Advances in Computer, Signals and Systems (2021) 5: 48-54
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2021.050107
ISSN 2371-8838

Improved Algorithm for RFID Indoor Positioning

Jiangyu Yan

School of Electrical and Electronic Engineering, North China Electric Power University, Beijing, 102206, China

185196587@qq.com

Keywords: indoor positioning, RFID, VIRE algorithm, positioning algorithm

Abstract: For indoor RFID localization, the LANDMARC algorithms and the VIRE algorithms are two typical RFID indoor localization algorithms. This paper analyzes the characteristics of the two algorithms and proposes an improved RFID indoor positioning algorithm. The proposed algorithm uses numerical interpolation to calculate the RSSI of virtual tags, uses double-weight fusion to obtain the weight of candidate tags, and combines the centroid method to locate the unknown tag. Simulation results show that the proposed algorithm improves the positioning accuracy.

1. Introduction

Indoor positioning technology is a technique for target positioning in indoor environments. In outdoor environments, satellite navigation and positioning systems (such as GPS and Beidou) and cellular positioning can accurately and quickly determine the target position. However, in the indoor environment, due to electromagnetic shielding, interference, and other reasons, the positioning accuracy of these two methods is not enough to meet the needs of indoor user positioning. RFID (Radio Frequency Identification) is a technology that uses radio waves for the intelligent identification of non-contact objects. RFID has the advantages of non-contact, non-line of sight, high sensitivity, and low cost, becoming the preferred technology for indoor positioning solutions [1-3].

RFID positioning is divided into ranging-based and range-free positioning methods. In the ranging-based positioning method, each tag measures the signal propagation time or phase between the tag and the reader, and calculates the distance or angle between the tag and the reader. At last, the reader determines the tag's location by using triangulation and other methods. The main ranging-based methods [4] are the Time of Arrival (TOA), the Time Difference of Arrival (TDOA), and the Angle of Arrival (AOA), and Phase Difference of Arrival (PDOA). However, due to the accurate measurement of distance and angle requires complex and expensive circuits, the ranging-based positioning methods have a high cost. So it is unsuitable for the precise RFID indoors positioning.

On the other hand, the range-free methods do not need the measuring circuit and locate tag according to the received signal strength indication (RSSI). Instead, these methods obtain the RSSI of the unknown tag and compare it with the RSSI of the reference tag with the known accurate position. Then, the accurate position of the unknown tag is obtained by executing the positioning algorithm. Therefore, the range-free method has a strong anti-interference ability and is suitable for the accurate positioning of indoor objects. It is also the main research method of RFID positioning

[5].

This paper first introduces the classical LANDMARC algorithm and VIRE algorithm, analyzes their advantages and disadvantages, proposes an improved RFID indoor positioning algorithm. Then, the improved algorithm is verified by simulation.

2. RFID indoor positioning algorithm

In the RFID positioning algorithm, RSSI is the most common method to determine the object's location. Therefore, the RSSI-based localization algorithm must first measure the RSSI value of the reference tag and the unknown tag [6]. We use a logarithmic distance model to calculate the propagation loss of wireless signals in free space, as follows:

$$PL(d) = PL(d_0) - 10nlg\left(\frac{d}{d_0}\right) + X_\delta \tag{1}$$

where in, d represents the linear distance from the reader to the tag. d_0 represents a reference distance; n is the path loss factor related to the environment, a general value 2.2. Is X_δ is Gaussian noise with a variance of δ and mean of 0. Its value range is 4 ~ 10.

The reader uses the formula (1) to deduce the distance d based on the RSSI received by the unknown tag, combining geometric knowledge to locate the tag. However, the indoor propagation of wireless signals will receive various interference, resulting in RSSI and distance relationship not strictly adhering to the formula (1), resulting in a large measurement error. It is not feasible to simply locate a tag according to RSSI.

