Applied Mathematics and Nonlinear Sciences 6(2) (2021) 43–52

Applied Mathematics and Nonlinear Sciences

https://www.sciendo.com

# Cartesian space robot manipulator clamping movement in ROS simulation and experiment

Longtao Mu[1]†, Yunfei Zhou[2], Tiebiao Zhao[3]

[1]School of Mechanical Engineering, Shaanxi Polytechnic Institute, Xianyang 712000, China

[2]School of Management, Xi'an University of Finance and Economics, Xi'an 710100, China

[3]School of Engineering, University of California Merced, Merced, CA 95340, USA

**Submission Info**

Communicated by Juan Luis García Guirao
Received September 29th 2020
Accepted November 30th 2020
Available online February 15th 2021

**Abstract**

This paper studies the robot arm sorting position control based on robot operation system (ROS), which works depending on the characteristics of the robot arm sorting operation using the top method, to automate the sorting operation and improve the work efficiency of workpiece sorting. Through the ROS MoveIt! module, the sorting pose and movement path of the robotic arm are planned, the inverse kinematics of the sorting robotic arm is solved, and the movement pose characteristics of the sorting robotic arm are analysed. The robot arm model was created using Solidworks software, and the URDF model file of the robot arm was exported through the sw2urdf plugin conversion tool, and the parameters were configured. Based on ROS for 6-degree-of-freedom (DOF) robot motion simulation, random extended tree (RRT) algorithm from open motion planning library (OMPL) is selected. The robot motion planning analysis and sorting manipulator drive UR5 manipulator. The results show that the sorting pose and motion trajectory of the robot arm are determined by controlling the sorting pose of the sorting robot arm, and the maximum radius value of the tool centre point (TCP) rotation of the robot arm and the position of the workpiece are obtained. This method can improve the success rate of industrial sorting robots in grabbing objects. This analysis is of great significance to the research of robots' autonomous object grabbing.

**Keywords:** industrial robot; universal robot; robot operation system; kinematic simulation
**AMS 2010 codes:** ???

## 1 Introduction

The research of robots on its various applications involves various fields such as military, aerospace, medicine, agriculture, industrial products, etc. It has achieved rapid development, and has gradually become a hot topic in various fields [1,2]. In recent years, the number of robots in the manufacturing industry has increased

†Corresponding author.
Email address: clever19880817@163.com

ISSN 2444-8656 doi:10.2478/AMNS.2021.1.00011

rapidly by 500,000 units, including finishing of manufacturing links/operations such as welding, spraying and assembly, loading and unloading, and sorting, and has partially replaced manual operations [3,4]. Robot applications will arrive in many fields, replace heavy labour, and realise automation and intelligence in the industrial field.

Currently, universities and research institutes have conducted robotics research and achieved important results. Literature [5] analysed the positive and negative solutions of the kinematics of the robotic with three-dimensional simulation technology and established the visual interface of the robotic arm sorting system, and realised the robotic operation simulation. The literature [6] analysed the prevention of accidents that may be caused by the entity during the operation of the robot in kinematics simulation [7,8]. Literature [9] developed the three-dimensional configuration module, motion simulation module, communication module and main control module of the robot with the surrounding environment, and completed the kinematics simulation of the robot. Aiming to solve the problem of sorting, the literature [10] studied the robot kinematics and established the human-computer interaction interface of the six degree-of-freedom (DOF) manipulator. Literature [11] proposed the inverse solution method and robot calibration technology to realise the eye-in-hand calibration of the robot arm. The joint angle was obtained from the positive solution, and the robot calibration technology can perform error compensation [12, 13]. The above-mentioned methods mainly carry out simulation analysis for traditional simulation modules. The methods do not use robot operating system (ROS) for simulation and physical robot communication for its research work.

With the development of Artificial Intelligence (AI), the research level of robot technology has been improved rapidly, and the combined computer vision and robot technology has become the current research hotspot. The convolutional neural network (CNN) [14] was proposed by Hinton et al., and has achieved success in object recognition, object detection and other fields. Using of computer vision technology to identify objects, and controlling the robot to grasp has become an important field of research [15–17]. In order to map between the coordinate information of the object in the three-dimensional space and the pixel information in the image, the camera was calibrated by this method [18]. Reference [19] connects the robot system and the vision system through eye-to-hand calibration, and then obtains the three-dimensional coordinates of the object in the robot base coordinate transformation, and completes the grasping task of the robot [20,21]. The deep learning was focussed for robot grasping in the literature from the aspect of robot grasping research [13,22]. Levine et al., collected the experimental data of robot grasping and trained the neural network model [23]. The objects were grasped by robot successfully using the convolution neural network and the robot completed the grasping action in virtual simulation.

At present, workpieces of complex sorting remain in manual operation phase except the contour occlusion of the workpiece operation. There is a large complex industrial device error and only low success is achieved. Based on the proposed ROS Cartesian space, the experiment achieved an industrial robot for the automatic sorting of complex parts. The use of industrial robot was accomplished in the established simulation environment in ROS software. The application manipulator solved inverse kinematics and achieved the Cartesian space target workpiece. In addition, the use of the ROS Protocol Standard is a great significance for industrial production. By establishing the robot sorting operation, the simulation experiments proved that the proposed method has good stability and accuracy. Section 1 introduces the motivation of this research, and few relevant backgrounds. Section 2 surveys the Robot System Operation, Universal robot (UR) and the software. Section 3 outlines our experiment and the corresponding results, discusses some aspects of operating the robot error and graph as illustrated in graphs. Finally, Section 4 concludes the paper.

