

Reducing location error in Wireless Sensor Networks

T.Sathya PG Scholar¹, S.Saranya Assistant Professor²

¹(Computer science and Engineering, SRM University, Chennai, Tamilnadu, India)

²(Computer science and Engineering, SRM University, Chennai, Tamilnadu, India)

Abstract: Localization is the basic problem in the wireless sensor networks. All the result data get wasted if the node is placed in the wrong position. Range based and Range free algorithms are used for locating the node in the wireless sensor network. But in the existing system they will use any one of the algorithm. The method CDL- combined and Differentiated Localization approach inherits the properties of both Range based and Range free algorithm. CDL provides the location accuracy efficiently but still it has location error in large scale networks. To overcome this problem the novel, geographic routing method named Conditioned Mean Square Error Ratio (CMSER) routing, intended to route packets when the localization error is occurred.

Keywords: Geographic routing algorithm, Localization, ranging quality, received signal strength indicator (RSSI), wireless sensor network (WSN).

I. Introduction

A fundamental problem in sensor network is localization identifying the location of nodes. Location information is used to identify the nodes and to route the packets with minimum distance to accurate location. All the sensing data will get waste when the node is placed in the wrong position. The Global Positioning System (GPS) consists many localization approach but they get fails in indoors and under the ground and in forest. Manual configuration is not feasible for large scale networks or networks where the nodes are moving. Providing hardware in all localization is very cost expensive. To locate the node in the wireless sensor network currently two approaches are used for localization. They are Range free and range based approach using RSSI. In practical situation both the algorithm does not able to address the issue of the localization because of dynamic environments. Range-free algorithm takes only network connectivity information and for calculating the distance it use the number of hops between the nodes. The other methods used to calculate the distance between nodes by either using the Time-Of-Arrival (TOA), Time-Differential-Of-Arrival (TDOA), Angle-Of-Arrival (AOA), or Received-Signal-Strength-Indicator (RSSI) technologies. Range based algorithm makes no assumption about the connectivity or availability of the node information in the WSN. Because Range based algorithm have hardware limitations in WSN devices, but the result in range-free localization are being pursued as a cost-effective compared with range-based approaches. To solve the limitations in Range based and Range free algorithm we propose the solution CDL, Combined and Differentiated Localization approach. CDL inherits the advantages of range based and range free algorithm. Initially it calculates distance by using the DV-hop localization algorithm, and then it improving the ranging quality and location accuracy process by double filtration and calibration process. It used to reduce the location error by giving the location accuracy.

1. Range-free algorithms:

Do not use radio signal strengths or distance measurements and do not need any special hardware. Range-free algorithms require that each node knows

- 1) Which nodes are within radio range.
- 2) Their location estimates.
- 3) The radio range of sensors.

No other information is need for localization. Thus, range free techniques are more cost-effective because they do not require any hardware for sensors, but use less information than range-based algorithms.

2. Range-based algorithms:

Ranging-based localization algorithm gives better localization result than range-free algorithm. Ranging quality finds the overall localization accuracy of the each node. To achieve better location accuracy in the WSN recently developed algorithms focused more on error control in the WSN. Some of those methods enhance the localization accuracy by deliberately reducing the contribution of error-prone nodes to the localization accuracy process.

Range free and range based both has limitations in locating the node.

a. Non-uniform Deployment:

When more sensor nodes are deployed in region like forestry, in the monitored area, some nodes have more than 15 neighbors, while some have less than 3 neighbors. Range free localization has a non-uniform deployment incurs errors. Fig.01. Represents WSN in the GreenOrbs deployment in the woodland.



Fig.01. Non uniform deployment of node

b. Irregularity of RSSI: Terrain and obstacles affect the signal propagation between the nodes in the WSN. The distance between node pairs differs because of the non uniform deployment of the node. Fig.02. represents RSSI of different node pairs.

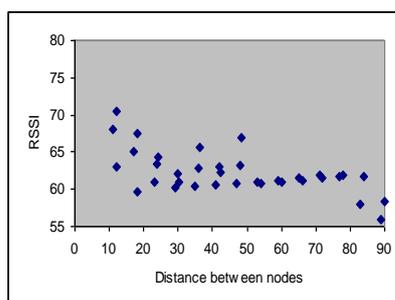


Fig.02. Irregularity of RSSI

c. Asymmetric and Dynamics of RSSI:

The RSSI between the links (say) AB and BA is differing in same time. The RSSI between two nodes is asymmetric in same time. No two links have same RSSI. Fig.03. Represents the RSSI between nodes A and B over time.

