-world applications.

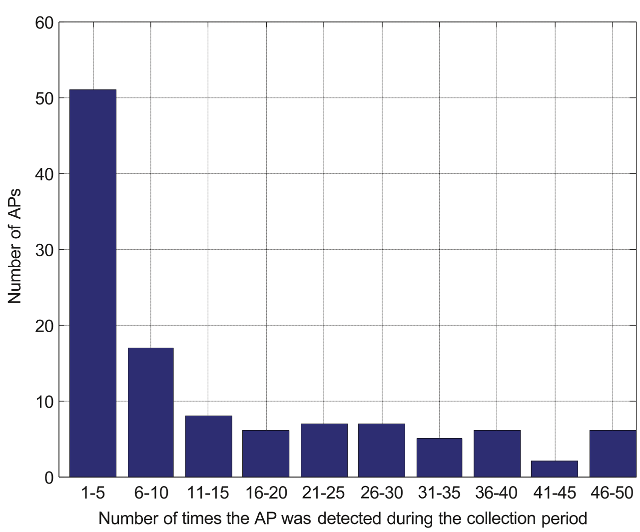

Fig. 1 A histogram of all APs detected at a reference location

Based on the above analysis, in this paper, we propose a novel indoor localization algorithm MAPS. This indoor localization algorithm is based on multiple AP selection method. MAPS can effectively solve the above problems. And MAPS could get a reliable APs subset by multiple AP selection method, which makes the signal to be more stable, and to obtain higher positioning accuracy. At the same time, MAPS can effectively reduce the computation complexity of localization algorithm, and improve the positioning performance.

The framework of MAPS still uses the traditional indoor localization algorithm. The MAPS is also divided into two phases: offline phase and online phase. The most important change of MAPS is to add two AP selection steps in the offline phase. In both steps, MAPS filters AP sets at different levels. Our aim is to study the effect of our new AP selection method on the performance of traditional localization algorithm.

In traditional offline phase, the location environment is first divided into square grids, and the center of the each grid is used as the reference location of this grid. Then, collecting RSSI information of APs in these reference locations. Using these APs' RSSI information as the fingerprint for each reference location, and make up the location fingerprint database. In MAPS, we add the first AP selection step here. In this step, we delete the APs information with less frequency from the RSSI information at each reference location from the fingerprint database.

Then, MAPS also uses k-means algorithm to cluster reference locations, and builds a decision tree for every location cluster. After clustering reference locations, MAPS re-selects APs subset for every cluster, obtain a better secondary APs subset for every cluster. This can further improve positioning performance.

In general, MAPS can be also divided into two phases: offline phase and online phase. In the offline phase, the MAPS contains five steps, collecting AP data, AP selection, location clustering, AP re-selection and building decision tree.

3.1 Collecting AP data

Collecting AP data is the basis of MAPS. This step includes the following process: First, the localization environment is divided into square grids, and the center of the grid is used as a reference location. Then, collecting AP data for a period of time at each reference location.

3.1.1 Selecting pre-selected AP Set

In this sub-step, we aim to delete APs that appear infrequently from the set of detected APs during the collection period of AP data, and form a stable AP subset. The details of the process are as follows:

1) Obtain the primary pre-selected AP subset.

   Using $\overline{AP_i}$ represents AP set detected at reference location i,

   $\overline{AP_i} = \left\{AP_i^1, AP_i^2, AP_i^3, ..., AP_i^f\right\}$ AP_i^j denotes AP j detected at reference location i, and f is APs number detected at reference location i. Counting the number of times n_i^j that AP_i^j is detected at reference location i during the data collecting period, so we can get APs set of detection times $\overline{n_i}$, $\overline{n_i} = \left\{n_i^1, n_i^2, n_i^3, ..., n_i^f\right\}$. Such as AP_i^j is detected 30 times at reference location i during the data collecting period, so n_i^j is equal to 30.

   Then, we set a threshold $th1$. If n_i^j less than $th1$, we delete AP_i^j from $\overline{AP_i}$. Finally, we merge all set $\overline{AP_i}$, and delete redundant AP in it. At last, we use the rest non-repeat AP to form primary pre-selected AP subset $PRAP$. $PRAP = \left\{AP_1, AP_2, AP_3, ..., AP_g\right\}$, and g is the number of APs that satisfy the above-mentioned condition in localization environment.

