This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

A Fine-grained Localization Algorithm in Wireless Sensor Networks

Xueyong Xu*, Liusheng Huang†, Xiang Li‡, Jichun Wang§ and Hongli Xu¶
Department of Computer Science and Technology, University of Science and Technology of China, Hefei, China
Suzhou Institute for Advanced Study, University of Science and Technology of China, Suzhou, China
{xxyyeah*, christli‡, jichunw§}@mail.ustc.edu.cn {lshuang†, xuhongli¶}@ustc.edu.cn

***Abstract*—In recent years, many localization algorithms have been proposed for wireless sensor networks, in which the hop-count based localization schemes are attractive due to the advantage of low cost. However, these approaches usually utilize discrete integers to calculate the hop-counts between nodes. Such coarse-grained hop-counts make no distinction among one-hop nodes. More seriously, as the hop-counts between nodes increase, the cumulative deviation of hop-counts would become unacceptable. In order to solve this problem, we propose the concept of fine-grained hop-count. It is a kind of float-type hop-count, which refines the coarse-grained one close to the actual distance between nodes. Based on this idea, we propose a fine-grained localization algorithm (AFLA). In AFLA, we first refine the hop-count information to obtain fine-grained hop-counts, then use the Apollonius circle method to achieve initial position estimations, and finally further improve the localization precision through confidence spring model (CSM). We conduct the comprehensive simulations to demonstrate that AFLA can achieve 30% higher average accuracy than the existing hop-count based algorithm in most scenarios and converge much faster than the traditional mass-spring model based scheme. Furthermore, AFLA is robust to achieve an approximate 35% accuracy even in noisy environment with a DOI of 0.4.**

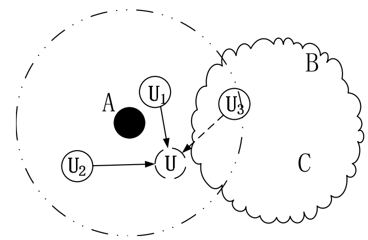
***Index Terms*—Fine-grained, Localization, Apollonius circle, Confidence spring model, Wireless sensor networks**

I. Introduction

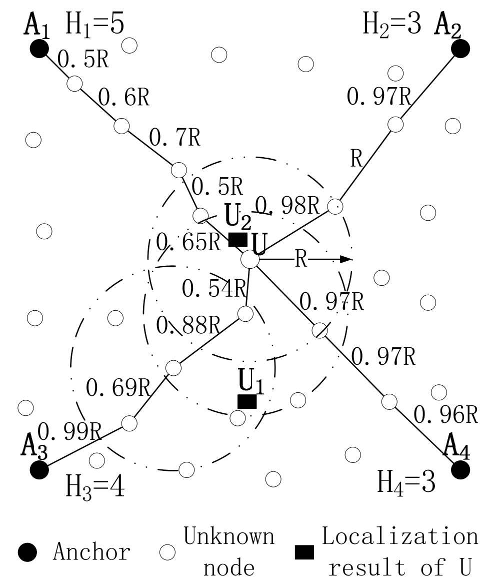
Geographic information is one of the most important attributes of data stream in wireless sensor networks. For most applications in wireless sensor networks, sensors' data is useless without physical coordinates. In the past few years, a large number of localization algorithms have been proposed to provide per-node position information for sensor networks.

Hop-count information is often used in range-free localization algorithms. Through collecting hop-count information, the nodes can be located. DV-Hop [1] and Amorphous Positioning [2] are classic algorithms falling in this category. In DV-Hop, position unknown nodes find out their hop-counts to at least three anchors. Anchors estimate the average distance per hop by exchanging messages with other anchors and send it to the unknown nodes. Then unknown nodes use multilateration to calculate their own coordinates. Although these hop-count based algorithms can achieve not bad accuracy, they still have many drawbacks. We take DV-Hop as the example to discuss this issue in detail.

