논문 2008-03-14

Map Building and Localization Based on Wave Algorithm and Kalman Filter

Dilshat Saitov, Jeong Won Choi, Ju Hyun Park, Suk Gyu Lee*

Abstract : This paper describes a mapping and localization based on wave algorithm[11] and Kalman filter for effective SLAM. Each robot in a multi robot system has its own task such as building a map for its local position. By combining their data into a shared map, the robot scans actively seek to verify their relative locations. For simultaneous localization the algorithm which is well known as Kalman Filter (KF) is used. For modelling the robot position we wish to know three parameters (x, y coordinates and its orientation) which can be combined into a vector called a state variable vector. The Kalman Filter is a smart way to integrate measurement data into an estimate by recognizing that measurements are noisy and that sometimes they should ignored or have only a small effect on the state estimate. In addition to an estimate of the state variable vector, the algorithm provides an estimate of the state variable vector uncertainty i.e. how confident the estimate is, given the value for the amount of error in it.

Keywords : Mobile robot, simulation, wave algorithm, navigation, mapping, localization

I. Introduction

An autonomous mobile robot works well if it is equipped with various kinds of expensive sensors, but it may prove to be too expensive and have less fault tolerance. Since multi-robot systems are characterized by distributed control, autonomy, enhanced fault tolerance, and communication, we decided to use a team of mobile robots for building a map in our simulation. Each robot is assigned a task such as building a map of its local position, and the robots combine their data into shared maps. The robots coordinate their exploration strategies to maximize the efficiency of their exploration using these shared maps. This algorithm of navigation and mapping was named "Wave algorithm".

* Corresponding Author
Received : 2008. 4. 23., Accepted : 2008. 6. 13.
Lee Suk Gyu : Yeungnam univ.

There are many approaches are used for defining local position of a robot. One of them is SLAM (Simultaneous Localization and Mapping).Since, there is the mapping and localization algorithm – Wave Algorithm, we decided to use Kalman Filter for simultaneous localization. Thus, we tried to make our own SLAM algorithm in which Wave Algorithm and Kalman Filter are combined.

II. Wave Algorithm

Now we will discuss the base algorithm in our research which is called wave algorithm [4,5]. The main idea of this exploration strategy is to drive to the closest location where the robot can gather information about a cell that has not been sufficiently explored. We show these strategy explanations on the following figure. It provides short trajectories for single robot exploration tasks. First, a robot explores the area around itself and then seeks a minimum distance to the next point.

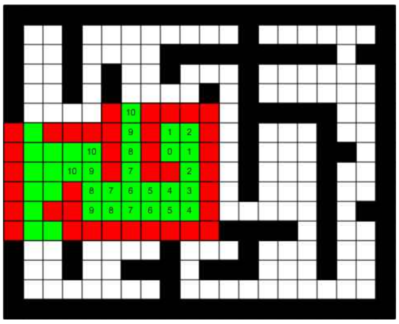

Fig. 1. A simulation of the wave algorithm

In Fig. 1, a robot staying at cell number 0 checks all available cell around it and marks each cells from 1 to n. Then robot goes to the lowest number and checks available cells again.

Table 1. The pseudo code of the Wave Algorithm

```
wM [ cX, cY ] = 0;
i = 0;
exist = false
 do
{
 exist = false;
  for (x = 0; x < width; x ++)
    for (y = 0; y < width; y ++)
      if (dM [x, y] is opened and free)
     and
          wM [x, y] is unassigned and
          neighbor of wM [x, y] is i)
    then (wM [x, y] = i +1; exist = true;)
   i = i + 1;
 }
while  (exist);
```

III. Previous Research and Results

In our previous research 11], we used the wave algorithm with directional priority. After that, we compared our results with the results of [10]. Let us consider in general, the algorithm used in [10] and then more deeply ours, to show efficiencies and inefficiencies of each algorithm. So, we considered 3 methods for map building according to [10]: the uncoordinated, coordinated and optimized methods.