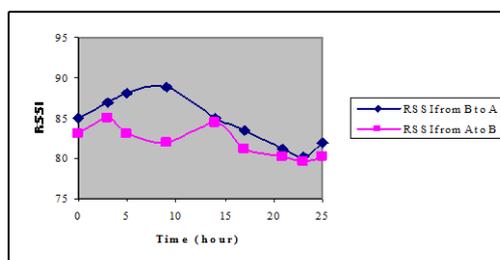


Fig.03. Asymmetric RSSI

In the WSN, the node's ranging quality includes two aspects:

- 1) Location accuracy of the reference node.
- 2) Accuracy of range measurements.

To solve the problem in Range based and Range free algorithm, CDL, a Combined and Differentiated Localization method is used.

II. CDL Design

The CDL, a Combined and Differentiated Localization approach. CDL solves localization process with a more accuracy, efficiency and consistent.

CDL consists of

- ❖ Virtual Hop Localization.
- ❖ Local Filtration.
- ❖ Calibration.

It starts localization by method DV-hop, and then it keeps improving then ranging quality and accuracy iteratively throughout the localization process. *Virtual Hop Localization* use to calculate the distance using Range free algorithm, *Local Filtration* picks good nodes with good neighbors. *Calibration* finds the node with the best range measurement. Fig.04 illustrates the CDL architecture diagram with workflow.

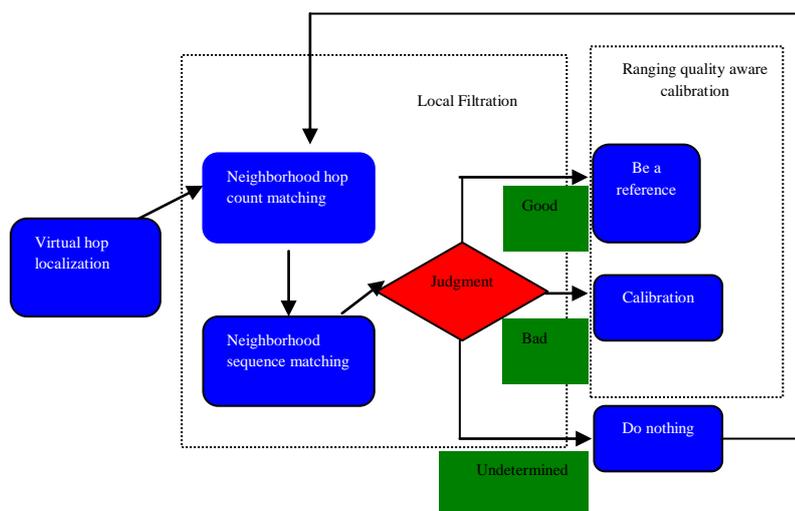


Fig.04 Workflow Diagram

A. Virtual-hop localization:

It first calculates the node locations. Compared with the DV-Hop algorithm, each node, counts the virtual hops instead of DV-hops, compensating for the errors in the non uniform deployment problem in the Range free algorithm. Virtual Hop Localization is an advanced version of the traditional hop Count based localization. In the traditional method distances of each node is calculated by distance from each node to the landmarks node (node which one knows their co-ordinates when it is deployed in the network).

Problem in Range Free Localization Algorithm:

Range Free Algorithm is based on the network Connectivity measurements. Sensor Node will be randomly distributed in the WSN. When the node is represented in the graph each sensor is represented by a node, and two nodes are connected by the edge if they can be communicated with the node within the one hop distance. It is possible to move a sensor node over nonzero distance without changing the set of its 1-hop neighbors.

DV-hop is one of the Range free algorithms that use the network connectivity information to estimate the distance between the nodes. The distance is calculated by, every node counts its hop count value from the land marks node. Nodes with the same hop count value will have different distances from landmarks. Fig.05. nodes are having different distances within the same hop count value.

