Information Sciences 280 (2014) 64–81

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Function-segment artificial moment method for sensor-based path planning of single robot in complex environments

Wang-bao Xu [a,b], Xue-bo Chen [b,*], Jie Zhao [c], Xiao-ping Liu [b,d]

[a] School of Control Science and Engineering, Dalian University of Technology, Dalian 116024, PR China
[b] School of Electronics and Information Engineering, Liaoning University of Science and Technology, Anshan 114051, PR China
[c] State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150001, PR China
[d] Department of Electrical Engineering, Lakehead University, 955 Oliver Road, Thunder Bay, ON P7B 5E1, Canada

ARTICLE INFO

Article history:
Received 11 April 2013
Received in revised form 7 March 2014
Accepted 27 April 2014
Available online 8 May 2014

Keywords:
Mobile robot
Motion control
Sensor-based path planning
Obstacle-avoidance
Artificial moment method
Attractive point

ABSTRACT

A function segment artificial moment method is proposed for sensor-based path planning of a single robot in complex environments. Similar to the existing artificial moment method, an *attractive point* is obtained first at each step for guiding the robot to move along a shorter path. Then, the robot moves one step to the next position under the guidance of the *attractive point* and the control of an artificial moment motion controller. Different from the existing one, *attractive function segments* are used for *attractive points* and the artificial moment motion controller is improved. As each *attractive function segment* can be used for several steps, the computational burden for *attractive points* is reduced considerably. Furthermore, positive and negative *key obstacle function segment* sets are proposed to determine whether the target is reachable when a new *attractive function segment* is required. If the target is reachable, then, a motion direction for the robot is determined by using the *key obstacle function segment* sets. Simulation results indicate that the proposed method is effective and can yield better solutions in complex environments.

1. Introduction

Path planning in the presence of obstacles is an important problem in robotics [1,10,22] as it has a wide range of applications in industries, hospitals and offices, including the fields such as military, simulation and computer aided design [1,19,22,25,26,31]. The basic path planning problem has several classifications. A common one includes global (off-line) and local (on-line or sensor-based) planning [22,23,26,31]. In global planning, a robot knows the whole environment beforehand and plans the entire path priori to its movement [22,23,31]. On-line or sensor-based path planning for a mobile robot, however, is concerned with steering it from an initial position to a final one without any priori knowledge of the environment [8,12,22,23,31].

In a realistic setting, path planning of a robot cannot be based on complete a priori knowledge of the environment [24]. Therefore, much attention has been paid to sensor-based path planning of mobile robots in unknown, complex and dynamic environments [8,12,9,23,24,30,31], and this paper focuses also on this topic.

Among global sensor-based path planning methods, the visibility graph [10,14,19,31,32], the Voronoi diagram [10,19,31,33,35], and the tangent graph [15,20] are well-known. The main idea of the methods consists of capturing the

* Corresponding author. Tel.: +86 412 5928176.
E-mail address: xuebochen@126.com (X.-b. Chen).

http://dx.doi.org/10.1016/j.ins.2014.04.045
0020-0255/

connectivity of the free space into a network called roadmap. Once the roadmap is built, it is used as a set of standardized path, and the path planning is reduced to find a path between an initial and a final point on the roadmap.

The above methods rely on an explicit representation of obstacles in the configuration space, which may result in an excessive computational burden. Instead of using an explicit representation of the environment, sampling based algorithms, such as probabilistic roadmaps [2,17,19] and rapidly-exploring random trees [2,5,19,28], build a roadmap (a graph of feasible trajectories) by connecting points sampled from the obstacle-free space using the collision checking module. The roadmap is then used to construct the solution to the original motion-planning problem.

In recent years, many evolutionary algorithms, such as particle swarm optimization, fuzzy logic method, genetic algorithm, and membrane computing, are employed not only for mobile robots control [3,4,6] but also for the path planning problem [9,13,25,26,31]. Better solutions for path planning are often yielded by applying these algorithms.