Figure 1(a) shows a general case of one-hop range. In this situation, the hop-count values of $A-U_1$, $A-U_2$ and $A-U_3$ are 1. If the hop-counts between U_i (U_1, U_2 and U_3) and other anchors (B, C) are the same, these unknown nodes will be located in the same position. Figure 1(b) is a nonuniform case of multi-hop range. In this scenario, the hop-counts between U and anchors (A_1, A_2, A_3 and A_4) are (5, 3, 4, 3) in DV-Hop algorithm. The hop-counts are discrete integers. In fact, the corresponding hop-counts of actual distance are (2.95, 2.95, 3.1, 2.9). Obviously, the hop-counts of $U-A_1$ and $U-A_4$ in DV-Hop have cumulative deviation comparing to the actual float-type hop-counts. As a result, DV-Hop locates the unknown node U in point U_1 with the estimation error of 140%R, while it can achieve a precision of 9%R (U_2) by utilizing the actual distance information. Therefore, low estimation error can be achieved if the hop-counts are close to their actual distance.



(a) One-Hop Range



(b) Multi-Hop Range: H_i is the hop-counts between U and anchor A_i and the float numbers of the lines are actual distance.

Fig. 1. Drawbacks of Hop-count based Localization Algorithms

From figure 1, we can find out problems in hop-count based approach. The hop-counts based on discrete integers can not reflect the exact distance. In the case of one-hop range, the nodes within one hop are not differentiated, while cumulative deviation occurs in the situation of multi-hop range. In a word, coarse-grained hop-counts result in the rising of estimation error. Hence, we propose a fine-grained hop counting method, in which float numbers are used to measure the hop-counts

978-1-4244-2948-6/09/$25.00 ©2009 IEEE

Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.

between nodes in a more precise manner. The hop-counts are refined towards their actual distance.

Besides refining the coarse-grained hop-counts, we can make use of the information of relative positions between nodes to further reduce the estimation error. As we know, spring model [3] is often used for improving localization accuracy by utilizing neighboring relationship. But the relationship is usually acquired through distance measurement which needs extra device. In this paper, based on the fine-grained hop-counts, we propose an efficient confidence spring model without any measurement. In this model, we adjust the moving step dynamically according to the correlated neighboring information rather than the fixed one in [3]. As a result, our approach can both reduce the estimation error greatly and converge quickly. Moreover, our approach is robust in noisy environment. We will demonstrate these in the simulation section.

In this paper, we focus on minimizing the estimation error and convergence rate. The main contributions of this paper are as follows.

1) We propose the concept of *fine-grained hop-count*, a kind of float-type hop-count, which is close to actual distance and contributes in reducing the estimation error.

2) We propose a confidence spring model with the evaluating mechanism of confidence level. The model converges quickly through adjusting the moving step dynamically.

3) We propose a robust range-free localization algorithm AFLA with high localization accuracy and fast convergence rate.

The rest of the paper is organized as follows: Section II summarizes related work on localization in wireless sensor networks. Section III describes the detailed process of AFLA. Section IV gives the performance evaluation. Finally, we conclude our work in Section V.

II. Related work

The localization algorithms are generally divided into two categories: range-based and range-free. In range-based algorithms, nodes estimate their distance to anchors through varieties of techniques, such as Received Signal Strength Indicator (*RSSI*) [4], Time of Arrival of Signal (TOA) [5], Time Difference of Arrival of Signal (TDOA) [6] and Angle of Arrival (AOA) [7]. Although the range-based approaches can achieve high accuracy, depending on specific hardware limits their applicability for large scale sensor networks. Range-free algorithms do not need special devices. The cost-efficient advantage makes them more attractive than range-based methods. In the past few years, many classic range-free schemes (DV-Hop, Amorphous Positioning and APIT [8] et al.) have been proposed for applications in wireless sensor networks.

Recently, researchers have proposed novel algorithms which do not belong to these two categories strictly. Ziguo Zhong et al. [9][10] proposed MSP scheme which estimates each sensor node's two-dimensional location by processing multiple one-dimensional node sequences. The sequences are through detecting localization event order. Hence, MSP does not require extra costly hardware but a generator of localization event, which makes it different from both range-based and range-free schemes. Kiran et al. [11] proposed a sequence-based localization algorithm which constructs the location sequence table that maps location sequences to the corresponding regions. The unknown node finds out its location as the center of corresponding region which is determined by its received sequence of strength of signals. Although *RSSI* values are used here, the mapping of distance and *RSSI* is not necessary to be concerned in this algorithm. So this scheme is different from *RSSI* based localization. In this paper, we are inspired partly by their work. We also simply collect *RSSI* information and do not take measurement to obtain the correspondence between the distance and *RSSI*. For instance, in TinyOS system [12], we are able to read the *RSSI* value of a message from Micaz node (by CrossBow Company [13]) directly. Hence, special device is not required when only acquiring the *RSSI* values.

