Mobile Robots Navigation, Mapping, and Localization Part I

Lee Gim Hee
DSO National Laboratories, Singapore

Marcelo H. Ang Jr.
National University of Singapore, Singapore

INTRODUCTION

The development of autonomous mobile robots is continuously gaining importance particularly in the military for surveillance as well as in industry for inspection and material handling tasks. Another emerging market with enormous potential is mobile robots for entertainment.

A fundamental requirement for autonomous mobile robots in most of its applications is the ability to navigate from a point of origin to a given goal. The mobile robot must be able to generate a collision-free path that connects the point of origin and the given goal. Some of the key algorithms for mobile robot navigation will be discussed in this article.

BACKGROUND

Many algorithms were developed over the years for the autonomous navigation of mobile robots. These algorithms are generally classified into three different categories: *global path planners*, *local navigation methods* and *hybrid methods*, depending on the type of environment that the mobile robot operates within and the robot's knowledge of the environment.

In this article, some of the key algorithms for navigation of a mobile robot are reviewed. Advantages and disadvantages of these algorithms shall be discussed. The algorithms that are reviewed include the *navigation function*, *roadmaps*, *vector field histogram*, *artificial potential field*, *hybrid navigation* and the *integrated algorithm*. Note that all the navigation algorithms that are discussed in this article assume that the robot is operating in a planar environment.

GLOBAL PATH PLANNERS

Global path planning algorithms refer to a group of navigation algorithms that plans an optimal path from a point of origin to a given goal in a known environment. This group of algorithms requires the environment to be free from dynamic and unforeseen obstacles. In this section, two key global path planning algorithms: *navigation functions* and *roadmaps* will be discussed.

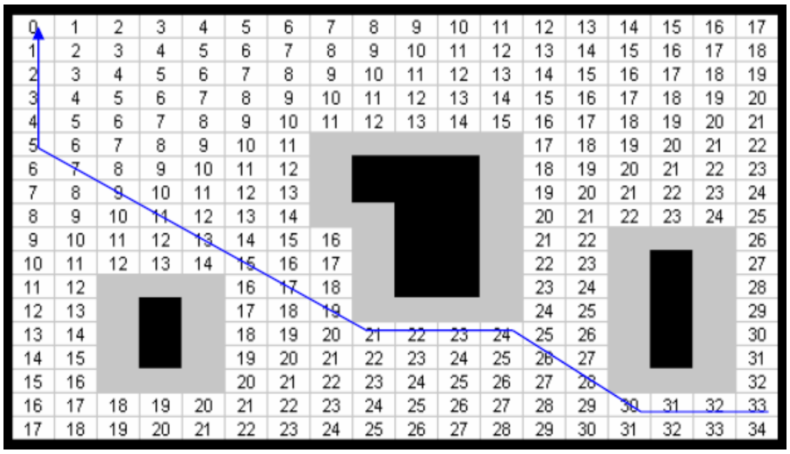
Navigation Functions

The most widely used global path planning algorithm is perhaps the navigation function computed from the "wave-front expansion" (J.-C Latombe, 1991; Howie Choset et al, 2005) algorithm due to its practicality, ease in implementation and robustness. The navigation function N is the *Manhattan distance* to the goal from the free space in the environment. The algorithm requires information of the environment provided to the robot to be represented as an array of grid cells.

Figure 1. The shaded cells are the 1-Neighbor of the cell (x, y) and the number shows the priority of the neighboring cells

2	3	4
1	(x, y)	5
8	7	6

Figure 2. Path generated by the navigation function



The navigation function assigns a numeric value *N* to each cell with the goal cell having the lowest value, and the other unoccupied cells having progressively higher values such that a steepest descent from any cell provides the path to the goal. The value of the unoccupied cell increases with the distance from the goal. Each grid cell is either free or occupied space denoted by gC_{free} and $gC_{occupied}$. First, the value of *N* is set to '0' at the goal cell gC_{goal}. Next, the value of *N* is set to '1' for every 1-Neighbor (see Figure 1 for the definition of 1-Neighbors) of gC_{goal} which is in gC_{free}. It is assumed that the distance between two 1-Neighbors is normalized to 1. In general, the value of each gC_{free} cell is set to N+1 (e.g., '2') for every unprocessed gC_{free} 1-Neighbor of the grid cell with value *N* (e.g., '1'). This is repeated until all the grid cells are processed.

Finally, a path to the goal is generated by following the steepest descent of the *N* values. To prevent the path from grazing the obstacles, the grid cells which are less than a safety distance α from the obstacles are omitted in the computation of the navigation function. Figure 2 shows a path generated by the navigation function. The black cells are the obstacles and the grey cells are the unsafe regions.

Roadmaps

A *roadmap* is a network of one-dimensional curves that captures the connectivity of free space in the environment (J.-C Latombe, 1991; Danner et al, 2000; Foskey et al, 2001; Isto P., 2002; T. Siméon et al, 2004; Xiaobing Zou et al, 2004; Howie Choset et al, 2005; Bhattacharya et al, 2007). Once a roadmap has been constructed, it is used as a set of standardized paths. Path planning is thus reduced to connecting the initial and goal positions to points in the roadmap. Various methods based on this general idea have been proposed. They include the *visibility graph* (Danner et al, 2000; Isto P., 2002; T. Siméon et al, 2004), *Voronoi diagram* (Foskey et al, 2001; Xiaobing Zou et al, 2004; Bhattacharya et al, 2007), *freeway net* and *silhouette* (J.-C Latombe, 1991; Howie Choset et al, 2005).

The *visibility graph* is the simplest form of *roadmap*. This algorithm assumes that the environment is made up of only polygonal obstacles. The nodes of a *visibility graph* include the point of origin, goal and all the vertices of the obstacles in the environment. The graph edges are straight line segments that connect any two nodes within the line-of-sight of each other. Finally, the shortest path from the start to goal can be obtained from the *visibility graph*.

Advantages and Disadvantages

The advantage of the *navigation functions*, *roadmaps* and other global path planning algorithms is that a continuous collision-free path can always be found by analyzing the connectivity of the free space. However,