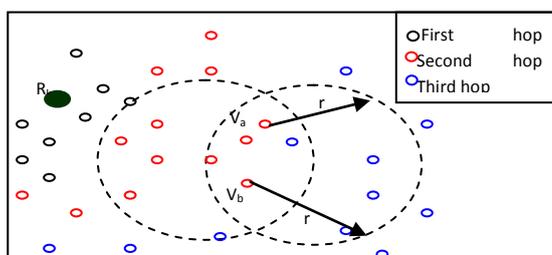


Fig.05. same hop counts have different distances.

Fig 05 shows sensor node within the third hop away from the land mark node R_k . For example node v_a and v_b are in two hop distance from the land mark node R_k . But v_a is closer to the land mark node than v_b . So DV hop does not provide the location accuracy.

Virtual-Hop:

The *virtual-hop-count*, V_{jk} , to represent the distance between a node v_i and a landmark R_k . Within the nodes with the same hop count to R_k , nodes closer to R_k , should have a smaller V_{jk} . Table 1 lists the symbols and notations used in this paper. Each v_j computed V_{jk} by

$$V_{jk} = \frac{1}{P_{jk}} \sum_{P_{ik} \ni V_i} V_{jk} + L_{jk} \quad (1)$$

TABLE 1
SYMBOLS AND NOTATIONS

Sy	Definition
V_{jk}	Virtual hop count from node R_k to node v_j
P_{jk}	$\{ v_i h_{jk} = 1 \text{ and } h_{ki} < h_{kj} \}$
N_{ik}	$\{ v_i h_{jk} = 1 \text{ and } h_{ki} > h_{kj} \}$
L_{jk}	Last virtual hop count
h_{jk}	Hop count from node v_i to node v_j

V_{jk} consists of two parts.

The first part is the average virtual hop count of node's v_j previous-hop neighbors.

The second part is the last virtual hop count that is, the incremental virtual-hop-count from v_j 's previous-hop neighbors to v_j , denoted by L_{jk} .

Here, a node v_j 's previous-hop neighbor is defined as a neighboring node whose hop count to landmark R_k is just one hop less than v_j , (denoted by in Table I). v_j 's next-hop neighbor is defined as a neighboring node whose hop count is just one hop more than v_j (denoted by N_{jk} in Table I).

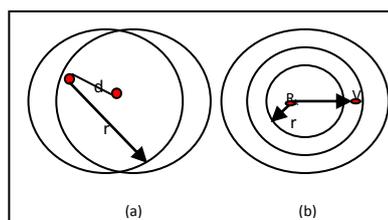


Fig.06.Virtual hop distance

(a) distribution of node distances; (b) neighbors with different hop counts.

Fig. 06(b) shows the node's relationship of neighbors with different hop counts. The concentric circles separately denotes the location boundary of 1 hop, 2 hop, 3 hop distance from the landmark node R_k . The circle denotes the communication range of v_i who is a 2-hop neighbor of R_k . The intersection, denoted as $A(P_{ik})$, of the circle and the small circle (centered at R_k) is the region where v_i 's previous hop neighbors present. The intersection denoted as $A(N_{ik})$, of the circle centered at R_k is the region where v_i 's next hop neighbor locate.

For any node v_i , the distance between the node and landmark R_k (denoted by d) satisfies $r < d < 2r$, it has two hops distance to landmark node R_k . Two different distances with the same hop count is close to r . For virtual hop, such two nodes have different virtual hop counts values. Thus virtual hop used to solve the problem "same hop counts have different distances".

B. Local Filtration

In filtration, CDL uses two filtering methods to identify good nodes. The good nodes are whose location accuracy is satisfactory. *Neighborhood hop-count matching* filters the bad nodes. The bad nodes are identified by the hop counts with its neighbor's node. *Neighborhood sequence matching* distinguishes good nodes from bad ones based on RSSI sequence and distance sequence.

Model-Based Filtration:

Filtration is used to examine the efficiency of location by all the nodes will experimented without any differentiating between good nodes and bad nodes. This model-based calibration is called *indiscriminate*

calibration. Using such calibration, every node's location is adjusted based on the distances to neighbors converted by RSSI value, using the log-normal shadowing model. Model-based filtration is infeasible, considering the estimated localization error and irregularity of RSSI.