3.1 Uncoordinated method

Regarding [10], the authors used value iteration which is a kind of dynamic programming algorithm. The purpose of the method is the determination of the cost of reaching the current frontier cells. In their approach, the cost for traversing a grid cell (x, y) is proportional to its occupancy value $P(occ_{xy})$.The minimum-cost path is computed using the following two steps.

3.1.1 Initialization

The grid cell that contains the robot location is initialized with 0, all others with ∞ [10] :

$$V_{x,y} \leftarrow \begin{cases} 0, & \text{if } \langle x,y \rangle \text{ is the robot position} \\ \infty, & \text{otherwise} \end{cases} \quad (1)$$

Here, $V_{x,y}$ is value of a grid cell.

3.1.2 Update loop

For grid cells (x, y), the update loop has the form of equation (2)

$$V_{x,y} \leftarrow \min_{\substack{\Delta x = -1,0,1 \\ \Delta y = -1,0,1}} \left\{ V_{x+\Delta x, y+\Delta y} + \sqrt{\Delta x^2 + \Delta y^2} \times P(occ_{x+\Delta x, y+\Delta y}) \right\} \quad (2)$$

Value iteration updates the value of all grid cells by the value of their best neighbors, plus the cost of moving to this neighbor. Cost is here equivalent to the probability $P(occ_{x,y})$ that a grid cell is occupied times the distance of the cell.

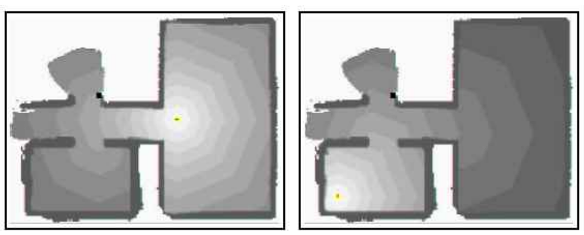

Fig .2 Typical value functions obtained for two different robot positions. The black rectangle indicates the target points in the unknown area with minimum cost.

3.2 The coordinated method

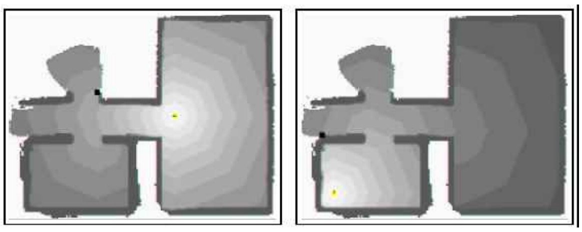

Fig. 3 Target positions obtained using the coordination approach. In this case the target point for the second robot is to the left in the corridor

Fig. 3 depicts the efficiency of the coordination technique [8]. According to this method, two robots with the same costs must choose different target cells. However in the uncoordinated technique, robots would choose the same target position (see Fig.2).So, regarding their simulation, the robot on the left picture chose the nearest point for further investigation. Since robots exchange data permanently, robot 2 in the right picture, exactly knows the chosen destination of robot 1 and chooses the other nearest point.

3.3 The optimized method

The last considered approach in [10] was the optimized method. For instance, consider the situation depicted in Fig.4. Two robots are investigating a corridor in two adjacent rooms. The grey area has already been investigated. The assignment resulting from an application of their algorithm is depicted in the left image of this figure. Suppose both target points a and b have the same utility. Then in the first round the algorithm assigns robot 2 to a since this assignment has the least cost of all other possible assignments. Accordingly, in the second round, 1 is assigned to b. If we assume that both robots require the same amount of time to explore a room, this assignment is clearly sub-optimal. A better assignment is shown in the right image of Fig. 4.