2.1 LANDMARC algorithm

The LANDMARC algorithm introduces reference tags because there are many interference factors in the indoor environment, leading to decreased positioning accuracy [7].The algorithm sets several reference tags at the same distance in the positioning region. Then, the reference tags are arranged in a matrix, and their position is accurately measured. The LANDMARC algorithm calculates the difference in RSSI between the unknown tag and the reference tags, selects the closest reference tags. Finally, the algorithm obtains the position information of the unknown tag by weighted calculation.

LANDMARC algorithm positioning steps:

1) There are M reference tags and N readers. The ith reference tag receives RSSI from each reader, constituting its RSSI vector $R_i = (R_{i1}, R_{i2}, \cdots, R_{iN})$. RSSI vector to be measured by the unknown tag is $R_T = (R_{T1}, R_{T2}, \cdots, R_{TN})$.
2) Calculate the difference between the unknown tag and each reference tag based on the RSSI vector. According to formula (1), RSSI is proportional to distance. So, if the difference between RSSI is small, the two tags are far apart. Otherwise, it indicates that the two tags are close. The formula is as follow:

$$\mathrm{E_j} = \sqrt{\sum_{i=1}^{N} \left(R_{Ti} - R_{ji}\right)^2} \qquad j = (1,2,3,\cdots M) \tag{2}$$

3) Set a threshold E_φ and select the k reference tags for $E_j < E_\varphi \; 1 \le j \le k$.
4) Calculate the weight. w_i is the weight of the ith candidate reference tag. The smaller the difference between tags, the closer the distance, so the weight should be higher. Calculation

formula of weight w_i is as follow:

$$w_i = \frac{\frac{1}{E_i^2}}{\sum_{j=1}^{k}\left(\frac{1}{E_j^2}\right)} \tag{3}$$

5) Location estimation. The coordinates (x, y) of the unknown tag are obtained by weighted summation of k candidate reference tags:

$$(x, y) = \sum_{i=1}^{k} w_i(x_i, y_i) \tag{4}$$

2.2 VIRE algorithm

The LANDMARC algorithm is a classical range-free algorithm. It uses the shortest Euclidean distance to select k reference tags near the unknown tag and then determine the unknown tag's coordinates by weighted summation. However, due to the multipath effect of wireless signals in the indoor environment, there is a large error between RSSI and distance measurement. Therefore, it is necessary to set more and more dense reference tags to improve accuracy. This results in higher cost.

The VIRE algorithm [8] improves the LANDMARC algorithm by inserting virtual tags evenly between reference tags. The relationship between virtual and reference tags is shown in Figure 1.

There are four reference tags (R1, R2, R3, and R4) in the positioning region. The virtual tags are even, equidistant, and meshed between the four tags. These four reference tags' locations and RSSI are known. Virtual tags reduce deployment costs and improve positioning accuracy. We set the $n \times n$ virtual tag array within the rectangular area with these four reference tags as vertices.

