Expert Systems With Applications 62 (2016) 104–115

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Neural networks based reinforcement learning for mobile robots obstacle avoidance

Mihai Duguleana[a,*], Gheorghe Mogan[b]

[a] Department of Automotive and Transport Engineering, Faculty of Mechanical Engineering, University Transilvania of Brasov, Str. Universității nr. 1, Brasov 500036, Romania.

[b] Department of Automotive and Transport Engineering, Faculty of Mechanical Engineering, University Transilvania of Brasov, Brasov 500036, Romania.

ARTICLE INFO

Article history:
Received 29 October 2015
Revised 9 June 2016
Accepted 9 June 2016
Available online 11 June 2016

Keywords:
Obstacle avoidance
Neural networks
Q-learning
Virtual reality

ABSTRACT

This study proposes a new approach for solving the problem of autonomous movement of robots in environments that contain both static and dynamic obstacles. The purpose of this research is to provide mobile robots a collision-free trajectory within an uncertain workspace which contains both stationary and moving entities. The developed solution uses Q-learning and a neural network planner to solve path planning problems. The algorithm presented proves to be effective in navigation scenarios where global information is available. The speed of the robot can be set prior to the computation of the trajectory, which provides a great advantage in time-constrained applications. The solution is deployed in both Virtual Reality (VR) for easier visualization and safer testing activities, and on a real mobile robot for experimental validation. The algorithm is compared with Powerbot's ARNL proprietary navigation algorithm. Results show that the proposed solution has a good conversion rate computed at a satisfying speed.

1. Introduction

Path planning is one of the key elements of autonomous mobile robots. Since the introduction of mobile robotic platforms back in the '50s, the main desiderate sought by most researchers in motion planning is the development an algorithm capable of providing collision-free trajectories. The subject was divided into two separate research areas, based on the type of environment information which is used by the mobile robot (de Berg, van Kreveld, Overmars, & Schwarzkopf, 2000).

The first approach uses global knowledge of the environment, meaning that at each moment, the robot has complete information about its location, movement capabilities, obstacles and target. This raises additional problems related to localization. Based on a good localization technique, the robot can determine precisely its position with respect to the changing environment (usually, a configuration space C is employed to describe all possible configurations of robot; presuming the navigation takes place in a 2D workspace, C is divided in 2: the obstacles space – C_{obs}, and the free space – C_{free}). Navigating in C_{free} can be achieved through a wide variety of algorithms (such as SLAM – Simultaneous Localization And Mapping (Leonard & Durrant-Whyte, 1991), Wireless Localization based on RSSI (Stoep, 2009), particle filter localization (Dellaert, Fox, Burgard, & Thrun, 1999) and others) and sensorial systems (GPS (Montiel & Sepúlveda, 2014), camera networks, environment markers and so on). As one can infer, it is possible to know in advance if the goal is reachable, which makes this a perfect candidate for artificial neural networks (ANNs).

The second approach uses local information retrieved by range sensors (sonar (Kim & Kim, 2011), laser (Surmann, Nüchter, & Hertzberg, 2003)), infrared sensors (Alwan, Wagner, Wasson, & Sheth, 2005) or video cameras (Seder & Petrovic, 2007). Aside from the fact that there is no guarantee of convergence, one of the main issues which needs to be solved by scientists is the identification and avoidance of local minima. In most cases, this approach doesn't guarantee convergence. Thus, we settle to use global information within this study.

Over the last decades, many researches dealt with the path planning problem. Various types of solutions were proposed: grid-based, potential fields, geometrical or based on artificial intelligence (AI). Grid-based methods involve the overlay of a grid over the C space. In order to obtain a valid path, all grid cells (or grid points) must therefore be included in C_{free}. One of the most common grid-based algorithms in motion planning is VFH (with its variants, VFH+ and VFH*) (Borenstein & Koren, 1991; Ulrich & Borenstein, 1998). Potential fields model C after a potential function: obstacles are seen as repulsive entities while the target is seen as an attractive center. The workspace is regarded as an

* Corresponding author. Fax: +40 268 418967.
E-mail addresses: mihai.duguleana@unitbv.ro (M. Duguleana), mogan@unitbv.ro (G. Mogan).

http://dx.doi.org/10.1016/j.eswa.2016.06.021

isolated universe, which works towards minimizing its potential energy, thus pushing the moving entity (the mobile robot) to the goal (Borenstein & Koren, 1991). Among geometrical methods, the most used are cell decomposition and the visibility graph (Barraquand & Latombe, 1991). The visibility graph is constructed based on clusters of inter-visible points within C_{free}. Using a traversing algorithm such as Dijkstra, A* (Dechter & Pearl, 1985) or D* (Stentz, 1994), the shortest or optimal paths are computed (Lozano-Pérez & Wesley, 1979). Latest research in motion planning employs the use of artificial learning techniques. Q-learning has been used in Jaradat, Al-Rousan, and Quadan (2011) to achieve motion planning in dynamic environments. The authors limit the number of states from the states space, thus reducing the size of Q-table and indirectly, the computation time; however, convergence is not guaranteed. Inspired by bird flocking, particle swarm optimization (PSO) is also widely used in motion planning (Qin, 2004). Each particle from the space of solution candidates tries to achieve the goal optimally, and improves its "experience" after every new iteration, based on its trajectory history and on the "experience" of other neighboring particles. Another widely exploited motion planning method is fuzzy logic (Reignier, 1994). Last but not least, neural networks were used to achieve obstacle-free trajectories (Dezfoulian, Wu, & Ahmad, 2013; Fierro & Lewis, 1998). Although many of these AI methods show promising results, just a few actually target the avoidance of dynamic obstacles, and even less fewer implement the proposed motion planning solution in real testing environments for experimental validation.