In this paper, we also refer to the knowledge of Apollonius circle, which is often applied for localization [14][15]. In [15], localization is converted to a problem of obtaining the hop-count-ratios which map to Apollonius circles. In our work, we refine the information collected before utilizing Apollonius circle so as to achieve initial localization results with higher accuracy. Moreover, in [15], multiple power levels are used to improve the accuracy which brings great communication cost. But in our work, we use the information of relative positions between sensors which is also often considered in localization schemes (AFL [3], MDS [16] et al.). AFL first determines relative coordinates of nodes. Then it uses mass-spring model to converge to a coordinate assignment. However, AFL knows the distance between neighbors often through measurement. In this paper, we do not measure the distance between neighbor nodes. Instead, we solve this problem by initial localization and then utilize the CSM scheme to refine with high convergence rate.

III. Process of AFLA

AFLA consists of four phases—collecting information, refining information, locating initially and refining iteratively. In the following four subsections, we will describe the four steps in detail.

A. Collecting information

In AFLA, we need to gather information as: the shortest paths to anchors, the neighboring set and the *RSSI* values corresponding to nodes in neighboring set. In order to acquire all these information, AFLA conducts a similar broadcasting course to DV-Hop, but with some differences in message content and extracting method. In the next two paragraphs, we will discuss how to collect information in details.

1) The shortest path: Each unknown node collects the shortest length to every anchor and its parent node in this path. Take Anchor A and unknown node X for example. When X receives the broadcast message from node Y, it compares its current shortest length to A with that in newly received message. If the latter is smaller, X sets Y as its parent node and its length added by 1 to that in the received message. Then X broadcasts new message. When no broadcast exists in network, the unknown nodes transmit the message collected back to server where all information is merged to shortest paths.

2) Neighboring set and the corresponding RSSI values: If unknown node X receives message from node B during the process of information collection, then we put node B into X's neighboring set. Meanwhile, X reads the *RSSI* value of the message directly just as what is mentioned in the section of related work.

Obviously, the course of collecting information takes less communication cost compared to that of the classic hop-count

Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.

based algorithms. Although extra step of transmitting message back to server is needed, there is no need to flood average one-hop distance estimations of anchors in our scheme. Definitely, flooding requires more communication expense.

B. Refining information

Once finishing gathering information, we refine it in this phase. Suppose there is a shortest path from unknown node X to anchor A: X-G-F-B-A. From the previous analysis of figure 1, we know it is necessary to reevaluate each interval of the path to get new fine-grained hop-counts close to the actual distance. We assume that the neighboring set of A contains B, C, and D. Moreover, the corresponding *RSSI* values of the messages that node B, C and D received from node A in last phase are respectively $RSSI_B, RSSI_C, RSSI_D$ and the radio propagation model is path loss model [17]. If the mean received power at distance d is $\overline{P_r(d)}$, the mean received power at reference distance d_0 is $P_r(d_0)$ and β is the path loss exponent, we can calculate $\overline{P_r(d)}$ as follow

$$\frac{P_r(d_0)}{\overline{P_r(d)}} = \left(\frac{d}{d_0}\right)^{\beta} \tag{1}$$

Given the values of $P_r(d_0)$ and d_0, we can calculate the distance between nodes directly according to $\overline{P_r(d)}$. But as a matter of fact, there must be at least two anchors within each other's range in order to obtain the two reference values. Obviously, the approach has limitations on the topology of networks. It brings problems especially in sparse networks without two anchors in range. However, we do not use the *RSSI* information to implement localization directly. Hence, we do not need to know the two reference values because it is easy to get the value of parameter β in different environments according to table I.

