Collision-Based Localization for Small Scale Mobile Wireless Sensor Networks

Xinan Yan, Lei Shang, Alei Liang, Haibing Guan
Embedded System Lab, School of Software, Shanghai Jiao Tong University, China
fert130@yahoo.com.cn, hbguan@sjtu.edu.cn

Abstract — Location awareness is important in most of the wireless sensor networks. Many localization schemes have been proposed for both static and mobile wireless sensor networks. However, most of them aim at localizing the sensor nodes with global coordinate by using additional ranging hardware or anchors knowing its global position. These schemes may suffer from the problem of no reference coordinate system when there is no anchor knowing global coordinate. This paper proposes a distributed localization mechanism based on collision among mobile nodes for small scale mobile wireless sensor networks. And the scheme is range-free and self-organized. In our approach, nodes work as swarms to move and communicate with others. And finally, a relative coordinate system is established among the nodes. The algorithm has been evaluated based on simulator. The simulation results show that our scheme can finish the establishment of the relative coordinate system in finite time and the increase of nodes will speed up the process.

I. INTRODUCTION

Wireless sensor networks (WSN) can be widely used in many real world applications such as military, home, industry, environment and etc [1]. And in most wireless sensor networks, especially in target detection, coverage calculation, mapping, data routing and etc, nodes’ position information is important and necessary [2]. However, the most widely used positioning technique in traditional applications, Global Positioning System (GPS) [3], does not cost effective in WSN. Because there are usually lots of nodes, the cost will be expensive if every node is equipped with a GPS receiver. Besides, GPS does not work well in cluttered urban area or under dense foliage, even disable on the Mars.

To overcome the limitations of GPS, extensive localization mechanisms by applying ranging techniques, called range-based schemes, such as time difference of arrival [5], angle of arrival [6], time of arrival [7] and radio signal strength indicator [4], have been proposed for localization in WSN. Though this kind of localization mechanism has high accuracy, each node has to be equipped with additional hardware for ranging which will consume more energy and production cost.

Researchers also developed range-free methods which is independent of special equipment for ranging. Most of these schemes use anchors which are equipped with devices can tell the global position such as GPS. Some schemes are focus on the localization for the static wireless sensor networks [8, 9, 10, 11, 12]. However, the amount or dense of the anchors and the deployment are the big problems when the topology of networks are unknown. To solve these problems, mobile anchors are introduced for localization [13, 14, 15]. However, they won’t work effectively if the nodes are moving. With considering nodes’ movement capability, localization problem is becoming more difficult. Accordingly, there are limited research efforts to resolve the localization for mobile sensor networks [16, 17, 18].

Most of these algorithms use anchors to locate the mobile nodes’ global position in global coordinate system and the radio range used in many schemes is approximated as one standard circle. However, in real world the radio range is always irregular and radio intensity is instable. In many applications, there is no need or even no possibility to locate global coordinate for nodes. For example, in the NASA’s Mars Tumbleweed project [19], establishing fixed landmark or positioning from space shuttle is proposed by Kenneth Lodding (mentioned in [16]) to solve the localization problem because GPS does not work on Mars. And on the earth, in the case of that the nodes with mobility are scattered in unknown area where no global coordinate
reference system is available, establishing relative positions and creating self coordinate system are important and necessary to many applications such as map detection.

This paper presents a distributed self-organized localization method for small scale mobile wireless sensor networks to create a relative coordinate system among nodes without using anchors or any additional hardware for range information. In our approach, we focus on the scenario that the wireless sensor networks’ scale is small where nodes are mobile and there is no centralized computing station, no ranging devices, no global coordinate reference system and no anchors. The most important point is that nodes will act as swarm to finish localization work based on “collision”, which means that two nodes just meet.

Next we provide a description of our localization scheme in detail in Section 2. And section 3 gives the simulation result and analysis. Finally, Section 4 concludes and gives some further work direction.

II. COLLISION-BASED LOCALIZATION

A. Environment and Assumption

This paper considers the system environment where nodes are mobile and there are no anchors or any other global coordinate system reference. The nodes without any additional hardware for localization are initially scattered randomly in a small scale with unknown their own or others’ locations. With our approach, in the specified scale, the nodes will move randomly and work as swarm to create a relative coordinate system for specified percent of them based on collision and coordinate adjustment.