A multi-layer neural network is able to map non-linear functions (Hecht-Nielsen, 1987). This feature can be used in conjunction with reinforcement learning in order to solve the path planning problem, given prior knowledge of the environment. For this specific case, Q-learning (Russell & Norvig, 2002) was used with the following function that quantifies the quality of a state-action:

$$Q : S \times A \rightarrow \mathbb{R} \tag{1}$$

where Q is the set of solutions, S is the set of states and A is the set of actions. The cost, or better said, the reward for a collision-free trajectory is given if the mobile robot reaches the goal. In other words, the proposed solution samples each state, action and result from the workspace as an underlying probability distribution which helps in calculating the reward parameter. For fast convergence, the solution makes further use of a feed-forward neural network. Thus the proposed, solution usually find a collision-free trajectory from the first few epochs.

The motion planner is implemented in VR for initial testing and efficient visualization, and after achieving satisfying results, on a real mobile robot: PowerBot from Mobile Robots ("PowerBot website", 2015).

2. Literature overview

This study encompasses aspects from multiple research areas. A brief resume of current achievements in these areas is proposed bellow.

2.1. Obstacle avoidance of mobile robots in dynamic environments

Avoiding collisions with moving obstacles is a challenging task. In order to solve this problem, a large number of algorithms using both local and global knowledge were proposed by researchers.

Knowing only local information about the working environment presumes, in most cases, the usage of a reactive approach, such as directional or velocity-based methods. Directional methods calculate geometrically the robot's trajectory (Khatib, 1986; Minguez & Montano, 2004). Knowing the exact coordinates of the robot and of the obstacles, the path planner can simply calculate the Euclidian distance at each time instance, and by setting a lower limit to this variable, the robot can move on collision-free trajectories (Asano, Guibas, Hershberger, & Imai, 1985). Velocity-based methods consider the kinetic energy of the robot and of the closest recognized moving obstacles, and use this data in trajectory generation (Large, Laugier, & Shiller, 2005). The most used velocity-based method is Dynamic Time Window, introduced back in 1997 (Fox, Burgard, & Thrun, 1997). One of the biggest issues with reactive methods is that they need a good sensorial system which can produce accurate position coordinates for any local obstacles. Latest studies use video cameras to get environment information and to estimate the dynamics of the scene. For example, a single camera can be used to either detect landmarks and environment cues, or based on an algorithm such as the Block-Based Motion Estimation (Kim & Do, 2012), to detect and classify moving obstacles. Multiple cameras provide stereoscopic vision, making depth perception much easier (Chilian & Hirschmüller, 2009). Another type of sensor introduced on the market in the last decade is the time-of-flight (TOF) camera (May & Werner, 2006), a hybrid between laser range sensors and classic video cameras.

Usually, obstacle avoidance algorithms based on local information of an environment with a fairly large amount of obstacles rely on selecting the obstacle which is most likely to collide with the robot. This strategy is however hard to implement on real mobile robots, since the selection process itself is subject to many questions such as:

1. Are all the obstacles properly sensed?
2. Is this the closest obstacle?
3. Is this the most dangerous obstacle?
4. What if there are 2 or multiple obstacles closing at the same time?

Considering a global representation of the dynamic environment is available, some of the most used navigation algorithms rely on variations of the potential field method. Cases include specific situations when for example both the target and the robot are moving (Ge & Cui, 2002; Huang, 2009), or the use of harmonic functions in order to completely eliminate the local minima (Kim & Khosla, 1992). Others use an integrated representation of the workspace (Savkin & Wang, 2014) or analytical approaches (Qu, Wang, & Plaisted, 2004) to achieve collision-free trajectories.

One of the main issues that drifted researches away towards unconventional motion planning algorithms is the computation time. Lately, several studies employ the use of AI, since many methods converge faster, are easier to implement and produce in some cases more satisfying results.

2.2. Path planning with artificial intelligence techniques

There are many artificial intelligence techniques used for solving path planning. Among these, fuzzy logic was the first to be used (Reignier, 1994; Saffiotti, 1997; Yen, 1995). Fuzzy logic is great for static workspaces, but produces weak results in dynamic environments. Also, fuzzy-computed trajectories are not optimal. Genetic algorithms (GAs) followed shortly (Sugihara & Smith, 1997). Due to their specific, GAs are great at finding global optimal trajectories. However, they do not scale well with highly complex environments, and finding a good fitness function for the motion planning problem is rather difficult. Hybrid methods emerged, which used classic algorithms such the potential fields, together with AI techniques (such as GAs), for improving the solution (Vadakkepat, 2000). However, the path planning performance is still weak due to the limitations imposed by the potential field model used in the study. PSO also started to be used for achieving collision-free robot navigation (Kennedy, 2010; Nasrollahy, 2009). However, the results