Adaptive Occupancy Grid Mapping With Clusters

Byoung-Gi Jang and Tae-Yong Choi and Ju-Jang Lee
Electrical Engineering and Computer Science
Korea Advanced Institute of Science and Technology(KAIST)
373-1, Guseong-dong, Yuseong-gu, Daejeon, 305-701, Korea
ee185zzang@kaist.ac.kr, ermzace@odyssey.kaist.ac.kr, jjlee@ee.kaist.ac.kr

Abstract

In this paper, we describe an algorithm for acquiring occupancy grid maps with mobile robots. The standard occupancy grid mapping developed by Elfes and Moravec in the mid-Eighties decomposes the high-dimensional mapping problem into many one-dimensional estimation problems which are then tackled independently. Because of the independencies between neighboring grid cells, it often generates maps that are inconsistent with the sensor data. To overcome it, we propose the cluster which is a set of cells. The cells in the clusters are tackled dependently with another occupancy grid mapping with EM algorithm. The occupancy grid mapping with EM algorithm yields more consistent maps, especially in the cluster. As we use mapping algorithm adaptively with clusters according to the sensor measurements, our mapping algorithm is faster and more accurate than the previous mapping algorithms.

Keywords : occupancy grid, mobile robotics, mapping, Bayes rule, cluster

1 Introduction

Robotic mapping has been a highly active research area in robotics and AI for a few decades. Robotic mapping addresses the problem of acquiring spatial models of physical environments through mobile robots. There are a number of mapping algorithms. However, the occupancy grid mapping is more popular than others, since it has the reputation of being extremely robust and easy to implement. Once mapped through occupancy grid mapping, they enable various key functions necessary for mobile robot navigation, such as localization, path planning, collision avoidance, and people finding.

Occupancy maps have been built using various sensors, such as sonar sensor, laser range finders, and stereo vision, etc. However, all these sensors are subject to errors often referred to as measurement noise. In addition, sonar sensors cover an entire cone in space and form a single sonar measurement it is impossible to say where in the cone the object is. The sonar sensors are also sensitivity to the angle of an object surface relative to the sensor and the reflective properties of the surface. The above properties of sensors make a mapping problem be difficult and lead inconsistent map.

The occupancy grid mapping resolves such problems by generating probabilistic maps. As the name suggests, occupancy grid maps are represented by girds. Namely, they decompose the high-dimensional mapping problem into many one-dimensional estimation problems which are then tackled independently. Because of the independency of neighboring cells, they often generate maps that are inconsistent with the data, particularly in cluttered environments.

To overcome it, we define the cluster which is a set of cells. The cluster is the region that has the high probability to be inconsistent with the sensor data when the standard occupancy grid mapping is used. Existing occupancy grid mapping algorithms do the task with the emphasis on individual cells. However, our approach maps with the emphasis on clusters. As making the cluster and choosing the optimal mapping algorithm according to the sensor measurements, maps generated by our approach more accurate than ones generated by the previous occupancy grid mapping algorithm. Our mapping algorithm is also as fast as the standard occupancy grid mapping algorithm.

2 Standard Occupancy Grid Mapping

The Standard occupancy grid mapping approach (Elfes, 1989; Moravec, 1988)[2][3] constitutes two algorithms mainly. First, it decomposes a multidimensional (typically 2D or 3D) tessellation of space into many independent cells. Second, each cell calculates a probabilistic estimate of its state. To calculate this estimate, techniques such as Bayesian reasoning are then employed on the grid cell level. And each cell is tackled independently.