2) Obtain the final pre-selected AP subset.

   Calculate the sum times of AP_l in $PRAP$ at all reference locations, marked as N_l. For example, suppose AP_l only appears in reference location i and j, the number of times that AP_l was detected in the above 2 reference locations is num_i and num_j respectively. So N_l is equal to the sum of num_i and num_j. Therefore, we can get the set $\overline{N}$, where value corresponds to the sum times that each AP in set $PRAP$ is detected, $\overline{N} = \left\{N_1, N_2, N_3, ..., N_g\right\}$.

Then, we set another threshold $th2$. If N_l is less than $th2$, we delete AP_l from $PRAP$. Those remaining AP make up the final pre-selected AP subset $FPAP$, $FPAP = \{AP_1, AP_2, AP_3, ..., AP_h\}$, and h is the size of $FPAP$.

3.1.2 Obtaining final target AP set

In [4], the author proposed InfoGain algorithm to select AP set. InfoGain algorithm uses APs information gain to represent the APs discriminative power to location. The more discriminative power, the more important AP is. We will continue to use this method in this paper. Information gain is calculated as follow:

$$InfoGain\left(AP_i\right) = H\left(G\right) - H\left(G|AP_i\right). \tag{1}$$

where $InfoGain\left(AP_i\right)$ is AP_i's discriminative power. $H\left(G\right)$ is the entropy of all reference locations without know AP_i's RSSI information.

$$H\left(G\right) = -\sum_{i=1}^{p} P\left(G_i\right) \log P\left(G_i\right). \tag{2}$$

where G_i is reference location, and p is the number of reference location. $H\left(G|AP_i\right)$ is the conditional entropy of location given APs RSSI information.

$$H\left(G|AP_i\right) = -\sum_{v}\sum_{j=1}^{p} P\left(G_j, AP_i = \mathrm{v}\right) \log P\left(G_j|AP_i = \mathrm{v}\right). \tag{3}$$

where v is RSSI value of AP_i. $P\left(G_j, AP_i = \mathrm{v}\right)$, and $P\left(G_j|AP_i = \mathrm{v}\right)$ can be obtained based on collected AP data.

In this step, we calculate information gain of every AP in set $FPAP$. Choosing the first k APs with the largest information gain to form fingerprint AP set $FingerAP$, $FingerAP = \{AP_1, AP_2, AP_3, ..., AP_k\}$.

Then, based on the $FingerAP$, we reconstruct the fingerprint of every reference location from the original AP data, and obtain the fingerprint database at last. Set F_j is the location fingerprint of reference location j,

$F_j = \left\{RSSI_1^j, RSSI_2^j, RSSI_3^j, ..., RSSI_k^j\right\}$, where $RSSI_i^j$ is RSSI of AP_j in reference location i.

From formulas (2) and (3), we can know that we need to know the RSSIs probability distribution of every AP, when calculating $H\left(G|AP_i\right)$. However, when collecting APs data, we find some APs only are detected occasionally. So there is a few data of these APs, as Fig. 1. Therefore, for these APs, we can't get reliable APs RSSI probability distribution. Paper [4] does not consider this problem, when the author proposes the InfoGain algorithm. In this case, only using information gain to select AP subset that not always work well. In our algorithm, AP selection contains two step. The first step deletes these APs, which are detected occasionally during AP data collecting period, and obtains pre-selected AP set. The second step gets final target AP set based on the InfoGain algorithm. These APs are stable in the pre-selected AP set, and through this AP set we can get good probability distribution through RSSI data that was collected. Finally, our MAPS algorithm eliminates these unstable APs, and makes better use of information gain algorithm.