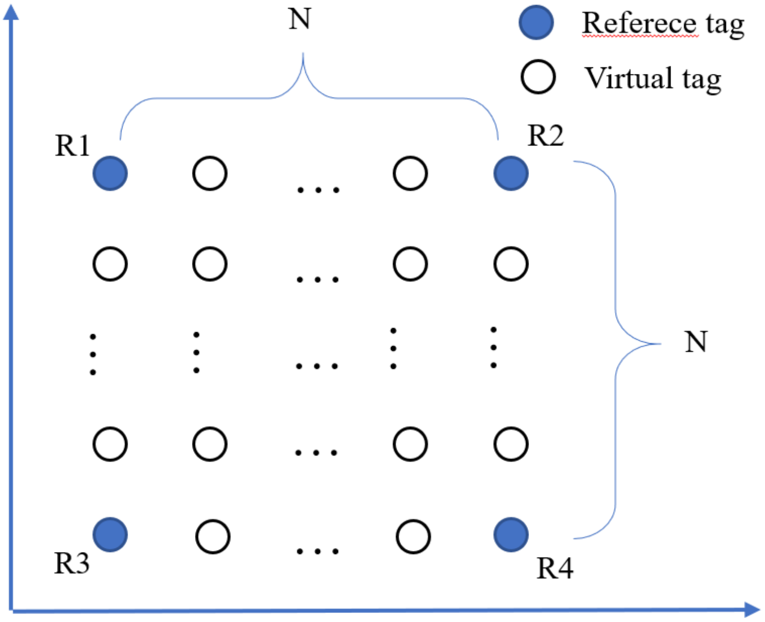

Figure 1: Virtual tags and reference tags

The VIRE algorithm is characterized by introducing virtual tags, which can improve the positioning accuracy without adding additional reference tags. It is necessary to determine the RSSI value of the virtual label through the usual method to construct a complete virtual positioning map.

3. Improved VIRE positioning algorithm

Aiming at the deficiencies of the LANDMARC algorithm, this paper improves the VIRE algorithm. These improvements include the RSSI generation of virtual tags, the weight of candidate tags, and the location estimation method of the unknown tag estimation method.

3.1 The RSSI calculation of virtual tag

Suppose $n \times n$ virtual tags are set in a rectangular region determined by four reference tags. RSSI values of virtual tags are calculated by numerical interpolation.

RSSI calculation formula of virtual label abscissa:

$$R_k(T_{p,b}) = R_k(T_{a,b}) + (p - a) \times \frac{R_k(T_{a+n,b}) - R_k(T_{a,b})}{n} \tag{5}$$

RSSI calculation formula of virtual label ordinate:

$$R_k(T_{p,b}) = R_k(T_{a,b}) + (q - b) \times \frac{R_k(T_{a+n,b}) - R_k(T_{a,b})}{n} \tag{6}$$

$$a = \left[\frac{i}{n}\right], b = \left[\frac{j}{n}\right], \qquad 0 \le p = i\%n \le n - 1, \qquad 0 \le q = i\%n \le n - 1$$

wherein, $R_k(T_{i,j})$ represents the virtual RSSI of the kth reader at the coordinate position (i, j). The VIRE algorithm uses the adjacent map method to determine candidate tags. It takes each reference tag (including virtual tags) as the center to divide the positioning region into several subregions and compares the RSSI of unknown tags and each subregion. If the comparison results are less than the predefined threshold, the subregion center is placed in the candidate tag list. The algorithm traverses all virtual tags and selects a total of k candidate tags. This method can select the large probability region, filters out the small probability region, and improves the positioning accuracy.

3.2 Improvement to weight

In order to obtain more accurate results, this paper introduces weights $\mathrm{W}_{1\mathrm{i}}$ and $\mathrm{W}_{2\mathrm{i}}$. $\mathrm{W}_{1\mathrm{i}}$ represents the weight of each candidate tag. $\mathrm{W}_{2\mathrm{i}}$ refers to the density of virtual tags. Their calculation formulas are as follows:

$$\mathrm{W}_{1\mathrm{i}} = \sum_{h=1}^{N} \frac{|R_h(T_i) - \theta_h(R)|}{h \times R_h(T_i)} \tag{7}$$

$$\mathrm{W}_{2\mathrm{i}} = \frac{p_i}{\sum_{i=1}^{k} p_i} = \frac{n_{ci}}{\sum_{i=1}^{k} n_{ci}} \tag{8}$$

$R_h(T_i)$ represents the RSSI of virtual tag i obtained from reader h. $\theta_h(R)$ is the RSSI received from reader h for the unknown tag. p_i denotes the ratio of continuous region number to total region number. n_{ci} represents the total number of continuous regions obtained by the intersection of adjacent maps. Finally, the formula of the weight of k candidate tags is as follow:

$$W_i = \alpha W_{1i} + (1 - \alpha) W_{2i} \tag{9}$$

In this paper, the value of α is 0.5.