Global sensor-based path planning methods can guarantee the global convergence and generate a shorter path. The shortcoming is the heavy computational burden because the tracking path needs to be frequently recomputed as new information is discovered continuously in the local path planning.

In local sensor-based path-planners, the Artificial Potential Field method (APF) [11,12,18,22,23,29,34,40,41] is influential, which uses local sensory information in a purely reactive fashion. Here a repulsive potential field to force the robot away from obstacles and an attractive one to drive the robot toward a target are used to generate a force. The force equals to the negative gradient of the total potentials and makes the robot move from the position with the higher potentials toward that with the lower. The method can be used directly in realistic dynamical environments. It is not required to build a configuration-space and to take additional measures to deal with dynamical obstacles. As such, it is much simpler to be implemented. However, it often suffers from the "local minimum" problem, yields low quality solutions, and cannot guarantee the global convergence in some complex environments.

The wall following or "bug" algorithm [12,21,24,27,34,41] is an important midway approach as it combines local planning with global information. Thus, the global convergence can be guaranteed to a certain extent. Two reactive modes of motion: moving toward the goal and following an obstacle boundary, are used. In order to guarantee convergence to the target, the transition between the two modes is governed by a criterion to ensure that the distance to the target decreases monotonically.

In the bug family, the tangentbug algorithm [16,27] is important as it uses a local tangent graph for choosing the local optimal direction while moving toward the target, for making local shortcuts and for testing a leaving condition while moving along an obstacle boundary [16,24]. The tangentbug algorithm often produces shorter path than other bug algorithms [27].

However, the performances of the traditional bug algorithms depend on the choice of the initial motion direction. If the choice is "wrong", then, it may produce a catastrophically long path, while in reality there is a much shorter path.

For example, in the environment shown in Fig. 1, where the only door lies to the right of *R*'s start position, if the wall is very long and *R* moves to the left forever, then, the target *T* will never be reached, as in [24].

Another example is the environment shown in Fig. 2, where a catastrophically long path will be produced if *R* moves to the right forever. As remarked in [10], a short path is important for many applications.

To avoid the above situations, the cautiousbug algorithm is proposed in [24], where the spiral search strategy is used in the wall-following mode. As such, the performance is not dependent on the choice of the initial motion direction. However, a longer path is produced on average.

Inspired by the artificial moment method for multi-robot formation control [36,37], a midway approach, artificial moment method, is developed in [38] for sensor-based path planning of a single robot. An *Optimal Way Representative point* (*OWRpoint*), which is on the shortest *free block-wall way*, is computed first by using the full environmental knowledge at the current time. Then, *R* moves one step to the next position under the guidance of the *OWRpoint* and the control of an *artificial moment motion controller*. Meanwhile, in order to guarantee *R*'s safety and to decrease the computational burden for *OWRpoint*s, *N-type* and *B-type artificial obstacle segment*s may be set.

In the artificial moment method, pre-processing of the environment and pre-planning of a complete feasible path to reach *T* at any step are not required, so the computation burden is less than that of the roadmap-based one. Because almost all the problems encountered in the APF method have been solved by the artificial moment motion controller, the method can be used in narrow, complex and dynamical environments. As more environmental information has been used for path planning, the solutions obtained by the method are often better than those by the bug algorithms.

In [39], the artificial moment method in [38] is extended to multi-robots, where each robot uses one path planner. If a robot has *coordinated companions*, then, according to the idea of pair-wise coordinated control [7], *coordinated moments*

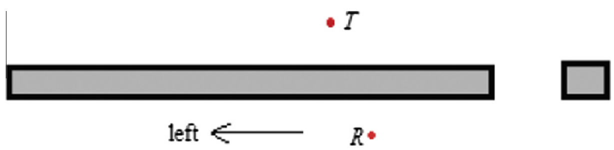

Fig. 1. *R* Cannot reach *T* if it moves to the left forever.