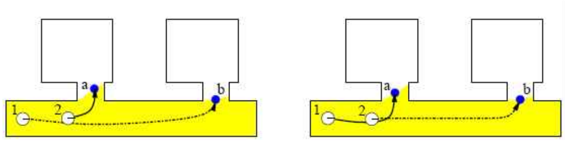

Fig .4 The trajectories depicted in the left image that result from algorithm 1 are sub-optimal. If robot 1 moves to point a and robot 2 moves to the location b as illustrated in the right figure, the time needed to finish the exploration task is reduced, since the maximum time needed to reach the rooms is lower.

By directing robot 1 to the left room and robot 2 to the right room, the whole team can finish the job earlier, because the time required to reach the rooms is reduced. As already mentioned above, one approach to overcome this problem is to consider all possible combinations of target points and robots. The authors [10] want to minimize the trade-off between the utility of frontier cells and the distance to be traveled. However, just adding the distances to be traveled by the two robots does not make a difference in situations like that depicted in Fig. 4 To minimize the completion time we therefore modify the evaluation function so that it considers squared distances to choose target locations

$$\arg\max_{(t_1,\ldots,t_n)} \sum_{i=1}^{n} [U(t_i | t_1,\ldots,t_{i-1},t_{i+1},\ldots,t_n) - \beta \times (V_{t_i}^{i})^2] \quad (3)$$

Here, $t_1 \cdots t_n$ are already investigated cells, U is utility of frontier cell t_n , V is the cost for obtaining frontier cell, i is a concrete robot, β is importance of U versus V.

Ⅳ. The wave algorithm with directional priority

In this section, we consider the algorithm used in [11], in other words our proposed algorithm. In general, the wave algorithm also uses costs and frontier cells when choosing the best target point just as the uncoordinated algorithm [10]. But instead of the “Optimized method” described in section 4.3, we assume that each robot has its own direction on the plane. This means that the probability of their choosing the same point is very low from early on. Notice that in [11] the communicational range was assumed as unlimited.

Fig. 5 depicts four different environments for simulations.

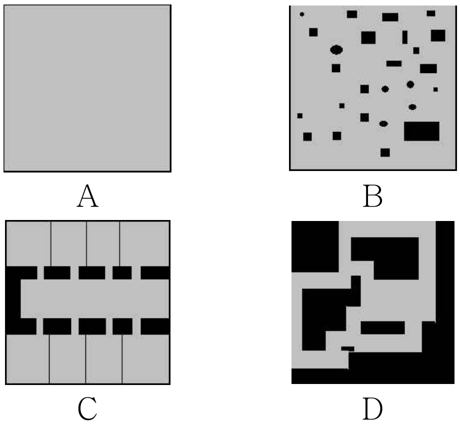

Fig.5 Maps used for the simulation experiments
A) An empty environment
B) An unstructured environment
C) An office environment
D) A corridor environment

The following screen shots of simulations are based on the represented algorithm described in this paper. We may see 9 robots with the same range for their sensors. The size of the map is 200*200 pixels.