3.2 Location clustering

In the online phase, the fingerprint-based indoor localization algorithm uses the fingerprint information that obtained currently to match all fingerprints in the fingerprint database. It is clear that the localization time is proportional to the size of the fingerprint database, that is, to the number of the reference locations in the localization environment. Therefore, in order to save localization time, we can divide all reference locations into some clusters. In the process of localization, current fingerprint simply needs to determine which cluster it is in, and match with the fingerprints within this cluster. So clustering can effectively reduce the localization time. In MAPS, we use a classical cluster algorithm, k-means algorithm [12], to cluster reference locations based on location fingerprint. Suppose there are M reference locations in location environment, and L clusters, and C_j is the center of cluster j, $C_j = \left\{c_1^j, c_2^j, c_3^j, ..., c_k^j\right\}$, where k is cardinality of $FingerAP$.

The process of location clustering is as follows:

1) Randomly selecting L locations as cluster centers $C_j = F_j$, so $C_j = F_j = \left\{RSSI_1^j, RSSI_2^j, RSSI_3^j, ..., RSSI_k^j\right\}$.
2) According to Euclidean distance between reference locations and all cluster centers, the reference locations are divided into L clusters. Such as, when determining which cluster the reference location i belongs to, calculate Euclidean distance between reference location i to every cluster centers. Then, reference location i will be divided to the cluster L with the shortest Euclidean distance to reference location i. Euclidean distance is as defined as follows:

$$Dis\left(F_i, C_j\right) = \sum_{h=1}^{k}\left(c_h^j - RSSI_h^i\right)^2. \tag{4}$$

where $Dis\left(F_i, C_j\right)$ is the distance between reference location i and cluster j.
3) Update the center location of each cluster. When all reference locations are divided into clusters, the average location of fingerprints in each cluster is be used as the new cluster center. Suppose cluster j contains w reference locations, so new cluster center can be calculated by the following formula:

$$C_{jnew} = \frac{1}{w} \times \left\{\sum_{h=1}^{w} RSSI_1^h, \sum_{h=1}^{w} RSSI_2^h, ..., \sum_{h=1}^{w} RSSI_k^h\right\}. \tag{5}$$

4) Determines whether to stop clustering iteration and updates each cluster center. Calculate the Euclidean distance between the new cluster center, and the old cluster center. If the Euclidean distance less than a certain threshold, stop iteration, and let C_j equal to C_{jnew}, else let C_j equal to C_{jnew}, and back to step 2), continue the iteration.

3.3 AP reselection and make up decision tree

After determination of clustering, our algorithm re-selects AP set that makes up each cluster's fingerprint again. Before clustering the reference locations, we get the $FingerAP$ set based on our AP selection method in Section 3.1.2. MAPS aims to obtain a better AP set, which has high discrimination. So $FingerAP$ is a reliable AP set for the indoor localization environment. However, our localization algorithm divides all reference locations into some clusters. Each cluster has its characteristics, and there are some differences between clusters. Therefore, if the fingerprint is formed based on the same AP set, it is not conductive to represent different features between clusters. In order to solve this problem, the MAPS add a new step of re-selecting AP set for each cluster. Therefore, through the re-select step, each cluster has a different AP set. The re-select process of AP set within a cluster is described below:

Suppose the set $Cluster_i$ is composed of reference locations that divided into cluster i. So $Cluster_i = \left\{Loc_1^i, Loc_2^i, Loc_3^i, ..., Loc_K^i\right\}$, where Loc_j^i represents the jth reference location, and K is the number of reference locations in cluster i. According to AP data collected at $Cluster_i$, we calculate each information gain of AP in $FPAP$, and obtain the re-selection AP set of cluster i, $ClusterAP_i = \left\{AP_1^i, AP_2^i, AP_3^i, ..., AP_c^i\right\}$, where c is the number of AP selected from $FPAP$.

After AP re-selection, we build the decision tree for each cluster according to C4.5 algorithm [13] based on $ClusterAP$.

3.4 Online phase

When the decision tree for each cluster is established, this means the end of the offline phase, and MAPS can be used to determine the user's location. The process of localization is described below:

Suppose the user's current localization data is T,

1) Determine which cluster the T belongs to. Based on set $ClusterAP$, we construct location fingerprint $Tfinger = \left\{\widetilde{AP}_1, \widetilde{AP}_2, \widetilde{AP}_3, ..., \widetilde{AP}_c\right\}$ of T. Then, we calculate Euclidean distance from $Tfinger$ to each cluster, and choose the cluster with the smallest Euclidean distance as Ts target cluster. The location of T can be got from the target cluster.
2) Localization based on the decision tree. In the previous sub-step, we obtain the target cluster that T belongs to. Then we use the traditional C4.5 decision tree algorithm to determine T's final location.

