Study on Wireless Sensor Networks Localization Algorithm Based on DV-Hop

Die Hu,Guangcong Liu,He Li

School of Computers，Guangdong University of Technology，Guangzhou 510006, China

Keywords: Wireless Sensor Networks,DV-Hop Localization Algorithm,Collinearity

Abstract. In the DV-Hop localization algorithm of wireless sensor networks, positioning error of the unknown node is larger if it obtain the average hop distance from the nearest anchor nodes.In order to improve the positioning accuracy of the node, this paper improves the DV-Hop algorithm. The improvement of thought is two correction of the average hop distance.The simulation results show that compared with the traditional DV-Hop algorithm, the improved algorithm can improve the positioning accuracy of the nodes, and the positioning error is reduced about 10%.

1. Introduction

Wireless sensor networks need work under the complex environment or the area that humans can not reach in most cases , so that the position of nodes[1,2] is not controllable because of the aircraft throwing. But the research need to get the specific sources of information. So that localization problem of node in wireless sensor networks is one of the key problems which must be solved.

In this paper, we improve the DV-Hop algorithm .Two amendments is used on the average hop distance to reduce the error because of the problem that obtaining average hop distance from nearest anchor nodes will lead to larger errors.

2. DV-Hop Localization Algorithm

Each anchor node calculates its own average hop distance by sing the position informationof the other anchor nodes and hop count information:

$$aHops_i = \frac{\sum\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum h_i} \quad (1)$$

Among them, $aHops_i$ is the average hop distance of anchor node I. (x_i, y_i) and (x_j, y_j) I are the coordinates of anchor nodes. h_i is the hops between the anchor node i and j（i≠j）.

The unknown node only records the average distance information received by the nearest anchor node, and computes the distance between the unknown node and the anchor node.

$$d_i = h_i \times aHops_i \quad (2)$$

Among them, h_i is the hops between the unknown node and anchor node i.

After geting the distance between unknown node and anchor node, maximum likelihood estimation method is used to calculate the coordinates of unknown nodes.

3. Improvement of DV-Hop Algorithm

Improvement thought is: Weighted average hop distance for two times . Firstly, judging whether has the anchor node that the hop is 1 around the unknown node when calculating the average hop distance. If there is, sum of the average hop distance that anchor node’s hop is 1 and mean value of them. If there is no anchor node in the one hop range, average hop is still in the traditional DV-Hop get nearest anchor node distance, geting average hop distance from the nearest anchor node by the tradition.This is recorded as the initial average hop distance. Secondly, the number of unknown nodes around the anchor nodes directly affects the positioning accuracy. We weighted average hop distance by the number of unknown nodes in one hop . This is recorded as the second average hop distance. Finally, the final average hop distance of the unknown nodes is obtained by taking the average value of two results.

The first step: the initial average hop distance.

The initial average hop distance of the unknown node is $a_1 Hops_i$, and it represents the mean of average hop distance of all anchor nodes in one hop range. The calculation formula is as follows:

$$a_1 Hops_i = \begin{cases} \dfrac{\sum_{j=1}^{n} aHops_j}{n} & (j > 0) \\ aHops_k & (j = 0) \end{cases} \quad (4)$$

Where n represents the total number of anchor node that unknown node is one. $aHops_i$ represents the average hop distance of anchor node i. $aHops_k$ is the average hop distance of unknown nodes obtained from nearest anchor node K . $a_1 Hops_i$ is the initial average hop distance.

The second step: second average hop distance.

The average hop distance of the anchor nodes is weighted according to the number of unknown nodes in one hop. Therefore, the weighted average hop distance of anchor nodes is as follows.

$$SUM = \sum_{i}^{m} n_i \quad (5)$$

$$\sigma_i = n_i / SUM \quad (6)$$

$$a_2 Hops_i = \sum_{k=1}^{m} (\sigma_k \times aHops_k) \quad (7)$$

Where n_i is the number of unknown nodes in one hop of anchor node i.The sum of unknown nodes in one hop of all anchor nodes. M represents the total number of anchor nodes. σ_i is the weight of anchor node i. $aHops_k$ is the average hop distance of anchor node K . $a_2 Hops_i$ is average hop distance of unknown node i which is weighted by anchor node.