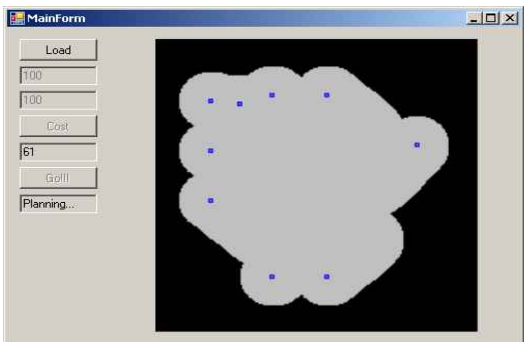

Fig. 6 Exploration of an empty environment

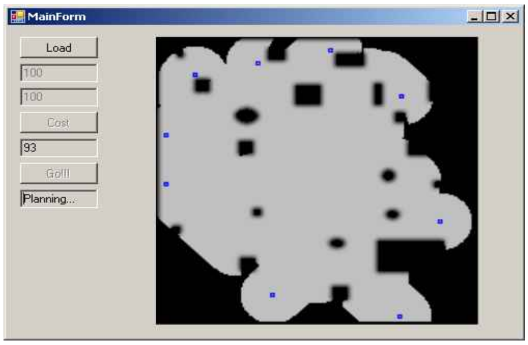

Fig. 7 Exploration of an unstructured environment

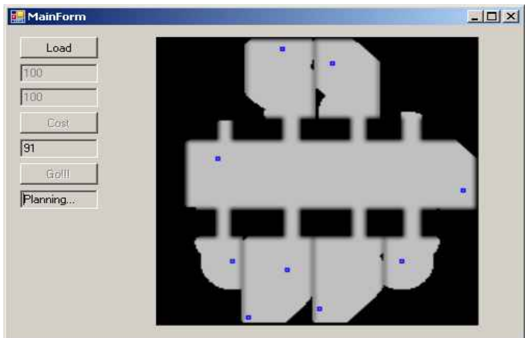

Fig. 8 Exploration of an office environment

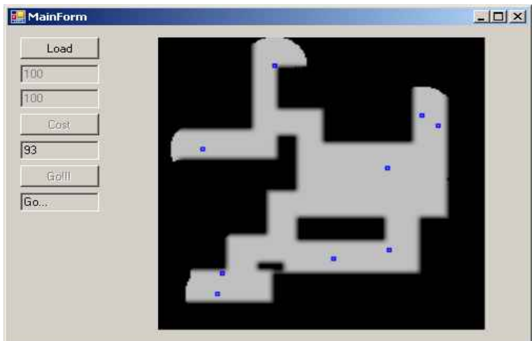

Fig. 9 Exploration of a corridor environment

For accurate mapping we consider each explored part of a map by a robot as a separated matrix [6]. Let us assume that A and B are two positive real numbers. A * B map is a function

$$m : [0, A] * [0, B] \rightarrow R \tag{4}$$

In equation 4, A and B represent rows and columns of the matrix, respectively. The next step is the definition of a transformation used to try different relative placements of two maps to find a good merging. We assume that the location of a point in the plane is expressed in homogeneous coordinates, i.e., the point (x, y) is represented by the vector [x y 1]T, where the trailing upper T indicates the transpose operation.

Let t_x, t_y, and ϕ be three real numbers. The transformation associated with t_x, t_y and ϕ is the function

$$Tt_x, t_y\varphi(x, y): R^2 \rightarrow R^2 \qquad (5)$$

defined as follows.

$$Tt_x, t_y\varphi(x, y): \begin{bmatrix} \cos\varphi & -\sin\varphi & t_x \\ \sin\varphi & \cos\varphi & t_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad (6)$$

Here, the transformation given in (5) corresponds to a rotation about the origin of the point (x, y) of ϕ, followed by a translation of (t_x, t_y). In the following sections of this paper, we see in the results, that the mapping algorithms produce occupancy grids where rotation and translation transformations are seemingly sufficient for merging real-world data.

For building a shared map using two small maps, the best overlapping between these two maps is needed. Let m_1 and m_2 be two maps in I_{A*B}. The overlapping between maps can be described as the following equation

$$\omega(m_1, m_2) = \sum_{i=0}^{A-1}\sum_{j=0}^{B-1} Eq(m_1[i,j]; m_2[i,j]) \qquad (7)$$

where Eq (a; b) is one when a = b and zero otherwise. The overlapping function ω measures how much the two maps agree.

