Robotics and Computer-Integrated Manufacturing 32 (2015) 65–71

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

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

On generating the motion of industrial robot manipulators

K. Kaltsoukalas, S. Makris, G. Chryssolouris *,[1]

Laboratory for Manufacturing Systems and Automation, University of Patras, Greece

ARTICLE INFO

Article history:
Received 17 October 2013
Received in revised form
19 September 2014
Accepted 8 October 2014
Available online 29 October 2014

Keywords:
Path planning
Industrial robot motion
Grid search

ABSTRACT

In this study, an intelligent search algorithm is proposed to define the path that leads to the desired position and orientation of an industrial robot's manipulator end effector. The search algorithm gradually approaches the desired configuration by selecting and evaluating a number of alternative robot's configurations. A grid of the robot's alternative configurations is constructed using a set of parameters which are reducing the search space to minimize the computational time. In the evaluation of the alternatives, multiple criteria are used in order for the different requirements to be fulfilled. The alternative configurations are generated with emphasis being given to the robot's joints that mainly affect the position of the end effector. Grid resolution and size parameters are set on the basis of the desired output. High resolution is used for a smooth path and lower for a rough estimation, by providing only a number of the intermediate points to the goal position. The path derived is a series of robot configurations. This method provides an inexperienced robot programmer with flexibility to generate automatically a robotic path that would fulfill the desired criteria without having to record intermediate points to the goal position.

1. Introduction

In the recent years, there is an increasing need for flexible manufacturing systems, capable of adapting to different market demands and product-mix changes [1]. The dynamic environment in production requires an increasing number of reconfigurations on assembly manufacturing resources [3]. In automated assembly systems such as robots, the flexibility is normally restricted due to the high programming effort required in order for robot trajectories to adjust to different assembly cell layouts. Experienced robot programmers have to spend considerable time in order to optimize the robotic paths for each specific application by using conventional programming methods. A method that is widely used is programming by demonstration, where the intermediate points to the goal position are recorded by sequentially moving the robot to each position using the teach pendant. The robot's final path is generated by connecting the recorded points via a robot controller, which tries to pass through all the points by taking into consideration the dynamic constraints of the robot. The robot's final trajectory is highly dependent on the points recorded and the experience of the respective programmer, who has carried this out. Automatic path planning for robotics poses the question as to how a robot can move from its initial to the final position and has been investigated during the last decades mainly focusing on path planning for collision avoidance.

One of the techniques for motion planning is the construction of approximate models by sampling their configuration space. Over the last few years, there has been a lot of work carried out for the improvement of sampling based motion planning algorithms. It is hard to define a single criterion that can classify all planners in distinct categories. The classical separation is between roadmap-based planners and tree-based planners [4].The probabilistic roadmap path planning was introduced in [5] as a new method of computing collision-free paths for robots. The method proceeds in two phases: those of learning and query. In the learning phase, a probabilistic roadmap is constructed by generating the robot's random free configurations and connecting them using a simple motion planner, also known as a local planner. Different approaches have been used to address a variety of problems. In [6], two different methods for constructing and querying roadmaps are suggested for the motion planning of deformable objects. Another two deformation techniques that can be applied to the resulting path are also presented. The obstacle probabilistic roadmap method is introduced into [7], where several strategies for node generation are described and multi-stage connection strategies are proposed for cluttered 3-dimensional workspaces. In [8], a randomized planner is described for planning CF-compliant motion between two arbitrary polyhedral solids, by extending the probabilistic roadmap paradigm for planning collision-free motion to the space of contact configurations. The key to this approach is a novel sampling strategy of generating random CF-compliant configurations.

* Corresponding author.
E-mail address: xrisol@lms.mech.upatras.gr (G. Chryssolouris).
[1] Tel.: +30 2610 997262.

http://dx.doi.org/10.1016/j.rcim.2014.10.002
0736-5845/

The concept of Rapidly-exploring the Random Tree is introduced in [9]. The basic idea is that an initial sample (the starting configuration) is the root of the tree and newly produced samples are then connected to the samples already existing in the tree. In [10], two Rapidly-exploring Random trees (RRTs) were rooted at the start and during the goal configurations. Each one of the trees explores the space around it and also advances towards each other through the use of a simple greedy heuristics. Although it was originally designed that motions be planned for a human arm (modeled as a 7-DOF kinematic chain), in the automatic graphic animation of collision-free grasping and manipulation tasks, the algorithm has been applied to a variety of path planning problems. Tree-based planners have proven to be a good framework for dealing with real-time planning and re-planning problems. In [11], a re-planning algorithm is presented for repairing Rapidly-exploring Random Trees when changes are made to the configuration space. Instead of abandoning the current RRT, the algorithm efficiently removes only the newly-invalid parts and maintains the rest. Dynamic obstacle avoidance has been investigated for the mobile robots found in industrial environments in [12]. However, industrial manipulators are typically programmed to execute predefined paths. The two main categories of robotic programming methods are those of online programming and offline programming.

