2018 18th International Conference on Control, Automation and Systems (ICCAS 2018)
Oct. 17~20, 2018; YongPyong Resort, PyeongChang, GangWon, Korea

Motion planning with obstacle avoidance of an UR3 robot using charge system search

Chih-Jer Lin[1*] and Ming-Yen Li[2]

[1] Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, 10608, Taiwan (cjlin@ntut.edu.tw) * Corresponding author

[2] Graduate Institute of Automation Technology, National Taipei University of Technology, Taipei, 10608, Taiwan

Abstract: **For a cyber-physical system (CPS) of a future intelligent factory, a robotic manipulator is requested to co-work with human efficiently and safely in an environment with flexible arrangements. Therefore, an autonomous path planning of robotic manipulator is the most necessary issue to be resolved for the factory automation. For the robotic manipulator, optimizations and artificial intelligence (AI) methods are widely used to investigate the autonomous dynamic path-planning tasks with obstacle avoidance. Among these methods, the Rapidly Exploring Random Tree (RRT) algorithm has been widely used in path planning for a complex environment, because the RRT algorithm has the advantages of perfect expansion, probability completeness, and fast exploring speed. However, for some practical cases, the existing RRT algorithm may obtain a discontinuous solution of the angular trajectory. To solve the above problem, we studied a particle swarm optimization with the charge search system (CSS) to find the optimal path planning with obstacle avoidance. The steps of the proposed method are mentioned as follows: (1) establish the configuration space with the obstacle regions, (2) formulate the motion planning with obstacle using the CSS method and (3) use the PSO method to solve the path planning problem. Finally, the simulation of the path-planning task with obstacle avoidance is visually illustrated using the software RoboDK and the proposed method is implemented by the real-time experiments of the UR3 robot.**

Keywords: Motion planning, obstacle avoidance, particle swarm optimization, Charge System Search.

1. INTRODUCTION

The digital transformation with smart manufacturing is on the forefront currently and Industry 4.0 is a promising approach based on integration of the business and manufacturing processes, as well as integration of the company's value chain, supplier chain and customers. There are some key technical aspects addressed by the applications in Industry 4.0, such as Cyber-Physical Systems (CPS) and industrial Internet of Things (IoT) to the industrial production systems [1]. The CPS is defined as transformative technologies for managing interconnected systems between its physical assets and computational capabilities [2]. Furthermore, the transformation from today's factories into an Industry 4.0 fact would obtain a significantly economic potential by integrating CPS with production, logistics and services in the current industrial practices. For the CPS of the robotic manipulators, the path planning on autonomous obstacle avoidance has been given much more attention in a static environment [3] or in a dynamic environment [4].

Path planning of a robotic manipulator is defined to obtain a non-collision continuous path of its end-effector from an initial pose to a target pose in the configuration space [5]. Among the path-planning methods, the RRT algorithms has been widely used in the dynamic path planning for a complex environment. The RRT algorithm has the properties of a fast exploring speed, probability completeness, and perfect expansion; therefore, there is no need to map the obstacle from the task space to the configuration space [6]. For a path planning issue with a dynamic unstructured environment where lots of obstacles are randomly distributed, the conventional RRT algorithm cannot obtain a well solution, because the obstacle constraint may make the path generated by the RRT algorithm contain many unnecessary breakpoints [7]. Motion planning with obstacle avoidance is defined to obtain a non-collision motion trajectory of a robotic manipulator from the initial pose to a target pose in the task space. That is, the robotic manipulator does not collide the obstacles for the whole motion process. Therefore, the obstacle region of the configuration space is needed to be obtained via the simulation of all possible poses; then, the motion planning problem is to find a possible path from an initial configuration to a final configuration, such that the robot does not collide with any obstacle as it traverses the path [8].

In this study, studied a particle swarm optimization with the charge search system (CSS) to find the optimal path planning with obstacle avoidance. The steps of the proposed method are mentioned as follows: (1) establish the configuration space with the obstacle regions, (2) formulate the motion planning with obstacle using the CSS method and (3) use the PSO method to solve the path planning problem. Finally, the simulation of the path-planning task with obstacle avoidance is visually illustrated using the software RoboDK and the proposed method is implemented by the real-time experiments of the UR3 robot.