V. Localization in multi-robot system

As previously said to model robot position we need x and y coordinates and its orientaion. These three parameters can be combined into a vector called a state variable vector. The state variable vector is usually denoted X. For calculating this vector at any time interval we use the equation:

$$\begin{pmatrix} x \\ y \\ \theta \end{pmatrix}_{k+1} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix}_k + \begin{pmatrix} \triangle & 0 & 0 \\ 0 & \triangle & 0 \\ 0 & 0 & \triangle \end{pmatrix}\begin{pmatrix} \nu_x \\ \nu_y \\ \nu_\theta \end{pmatrix}_k \qquad (8)$$

$$X_{k+1} = AX_k + Bu_k$$

The state variable vector summarizes the relevant information from the system at any time and describes how the system changes as a function of time and input. The k and k+1 subscripts represent the time of the vector. In this case, Bu is the input that we receive due to the robot walking (distance = speed x time) and A is the identity matrix. In general, u is the input vector, A is a matrix relating the state variables at the previous time step to the state variables at the current time step and B is a matrix relating the input to the state variables.

If we have a measurement of the x and y positions and orientation we could write these measurements in the form:

To generalize any system which is finite dimensional, causal (the output does not depend on future input values) and time invariant can be described by a set of variables which constitute the important aspects of the system's internal behaviour. These variables can be combined into a vector which at any time contains the information required to characterize the system. This type of model is known as a state space model and is commonly used in control system modelling and signal processing. The equations above represent a linear, discrete time state space model.

There are some SLAM steps:

First of all we have to define robot's initial position as the root of the world coordinate space or start with some preexisting features in the map with high uncertainty of the robot position. The following step in SLAM is prediction. When the robot moves, motion model provides new estimates of its new position and also the uncertainty of its location –positional uncertainty always increases. After prediction follows measurement. It consists of two parts: (a) adding new features to map, (b) re-measuring previously added features. The last step is repetition of prediction and measurement.

Thus, we combined the navigation and mapping algorithm, also known as wave algorithm and Kalman Filter.

Following figures depict the simulation of this combination.

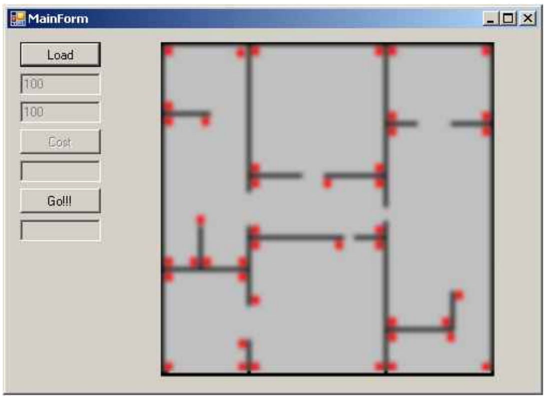

Fig. 10 Loaded map with fixed artificial landmarks.

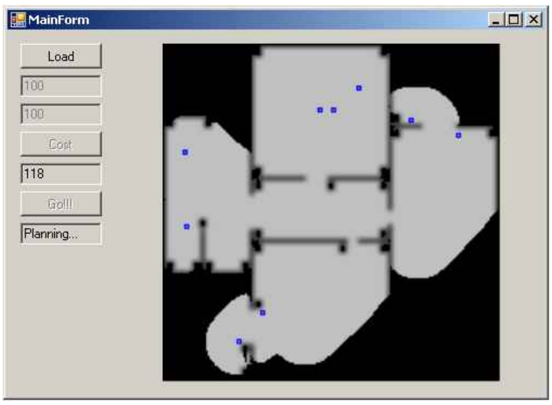

Fig. 11 Simulation of the combined algorithms

Fig. 10 depicts our program with loaded map. There are number of points which are fixed. These points are artificial landmarks. By referring to these landmarks, robot updates, makes measurements, predicts its own position.

VI. Conclusions

We have proposed a new approach to the multi robot simultaneous localization and mapping problem, which enables groups of robots to explore a single environment map. It consists of two parts: wave algorithm and Kalman filter, which are responsible for navigation and localization respectively. Wave algorithm is based on defining and directing a robot to a closest unknown point on a map. Kalman filter is based on modeling state variable vector, which depends on x, y coordinates and orientation of the robot.

So, we combined these two algorithms and found our approach to be highly reliable to navigate and localize robots positions. According to our simulation results, this approach might be much more useful for map building in real-world in multi-robot systems.

