Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Supplementary materials for

Jin WANG, Shengjie LI, Haiyun ZHANG, Guodong LU, Yichang FENG, Peng WANG, Jituo LI, 2023. A distributed variable density path search and simplification method for industrial manipulators with end-effector's attitude constraints. *Front Inform Technol Electron Eng*, 24(4):536-552. https://doi.org/10.1631/FITEE.2200353

1 Collision detection based on SDF

Collision models of obstacles are constructed based on signed-distance-field (SDF). SDF consists of a series of sampling point clouds, and each point of SDF has a signed number, whose absolute value is the shortest distance from the obstacles. If the sampling point is outside the obstacle, the signed number is positive; otherwise, it is negative. In this way, collision detection can be carried out quickly by comparing the value of the sampling point with the radius of a collision sphere, as shown in Fig. S1, where SN_1 and SN_2 represent the signed values of different sampling points.

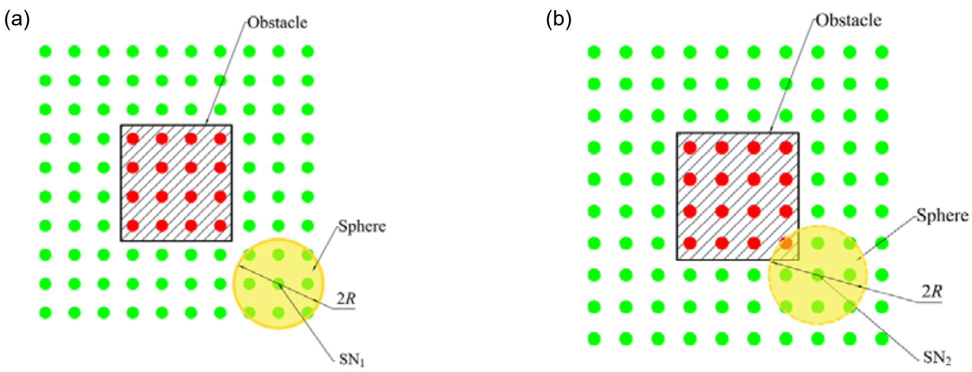

Fig. S1 Schematic of collision detection based on SDF: (a) no collision ($SN_1>R$), (b) a collision ($SN_2\leq R$)

2 Existing path simplification approach

Since the path planning in this study is based on grid-based search, there are redundant bends in the path. To achieve optimal planning results, it is necessary to remove the redundant path points. Existing methods tend to start with the farthest nodes in terms of connection relationships, such as the method of Fu et al. (2018):

1. Define node p_i at the starting point;
2. Establish the local paths between p_i and subsequent non-adjacent path points p_j;
3. Find the collision-free local path which contains node p_j with the maximum subscript j, use p_i–p_j to replace the corresponding segment of the initially planned path, and then assign j to i;
4. If there is no collision-free local path, then i=i+1;
5. Repeat steps 2–4 until p_j reaches the last point.

The process is shown in Fig. S2.

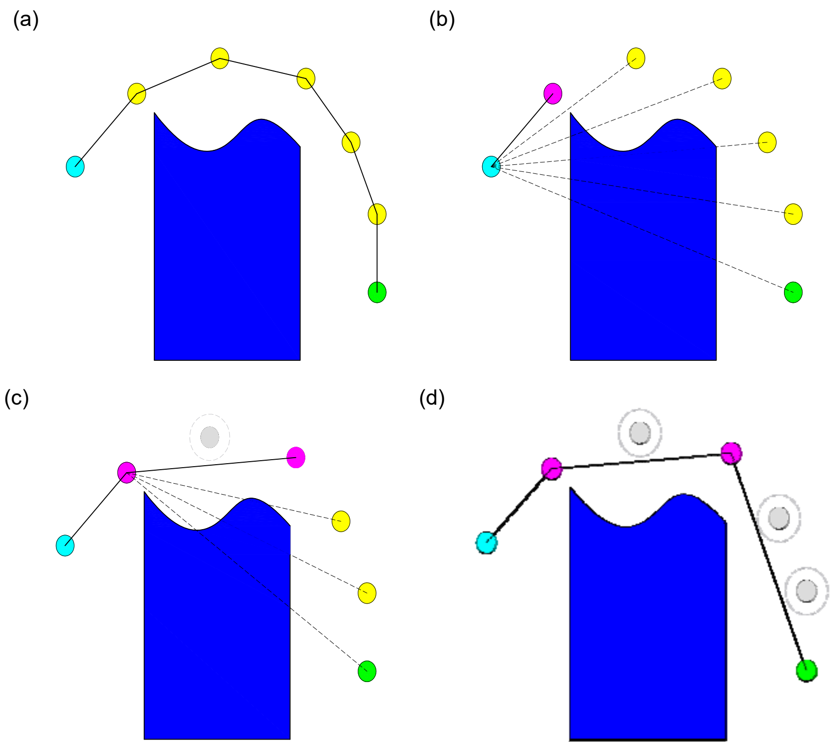

Fig. S2 Simplification method of Fu et al. (2018): (a) original path; (b) finding the next path point; (c) eliminating redundant points; (d) completing the simplification

