

# RSSI Based Path Finding of Mobile Robot in Sensor Network

Sung Gil Wee, Jung Won Choi\*, Suk Gyu Lee, Ju Hyun Park

Department of Electrical Engineering, Yeungnam University, Korea

\*Department of Electronic Engineering, Kumoh National University, Korea

Tel: +82-53-810-3923, E-mail: dolphin89 @empal.com

**Abstract** - We propose a new path finding approach for a mobile robot using the RF strength in a sensor network. In experiments based on the proposed method, a mobile robot finds its location, heading direction, and the shortest path under an indoor environment. The mobile robot is equipped with a base station to receive data from the environment around each node. When the robot moves from one point to another, it checks the strength of the received signal to find its path and uses a Kalman filter to reduce the error of the tangent. The experimental results show the effectiveness and applicability of the proposed method in path finding of a mobile robot.

## 1. Introduction

The development of mobile robot technologies combined with sensor networks has become one of the most prominent technological driving forces. The use of these robots working with sensor networks embedded in the environment has broadened conventional industries and has become the basis of new industries. Sensor nodes act as signposts for the robot to follow, thus obviating the need for a map of localization on the part of the robot. The objective of this study, described in this paper, is to improve the information acquisition ability of a mobile robot for use in its navigation by using a sensor network. There are some research efforts on sensor network systems for localization using different kinds of sensors[1-3]. Their localization errors have a range from 1 to 3 m. To reduce the localization error, the approximate position of the mobile robot and its surround information has been used by Batalin et al. [4]. The sensor network needs subsidiary techniques for collecting accurate information during particular circumstances such as fire and other dynamic situations. In the situation described in [4], to find a fire source, the mobile robot scatters inexpensive sensor nodes and gets environmental information such as temperature and humidity, etc. By using static and(or) dynamic mote sensors, it is possible to prevent some accidents such as fire, gas leakage, etc. There have been several papers concerning the localization of mobile robots equipped with sensors such as vision systems, mote systems, IR sensors, ultrasonic sensors,

and laser and sensor networks[5-8].

In this study, we considered an inexpensive sensor network with cheap sensor nodes to find the path of a mobile robot.

## 2. Sensor Network

Table 1 shows the measurement of distance between nodes using signal strength. After regulating the power parameter for the minimum value of the RF signal strength, we recorded the measurement of distance through the signal strength. For controlling the RF signal strength, we modified the NesC program in TinyOS, an event-driven operating system for sensor networks. The TOS component transmits the measured strength of the signal to CPU and the CntToLedsAndRfm component transmits it to the TOS component.

Table 1 Measurement of distance using signal strength

Distance (cm)	Average strength of each node			
	no.1	no.2	no.3	no.4
10	218	229	233	229
20	226	229	235	231
30	228	226	235	227
40	231	230	234	229
50	230	231	234	231
60	235	229	237	231
70	235	233	239	234
80	234	233	243	239
90	249	236	239	239
100	246	246	249	246

When a mobile robot is in a detectable range, it can estimate the distance according to the signal strength value indicated in Table 1. Fig. 1 shows features of the RSSI depending on the distances. We suggest finding the real coordinates using the average strength of the RF signal for each node as shown in Table 1. Actually, it is difficult because strength is not a gap yet, as shown in Fig 1.

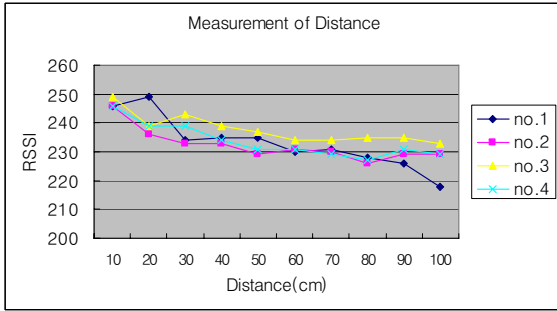


Fig. 1 Features of the RSSI

### 3. Movement of the Mobile Robot

#### 3.1 Heading of the robot

The robot communicates with nodes in the network locally, and makes navigation decisions based on which node it is near.