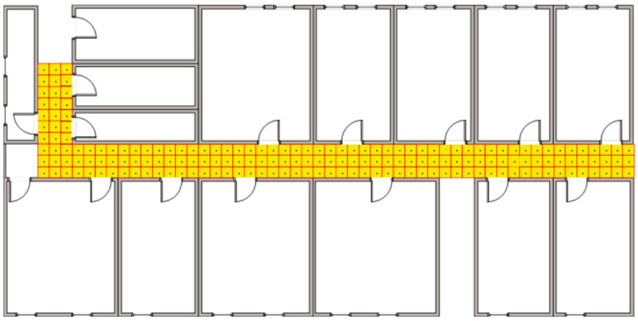

Fig. 2 Experimental testbed

4 Experimental evaluation

In this section, we describe our experimental environment and assess the performance of the MAPS.

We conducted the experiment in the fourth-floor corridor of the main building of Xidian University, as Fig. 2 showed. This experimental environment includes 177 reference locations, and every reference location is the center of a 0.8m*0.8m grid. In the phase of collection AP data, we run 50 scan cycles at every reference location. Each cycle lasts 6 seconds. And we can detect more than one hundred APs at every reference location in our experimental environment, and even more than two hundred APs at some reference locations.

In the above experimental environment, that the number of location clusters is 5, Fig. 3 shows that the positioning ability of MAPS varies with the number of APs under different location errors. From Fig. 3, we can see that MAPS has good positioning ability. Specifically, when the location error ranges from 0.8m to 1.6m, the positioning ability increases by 18% on average. When the location error ranges from 1.6m to 2.0m, the positioning ability increases by 12% on average. The MAPS achieves the best performance when the location error within 2m. The 90% of the test results can reach the positioning accuracy below 2m when the number of APs is 14.

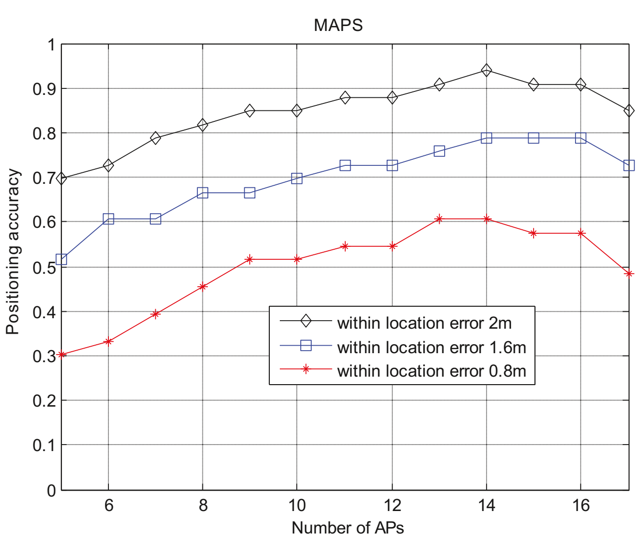

Fig. 3 The performance of MAPS within different location error under different number of APs

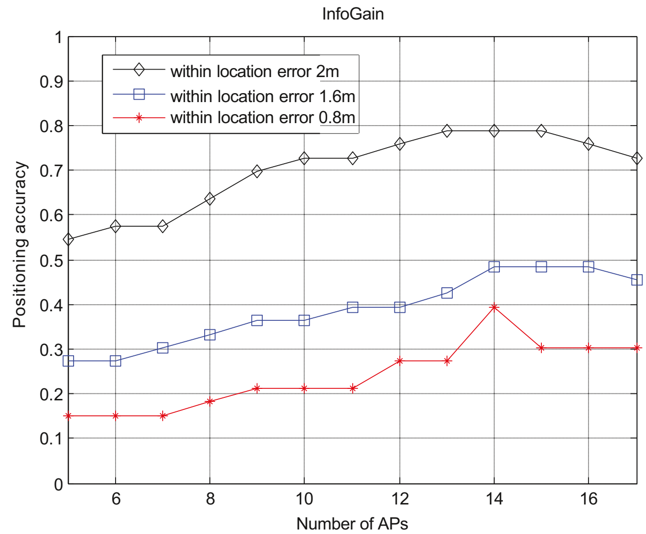

Fig. 4 The performance of InfoGain algorithm within different location error under different number of APs