2. SYSTEM DESCRIPTION

In this study, a six DOF robotic arm (UR3) is used for the simulation and implementation, as shown in Fig. 1.

The UR3 is a robot with an effective load of 3 kg and a weight of 11 kg. It allows all wrist joints to rotate 360 degrees and the end joints can be rotated infinitely, as well as a unique force sensing device to ensure safety.

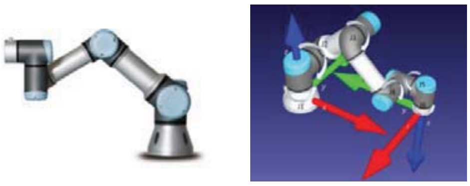

Fig. 1 (a) UR3 robot (b) Coordinate system

In this study, we establish the virtual simulation environment using RoboDK. RoboDK is an offline programming and simulation software for industrial robots. It has an intuitive user interface and it is easy to set up a virtual working environment, create a coordinate system and input the robot's joint trajectories to simulate. RoboDK supports many robotic arm and its tool modules can be easily implemented in many robotic arms, such as machining, welding, cutting, painting, and inspection. Its post-processing program supports many robotic controllers such as ABB, Fanuc LS, Kuka and the Universal Robots (UR3 or UR5), where the UR3 is used in this study. The code of RoboDK can be programmed using the language Python. Via the Python code, we can set up robots, place obstacles, set coordinate systems, set target points, and achieve simulation operations. In this study, the proposed algorithm is programmed in Matlab and RoboDK is used to illustrate the simulation for path planning with obstacle avoidance; after the simulation. After confirmation of the simulations, the path-planning results is transmitted to RoboDK, which is connected to the UR3 controller through TCP/IP; then, the real-time implements are implemented using the UR3 robot using the self-developed m-code. The block diagram is shown in Fig. 2.

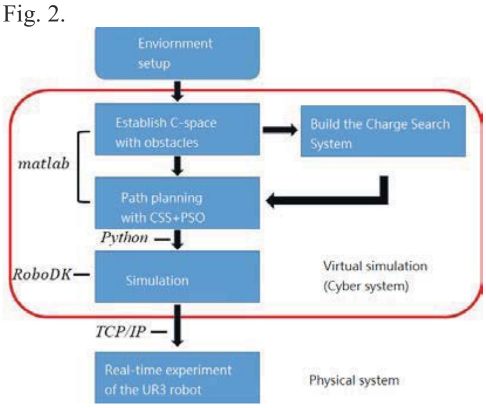

Fig. 2 block diagram of the proposed method.

3. PATH PLANNING WITH OBSTACLE AVOIDANCE

3.1 Modeling of Robot

The kinematic model of the robot is most widely presented using Denavit-Hartenberg (DH) convention. The set of four parameters relates the transformation between two successive Frames by $\alpha_i, a_i, \theta_i, d_i$. The UR3 robot installed at our lab is shown in Fig. 1. The robot can have a gripper attached to the flange of the robot. However, in this study, we only discuss the path planning with obstacle for the flange without considering the gripper. Table 1 shows the DH parameters of the UR3 robot.

Table 1 www.universal-robots.com robot [9].

J	αi(deg)	ai(mm)	θi(deg)	di(mm)
J1	0.0	0.0	0.0	118.0
J2	90.0	0.0	180.0	0.0
J3	0.0	243.7	0.0	0.0
J4	0.0	213.0	0.0	110.4
J5	-90.0	0.0	0.0	83.4
J6	90.0	0.0	180.0	82.4

Then the position and orientation of the end-effector in the inertial frame are given by

$$P_0 = {}^0A_1\,{}^1A_2\,{}^2A_3\,{}^3A_4\,{}^4A_5\,{}^5A_6 P_6 \qquad (1)$$

where P_6 is a coordinate for a given point is expressed relative to the frame 6. If $P_6 = [0\ \ 0\ \ 0\ \ 1]^T$ is substituted into Eq. (1), then we have the forward kinematic function of the robot as follows.

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \boldsymbol{f}(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \qquad (2)$$

, where (x,y,z) represents the coordinate of the flange center with respect to the world frame.