3 Setup of simulation cases

The simulation cases consist of a robot, a conveyor belt, and two shelves, simulating the processes of taking out the workpiece from the shelf and placing it on the conveyor belt. Based on this, the starting point, ending point, and positions of obstacles are changed to form 10 different scenes, some of which are shown in Fig. S3.

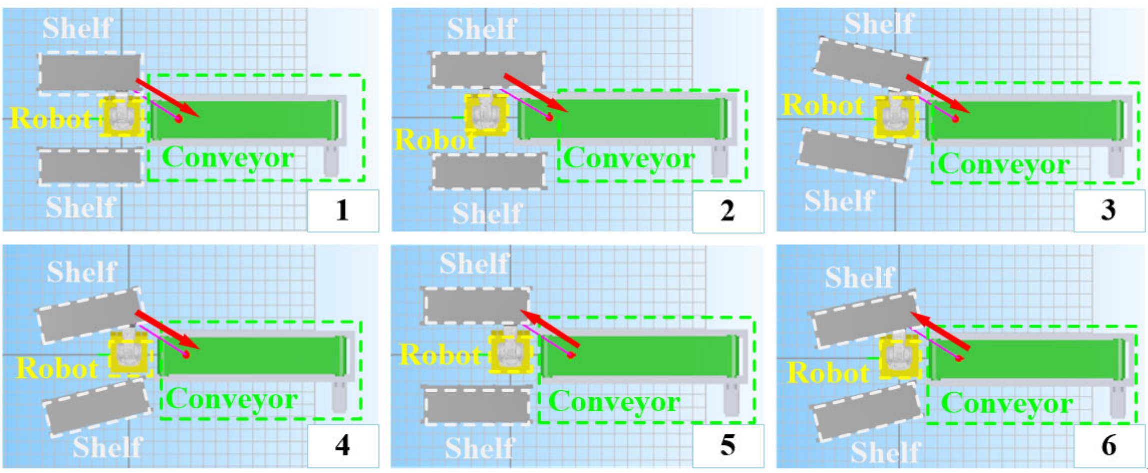

Fig. S3 Some simulation cases and their setup

4 Motion process in ablation study

In case 1, there is a robot, a CNC milling machine, a couple of aluminum brackets (obstacles), and a table. The robot was ABB IRB120, required to avoid obstacles and move to the CNC milling machine. The motion process is shown in Fig. S4. In case 2, the robot was required to complete loading and unloading actions. While waiting for processing, an obstacle was added so that the manipulator cannot return along the original path. The motion process is shown in Fig. S5.