Let m be the occupancy grid map. The grid cell has the index $\langle x, y \rangle$ to store a probabilistic occupancy, which is $m_{x,y}$. Occupancy grid maps are estimated from sensor measurements. Let $z_1, \cdots, z_T$ denote the measurements from time 1 through time T. The measurement is composed of a sonar scan and the robot pose at which the measurement was taken. The robot pose which is assumed to be known is xy coordinates of the robot and heading direction. Each measurement carries information about the occupancy of many gird cells. Thus, the problem addressed by occupancy grid mapping is the problem of determining the probability of occupancy of each grid cell $m_{x,y}$ given the measurements $z_1, \cdots, z_T$.

$$p(m_{x,y} \mid z_1, \cdots, z_T) \tag{1}$$

For computational reasons, it is common practice to calculate the *log-odds* instead of estimating the above posterior. The *log-odds* is defined as follows.

$$l^T_{x,y} = \log \frac{p(m_{x,y} \mid z_1, \cdots, z_T)}{1 - p(m_{x,y} \mid z_1, \cdots, z_T)} \tag{2}$$

The assumption in standard occupancy grid mapping is the static world and conditional independence given knowledge of each individual grid cell $m_{x,y}$. Two assumptions and Bayes rule allow us to simplify the posterior to following:

$$\begin{aligned} &p(m_{x,y} \mid z_1, \cdots, z_t) \\ &\quad = \frac{p(m_{x,y} \mid z_t) p(z_t) p(m_{x,y} \mid z_1, \cdots, z_{t-1})}{p(m_{x,y}) p(z_t \mid z_1, \cdots, z_{t-1})} \end{aligned} \tag{3}$$

Let $\overline{m}_{x,y}$ be freeness of the grid cell. The probability of the freeness of grid cell can be calculated as same way.

$$\begin{aligned} &p(\overline{m}_{x,y} \mid z_1, \cdots, z_t) \\ &\quad = \frac{p(\overline{m}_{x,y} \mid z_t) p(z_t) p(\overline{m}_{x,y} \mid z_1, \cdots, z_{t-1})}{p(\overline{m}_{x,y}) p(z_t \mid z_1, \cdots, z_{t-1})} \end{aligned} \tag{4}$$

By dividing (3) by (4) and adapting logarithm, the desired *log-odds* is expressed as follow:

$$l^t_{x,y} = \log \frac{p(m_{x,y} \mid z_t)}{1 - p(m_{x,y} \mid z_t)} + \log \frac{1 - p(m_{x,y})}{p(m_{x,y})} + l^{t-1}_{x,y} \tag{5}$$

Finally, the desired posterior occupancy probability $p(m_{x,y} | z_1, \cdots, z_T)$ can be recovered from the log-odds representation of the map.

Standard occupancy grid mapping does not take the occupancy of neighboring cells into account. It makes the crucial independence assumption that the occupancy of a cell can be predicted regardless of a cell's neighbors. Herein lies a major problem of the standard occupancy approach. This leads to incorrect map.

3 Adaptive Occupancy Grid Mapping With Clusters

This section presents an algorithm to improve the problems of the previous occupancy grid mapping. A key idea is adapting the cluster which is a set of cells. The cells in the cluster mean that they have the high probability to be inconsistent with the sensor data when the standard occupancy grid mapping is used. Unlike existing occupancy grid mapping algorithm, our approach does the mapping with the emphasis on the clusters. One cluster doesn't affect the others, since the cluster is independent each other. The occupancy of the cells in the cluster is calculated with the occupancy grid mapping proposed by Thrun in 2003[1]. Using Expectation Maximization algorithm, in short EM, the alternative mapping algorithm solves the mapping problem as maintaining the dependencies between neighboring cells. Hence, it leads to the more accurate maps than the standard occupancy grid mapping in the cluster. The clusters are made with the neural networks[4][5][6] which is a powerful tool in pattern recognition.

To make the cluster, we use the neighboring sensor measurements which are the input of neural networks.

$$P = [p_1, \cdots, p_R] \tag{6}$$

R is the number of the sensor measurements used. The output of neural networks, y, is $'1'$ if the region swept by the sensors is cluttered or erroneous place. Otherwise y is $'0'$. That is, if y is $'1'$, we assemble the cells in that region and make a new cluster.

The occupancy of cells out of cluster is calculated with the standard occupancy grid mapping algorithm explained in section 2. The binary occupancy of cells in the cluster is calculated with the alternative occupancy grid mapping proposed by Thrun.