3.2 Configuration-Space

In the path planning of the robot, every possible vector of joint variables is referred to as a configuration with respect to a complete specification of the location. As a result, the vector of joint variables q provides a convenient representation of a configuration. Therefore, the set of all possible configurations is referred to as the configuration space. We will denote the configuration space by C-space.

The forward kinematic relation is used to determine the position and orientation of the end effector given the vector of joint variables. For example, consider a two-link planar arm as shown in Fig. 3, a configuration of the robot can be represented by q = (θ_1, θ_2). For example, if there is only one obstacle in the workspace and both the robot end-effector and the obstacle are convex polygons [9], we can compute the obstacle region of configuration space, which is denoted by $C_{obstacle}$. Except for $C_{obstacle}$, the other space can be considered as C_{free} , which represents the collision-free region of the C-space.

For the UR3 robot manipulator, it has six DOF of the C-space, because the number of joints are six. However, its joints can be separated into two groups, which are three for positioning and three for orientation. The first group is $(\theta_1, \theta_2, \theta_3)$, because the first three joints will be dominant for positioning of the end-effector. The second group is $(\theta_4, \theta_5, \theta_6)$, because the last three joint axes intersect at a point o_c. Therefore, the reduced C-space $(\theta_1, \theta_2, \theta_3)$ with only three DOF is used in this study to increase the computation speed for the $C_{obstacle}$. For example, if there is an obstacle (as shown in Fig. 4) in the environment for the UR3 robot, then we can obtain the C-space as shown in Fig. 5, where the dot space represents the $C_{obstacle}$. Therefore, the path planning with obstacle avoidance is to find a path from the initial configuration to the destination without passing the space $C_{obstacle}$. If we can obtain the farther path away from $C_{obstacle}$, the UR3 robot has the better safety; however, the better safety may cause the worse performance due to the larger trajectory of the joint variables. Therefore, the next section will discuss the path planning methods using the charge search system and the particle swam optimization (PSO).

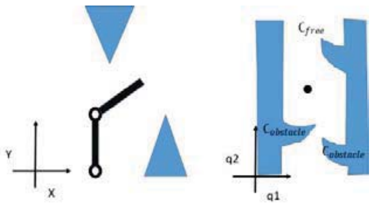

Fig. 3 An example of C-space for a two-link planar arm.

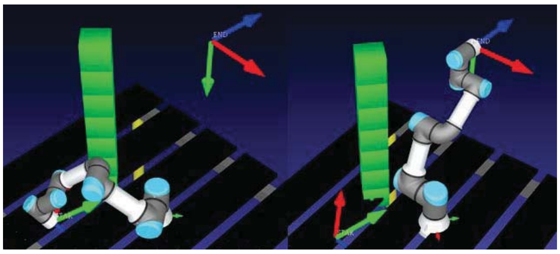
Fig.4. (a) Initial configuration (b) Final configuration.

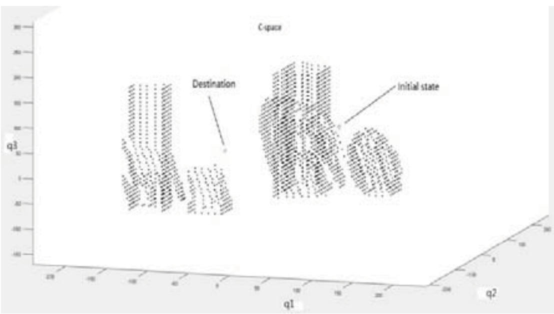

Fig. 5 An example of reduced C-space for the above case in Fig.4.

3. MAIN RESULT

In this study, we proposed a method called charge search system (CSS) to search the optimal path with obstacle avoidance for the C-space as shown in Fig. 6. The steps of the proposed method are mentioned as follows: (1) establish the configuration space with the obstacle regions, (2) the initial configuration is negatively charged and the destination is considered as a particle with positive charges. (3) the obstacles are negatively charged. Then, if we can design the suitable electricity for the initial, destination and the obstacles, then the path of the moving particle can generate the specified path with obstacle avoidance. The moving direction can be obtained based on the summation of the forces of repulsion and the forces of attraction according to the following equations.