The whole localization process is distributed and self-organized. And in our scheme, to simple the complexity, there is some assumption as following:

1) The scale is small and nodes can hear the whole network;
2) Each node has enough energy;
3) There are only four directions as one cross;
4) Each node move in line;
5) Each node knows its direction and distance it moves. This kind of mobile node is currently available [20];
6) The node can detect object which is close enough (thought as zero exposure) on its ahead, left and right by infrared or such simple techniques.
7) Each node has unique ID or MAC and one priority value, used in the creation of coordinate system.

B. Localization

In the specified scale with borders, there are N number of nodes scattered with random direction, and our aim is that at least M nodes (M < N) establish one relative coordinate system for themselves. Our localization approach takes effect as follows:

Step 1) Each node scattered establishes one 2-D coordinate system for only itself at the initial position and maintains one list, called ID-List, to record the nodes’ ID those in the same coordinate system. Each node generates one coordinate system with its current position as origin and head direction as positive x axis, which complies with right-handed coordinate system except the z axis. And the ID-List is initialized empty.

Step 2) Nodes move randomly and maintain updating self-coordinate. In our experiment, the mobility model is based on the Random Waypoint model [21]. Each node randomly selects direction and length it moves. After passing the length in that direction, the node randomly generates new direction and length. When moving, each node keeps its coordinate updating in its self-coordinate system, since it knows moving direction and distance.

Step 3) When two nodes meet each other, they disable the ability of detection and turn direction in order to make sure that they are in the position of back to back.

![Figure 3.1 Four kinds of collision situation](image)

**Figure 3.1** Four kinds of collision situation

**Collision Situation**

Since each node has the ability of detecting objects on its front, left and right, when two nodes detect each other
close enough, we consider that they are in collision, that is, they meet each other. The situations of two nodes knocks together are enumerated in Figure 3.1, and Figure 3.2 shows the two nodes’ back to back situation after turning, in which red arrows mean left, ahead and right.

**Step 4**  One coordinate system selected from one of the two meet nodes in Step (3) is thought as the standard coordinate system. And the other node will change its coordinate into the standard coordinate system, if it is in different coordinate system.

**Standard Coordinate Selection**

The decision of the standard coordinate system is based on the nodes’ different priority values defined in Assumption (7). If the two meet nodes have different priority value, the high-priority node’s coordinate system is selected as the standard one. Then the low-priority node will change its coordinate into the standard coordinate system, performed in Step (5). Otherwise, if the priority values are the same, the two nodes will separate and take action as in Step (2).

**Step 5**  The low-priority node defined in Step (4) is called Min node, and the high-priority node is called Max node. Since they are in back to back position, so

\[
D_{\text{min new } x \text{ axis}} = D_{\text{max } x \text{ axis}} \quad (3.1)
\]

\[
X_{\text{min new}} = X_{\text{max}} + \sin(\sigma_x - 90) \quad (3.2)
\]

\[
Y_{\text{min new}} = Y_{\text{max}} + \sin(\sigma_y - 90) \quad (3.3)
\]

\[
D_{\text{min new head}} = D_{\text{max head}} + 180 \quad (3.4)
\]

where \( D_{\text{min new } x \text{ axis}} \) is the new direction of x axis of Min node; \( D_{\text{max } x \text{ axis}} \) is the direction of x axis of Max node; \( X_{\text{min new}} \) and \( Y_{\text{min new}} \) are \((x, y)\) coordinate of Min node in the standard coordinate system; \( X_{\text{max}} \) and \( Y_{\text{max}} \) are the \((x, y)\) coordinate of Max node; \( \sigma_x \) is the absolute value of the interval angle between Max node’s x axis and the Max node’s current head direction; \( \sigma_y \) is the absolute value of the interval angle between Max node’s y axis and the Max node’s current head direction; \( D_{\text{min new head}} \) and \( D_{\text{max head}} \) are respectively the current head direction of Min and Max node. Figure 3.3 gives one sample of Min node’s coordinate conversion.

With calculation by formula (3.1), (3.2), (3.3) and (3.4), the Min node finishes converting its coordinate to the Max node’s coordinate system. That is, the Min and Max node are in the same relative coordinate system.

![Diagram](image)

**Step 6**  The Min node set its priority value equal to the one of Max node. Same priority values means nodes are in the same coordinate system.

**Step 7**  Both Min and Max node notify the nodes, in the same coordinate system with them respectively, to update ID-List.