The third step: the two correction.The average hop distance of the node i is revised as:

$$a'Hops_i = \frac{(a_1 Hops_i + a_2 Hops_i)}{2}$$

Average hop distance of nearest anchor node is used in traditional DV-Hop algorithm, and large error is caused. The two amendment is carried out on average hop distance in the improved algorithm.

4. Simulation and Analysis

In order to verify the effectiveness of the improved algorithm, simulation is carried out on the platform of MATLAB, the 100 sensor nodes are randomly deployed in a square area of 100m*100m. Simulation results of DV-Hop algorithm, literature [3] and algorithm in this paper are compared and analyzed.

The measure standard is average positioning error:

$$\lambda = \frac{\sum \sqrt{(x_{rei} - x_{tei})^2 + (y_{rei} - y_{tei})^2}}{NR} \quad (8)$$

Among them, λ is average positioning error, (x_{rei}, y_{rei}) and (x_{tei}, y_{tei}) are respectively the estimated coordinates and real coordinates of unknown node, N is the total number of unknown nodes. R is the communication radius.

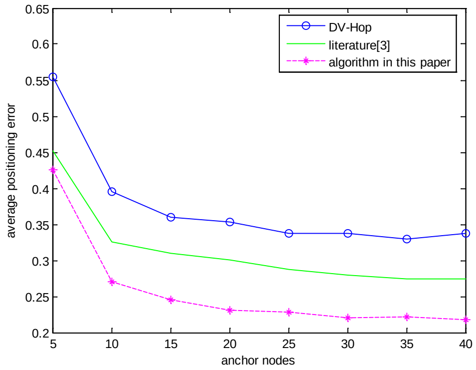

Fig.1 Schematic of node average positioning error under the change of anchor node

Figure 1 is the average positioning error of the unknown nodes is changed with the change of number of anchor nodes under the condition that the total node is 100 and communication radius is 25. The curve can be seen that the algorithm in this paper is obviously superior to the traditional DV-Hop algorithm and algorithm of literature [3]. The overall error is about 7% lower than that of the literature [3], which is about 15%. lower than the traditional DV-Hop.

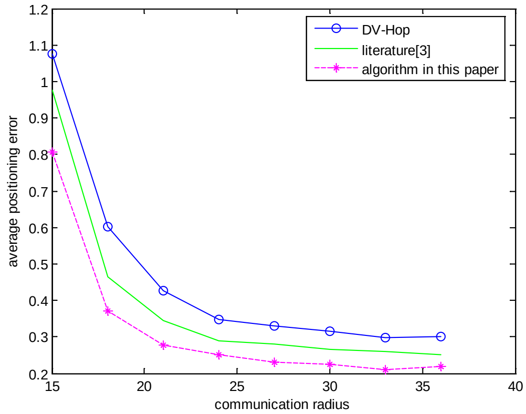

Fig.2 Schematic of node average positioning error under change of node radius

Figure 2 is the average positioning error of the unknown nodes is changed with the change of the communication radius under the condition that the total node is 100 and the number of anchor nodes is 25. It can be seen from the figure, the general trend is that with the increasing of communication radius, the average positioning error is reduced. The positioning error sharply declined when the communication radius increased from 15m to18m, and the rate of decline slowed down he communication radius increased from 18m to 24m. It is to be gentle after 24m. Overall, positioning accuracy of algorithm in this paper is improved by about 8% than that algorithm in the literature [3]. Compared with the DV-Hop algorithm, positioning accuracy improves about 15%.

5. Concluding Remarks

DV-Hop algorithm is researched in this paper, the unknown node use average hop distance of the nearest anchor node as the average hop distance of the whole network will lead to larger positioning error. In this paper, we propose a scheme for the two correction of the average hop distance. Through simulation and verification, the improved algorithm proposed in this paper improves the positioning accuracy and reduces about 15% of the positioning error.

References

[1] Li H,Liu G C,Hu D.Cluster-head Selection Mech-anism Based on Energy Ratio in Wireless Senso-r Networks[J]. Journal of Guangdong Universit-y of Technology, 2014,31(3):83-87.

[2] Peng Y, Wu D L, Guillaume Andrieux, et al. Re-search on energy-efficiency in wireless senso-r

networks based on minimum energy coding[J].Journal of Guangdong University of Technology,2014, 31(1) : 86-89.

[3] Wu N, Liu F A, Wang S X, et al. Algorithm for location nodes in WSN based on modifying hops and hopping distances[J]. Microelectronics &C-omputer, 2015,32(1):91-95.