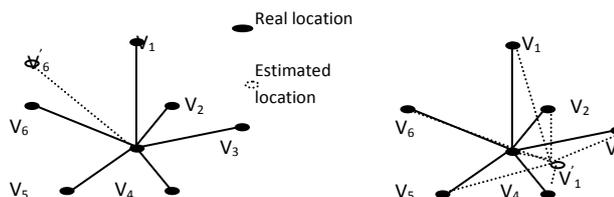


Fig. 07. (a) Good node with a bad neighbor. (b) Bad node with good neighbors.

We can calculate the distances between two nodes by two ways, for example v_1 and its neighbor v_2 . One way is to calculate the distance using their coordinates, say a_1 . The second way is converts the RSSI from v_1 to v_2 into a distance (RSSI distance) based on the log-normal shadowing model say a_2 . We expect $a_1=a_2$. Because of the errors of estimated coordinates and the error from the log normal shadowing model, there is some difference between a_1 and a_2 . By summing up $|a_1-a_2|$ corresponding to every neighbor, we can measure the *Aggregated Degree of Mismatches* (ADM) of v_1 .

ADM actually reflects the error of a node's estimated location. For example in Fig.07 (a), v_1 is a good node with six neighbors. Among them, v_6 only is a bad node. Let v'_6 denotes its estimated location. The ADM of v_6 is mainly caused by v'_6 . In Fig.07 (b), v_1 is a bad node with six good neighbors. The link to every neighbor contributes to the ADM of v_1 . Comparing these two figures, ADM of a bad node is typically higher than that of a good one. Thus, we may distinguish good nodes from bad ones by contrasting their ADM.

Neighborhood Hop-Count Matching:

In ADM, the first step is each node takes neighborhood hop-count matching to check whether the particular node is a *good node* based on local connectivity information. Hop count value is important to estimate the distance between two nodes. If a node's hop counts to its greatly differs with the neighbors then the distances calculated using the nodes estimated coordinates, will have a large error

Neighborhood Sequence Matching:

Though model-based filtration is infeasible, RSSI still have some useful information. The RSSI between two nodes decreases as the distance increases from the RSSI readings. Based on this observation, the filtration is called *neighborhood sequence matching*.

First, v_1 sorts its neighbors in descending order with the RSSI from them, sequence number will generated for each neighbor. By mapping the sequence numbers into v_1 , we get the sequence called *RSSI sequence*. Let denote it by RS_a .

Second, according to the coordinates, v_1 sorts its neighbors in the ascending order with the estimated distance between them, we get the second sequence called *distance sequence*. Let denote it by RS'_a .

If there is no noises in the environment then the value of RS_a and RS'_a should be same. If there is mismatch, it indicates that the node's estimated coordinates contains large error.

The mismatch between RS_a and RS'_a is caused by the following reasons:

- ❖ the location estimation errors,
- ❖ the irregularity of RSSI between v_1 and its neighbors.

To distinguish good nodes from bad ones, calculate the distance between RSSI sequence and distance sequence. To improve the filtration performance, need to suppress the irregularity of RSSI.

We define the matching degree M_i between the RSSI sequence and distance sequence as follows.

$$Cosdist = \frac{a_1 a'_1 + a_2 a'_2 + \dots + a_n a'_n}{\sqrt{a_1^2 + a_2^2 + \dots + a_n^2} \sqrt{a_1'^2 + a_2'^2 + \dots + a_n'^2}} \tag{2}$$

a_1, a_2, \dots, a_n , are the sequence numbers in RS_a , while a'_1, a'_2, \dots, a'_n are the sequence numbers in RS'_a . These two sequences are actually two different permutations of 1, 2... n. Thus, they are two equal sets. The cosine distance filtration reduces the influence of RSSI irregularity.

When a good node has bad neighbors with large location errors, the cosine distance between two sequences of a good node does not actually differ from that of a bad node. To solve this issue, the new method longest common subsequence (LCS) length ratio $\hat{\rho}_a$ is used. Let n denote the number of v_1 's neighbors. Then, $\hat{\rho}_a$ denotes the ratio of the length of the LCS between RS_a and RS'_a to n . The LCS length ratio of a good node is higher than that of a bad node. The LCS length ratio $\hat{\rho}_a$ is error-tolerant to interference of bad neighboring nodes with large location estimation errors. The boundary of $\hat{\rho}$ is between 0 and 1.