Keeping the experimental conditions unchanged, we also did several experiments to observe the performance of InfoGain algorithm within different location error. As Fig. 4 shows, the performance of InfoGain algorithm is relatively poor in our experimental environment. The main reason for this result is that in [4], their testbed is relatively simple, and only 25 APs can be detected in the environment. Duo to the number of AP is so small, there is less interference between APs in their experimental environment. Therefore, information gain algorithm could achieve good performance in their experiment.

But now WiFi is everywhere, for example, we can detect hundreds of WiFi APs in my campus or in a mall. Therefore, there is serious interference between WiFi APs, and at the same time the deployment location of APs is also extremely complex. Especially, at the data collection phase of APs, some APs are merely detected. As a result, we can't correctly get these probability distributions of these APs base on collected AP data. But the performance of InfoGain algorithm heavily depends on the probability distribution of APs. So only using information gain algorithm to choose APs can't obtain satisfactory results in our experimental environment. Our MAPS algorithm can delete unstable APs, and use reliable APs represent location fingerprint. Therefore, our MAPS algorithm achieves better performance.

In addition, from Figs. 3 and 4, there is an location accuracy decreasing when the number of AP excesses 16. This is because MAPS is one of the AP selection algorithms, and the core idea of MAPS is to select a more stable and suitable AP subset from AP set. In the process of various AP selection algorithms, the first step is to set an index to indicate the advantage and disadvantage of AP. The second is to determine a threshold of this index. During implementation, the corresponding value of this index will be calculated for every AP. The APs whose corresponding value is higher than the threshold will be selected as a valid subset for localization. The rest of the APs are discarded. As the number of AP gradually increases, the network environment becomes more and more complex, and the selection ability of various AP selection algorithm will be reduced to some extent. This is mainly because, as the number of AP increases, it will be difficult to distinguish between good and bad APs except some that are obviously better. This trend could also be seen from the simulation results of MAPS and InfoGain in this paper. The most direct solution to this problem is to design a better AP selection algorithm, and this paper is also based on this view. It could be seen from the simulation results that our MAPS has stronger selection ability and better positioning performance than InfoGain.

Under the condition that the number of location clusters is 5, we compare MAPS with InfoGain algorithm [4] within 2m location error. Similar results can be obtained when the location error is other values.

As shown in Fig. 5, MAPS performs better than InfoGain. Because essentially, MAPS adds multiple AP selection method to the framework of InfoGain. By comparison, we can clearly see the impact of the addition of multiple AP selection method on the performance of the localization algorithm. Obviously, multiple AP selection method can significantly improve the positioning accuracy of the algorithm. Further, we can observe from Fig. 5 that MAPS also has good stability. As the number of APs gradually increases, compared with the InfoGain, MAPS has maintained a performance advantage of more than 11%. This means that the multiple AP selection method has good stability.

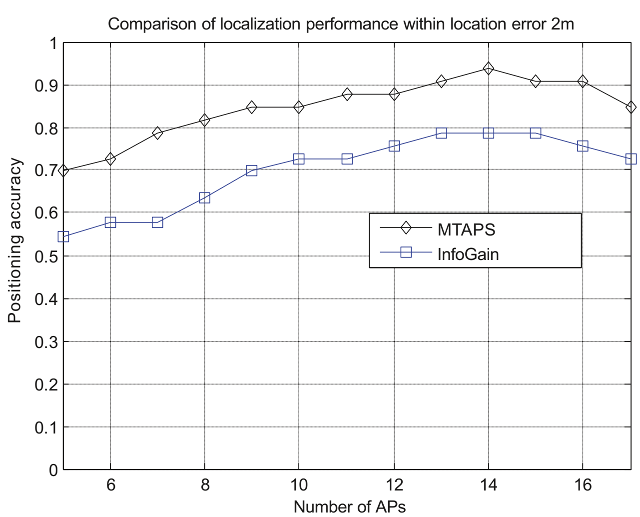

Fig. 5 Comparison of localization performance for MAPS and InfoGain under different number of APs