References

[1] G. Dissanayake, P. Newman, S. Clark, H. Durrant Whyte, and M. Csorba, "B.A solution to the simultaneous localization and map building (SLAM) problem", IEEE Trans. Robot. Autom., vol. 17, no. 3, pp. 229-241, Jun. 2001.

[2] J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai, "B.A practical, decision theoretic approach to multi robot mapping and exploration", Proc. IEEE RSJ Int. Conf. Intelligent Robots and Systems (IROS), pp. 3232-3238, 2003.

[3] G. W. Pulford, B.F. La Scala, "Map estimation of target maneuver sequence with expectation maximization algorithm", IEEE Trans. on Aerospace and Electronic Systems, vol. 38, no. 2, Apr. 2002.

[4] S. Williams, G. Dissanayake, and H. Durrant Whyte, "Towards multi vehicle simultaneous localization and mapping", Proc. IEEE Int. Conf. Robotics and Automation (ICRA), 2002, pp. 2743 - 2748, 2002.
[5] B. Yamauchi, "Frontier based exploration using multiple robots", Proc. of the Second International Conference on Autonomous Agents, 1998.
[6] A. Birk, and S. Carpin, "Merging occupancy grid maps from multiple robots", IEEE Trans. Robot. Autom., vol. 9 , pp. 1384 - 1397, Jul. 2006.
[7] S. Roumeliotis G. Bekey, "Distributed multirobot localization", IEEE Trans. Robot. Autom., vol. 18, no. 5, pp. 781 - 795, Oct. 2002.
[8] W. Burgard, M. Moors, and F. Schneider, "Collaborative exploration of unknown environments with teams of mobile robots", IEEE Trans. Robot. Autom., vol. 18, no. 5, pp. 781 - 795, Oct. 2002.
[9] C. Stachniss, O.M. Mozos, and W. Burgard, "Speeding up multi robot exploration by considering semantic place information", IEEE Trans. Robot. Autom., vol. 18, no. 5, pp. 581 - 587, May 2006.
[10] C. Stachniss, O.M. Mozos, and W. Burgard, "Coordinated multi robot exploration", IEEE Trans. Robot., vol. 21, no 3, pp 376-386, 2005.
[11] D. Saitov, U. Umirov, S.G. Lee, and J.H. Park, International Symposium on Mechatronics and Automatic Control Section. (ISMAC), pp. 116-122 Oct. 2007.

Biography

Dilshat Saitov

2005 : BS degree in Telecommunication Dept, TUIT, Tashkent.
2008 : MS degree in Dept. of Electrical Eng. Yeungnam Univ.
EMAIL : dilshat_saitov@yahoo.com

Jeong Won Choi

1995 BS degree in Dept. of Electrical Eng Yeungnam Univ.
1997 MS degree in Dept. of Electrical Eng Yeungnam Univ.
2002 Ph.D. degree in Dept. of Electrical Eng Yeungnam Univ.
2003~2006 Researcher, STX Heavy Industries Co.
Since 2006, Professor, School of Electronic Eng Kumoh National Institute of Technology.
EMAIL : consys@korea.com

Ju Hyun Park

1990 BS degree in Kyungpook National Univ.
1992 MS degree in Kyungpook National Univ.
1997 Ph.D. degree in POSTECH.
1997-2000 Researcher at ERC-ARC, POSTECH.
Since 2000, Professor, Yeungnam Univ.
EMAIL : jessie@ynu.ac.kr

Suk Gyu Lee

1979 BS degree in Dept. of Electrical Eng. Seoul National Univ.
1981 MS degree in Dept. of Electrical Eng. Seoul National Univ.
1990 Ph.D degree in Dept. of Electrical Eng, Univ. of California, Los Angeles.
Since 1982, Professor, Dept. of Electrical Eng. Yeungnam Univ.
EMAIL : sglee@ynu.ac.kr