(a) Repulsion force:

$$F_i(t) = \sum_{j=1}^{k} \frac{1}{\left|\vec{r}_i(t) - \vec{o}_j\right|^2} \tag{3}$$

, where $\vec{r}_i(t)$ is the current position of the particle and $\vec{o}_j(t)$ is the point of the obstacles.

(b) Attraction force:

$$A_i(t) = \frac{EK}{\left|\vec{r}_i(t) - \vec{G}\right|^2} \tag{4}$$

, where $\vec{r}_i(t)$ is the current position of the particle, $\vec{G}$ is the point of the destination and EK is a constant to be determined for the optimization.

(c) Resultant force:

$$T_i(t) = A_i(t) - F_i(t) \tag{5}$$

In the simulation, we can choose the different EK for different purposes. For example, the larger EK has the larger safety with the worse performance, because the trajectory will be longer. Therefore, the PSO is used to solve this optimization problem. Figure 7 illustrates the different result with the different parameters of EK [10].

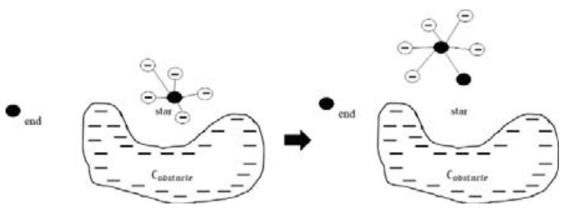

Fig. 6. Charge search system for the initial configur -ation and the destination [10].

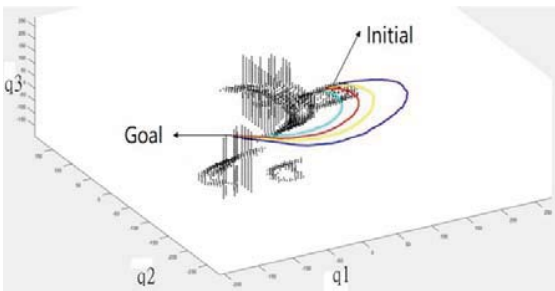

Fig. 7. Different results with the different EK [10].

4. EXPERIMENTS AND DISCUSSIONS

In the case study, the proposed method is used to obtain the path planning problem as shown in Fig. 8, where the green blocks are the obstacles to be avoided for the path planning. Figure 9-10 show the path planning result in C-space and the angular trajectory obtained using the proposed method. In this case, the orientation is determined by the last three joints and the other optimization problem is designed to control the orientation. Figure 8 (b) shows the simulation result for this case study and Figure 11 shows the implementation of the UR3 robot using the proposed method with the same initial configuration, destination and the obstacles.

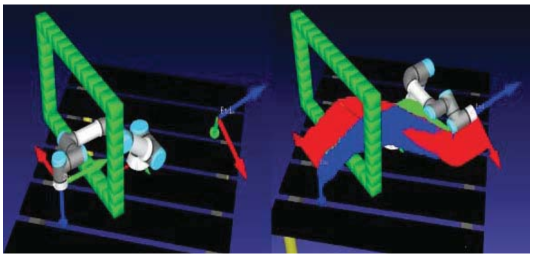

Fig. 8 (a) Initial configuration, (b) path planning result using the proposed method.

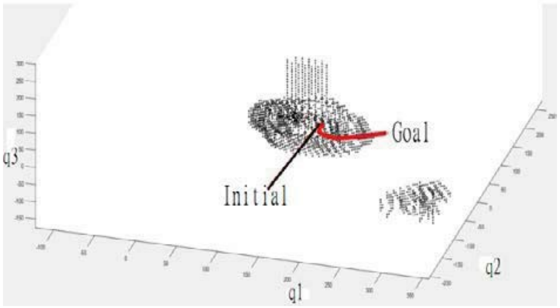

Fig. 9. Path planning result using the CCS in C-space.