In [13], an online path planning and programming support system is proposed for the transformation of the user's interaction into a simplified task that generates acceptable trajectories, applicable to industrial robot programming. In [14], a novel approach to robot programming using an Augmented Reality environment was proposed, offering flexibility and adaptability to different environments when an on-site robot programming approach was desired. The path planning methodology included a beam search algorithm to generate paths. In [15], there is a similar study, where the user is able to perform operations, namely via-points selection and modification, in order for a smooth and collision-free path to be achieved. An on-line robot motion planning approach that is based upon pre-computing the global configuration space (C-space) connectivity is proposed. In [16], the motion planner consists of an off-line stage and an on-line stage and the collision-free path is searched in this C-space by using the A^* algorithm under a multi-resolution strategy.

In this study, an intelligent search algorithm is proposed to define an industrial robot manipulator's path that leads to the desired position and orientation of the end effector. A maximum number of alternative configurations are selected and evaluated in each step until the desired configuration is approached within a predefined error. The alternative configurations are generated in a clever way giving emphasis to the joint angles that mainly affect the robot's position in the workspace. In the configuration space, there is a grid constructed to derive the robot's alternative configurations. A set of clever parameters are used to reduce the search space and increase the performance of the algorithm. In the evaluation of the alternatives, multiple criteria that would enhance the algorithm's flexibility to extend are used, in order for the different requirements, namely the shortest path, to be fulfilled.

2. Approach

For an industrial robot manipulator (usually six degrees of freedom), the path planning problem is described via three hierarchical levels as shown in Fig. 1. For a given starting and goal position, the requested paths include the robot's intermediate configurations, where each configuration is a set of six joint parameters.

2.1. Grid search of the alternative configuration

For an industrial robot manipulator with n degrees of freedom (n-DoF), the alternative configurations are defined from a set of n joint angles. If the possible values of each joint angle are equal to $2k+1$, with resolution $d\Theta$ (Fig. 2), the number of alternative configurations is given by the following equation:

$$\text{Number of alternative configurations } N=(2k+1)^n \tag{1}$$

For each joint angle that can be incremented, a $d\theta_n$ resolution has to be selected.

The number of alternative configurations increases for a robot with higher degrees of freedom and a larger grid (k) size. For this reason, the following parameters are used for the reduction of the alternative configurations, where a multi-criteria evaluation will be carried out as follows:

- *Decision Horizon* (*DH*): This parameter is taking values from one to n (DoF of the robot). Starting from the base of the robot, DH

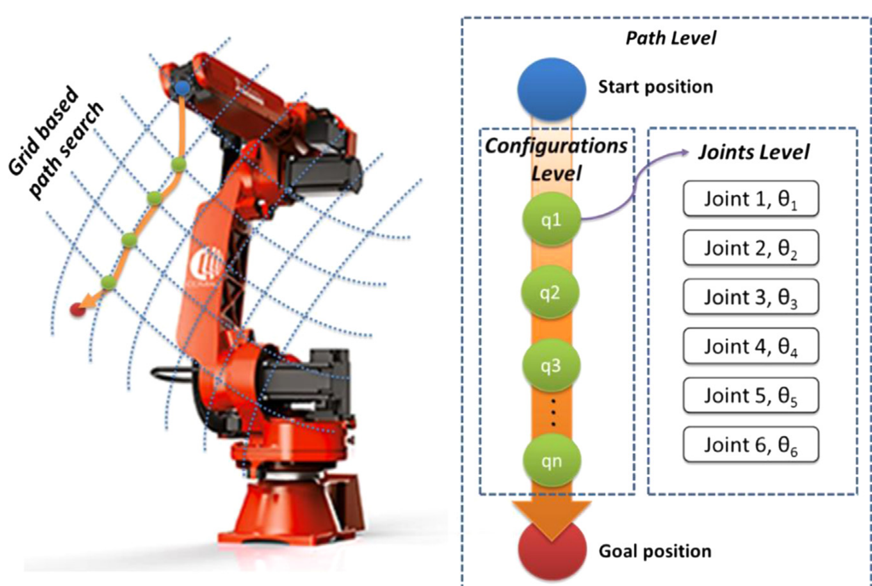

Fig. 1. Hierarchical levels for path planning problem (6 DOFs robot).