The mobile robot decides its movement according to the following procedure: The robot

1. Measures the exact distance through the signal strength (RSSI).
2. Calculates its own coordinates in the strength range among the nodes.
3. Moves based on the measured distance.
4. Calculates the moving distance by using two coordinates.
5. Estimates the real coordinates of itself by using distance.
6. Decides the vector direction from the angle( $\theta$ ) for the origin.
7. Decides the vector direction (heading) from the angle( $\theta'$ ) for the origin.
8. Moves to the destination.

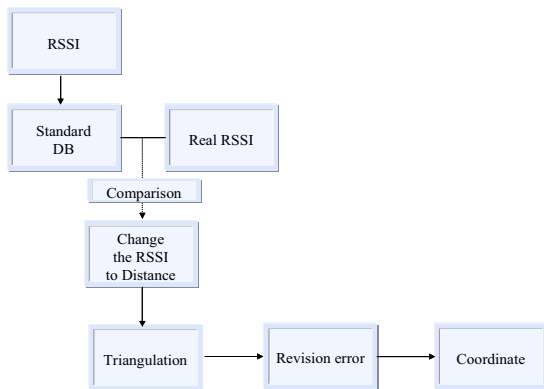


Fig. 2 The flowchart of the coordinates measurement using a RSSI

Figure 3 and 4 show how to decide the heading direction of the mobile robot. The right part of Fig. 3 represents the distance between two nodes according to the coordinates of the robot where  $\theta$  is the angle between the heading direction of the robot and the x axis of the coordinate. After controlling

the angle  $\theta$ , the robot decides to move to the next position. Equations (5) and (6) describe the distances in the x and y axes and  $\theta$ , respectively.

$$(x - a)^2 + (y - b)^2 = r_1^2 \quad (1)$$

$$(x - c)^2 + (y - d)^2 = r_2^2 \quad (2)$$

$$a' = \frac{-(a - c)}{(b - d)}$$

$$b' = -\frac{(r_1^2 - r_2^2) - (a^2 - c^2) - (b^2 - d^2)}{2(b - d)}$$

$$(1 + a'^2)x^2 + (-2a + 2a'(b' - b))x + (a^2 + (b' - b)^2 - r_1^2) = Ax^2 + Bx + C = 0$$

$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \quad (3)$$

$$y = \frac{-(a - c)x - (r_1^2 - r_2^2) - (a^2 - c^2) - (b^2 - d^2)}{(b - d) - 2(b - d)} \quad (4)$$

$$d_1 = |x_1 - x_2|, d_2 = |y_1 - y_2| \quad (5)$$

$$\theta = \arctan \frac{d_2}{d_1} \quad (6)$$

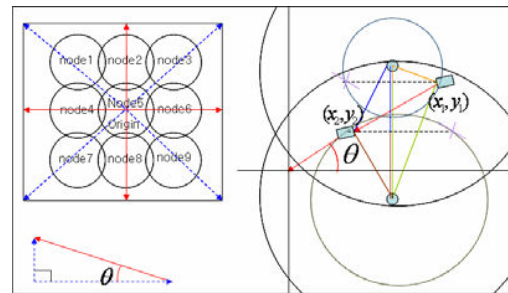


Fig. 3 Finding the heading by the vector angle's gap

The heading of the robot is determined by using the angle of the first coordinate and the angle  $\theta'$  of the second coordinate under the assumption that the goal node is known. We can calculate the coordinates by using Eq. (1) and Eq. (2). The first angle  $\theta$  and the second angle  $\theta'$  are calculated by Eq. (5) and Eq. (6).  $\theta'$  decides the heading direction of the robot for the goal position.

#### 3.2 The algorithm for finding the shortest path

The shortest path is defined by sending a packet from the first-source to the destination with minimum cost. The shortest path presents a relatively lower link where the link

values show a transmission capacity and traffic load. Since the proposed method for the shortest path is very simple and easy to scale, a mobile robot can find a shortest path to its destination wherever it is. If more than two routes are created, as shown in Fig. 5, the patterns are constructed by the following steps:

1. Give the node number.
- 2a. If the number of the destination > the node number of the starting point, 1)Compare the next nodes, 2)Select the biggest node, 3)using this number, then, move to the destination
- 2b. If the number of the destination < the node number of the starting point, a robot moves to find the lowest node number.