**Max Node:** Max node requests all the IDs in the ID-List of Min node. Then Max node adds these IDs including the Min’s ID to its ID-List, and notifies the nodes in its old ID-List to update by sending the new coming IDs.

**Min Node:** Min node makes similar process what Max node does to notify the nodes in its ID-List to update ID-List. Furthermore, these nodes change their coordinate into Min node’s new coordinate system. The process of changing the coordinate system of node T, which is in Min node’s ID-List before Step (5), is given as follows:

Define the coordinate system of Min node before Step (5) as Old Coordinate System (OCS) and the one after Step (5) as New Coordinate System (NCS). The rotated angle of the coordinate system from OCS to NCS is

\[
\Delta D = D_{\text{min new } x \text{ axis}} - D_{\text{min old } x \text{ axis}} \quad (3.5)
\]

where \( D_{\text{min old } x \text{ axis}} \) is Min node’s x axis direction in OCS. And in OCS, here is

\[
D_{T, \text{ old } x \text{ axis}} = D_{\text{min old } x \text{ axis}} \quad (3.6)
\]

In NCS, here is

\[
D_{T, \text{ new } x \text{ axis}} = D_{\text{min new } x \text{ axis}} = D_{T, \text{ old } x \text{ axis}} + \Delta D \quad (3.7)
\]
where \( D_{T,old,x_axis} \) and \( D_{T,new,x_axis} \) are T’s x axis direction in OCS and NCS respectively.

In OCS, vector \( \overrightarrow{MinT} \) is \((\Delta X_{old}, \Delta Y_{old})\). And here are

\[
\begin{align*}
\Delta X_{old} &= X_{T,old} - X_{min,old} \\
\Delta Y_{old} &= Y_{T,old} - Y_{min,old}
\end{align*}
\] (3.8)

\[
\begin{align*}
\Delta X_{old} &= r \cdot \cos(\alpha) \\
\Delta Y_{old} &= r \cdot \sin(\alpha)
\end{align*}
\] (3.9)

where \( r = |\overrightarrow{MinT}| \) and \( \alpha \) is the angle between Min node’s x axis direction and \( \overrightarrow{MinT} \) direction; \((X_{T,old}, Y_{T,old})\) is T’s coordinate in OCS, \((X_{min,old}, Y_{min,old})\) is Min’s coordinate in OCS.

After rotating \( \Delta D \) from OCS to NCS, and in NCS, the vector \( \overrightarrow{MinT} \) is \((\Delta X_{new}, \Delta Y_{new})\). Here are

\[
\begin{align*}
\Delta X_{new} &= X_{T,new} - X_{min,new} \\
\Delta Y_{new} &= Y_{T,new} - Y_{min,new}
\end{align*}
\] (3.10)

\[
\begin{align*}
\Delta X_{new} &= r \cdot \cos(\alpha + \Delta D) \\
\Delta Y_{new} &= r \cdot \sin(\alpha + \Delta D)
\end{align*}
\] (3.11)

where \((X_{T,new}, Y_{T,new})\) is T’s coordinate in NCS, \((X_{min,new}, Y_{min,new})\) is Min’s coordinate in NCS.

\[
\begin{align*}
\Delta X_{new} &= r \cdot \cos(\alpha) \cos(\Delta D) - r \cdot \sin(\alpha) \sin(\Delta D) \\
\Delta Y_{new} &= r \cdot \sin(\alpha) \cos(\Delta D) + r \cdot \cos(\alpha) \sin(\Delta D)
\end{align*}
\] (3.12)

\[
\begin{align*}
\Delta X_{new} &= \Delta X_{old} \cdot \cos(\Delta D) - \Delta Y_{old} \cdot \sin(\Delta D) \\
\Delta Y_{new} &= \Delta Y_{old} \cdot \cos(\Delta D) + \Delta X_{old} \cdot \sin(\Delta D)
\end{align*}
\] (3.13)

Then with (3.10) and the result of (3.13)

\[
\begin{align*}
X_{T,new} &= X_{min,new} + \Delta X_{new} \\
Y_{T,new} &= Y_{min,new} + \Delta Y_{new}
\end{align*}
\] (3.14)

And the T’s new head direction in NCS is

\[
D_{T,new,head} = D_{T,old,head} + \Delta D
\] (3.15)

where \( D_{T,old,head} \) is T’s old head direction in OCS.

With the calculation of node T’s \((X_{T,new}, Y_{T,new})\), \(D_{T,new,head}\) and \(D_{T,new,x_axis}\), T finishes its coordinate changing into NCS.