The matching degree between the RSSI sequence and distance sequence is:

$$M_i = \hat{\rho}_i \cdot \text{Cosdist}_i \quad (3)$$

M_i is used to differentiate the good nodes from bad nodes. When a small portion of RSSI readings has large errors, or a good node has some bad neighbors with large location errors, the matching degree cannot be inclined too much.

C. Ranging-quality Aware Calibration (RQAC):

The iterative process of filtration and calibration improves localization accuracy of each node. This process contains two design criteria. First, filtration must find many good nodes with high localization accuracy to facilitate calibration. Second, good nodes may have both good links and bad links. Only the good links (with small ranging errors) should enter into the calibration, while the bad links must be restrained. Filtration addresses the first criterion, while calibration addresses the second criterion.

RQAC Estimator:

The undetermined node contains both good nodes and bad nodes, but only the good node will be considered as a reference node. The node v_i , and the ranging quality of its neighbor v_j is simultaneously determined by *two factors*:

- ❖ the location accuracy of v_j , and
- ❖ the ranging error over the link from v_j to v_i .

RQAC estimates the ranging quality of a good node v_j with its good neighbors as follows:

$$g(i,j) = \left\{ \begin{array}{l} \alpha w_j (l_{ij} - d_{ij})^2, \quad |l_{ij} - d_{ij}| < \varepsilon \\ \ln (|l_{ij} - d_{ij} - \varepsilon + 1|) |l_{ij} - d_{ij}| \geq \varepsilon \end{array} \right\} \quad (4)$$

w_i and $|l_{ij} - d_{ij}|$ denote the ranging quality from w_i to v_i , and α is a constant parameter.

From the formula (4) range measurements to the node v_i are divided into two types based on their ranging quality. The range measurements with errors less than ε contribute more to the calibration process by taking the quadratic form of $|l_{ij} - d_{ij}|$. For a range measurement with an error not less than ε , its contribution is suppressed by taking the logarithmic form of $|l_{ij} - d_{ij}|$. Range measurements in the same type are also differentiated from others by taking the weights of reference nodes (w_i) into account. The same way, RQAC contributions the best range measurements, remove the interference of outliers, and suppress the ranges in between the nodes.

Problem in CDL:

Even though CDL produces good possible localization accuracy in an efficient manner, it will effectively work on the small scale networks. When the nodes are deployed in the large scale network in the wild, it contains location error. So it increases the packet transmission rate in the network.

In geographic routing performance, localization errors have a great impact. Routing algorithms have to address the presence of location errors. So we need a routing algorithm which addresses the location error, that algorithm should be able to analyze strictly the forwarding techniques, it is assumed that communication is not affected by any environment.