Let K_i the number of obstacles in the sensor cone of the i-th measurement. Let $D_t = \{d_{t,1}, \cdots, d_{t,K_t}\}$ denote the distances to these obstacles and ordered in increasing order. To describe the multiple causes of a sensor measurement z_i, the new variables, called correspondence variables, are defined as follow:

$$c_t = \{c_{t,*}, c_{t,0}, c_{t,1}, \cdots, c_{t,K_t}\} \tag{7}$$

Each of these variables corresponds to exactly one cause of the measurement z_t. If $c_{t,k}$ is 1 for $1 \leq k \leq K_t$, the measurement is caused by the k-th obstacle. If $c_{t,0}$ is 1, none of the obstacles were detected and the sensor returns a max-range reading. The random variable $c_{t,*}$ corresponds to the case where a measurement was purely random. The log-likelihood of all data and correspondences is written as follows:

$$\log p(Z, C|m) = \sum_t \log p(z_t, c_t|m) \tag{8}$$

Here Z denotes the set of all measurements and C is the set of all correspondences c_t for all data. Not calculating the probability of the correspondence variables but Maximization the likelihood of the data is important, since the probability of correspondence variables is unobservable. This is achieved by maximizing the expected log-likelihood $E[\log p(Z, C|m)|Z, m]$, where the expectation is taken over the correspondence variables C. The expected log-likelihood can be obtained as follows:

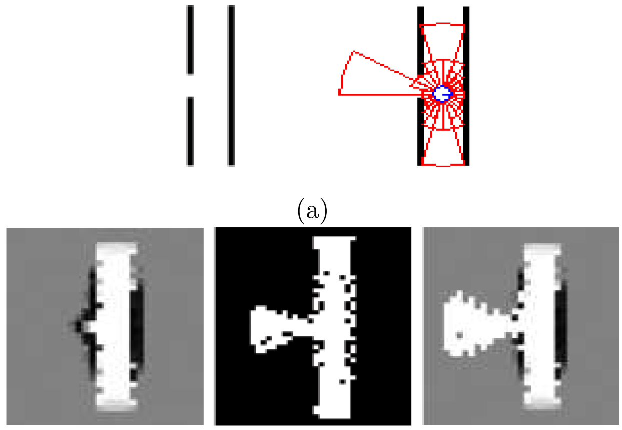
$$\begin{aligned}
&E[\log p(Z, C|m)|Z, m] \\
&\quad = \sum_t [E[\log p(c_t)|z_t, m] + \log \frac{1}{\sqrt{2\pi\sigma^2}} \\
&\quad -\frac{1}{2}[E[c_{t,*}|z_t, m] \log \frac{z_{max}^2}{2\pi\sigma^2} \\
&\quad +E[c_{t,0}|z_t, m] \frac{(z_t - z_{max})^2}{\sigma^2} \\
&\quad +\sum_{k=1}^{K_t} E[c_{t,k}|z_t, m] \frac{(z_t - d_{t,k})^2}{\sigma^2}]]
\end{aligned} \tag{9}$$

Maximizing the above expected log-likelihood is the final goal. To do this, *expectation maximization algorithm*, EM algorithm, is used. The EM algorithm is one such elaborate technique. The EM algorithm is a general method of finding the maximum-likelihood estimate of the parameters of an underlying distribution form a given data set when the data is incomplete or has missing values.

As the above way, we choose the optimal mapping algorithm according to the sensor measurements, namely clusters. Hence, maps generated by our approach are faster and more accurate than ones generated by the previous occupancy grid mapping algorithm.

