Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 44 (2016) 377 – 382

www.elsevier.com/locate/procedia

6th CIRP Conference on Assembly Technologies and Systems (CATS)

Automated and Cycle Time Optimized Path Planning for Robot-Based Inspection Systems

Marco Ulrich[a,*], Gregor Lux[a], Leïla Jürgensen[a], Gunther Reinhart[a]

[a]Institute for Machine Tools and Industrial Management, Technische Universität München, Boltzmannstr. 15, 85748 Garching, Germany

* Corresponding author. Tel.: +49-89-289-15576; fax: +49-89-289-15555. *E-mail address:* marco.ulrich@iwb.tum.de

Abstract

Robot-based inspection systems, consisting of a standard industrial robot and an optical 3D sensor, increasingly gain importance within production in order to quantify the quality of products. These systems show advantages in terms of costs, flexibility and in-line capability. Based on the inspection plan of a product, the robot path for the inspection system is currently planned manually which is a very time consuming process. Consequently, an automated path planning algorithm generating a time optimized and collision-free path would improve the flexibility of robot-based inspection systems. The presented approach shows an automated and cycle time optimized path planning algorithm for robot-based inspection systems. This is realized by the probabilistic roadmap method applied on all measurement poses in combination with the A* search algorithm for the determination of weighted paths. Finally, the optimization of the path is reduced to a traveling salesman problem which is solved by the Christofides heuristic.
© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of the organizing committee of the 6th CIRP Conference on Assembly Technologies and Systems (CATS)
Keywords: Industrial robot, Path planning, Visual inspection

1. Introduction

Costumers increasingly demand high quality of products, especially in the premium sector. A typical example for such products can be found in the automotive industry during the body construction [1]. In this context, product quality is often associated with the compliance of the geometry of parts. To counter the quality claim, ever-tightening tolerances are established. These product tolerances have to be monitored within the production process [2]. According to the requirements of quality assurance, robot-based inspection systems, consisting of an industrial robot and an optical 3D sensor, become increasingly important for inspection tasks. Standard industrial robots have advantages concerning costs, in-line capability and flexibility. Optical 3D sensors enable the inspection of quality determining characteristics. Thus, the combination of an industrial robot and a 3D optical sensor has the potential to ensure an efficient inspection of products within the production process [3].

In consideration of the workflow for an inspection task, the key for a successful use of such systems is the degree of automation. Currently, the robot path is planned manually based on the inspection plan. This can be a time consuming process depending on the amount of features of the measurement object. Furthermore, the complexity of a cycle time optimized path increases with the number of features. Consequently, the automated path planning for robot-based inspection systems on basis of the inspection plan has the potential to reduce the preparation time of the system as well as the cycle time of the inspection.

Hence, the following challenges have to be addressed in order to realize an automated path planning algorithm for robot-based inspection systems.

The robot cell contains different obstacles which can limit possible robot poses, including the robot itself as well as the test object. Thus, the environment has to be modeled to compute a valid path. In order to ensure an unambiguous representation, the robot kinematic is represented in a vector space defined by the joint angles of the robot (configuration space). The set of configurations without collision is called collision-free configuration space. Due to the six degrees of freedom of industrial robots, building a complete configuration space is computationally intensive. Because of that, the first challenge is to find a time efficient solution to this problem. Secondly, the fastest path between two given measurement poses within the configuration space has to be computed. Thirdly, the sequence of measurement positions has to be optimized according to the cycle time. This typically can be described as a traveling salesman problem which has to be solved.

2212-8271 © 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).
Peer-review under responsibility of the organizing committee of the 6th CIRP Conference on Assembly Technologies and Systems (CATS)
doi:10.1016/j.procir.2016.02.021

The most time-consuming step within path planning for industrial robots is the recurring check for collisions. This should be considered for example by minimizing the number of paths to be checked [4].

2. State of the art

2.1. Representation of the configuration space

Path planning problems can be solved using different methods [5]. However, every method requires a representation of the collision-free configuration space.

Often, roadmap methods are used which represent the collision-free configuration space by a graph or network of paths. Cell decomposition methods divide the configuration space into cells, this can be done exactly or approximatively. Potential field methods describe the environment as a field where the goal possesses an attractive potential and the obstacles repulsive forces. This method is specifically applicable for local path planning to avoid obstacles during runtime. The challenge of the other two methods for global path planning is the representation of the configuration space. For robots with multiple degrees of freedom the computation of this space is complex and time consuming. Therefore, several methods using random elements were developed [6] [7].