TABLE I
SOME TYPICAL VALUES OF PATH LOSS EXPONENT β

Environment		β
Outdoor	Free Space	2
	Shadowed urban area	2.7 to 5
In building	Line-of-sight	1.6 to 1.8
	Obstructed	4 to 6

Knowing the corresponding value of β (2 in simulations), we can express the power of messages that B, C and D received in the form d_0 and $P_r(d_0)$ according to formula (1). Before refining information, we need another parameter, the smallest one-hop range *RSSI* value in the networks. The distance where we collect this value is considered to be communication range r (the farthest node within range, the closest to range border). The unit of *RSSI* is dBm and the relation between dBm and mW is: $mW = 10^{dBm/10}$. Then we can convert *RSSI* values to power and use formula (1). So we have

$$\left(\frac{1}{r}\right)^{\beta} : \left(\frac{1}{d_B}\right)^{\beta} : \left(\frac{1}{d_C}\right)^{\beta} : \left(\frac{1}{d_D}\right)^{\beta} \tag{2}$$
$$= \overline{RSSI_{MinmW}} : \overline{RSSI_{BmW}} : \overline{RSSI_{CmW}} : \overline{RSSI_{DmW}}$$

From formula (2), we can obtain the values of d_B, d_C and d_D. Therefore, the fine-grained hop-counts of three intervals A-B, A-C and A-D are respectively d_B/r, d_C/r and d_D/r. We also reevaluate the fine-grained hop-counts of A-C and A-D for later use in the phase of locating iteratively. Finally, we add the reevaluated hop-counts of intervals along the path to get the fine-grained hop-counts.

C. Locating initially

Utilizing the fine-grained information obtained in previous phase, we are capable to conduct initial localization. Supposed there are M anchors $B_1, B_2 \cdots B_M$ in networks. The coordinate of anchor B_i is (X_{b_i}, Y_{b_i}). The hop-counts between the unknown node U (coordinate (X_u, Y_u)) and these M anchors are respectively $H_1, H_2 \cdots H_{M-1}, H_M$. So we can calculate the coordinate of U through the formulas as

$$\begin{aligned}
\frac{\sqrt{(X_{b_1}-X_u)^2+(Y_{b_1}-Y_u)^2}}{\sqrt{(X_{b_2}-X_u)^2+(Y_{b_2}-Y_u)^2}} &= \frac{H_1}{H_2} \\
&\vdots \\
\frac{\sqrt{(X_{b_1}-X_u)^2+(Y_{b_1}-Y_u)^2}}{\sqrt{(X_{b_M}-X_u)^2+(Y_{b_M}-Y_u)^2}} &= \frac{H_1}{H_M} \\
&\vdots \\
\frac{\sqrt{(X_{b_{M-2}}-X_u)^2+(Y_{b_{M-2}}-Y_u)^2}}{\sqrt{(X_{b_M}-X_u)^2+(Y_{b_M}-Y_u)^2}} &= \frac{H_{M-2}}{H_M} \\
\frac{\sqrt{(X_{b_{M-1}}-X_u)^2+(Y_{b_{M-1}}-Y_u)^2}}{\sqrt{(X_{b_M}-X_u)^2+(Y_{b_M}-Y_u)^2}} &= \frac{H_{M-1}}{H_M}
\end{aligned} \tag{3}$$

Each line of formula (3) corresponds to an Apollonius circle. Hence, it is easy to solve the equation to obtain (X_u, Y_u) through properties of Apollonius circle. Compared to classic hop-count based algorithms (DV-Hop et al.), AFLA does not need to determine the average one-hop distance estimations in this step. In addition, the initial position estimations are only concerned with the information collected by unknown nodes respectively. Hence, the situation that the inaccuracy of anchors' average one-hop distance estimations affects the localization results of all the unknown nodes in networks does not exist in AFLA.

D. Refining iteratively

In the phase of locating initially, we obtain initial position estimations. Before refining them iteratively through CSM, we introduce briefly about the traditional mass-spring model (TMSM) in the next paragraph.

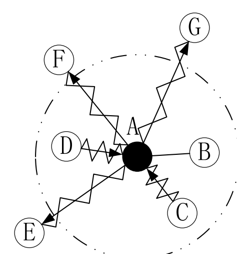


Fig. 2. TMSM

In figure 2, the nodes are displayed in points with coordinates obtained in the second phase. And the information collected shows that B, E, F and G all belong to the neighboring set of A, while C and D do not. Obviously, the position estimations in the second phase are not consistent with information collected. Nodes E, F, G should be in the range of A, while the results of localization show not. So pulling force exists between E, F, G and A. Similarly repelling force exists between C, D and A. In the course of one iteration, node A moves towards the direction of its resultant force. After all nodes in networks finish moving in the fashion similar to A, a new iteration continues until the power of the system is not

Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

larger than the threshold value. Before introducing CSM, we first define the following two terms.