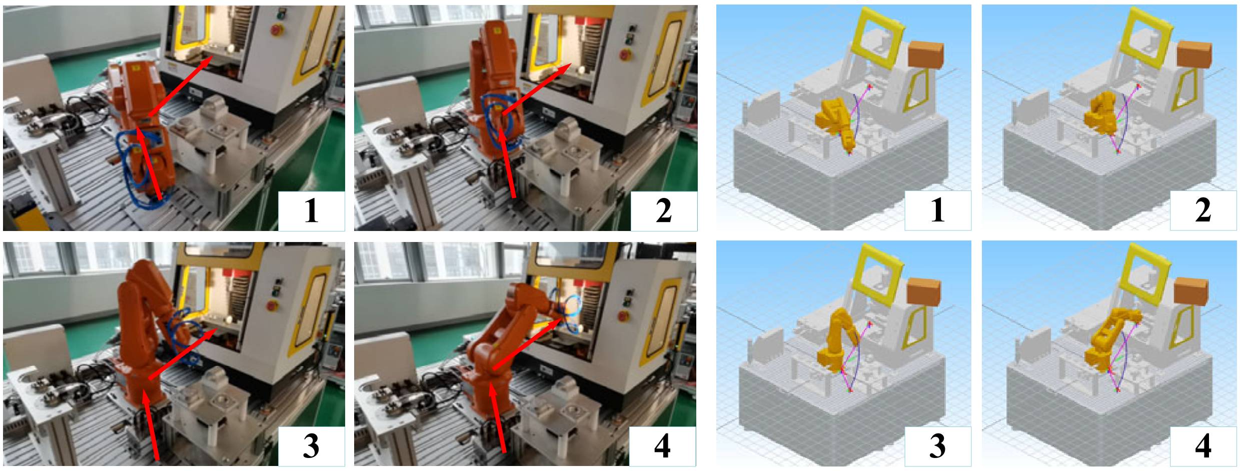

Fig. S4 Motion process of case 1

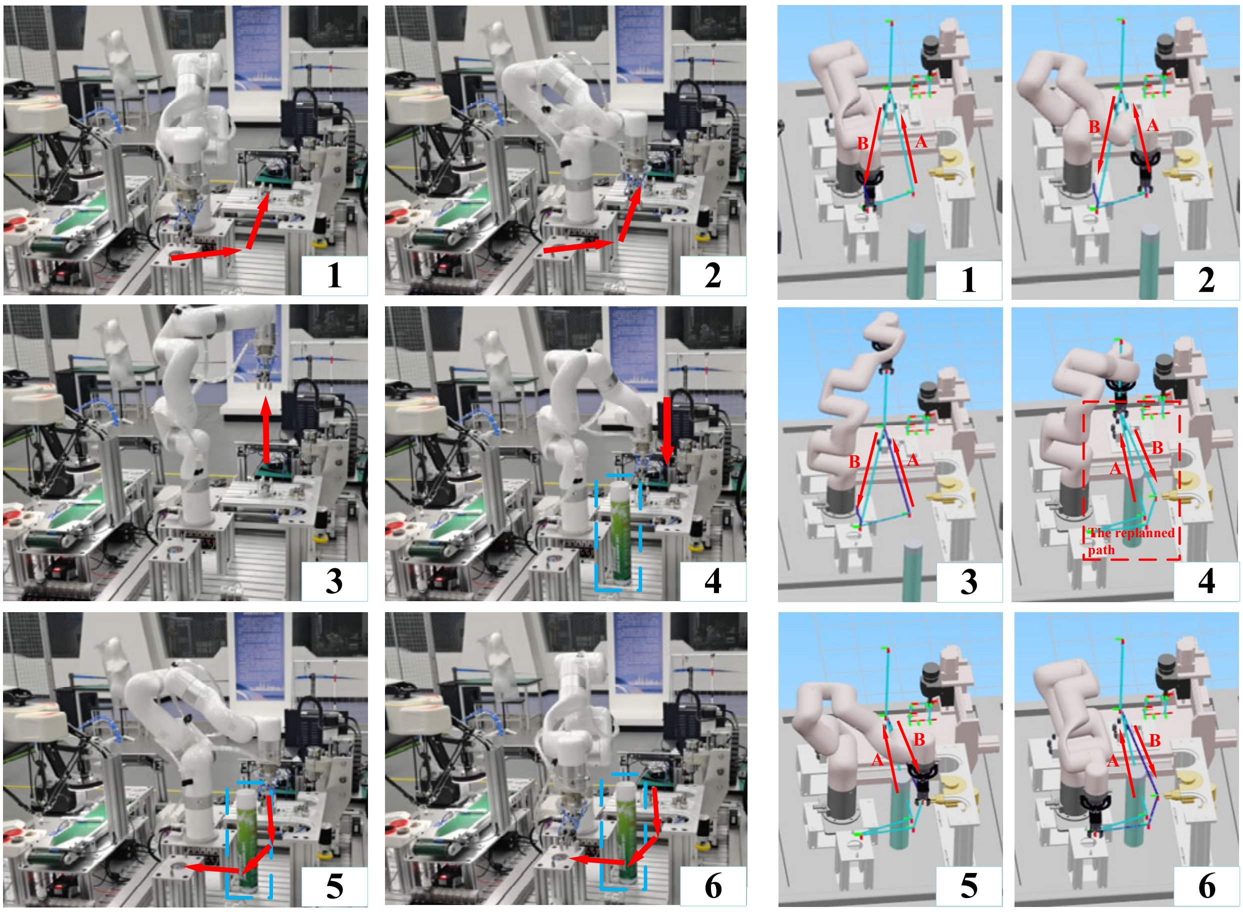

Fig. S5 Motion process of case 2, in which the additional obstacle and replanning results have been marked

5 Motion process in baseline comparison

By observing the motion process in simulation, it can be found that the algorithms of the three control groups (CHOMP, RRT-connect, and BFMT*) fail to maintain the correct attitude of the end-effector, as shown in Fig. S6.