4 Simulation

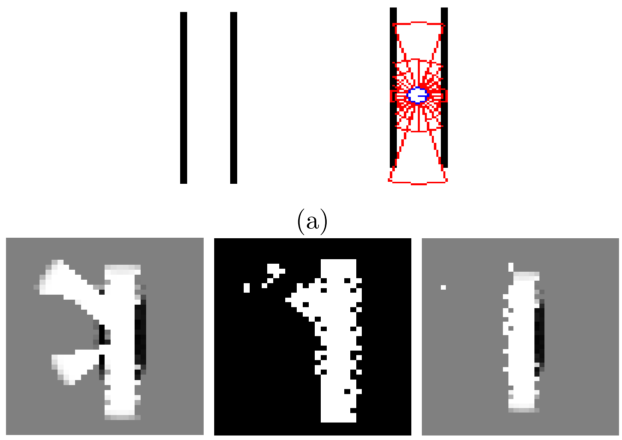
In order to test our approach, we applied our approach to learning grid maps using simulated data. Our main finding are that the maps generated our approach are more accurate and the approach has less time than the previous occupancy grid mapping algorithm, such as the standard occupancy grid mapping algorithm and the alternative occupancy grid mapping algorithm with EM.



(a)

time:0.172s (b) | time:48.656s (c) | time:5.047s (d)

Figure 1: Narrow open door without error



(a)

time:0.156s (b) | time:101.078s (c) | time:18.828s (d)

Figure 2: Corridor with error

The sensor measurements are gathered in a corridor while driving by an open door. The mobile robot is equipped with a circular array of 24 sonar sensors.

Figure 1(a) shows a narrow open door as a first example. The width of the door is two times wider than the width of mobile robot. Hence, the mobile robot can pass through the door, but it may be difficult to control. Figure 1(b) shows the result of the standard occupancy grid mapping algorithm. In the standard occupancy grid mapping, a narrow open door is not detected, but other places are similar to Figure 1(a). Figure 1(c) is obtained by the alternative occupancy grid mapping with EM. In Figure 1(b), the door is detected, but it takes much time to calculate. In Figure 1(d) generated by our approach, the door is detected and it takes less time than the occupancy grid mapping with EM. Figure 2 shows the result of corridor with the error measurements. Figure 2(a) is a simulated environment. As Figure 2(b) shows map of the standard occupancy grid mapping, map is incorrect because of the sensor error. In Figure 2(c), the alternative occupancy grid mapping detect incorrectly in one place though it is better than (b). Unlike Figure 2(b) and Figure 2(c), Figure 2(d) shows an accurate map. As Figure 2(d) is generated by our approach, the map is similar to the environment(a). Because of clusters, our approach is more accurate than the occupancy grid mapping with EM in erroneous place. Our approach takes also less time than others.

As a result, because our approach maps with the emphasis on the clusters, maps generated with adaptive occupancy grid mapping algorithm, more accurate and faster than the others.

5 Conclusion

In this paper, the adaptive occupancy grid mapping algorithm is proposed. Unlike existing occupancy grid mapping algorithm, our approach relies on the clusters. The clusters are the region that have the high probability to be inconsistent with the sensor data. Neural networks is used to make a cluster. According to the cluster, we use optimal occupancy grid mapping algorithm. As seeing in simulation result, we can map more accurate and faster than the previous occupancy grid mapping.

References

[1] Thrun, S, “Learning Occupancy Grid Maps with Forward Sensor Models,” *Autonomous Robots*,Vol. 15, pp. 111-127, 2003.

[2] A. Elfes, “Using Occupancy Grids for Mobile Robot Perception and Navigation,” *IEEE Computer*,Vol. 22, pp. 46-57, 1989.

[3] H.P Moravec, “Sensor Fusion in Certainty Grids for Mobile Robots,” *IEEE Computer*,Vol. 22, pp. 46-57, 1989.

[4] L, Fausett, “Fundamentals of Neural Netwoks: Architectures, Algorithms, and Application,” Prentice-Hall, 1994.

[5] Riedmiller, M, “A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm,” *IEEE Neural Network*,Vol. 1, pp. 586-591, 1993.

[6] C.H Kim, “Feature Extration Method for A Robot Map Using Neural Networks,” *AROB*, Vol. 1, pp. 256-258, 2002.