3.3 Improvements to location estimation

The k candidate tags can be connected to form a polygon. Its centroid is the estimated location of unknown tag. The calculation formula for unknown tag coordinates is as follows:

$$(x^*, y^*) = \left(\frac{\sum_{j=1}^{k} w_j x_j}{\sum_{j=1}^{k} w_j}, \frac{\sum_{j=1}^{k} w_j y_j}{\sum_{j=1}^{k} w_j} \right) \tag{10}$$

wherein, (x^*, y^*) is the estimated coordinates of the unknown tag; (x_j, y_j) refers to the coordinates of the candidate tag j.

4. Simulation and analysis

In order to verify the performance of the proposed algorithm, we used MATLAB R2018A for simulation. The simulation region is a square of $10 \times 10m^2$, as shown in Figure 2. We set up four readers in the four corners, and their coordinates are (0, 0), (0, 10), (10, 0) and (10, 10). Set the reference tag from the coordinates (1, 1) in the simulation region, with an interval of 1.5m. A 5×5 virtual teg matrix is set between every four reference tags. There are 100 unknown tags the simulation.

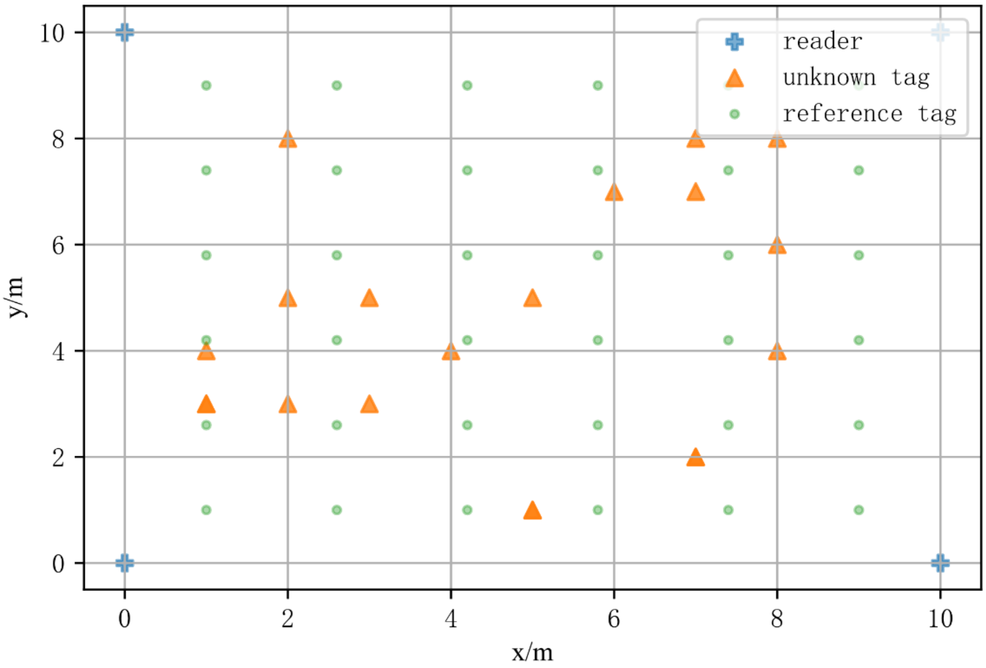

Figure 2: Simulation region

The performance of the evaluation algorithm uses an average square error:

$$\text{err} = \sqrt{(\hat{x} - x_0)^2 + (\hat{y} - y_0)^2} \tag{11}$$

In the formula, $(\hat{x}, \hat{y})$ is the estimated coordinates of the unknown tag according to our algorithm. (x_0, y_0) is the real coordinate of the unknown tag.