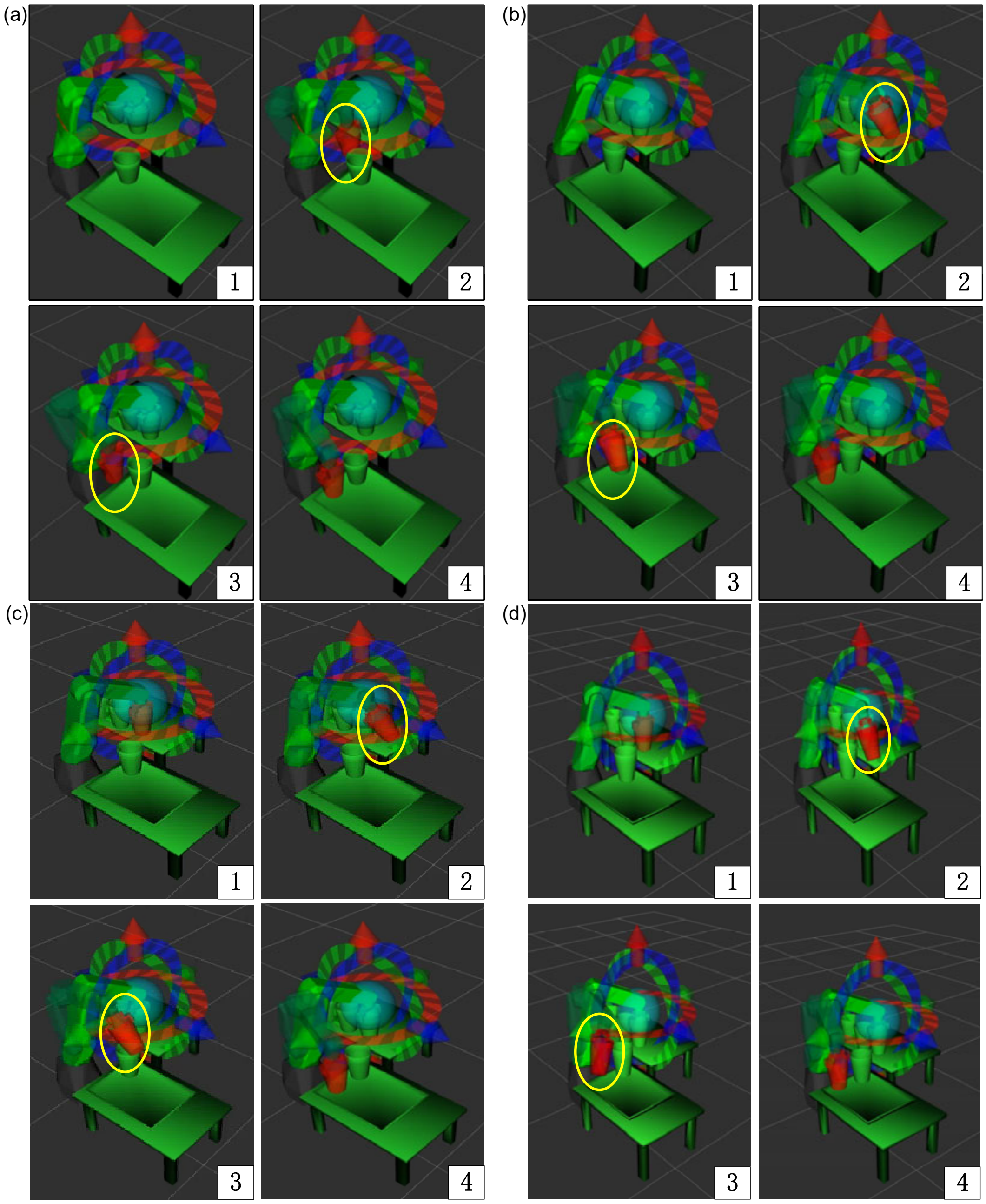

Fig. S6 Motion process comparison of case 3: (a) DVDP-AC; (b) CHOMP; (c) RRT-connect; (d) BFMT*

Reference

Fu B, Chen L, Zhou YT, et al., 2018. An improved A* algorithm for the industrial robot path planning with high success rate and short length. *Robot Auton Syst*, 106:26-37. https://doi.org/10.1016/j.robot.2018.04.007