An approach to global path planning for robots with multiple degrees of freedom operating in known static environments is the probabilistic roadmap method (PRM) [7]. It consists of a preprocessing of the free configuration space, after which the path planning problem can be solved by well-known algorithms.

During the learning phase, a set of collision-free configurations is randomly generated and interconnected by a local planner to a predefined number of neighbors. The resulting network whose edges correspond to feasible paths, may contain one or more connected components, depending on the time spent on preprocessing and the robot's free space. To get a better connectivity, optimization algorithms can be applied. In the query phase, the start and goal configuration are connected to the graph to search the roadmap for a sequence of edges connecting these two nodes to obtain an feasible path. This corresponds to the problem of path finding. However, a solution will usually differ from the optimal one, as the configuration space is constructed only partially.

2.2. Path finding

Once the collision-free configuration space is described as a graph, the shortest path between two nodes can be searched. An overview about common path finding algorithms is given in [9]: depth-first, breadth-first and best-first search, the algorithm of Dijkstra and finally the A* algorithm. All these approaches find a solution, if one exists. Especially the Dijkstra and A* algorithm are in the focus of research [10], as they promise the optimal path with a minimal computing time.

The algorithm of Dijkstra was developed in 1959 and always finds the shortest path between two given nodes or proves that no solution exists [11]. For this purpose, the costs $g(n)$ from the start node is assigned to each considered node n. Thereby the nodes with the smallest value of $g(n)$ are prioritized which guarantees an optimal path.

On this basis, the widely used A* algorithm was presented in 1968 [12]. The method finds a least-cost path between a start and a goal node. This is achieved by evaluating a cost function $f(n)$ of a node n to determine in which sequence the search visits nodes in order to expand the fewest possible nodes. The function $f(n)$ is the sum of the known costs $g(n)$ from the start node to n and the estimated costs $h(n)$ (also called heuristic function) from n to the goal node. The A* algorithm is complete, it will always find a solution if one exists. Furthermore, it computes the optimal path if the heuristic $h(n)$ does not overestimate the costs to the goal and is faster than the algorithm of Dijkstra [13].

2.3. The traveling salesman problem

To guarantee a cycle time optimized path, the measurement poses have to be sorted to obtain the fastest tour. Starting from the collision-free configuration space graph, this corresponds to solving a traveling salesman problem (TSP). Given a set of nodes along with the cost of travel between each pair of them, the TSP is about to find the cheapest way of visiting all the nodes and returning to the starting point [14].

The solving methods can be classified into two groups. The exact algorithms find the optimal solution for the TSP or prove that no solution exists. The computation time to find a solution depends exponentially on the number of considered nodes. [15]

The class of heuristics calculates an approximated solution. The advantage lies within the shorter computation time compared to exact algorithms [16]. Thereby, the quality of the solution depends on the chosen heuristic and is characterized by the ratio of the computed tour length to the optimal tour length. Different heuristics like the nearest neighbor algorithm, the nearest insertion heuristic, the Christofides algorithm or the PTAS of Arora [17] are analyzed in [18], [19] and [20].

The scope of this paper are industrial inspection systems with test plans consisting of hundreds of measurement poses. Hence, exact algorithms are not feasible. Considering the class of approximation algorithms, the Christofides heuristic calculates the best results, as it ensures a solution that is at most 1.5-times longer than the optimal solution [21]. The PTAS of Arora promises even better results, but a efficient implementation is not yet available [19].

3. Path planning

Typically, the path planning problem for robot-based inspection systems has to be solved according to given measurement poses of the robot and the CAD data of the static robot cell. Based on that, the path planning aims to generate an optimal path for the defined task. Automating the process is particularly useful if the cycle time and the duration for solving the problem manually is reduced by the automated process. Because of this, the approach is designed for a fast and flexible use.

3.1. Approach

Path planning can be structured in three major steps (see Fig. 1). In the first step, collision-free poses of the robot have to be determined which can be used for the generation of a path.

Download English Version:

https://daneshyari.com/en/article/1698624

Download Persian Version:

https://daneshyari.com/article/1698624

Daneshyari.com