Confidence Level ($CL_N^{[i]}$): The reliable level of node N's coordinate in the i^{th} iterative phase of spring model. Anchor always has the highest confidence level C_{MAX}. The confidence level of the unknown nodes are evaluated through their surrounding nodes. Their initial values are all 1.

History Information Reference Parameter ($HIRP_N^{[i]}$): A parameter refers to historical coordinate information of the unknown node N in the $(i-1)^{th}$ iteration of spring model.

Suppose the position estimation of unknown node X in last iteration (or the initial position of X) indicates: X is within the range of its m_1 neighbors (neighbor means other node around itself during the phase of collecting information), m_2 of which are anchors; X is beyond the range of its n_1 neighbors, n_2 of which are anchors. φ_j^{i-1} is the CL of the unknown node j in the $(i-1)^{th}$ iteration. μ equals to the CL of anchor. Let set Set_{mU} include the $(m_1 - m_2)$ unknown neighbors within the range of X's current position estimation and Set_{neighU} contain unknown nodes in X's neighboring set. Then its HIRP is given by

$$HIRP_X^{[i]} = \frac{\sum_{j \in Set_{mU}} \varphi_j^{i-1} + m_2 \cdot \mu}{\sum_{k \in Set_{neighU}} \varphi_k^{i-1} + m_2 \cdot \mu + n_2 \cdot \mu}, i \geq 1 \tag{4}$$

ψ is a positive integer constant less than C_{MAX}. λ is a positive constant less than 1. Max (100) is a value much larger than C_{MAX} in case there are nodes without any neighbor around it. Therefore, we have

$$CL_X^{[i]} = \begin{cases} \frac{1}{Max} & : \ HIRP_X^{[i]} = 0 \\ C_{MAX} - \frac{\psi}{(\chi^{[i]})^{\lambda}} & : \ HIRP_X^{[i]} \neq 0 \end{cases} \tag{5}$$

χ is given by

$$\chi^{[i]} = \begin{cases} \chi^{[i-1]} + 1 & : \ HIRP_X^{[i]} > 0.5 \\ \chi^{[i-1]} - 1 & : \ 0 < HIRP_X^{[i]} < 0.5 \\ 1 & : \ \chi^{[i]} < 1 \end{cases} \tag{6}$$

The initial value $\chi^{[0]}$ is 1. In this way, χ's value will become bigger and bigger if the value of $HIRP_X^{[i]}$ is larger than 0.5 in continuous several iteration rounds, and so is the value of $CL_X^{[i]}$. Figure 3 is a curve of formula (5) when $HIRP_X^{[i]} \neq 0$. The tendency of the curve confirms the reasonableness of formulas (5) and (6).

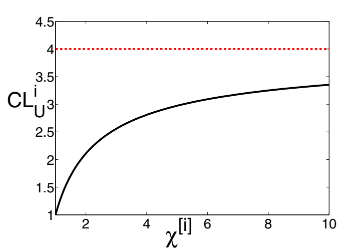


Fig. 3. $C_{MAX} = 4, \psi = 3, \lambda = 2/3, HIRP_X^{[i]} \neq 0$

By introducing the evaluating mechanism of confidence level into TMSM, we propose the confidence spring model (CSM) to refine the initial localization results. The main goal of CSM lies in such a fact that the more reliable the position estimations are, the less distance the nodes move, and vice versa. In other words, CSM aims to refine the position of reliable nodes in smaller range and that of unreliable nodes in larger range. In the following paragraphs, we will show the process of CSM.

Suppose the neighboring set of unknown node X is Set_{neigh} and the initial position estimations show the set of nodes around X is Set_{pos}. $AllSet_X = Set_{neigh} \bigcup Set_{pos}$. $\overrightarrow{V_{X,Y}}$ is the unit vector in the direction from node X to Y. $\hat{d_{X,Y}}$ represents the estimated distance between X and Y. It is calculated through the coordinates of X and Y's position estimations. $d_{X,Y}$ is the actual distance (If X and Y are neighbors, it is the value obtained in the fine-grained phase; else it is a value of communication range r) between X and Y. In CSM, we have the force applied to X by Y as

$$\overrightarrow{F_{X,Y}} = \overrightarrow{V_{X,Y}} \cdot \left(\hat{d_{X,Y}} - d_{X,Y}\right) \cdot CL_Y \tag{7}$$