5 Conclusion

In this paper, we propose an indoor localization algorithm MAPS. This algorithm can get a reliable AP subset to represent location fingerprint. In addition, MAPS re-select AP subset for each location cluster to get a special AP subset. Based on the above steps, the results of the experiments show that our algorithm has the better performance. On the other side, we also analyze the causes of the bad performance for information gain algorithm. Our algorithm can effectively solve the defect of information gain algorithm, and obtains satisfactory positioning performance.

Based on MAPS, we think that there are still many problems that should be further explored. (1) In the complex WiFi environment, there is still space for further development of multiple AP selection algorithm. On the basis of this algorithm, the efficiency of this algorithm could be improved by adding some refined and adaptive strategies. (2) Because of the independence of the algorithm in this paper, we could consider inserting the multiply AP selection algorithm in other indoor localization algorithms to improve the performance of these algorithms.

Acknowledgements The financial support of the program of Key Industry Innovation Chain of Shaanxi Province, China (2017ZDCXL-GY-04-02), of the program of Xi'an Science and Technology Plan (201805029YD7CG13(5)), Shaanxi, China, of Key R&D Program C The Industry Project of Shaanxi (Grant No. 2018GY-017), of Key R&D Program C The Industry Project of Shaanxi (Grant No. 2017GY-191) and of Education Department of Shaanxi Province Natural Science Foundation, China (15JK1742) are gratefully acknowledged.

References

1. Kaemarungsi K, Krishnamurthy P (2004) Modeling of indoor positioning systems based on location fingerprinting, Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies. IEEE, vol 2, pp 1012–1022
2. Youssef MA, Agrawala A, Shankar AU (2003) WLAN location determination via clustering and probability distributions. Proceedings of the 1st IEEE international conference on pervasive computing and communications, pp 143–150
3. Du L, Bai Y, Chen L (2007) Access point selection strategy for large-scale wireless local area networks. IEEE Wireless Communications and Networking Conference 2007:2161–2166

4. Chen Y, Yang Q, Yin J et al (2006) Power-efficient access-point selection for indoor location estimation. IEEE Trans Knowl Data Eng 18(7):877–888
5. Zhao Q et al (2010) An effective preprocessing scheme for WLAN-based fingerprint positioning systems, 2010 12th IEEE international conference on communication technology (ICCT)
6. Kushki A, Plataniotis KN, Venetsanopoulos AN (2007) Kernel-based positioning in wireless local area networks. IEEE Trans Mobile Comput 6(6):689–705
7. Zou G et al (2014) An indoor positioning algorithm using joint information entropy based on WLAN fingerprint, 2014 International conference on computing, communication and networking technologies (ICCCNT)
8. Laoudias C, Panayiotou CG, Kemppi P (2010) On the RBF-based positioning using WLAN signal strength fingerprints, 2010 7th workshop on positioning navigation and communication (WPNC)
9. Liang D, Zhang Z, Peng M (2015) Access point reselection and adaptive cluster splitting-based indoor localization in wireless local area networks. IEEE Internet of Things Journal 2(6):573–585
10. Zhou M, Wei Y, Tian Z et al (2017) Achieving cost-efficient indoor fingerprint localization on WLAN platform: A hypothetical test approach. IEEE Access 5:15865–15874
11. Jia B, Huang B, Gao H et al (2019) Selecting critical WiFi APs for indoor localization based on a theoretical error analysis. IEEE Access 7:36312–36321
12. Duda RO, Hart PE, Stork DG (2012) Pattern classification [M]. Wiley, New York
13. Quinlan JR (2014) C4. 5: programs for machine learning [M]. Elsevier