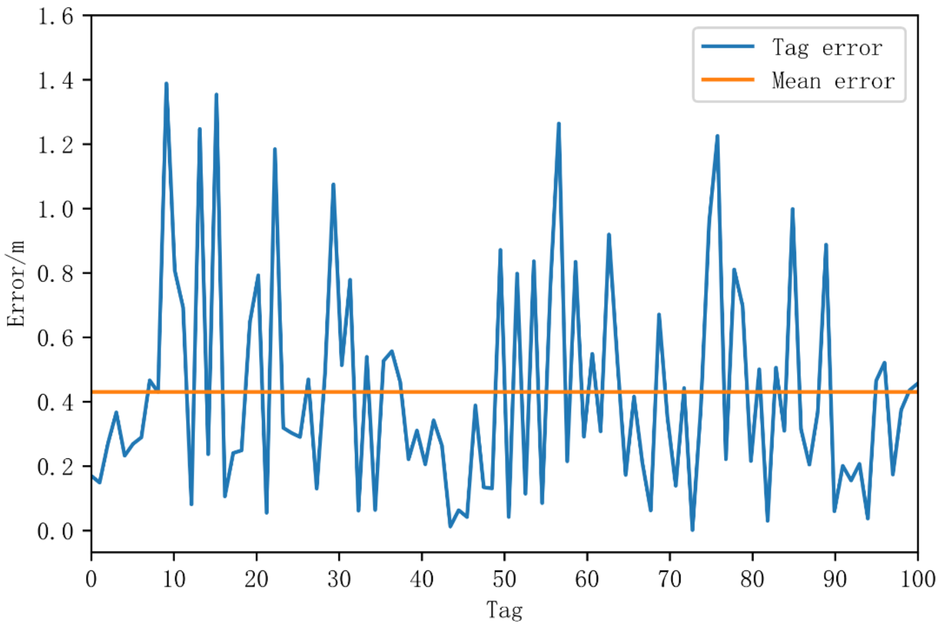

Figure 3: Positioning error

Through simulation, the average positioning error of the algorithm proposed in this paper is 0.431m; the maximum error is 1.389m, and the median error of 0.331m. The error distribution is shown in Figure 3. It can be seen from Figure 3 that in 100 simulations, 63% of the unknown tags had a positioning error of less than 0.5m, and 93% of the unknown tags had a positioning error of less than 1m.

5. Conclusion

Aiming at the accuracy problems faced by RFID in indoor positioning, this paper analyzes the characteristics of the LANDMARC algorithm and the VIRE algorithm. We propose an improved RFID-based indoor positioning algorithm. The algorithm uses numerical interpolation to calculate the RSSI of virtual tags, set the double weight for the candidate tag, and combines the centroid method to determine the coordinates of the unknown tags. The simulation results show that the improved positioning algorithm has higher positioning accuracy.

References

[1] Xu H, Wu M, Li P, et al. An RFID indoor positioning algorithm based on support vector regression[J]. Sensors, 2018, 18(5): 1504.

[2] López Y Á, de Cos Gómez M E, Andrés F L H. A received signal strength RFID-based indoor location system[J]. Sensors and Actuators A: Physical, 2017, 255: 118-133.

[3] Ma Y, Wang B, Pei S, et al. An indoor localization method based on AOA and PDOA using virtual stations in multipath and NLOS environments for passive UHF RFID[J]. IEEE Access, 2018, 6: 31772-31782.

[4] Zhu J, Xu H. Review of RFID-based indoor positioning technology[C]//International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. Springer, Cham, 2018: 632-641.

[5] Liu H,Darabi H,Banerjee P,et al. Survey of Wireless Indoor Positioning Techniques and Systems[J]. IEEE Transactions on Systems Man&Cybernetics Part C, 2007,37(6):1067-1080.

[6] Choi J S, Lee H, Lee S C, et al. Reducing localization ambiguity of immobile passive UHF RFID tagged physical objects[C]//2011 8th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). IEEE, 2011: 35-40..

[7] Ni L M, Liu Y, Lau Y C, et al. LANDMARC: Indoor location sensing using active RFID[C]//Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003). IEEE, 2003: 407-415.

[8] Wang H, Chen Y. Adaptive VIRE indoor positioning algorithm based on Kriging interpolation[J]. International Journal of RF Technologies, 2019, 10(3-4): 115-130.