#### 4. Experimental results

The parameters of the experiment with 5 mote sensors are shown for the shortest path planning of the robot as below.

1. The radius of the mote: 1m~1.5m
2. The dimension of the environment: 3 × 3m
3. The load: a base mote on the robot

It is difficult to measure distance according to signal strength because of specific features of RF signal. So the signal strength is set to the lowest value (3dbm) to reduce the refraction ratio of the RF signal. In addition, the distance according to the signal strength value makes the area limited (about 2m~3m). We can get the results shown in Table 1 which are applied to the experiment where any local sensor for finding a robot heading is used. The experimental apparatus consists of four mote sensors and one gateway mote. The distance information, through the signal strength of each node, is gathered through several experiments and the DB is used to reduce errors. The location of the robot is estimated by computing the distance from each node with the on-line data from the RSSI and the DB.

As shown in Table 2, the differences between the real coordinates and the measured coordinates are presented as errors. To measure its coordinates, a mobile robot needs to gather the RSSI values from each anchor node by a triangulation method. However, as shown in Table 2, the error between the gathered coordinates and the real coordinates is not negligible.

Table 2 Error from the real-coordinates

Real coordinate	Measured coordinate	Error
(0, 30)	(69, 69)	(69, 39)
(70, 80)	(71, 72.5)	(1, -7.5)
(130, 180)	(48, 90)	(-82, -90)

To compensate for the error, we propose a new way:

1. When the sum of the two data elements (data of one node is the two RSSI values ) is below 200, a robot chooses the largest one. On the other hand, when the sum is over 200,

it takes the average value of the two data elements.

2. If there is comparable data from one point (the coordinates of the point on the plane) at the same time, it takes the average value of the sum of the data.

3. If the error value is positive for each of the x and y coordinates, the measured coordinates are subtracted from the average error. In the opposite case, its measured coordinates are added to the average error.

Table 3 presents the revised error results where the values of the coordinates of the revised errors are reduced in comparison with the previous results.

Table 3 The revised errors

Real coordinate (cm)	Measured coordinate	Error	Revision coordinate	Revised error
(0, 30)	(69, 69)	(69, 39)	(6.5, 25)	(6.5, -5)
(70, 80)	(71, 72.5)	(1, -7.5)	(74.5, 72)	(4.5, -8)
(130, 180)	(48, 90)	(-82, -90)	(157.5, 154.5)	(27.5, -25.5)

Table 4 The angular errors in the real coordinates

Angle	Real coordinate	Before coordinate revision	After coordinate revision	Before error revision	After error revision
$\theta$	35.5°	60.3°	34.7°	-24.8°	0.8°
$\theta'$	42.7°	44.7°	45.6°	-2°	-2.9°

Table 4 represents the angular errors according to the real coordinates where  $\theta$  and  $\theta'$  are the angles for the first coordinates and the second coordinates with x=0 and y=0, respectively. The proposed method for revision shows the desirable results as shown in Table 4. However, in the case of  $\theta'$ , the angle of the previous revision error is better than that after the revision error angle, because the proposed way of revising with the average error was applied to this system. Although the revised error angle is bigger than the previous one, the result is also a close value to the real coordinate angle.

Fig. 11 represents the decision of the coordinate using the grid method. Based on the assumption that the coordinates of a mobile robot should be in accord with the DB coordinates, a mobile robot estimates the approximate coordinates to calculate the angle several times when moving.

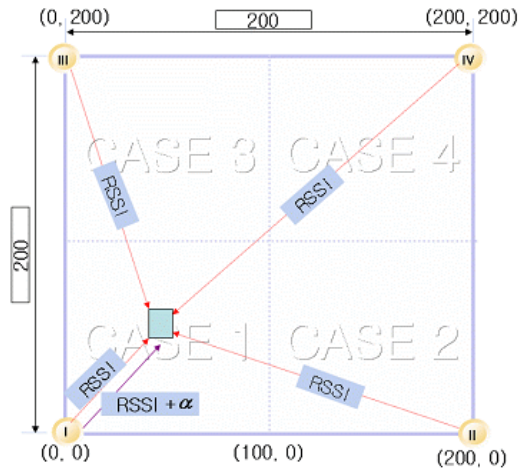


Fig. 4 Decision of the coordinate with the grid method

A robot updates the coordinates to enhance accuracy. In addition, a robot calculates the accurate heading angle to move to a destination by using shortest path rules. Table 5 shows the results of the grid method.