The resultant force on node X is equal to

$$\overrightarrow{F_X} = \sum_{Y \in AllSet_X} \overrightarrow{V_{X,Y}} \cdot \left(\hat{d_{X,Y}} - d_{X,Y}\right) \cdot CL_Y \tag{8}$$

The energy of X is given by

$$\begin{aligned} E_X &= \Sigma_{Y \in AllSet_X} E_{X,Y} \\ &= \Sigma_{Y \in AllSet_X} (\hat{d_{X,Y}} - d_{X,Y})^2 \cdot CL_Y^2 \end{aligned} \tag{9}$$

Finally, we have the total energy of the system as

$$E = \sum_{X \in Network\ W} E_X \tag{10}$$

In the course of one CSM iteration, the energy of X is calculated according to these formulas at first. In the CSM, it moves towards the direction of $\overrightarrow{F_X}$ and its shifting distance is $E_X^{\sigma} \times UL$. σ is an parameter less than 1 (1/5 in experiments) to avoid the moving range of nodes beyond the networks border. UL is the uncertainty level of X, which is the reciprocal of its confidence level. $\overrightarrow{V_X}$ is the unit vector of resultant force on X. So the moving strategy is given by

$$Cor_X^i = Cor_X^{i-1} + E_X^{\sigma} \cdot UL \cdot \overrightarrow{V_X} \tag{11}$$

In formula (11), X's self confidence level also contributes to the distance of movement. It is a kind of intuitive and reasonable consideration. For instance, if X's resultant force is large and its confidence level is also high, its shifting distance should be reduced apropos. The reduction avoids bad influence from the situation that X has reliable position but owns large resultant force, which is caused by the nodes surrounding it with greatly biased coordinates. From formula (11), it is easy to get the essence of CSM movement strategy: the moving step should be decided dynamically by its own confidence level and the correlated neighboring information.

After all unknown nodes finish moving, one iteration is over. We repeat the iterative process until the energy of system is less than the threshold value or the iteration times are larger than the limitation.

IV. Performance Evaluation

In this section, we conduct detailed simulations under the OMNET++ 3.3 platform for DV-Hop, IAFLA (initial localization phase of AFLA), TMSM* (TMSM based on the results of IAFLA) and AFLA in various network configurations.

A. Simulation Environment and Radio Model

Sensor nodes are deployed in a 100×100 square region. Two kinds of anchor placement strategies are adopted in our experiments: border and random. In the first strategy, the

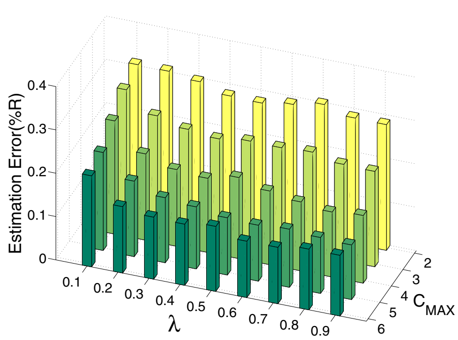
Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2009 proceedings.

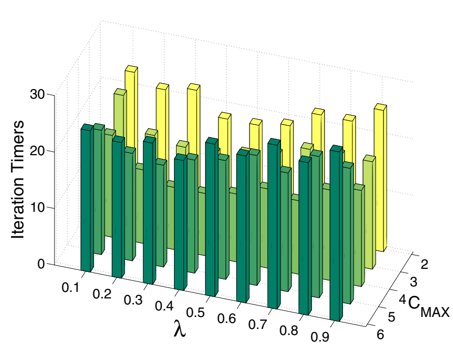
anchors are disposed around border, while randomly placed in the square area in the latter one. To conduct our evaluation as true to reality as possible, we use a general radio model RIM (Radio Irregularity Mode) [18] in IAFLA, TMSM* and AFLA. RIM uses DOI to present the irregularity of the radio pattern. We do not consider the DOI in DV-Hop for fairness because we collect more information in our scheme. In the following evaluation, we investigate the estimation error of DV-Hop, IAFLA, TMSM* and AFLA.