**Step 8** If there are more than M nodes that are in same coordinate system, one relative coordinate system among these M nodes is established and the localization is over. Otherwise, each node moves as described in Step (2).

**Potential Deadlock Phenomenon**

There is a potential problem that whether there are two groups of nodes that they will never integrate into each other and each group has one independent relative coordinate system.

In our approach, this kind of deadlock can be broken. Let’s assume that there are group GA and GB which meet the condition mentioned above. Since each node moves randomly in specified area, there is the possibility of one node from GA and another from GB to meet. Once collision happens, Step (3) to Step (7) make both nodes in GA and GB to be in same coordinate system. And simulation results (Section 3) approve that our approach is effective with no deadlock.

**III. PERFORMANCE EVALUATION**

The performance of the collision-based localization method is evaluated on the simulator which simulates all the assumption defined in section 2. Since the scenario is special and our main aim is to test whether the algorithm can terminate in finite time, we develop the suitable simulator. Figure 4.1 is just one profile of the simulator.

![Figure 4.1 User Interface of the Simulator](image)

**A. Simulation Settings**

Define the size of one node as one size unit. The scale of the simulation for localization area is 100 * 100 size unit\(^2\). Define one time unit as the time counting standard, in which every node finishes one atom action including moving one size unit, turning, sending one message and dealing one message. The aim of the simulation is to test whether this approach can terminate and how long it takes.

The total number of nodes and the least percent of nodes needed to be localized in simulation are \(\{10, 20,\)
30, …, 190, 200} and \{20\%, 30\%, …, 90\%, 100\%\} respectively. The mobile model of nodes is based on the Random Waypoint Model and the random movement will affect the time of the convergence, so for each pair of the total number of nodes and the goal percentage, the simulation will be executed 1000 times. The average time is considered as the approximate result. And in each simulation with the same total number of nodes, the initial positions of nodes are the same which is randomly generated at first time.

### Table 4.1 Sample of Simulation Results

<table>
<thead>
<tr>
<th>Localization Percentage</th>
<th>Total Number of Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
</tr>
<tr>
<td>20%</td>
<td>74.185</td>
</tr>
<tr>
<td>30%</td>
<td>94.345</td>
</tr>
<tr>
<td>40%</td>
<td>107.326</td>
</tr>
<tr>
<td>50%</td>
<td>119.145</td>
</tr>
<tr>
<td>60%</td>
<td>129.948</td>
</tr>
<tr>
<td>70%</td>
<td>139.079</td>
</tr>
<tr>
<td>80%</td>
<td>150.281</td>
</tr>
<tr>
<td>90%</td>
<td>172.152</td>
</tr>
<tr>
<td>100%</td>
<td>300.629</td>
</tr>
</tbody>
</table>

#### B. Simulation Results

One sample of simulation results are described in Table 4.1. It shows that with our approach, the relative coordinate systems can be established in limited time. And time cost is affected by the total number of nodes and the goal number of nodes we need.

1) Impact of localization percentage of nodes. Figure 4.2 indicates that with same amount of nodes, the time cost becomes higher with the increase of number of nodes needed to be in one coordinate system. In our approach, more nodes to be localized, more time cost for nodes moving to make more collision with others.

2) Impact of total number of nodes. Figure 4.3 shows that with same localization percentage, the increase of total number of nodes decreases the time cost for localization. More nodes, more possibility and number of collision happen, the speed of the establishment of the coordinate system is faster.

![Figure 4.2](image1.png)

![Figure 4.3](image2.png)

IV. CONCLUSION AND FUTURE WORK

This paper presented a distributed self-organized localization method based on nodes’ collision and communication for small scale mobile wireless sensor networks to create a relative coordinate system among the nodes. This mechanism is independent of additional hardware for acquiring range information. And there are no anchors with knowing the position information of global coordinate system. With this scheme, nodes work a little like swarm to establish one relative coordinate system among them in limited time under the condition of no global coordinate reference system. The simulation results show that the increase of nodes will speed up the process of the localization.

The future work will focus on how to decrease the
movement and communication among nodes for saving power. And the position error caused by movement or communication will be considered to make this method more robust.

ACKNOWLEDGEMENT

This work is supported by the Project, supported by the National Natural Science Foundation of China (Grant No. 60873209), and the Project supported by the Key Program for Basic Research of Shanghai (Grant No.08JC1411800).

REFERENCE