Table 5 Results of the grid method

After revision error coordinate	Grid coordinate	Anchor node coordinate
(6.5, 25)	(74.5,72)	(130, 110)
(74.5, 72)	(130, 110)	(200, 200)
$\theta$	34.7°	$\theta'$
		34.4°
		$\omega$
		52.1°

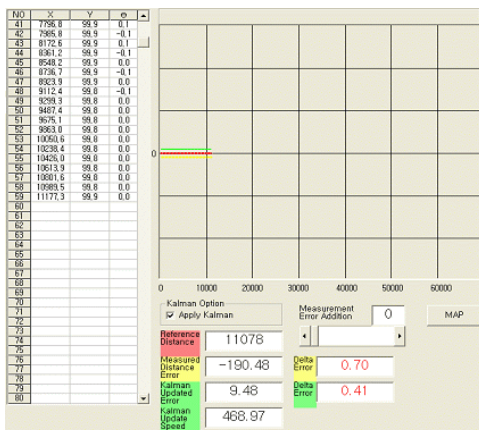


Fig. 5 The application of the Kalman filter

In this Fig. 5, the red, yellow, and green lines are the reference, the measured encoder, and the Kalman filter application, respectively. As a result of the experiment, when the process applies the Kalman filter, the error is close to zero. In the experiment, the Kalman filter is applied to reduce the error for a straight line interval. However, it cannot be used in the case of RF communication which is subject to noise.

## 5. Conclusions

When the mobile robot moves several times for reducing error and if the robot's coordinates accord with a grid DB, it has accurate coordinates by using an update. To avoid repetitive movement of a mobile robot for finding its coordinates in accord with the grid DB, we measure using a RSSI of the Grid DB for each value (below 10cm) of the coordinates of the experimental environment. The proposed method has some advantages over conventional approaches. First, when an event happens, it becomes available for any robot in the constructed environment. Second, controlling the algorithm is simple making it possible to be improved for a dynamic environment. Finally, we have presented the way of finding coordinates and the heading direction of a mobile robot using signal strength values without any local sensor.

## References

- Roy Want, Andy Hopper, Veronica Falcao, and Jon Gibbons, "The active badge location system." Transactions on Office Information System (TOIS), Vol. 10. No. 1, pp.91-102, Jan. 1992.
- Mica-Z Wireless Measurement System Datasheet, in [http://www.xbow.com/Products/Product\\_pdf\\_files/Wireless\\_pdf/MICAZ\\_Datasheet.pdf](http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAZ_Datasheet.pdf).
- Hyper\_Tmote\_system Datasheet, in <http://www.hybus.net/>.
- M.A. Batalin, G.S. Sukhatme, M. Hattig, "Mobile robot navigation using a sensor network", ICRA '04., Vol. 1, pp. 636-641, 2004.
- V. Kumar, D. Rus, and S. Singh, "Robot and Sensor Networks For First Responders", PERVASIVE computing pp. 24-33, Oct. 2004.
- A. Tsertou, R. Upadhyay, D. Laurenson, S. McLaughlin, "Towards a tailored sensor network for fire emergency monitoring in large buildings" Proceedings of the 1st IEEE International Conference on Wireless Emergency and Rural Communications, Rome, Italy, 30 Sept - 2 Oct. 2007.
- Andy Harter, Andy Hopper, Pete Steggle, AnyWard, and Paul Webster., "The anatomy of a contextaware application." Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom 1999), pp. 59-68, Seattle, WA, Aug. 1999.
- Y. Fukuju, M. Minami, H. Morikawa and T. Aoyama, "DOLPHIN: An Autonomous Indoor Positioning System in Ubiquitous computing Environment", Proceedings of the IEEE Workshop on Software Technologies for Future Embedded Systems, Hakodate, Japan, May 2003.