B. Evaluation

1) Different Parameters: Figure 4 shows the effort of different parameters on the estimation error and convergence rate of AFLA. The localization precision reduces as the values of C_{MAX} and λ grow up. But when C_{MAX} exceeds 4, the estimation error reduces slightly. The convergence rate achieves max in the central area. At the beginning, larger C_{MAX} and λ lead to the increasing of moving step. As a result, the nodes shift faster towards destination. However, if C_{MAX} and λ continue going up, the nodes might have great oscillation around their actual positions. Therefore, it instead takes more iteration times to converge. Consider the two factors synthetically, we choose $C_{MAX} = 4$ and $\lambda = 0.2$ for AFLA. Besides, we only consider the border placement strategy in this experiment. Because the random strategy itself is a variable factor with unstable effect on the results, which makes us hard to tell whether it is from random strategy or different parameters.



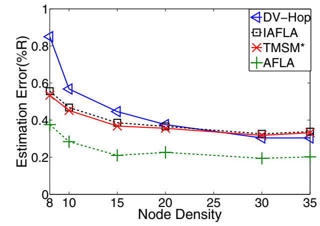
(a) Localization precision



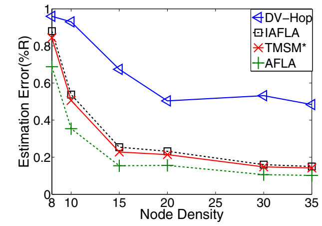
(b) Convergence rate

Fig. 4. Localization accuracy and convergence rate under varying parameters

2) Different Node Density: Figure 5 explores how the estimation error varies under different node density. When the node density increases, the unknown nodes can hear more nodes. Obviously, the hop-count information is more precise and it leads to the reduction of the estimation error in all these schemes. Furthermore, we have more relative position information for refinement in spring model based scheme. So the latter two schemes can achieve higher accuracy than the first two. In the following experiments, we can further demonstrate the good effect of utilizing relative position between nodes for localization.



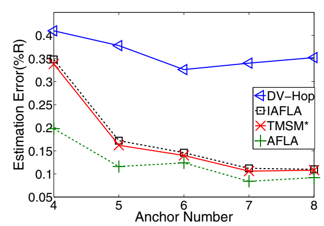
(a) Border



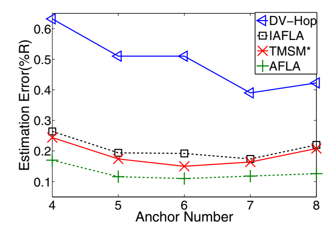
(b) Random

Fig. 5. Estimation Error vs Node Density

3) Different Anchor Numbers: Figure 6 presents the effect of different anchor number on these four algorithms. The estimation error will decrease in all these schemes when the number of anchors grows. It is intuitive because the unknown nodes know more certain information. This parameter is much more effective in IAFLA than DV-Hop. In IAFLA, we can get more exact fine-grained information with more anchors. Therefore, IAFLA has less estimation error.



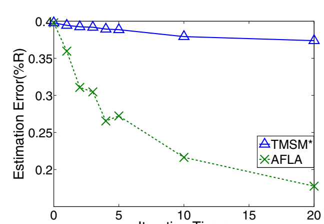
(a) Border



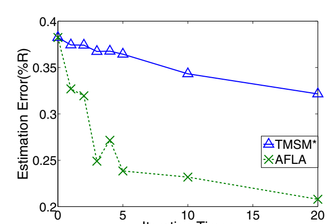
(b) Random

Fig. 6. Estimation Error vs Anchor Number

4) Different Iteration Times: From figure 7, we find AFLA shows great advantage in convergence rate compared to TMSM*. AFLA converges much more effectively than TMSM*. The results further demonstrate the reasonableness of CSM. In AFLA, CSM adjusts the moving step dynamically according to correlated neighboring information, while fixed moving step is used for refinement in TMSM*. In CSM, we distinguish nodes through their confidence level. Nodes in well condition contribute more, and vice versa. In TMSM*, the moving step of node is decided by its neighbor nodes, which are not differentiated dynamically. Inevitably, TMSM* refines the same initial results much more slowly than CSM.



(a) Border

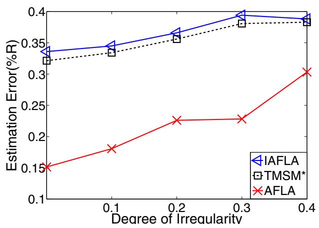


(b) Random

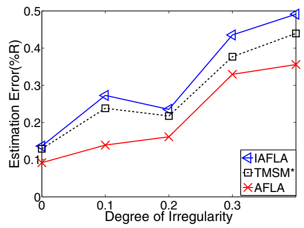
Fig. 7. Estimation Error vs Iteration times

Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.