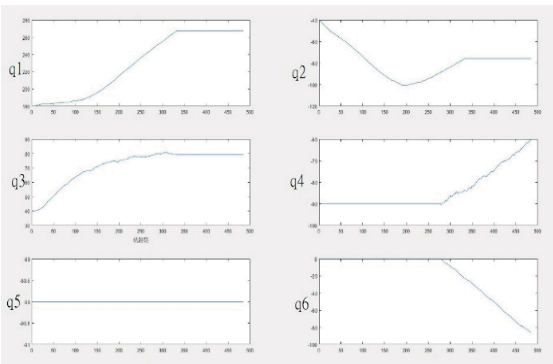

Fig.10 Angular trajectory obtained by the proposed.

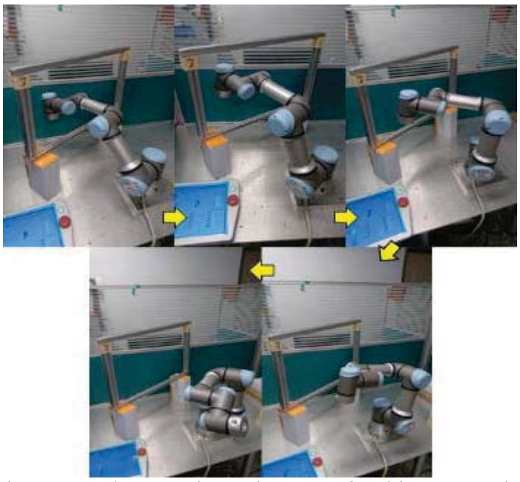

Fig. 11. Implementation using UR3 for this case study.

5. CONCLUSION

Because the path planning with obstacle is a very difficult problem for the robotic manipulators, the RRT algorithm has been widely used in path planning for a complex environment. However, for some practical cases, the existing RRT algorithm may obtain a discontinuous solution of the angular trajectory. To solve the above problem, this paper proposed a CSS method to find the optimal path planning with obstacle avoidance. Based on the resulting simulation of the path-planning task with obstacle avoidance, the proposed method can deal with the discontinuous problem of the RRT method and obtain the acceptable path planning result. The simulation is also visually illustrated using the software RoboDK. To confirm the consistence of the simulation and the actual implementation for the proposed method, the UR3 robot is used to perform the same path planning task. The experimental results validated the proposed method.

ACKNOWLEDGEMENTS

The authors would like to thank the National Science Council of the Republic of China, Taiwan for financial supporting this research under Contract No. MOST 106-2221-E-027-058.

REFERENCES

[1] A. Rojko, "Industry 4.0 Concept: Background and Overview," iJIM – Vol. 11, No. 5, pp. 77-90, 2017.

[2] R. Baheti, H. Gill, "Cyber-physical systems," Impact Control Technol , pp. 1-6, 2011.

[3] Z. C. He, Y.L. He, B. Zeng, "Obstacle avoidance path planning for robot arm based on mixed algorithm of artificial potential field method and RRT. Ind. Eng. J., Vol. 20, pp. 56-63, 2017.

[4] W. Ji, FF.Y. Cheng, D. Zhao, “Obstacle avoidance method of apple harvesting robot manipulator,” Trans. Chin. Soc. Agric. Mach., Vol. 44, pp. 253-259, 2013.

[5] L. Feng, J. H. Jia, “Improved algorithm of R RT path planning based on comparison optimization,” Comput. Eng. Appl., Vol. 47, pp. 210 -213, 2011.

[6] R.C. Luo, C. Kuo, “Intelligent seven-DOF ro bot with dynamic obstacle avoidance and 3-D object recognition for industrial cyber-physical system in manufacturing automation,” Proc. IE EE, Vol. 104, pp. 1102-1113, 2016.

[7] K. Wei, B. Ren, “A method on dynamic path planning for robotic manipulator autonomous obstacle avoidance based on an improved RRT algorithm,” Sensors, Vol. 18, 571, 2018.

[8] Mark W. Spong, Seth Hutchinson, and M. Vidyasagar, “Robot Modeling and Control,” JOHN WILEY & SONS, INC.

[9] www.universal-robots.com

[10] M.Y. Li, “Simulation and implementation of motion planning with obstacle avoidance for an universal robot,” (in Chinese), master thesis for Graduate Institute of Automation Technology, National Taipei University of Technology, 2017.