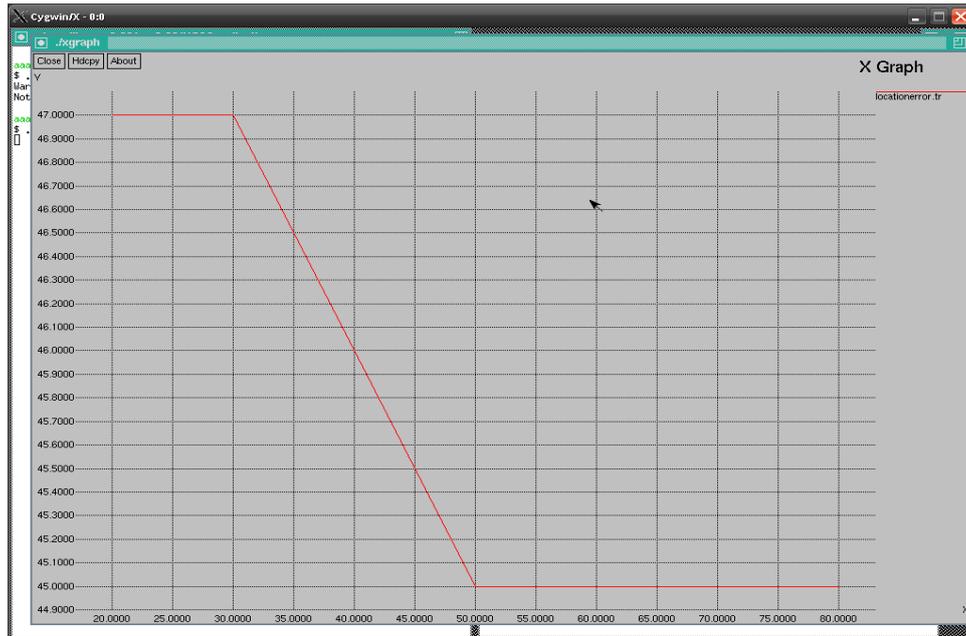


Fig.08 Reducing Location Error

The above result shows that location error has been reduced in CDL but still it has reduced location error.

III. Geographic Routing

The routing algorithm which address the location error in the large scale network is called, Conditioned mean square error ratio (CMSER). The aim of the algorithm is minimize the location error on the network.

CMSER algorithm is an alternative method to improve the performance while still coping with location errors. CMSER dismissing the possibility in backward progress

CMSER throughput grows higher without the lost packets traveling in the network for a large number of hops, thus reducing the overall power consumption of the network.

CMSER intended to efficiently make use of existing network information and to successfully route packets when localization is inaccurate. Next hop selection (in CDL) is based on the largest distance to destination (minimizing the number of forwarding hops) and on the smallest estimated error figure associated with the measured neighbor coordinates. It is found that CMSER outperforms other basic greedy forwarding techniques employed by algorithms such as most forward within range (MFR), maximum expectation progress (MEP) and least expected distance (LED). CMSER makes use of the notion of maximum progress to destination, but gives more importance to the probability of success when coordinates are affected by location error. As a consequence, the energy spent on lost routing packets is considerably decreased. While the paths of the received packets of CMSER may be longer, the routes of the lost packets are kept short, being surpassed only by MFR, which does not cope with location error at all.

IV. Conclusion

In the CDL design, experimental results show that CDL outperforms location with higher accuracy, efficiency, and consistent performance. But CDL doesn't solve the problem, location error in large scale network. The routing algorithm, CMSER, whose performance in terms of throughput is considerably better when compared to other basic routing algorithms MFR, MSER, MEP and LED. All the experimental results confirm that CMSER outperforms all other routing algorithms when the network objective is to increase packet delivery. Overall energy costs are also minimum. CMSER makes use of the notion of maximum progress to destination, but gives more importance to the probability of success when coordinates are affected by location error. While the paths of the received packets of CMSER may be longer, the routes of the lost packets are kept short, being surpassed only by MFR, which does not cope with location error at all.

References

- [1]. H. Kung, C. Lin, T. Lin, and D. Vlah, "Localization with snap-inducing shaped residuals (SISR): Coping with errors in measurement," in Proc. ACM MobiCom, 2009, pp. 333–344.
- [2]. Localization of Wireless Sensor Networks in the Wild: Pursuit of Ranging Quality
- [3]. Jizhong Zhao, Associate Member, IEEE, Member, ACM, Wei Xi, Student member, IEEE, ACM, Yuan He, Student Member, IEEE, Member, ACM, Yunhao Liu, Senior Member, IEEE, Xiang-Yang Li, Senior Member, IEEE, Lufeng Mo, and Zheng Yang, Student Member, IEEE, ACM
- [4]. Y. Shang, W. Rumi, Y. Zhang, and M. Fromherz, "Localization from connectivity in sensor networks," IEEE Trans. Parallel Distrib. Syst.,
- [5]. "Geographic Routing Resilient to Location Errors" Ana Maria Popescu, Naveed Salman, and Andrew H. Kemp
- [6]. M. Popescu, I. G. Tudorache, and A. H. Kemp, "Surveying position based routing protocols for wireless sensor and ad-hoc networks," International J. Commun. Networks and Inf. Security, vol. 4, no. 1, 2012.
- [7]. S. Kwon and N. B. Shroff, "Geographic routing in the presence of location errors," Computer Networks, vol. 50, pp. 2902–2917, 2006.