5) Different Degree of Irregularity: In this section, we analyze the effect of the DOI on the localization accuracy. Figure 8 shows the estimation error increases as DOI value growing up. It is obvious because *RSSI* value in the initial phase can be affected greatly by DOI. But part effect of DOI can be eliminated through the reciprocity of neighboring nodes in refinement phase. And AFLA can still obtain an approximate 35% accuracy with a DOI of 0.4. Hence, AFLA is robust in noisy environment.



(a) Border



(b) Random

Fig. 8. Estimation Error vs DOI

In the last four groups of experiments, we consider both two types of anchor placement: border and random. The simulation results show that our scheme is robust to the position of anchors, while DV-Hop is more sensitive to the disposal of anchors. Besides, AFLA can achieve 30% higher average accuracy than DV-Hop in most scenarios.

V. Conclusion

In this paper, we propose a novel localization scheme called AFLA. In this scheme, we first reduce the deviation of information collected through refining hop-count information. Then we utilize the fine-grained information to implement initial localization. Finally we refine initial results through CSM to obtain higher position accuracy. Investigating the results of simulations, we confirm AFLA not only gets more precise initial position estimations but also converges faster in the later course of further refinement than existing algorithms. Moreover, AFLA is robust to achieve good accuracy in noisy environment without any specific requirement on node deployment.

Acknowledgment

This paper is supported by the National Grand Fundamental Research 973 Program of China under Grant No.2006CB303006, the China Next Generation Internet Program, the Knowledge Innovation Project of the Chinese Academy of Sciences (CAS), the National Science Foundation of China under Grant No. 60803009 and 60803116.

References

[1] D.Niculescu and B. Nath. Ad hoc positioning system (APS). GLOBECOM'01, pages 2926-293, 2001.
[2] R. Nagpal, H.Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. IPSN'03, pages 333-348, 2003.
[3] N.B.Priyantha, H.Balakrishnan, E.Demaine, and S.Teller, Anchor-free distributed localization in sensor networks. Technical Report 892, MIT Laboratory for Computer Science, April 2003.
[4] P. Bahl and V. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. INFOCOM'00, pages 775-784, 2000.
[5] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. IEEE Personal Communications, 4(5):42-47, Oct. 1997.
[6] N.Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. MobiCom'00, pages 32-43, 2000.
[7] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AoA. INFOCOM'03, pages 1734-1743, 2003.
[8] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. MobiCom'03, pages 81-95, 2003.
[9] Ziguo Zhong and Tian He. MSP: multi-sequence positioning of wireless sensor nodes. SenSys'07, pages 15-28, 2007.
[10] Ziguo Zhong, Dan Wang, and Tian He. Sensor Node Localization Using Uncontrolled Events. ICDCS'08, pages 438-445, 2008.
[11] K.Yedavalli, and B. Krishnamachari. Sequence-Based Localization in Wireless Sensor Networks. IEEE Transactions on Mobile Computing, Volume 7, Issue 1, pages 81-94, Jan. 2008.
[12] www.tinyos.net
[13] http://www.xbow.com/Products/productdetails.aspx?sid=164
[14] Joseph Hoshen, The GPS equations and the problem of Apollonius, IEEE Transactions on Aerospace and Electronic Systems, Volume 32, Issue 3, pages 1116-1124, July 1996.
[15] Yang Sungwon, Yi Jiyoung and Cha Hojung. HCRL: A Hop-Count-Ratio based Localization in Wireless Sensor Networks. SECON'07, pages 31-40, June 2007.
[16] Yi.Shang and Wheeler.Ruml Improved MDS-based localization, INFOCOM'04, pages 2640-2651, March 2004.
[17] T. S. Rappaport, Wireless Communications: Principles and Practice, 2nd ed. Upper Saddle River, NJ: Prentice Hall PTR, 2002.
[18] Gang Zhou, Tian He, Sudha Krishnamurthy, John A. Stankovic, Models and solutions for radio irregularity in wireless sensor networks. ACM Transactions on Sensor Networks (TOSN), Volume 2, Issue 2, pages 221-262, May 2006.

Authorized licensed use limited to: Yonsei University. Downloaded on May 30, 2009 at 01:06 from IEEE Xplore. Restrictions apply.