## 2 Materials and methods

### 2.1 Development and application

ROS is an integrated development platform, composed of complex functions package, which includes a mechanical arm motion planning, operation control, 3D perception, kinematics modules, and the plugins that can be used in industrial, research and other fields. ROS of MoveIt! Mechanical arms control module can be implemented to control a robotic arm manipulator model, including modelling, kinematics, motion planning, and obstacle avoidance research.

### 2.2 Structure of simulation system for robot

Considering the model of the 6-DOF universal robot manipulator, the simulation platform of sorting system based on ROS is shown in Figure 1. The simulation platform of the sorting system is mainly composed of three units: workpiece platform unit, vision unit and manipulator body unit. The workpiece position and trajectory are used to place the workpiece in different positions. The vision part is composed of Kinect V2, camera bracket and computer processing software. The vision system unit calculates the position and posture of the robot and transmits calculated parameters to the control cabinet.

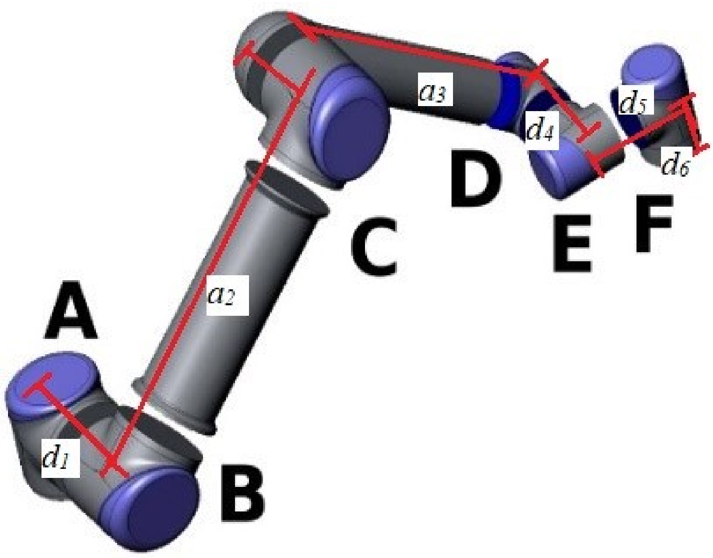



**Fig. 1** UR5 robotic arm with 6 degrees of freedom.
Note: Joint A: Base, B: Shoulder, C: Elbow and D, E, F: Wrist 3, 2, 1; Picture source: Universal Robot.

### 2.3 Manipulator simulation function package

The MoveIt! controller has three features. MoveIt! is a part of the ROS system and used to control the multi-joint robotic arm. It provides control plugins and tools for the robotic arm, which can realise the rapid configuration of the robotic arm control and encapsulate the thousands of APIs, secondary application on MoveIt! module. In addition, the three-dimensional visualisation tool RViz in ROS can reconstruct the working scenes of various industrial robotic arms in the simulation environment. In RViz, the interactive marker on the end-effector can be used to move the manipulator to the target position. Next, one can select a specific planning algorithm from the OMPL library from the drop-down menu in the planning library. Then, one can click the plan button – the Rviz window will appear from the initial position to the target position of the trajectory, and loop constantly plays.

Now, the simulation condition and process of ROS simulation system are described. First, the hardware part of ROS simulation that includes creating subclass, inheriting robotHW parent class are described and some functions and variables in URDF (Unified robot description format) are declared. The joint and motor data are initialised, and the corresponding data ($d_1$=89.2, $a_2$=-425, $a_3$=-392, $d_4$=109.3, $d_5$=94.75, $d_6$=82.5) are registered using the hardware interface. Second step is robot simulation motion planning layer. The motion planning layer plays a very important role in the autonomous grasping of manipulator. Finally, the path of the manipulator is a series of independent position points in the process of moving the end-effector to the target position. The trajectory is based on the path by adding speed (0-1.0 m/s), acceleration constraints and time parameters to make the manipulator move more smoothly.

### 2.4 Robotic arm modelling

Customising the URDF file was made from the robotic arm prototype. The structure shape and joint position of this robot arm are shown in Figure 1. The manipulator model used in the article is a UR5 manipulator produced by Universal Robot of Denmark, which has six degrees of freedom, excellent flexibility, safe and a unique and innovative cooperative manipulator. The traditional D-H (Denavit-Hartenberg) modelling method establishes a link coordinate system for each joint. The D-H parameters of the UR5 type robot arm are shown in Table 1.

**Table 1** D-H parameters of UR5 robotic arm.

| *Joint* | $\alpha_{i-1}$ | $v_{i-1}$/*mm* | $d_1$/mm | $\theta_{i/}$(°) | Joint range/(°) |
|---|---|---|---|---|---|
| A | $\alpha_1$=90 | 0 | $d_1$=89.2 | $\theta_1(0)$ | -360∼ 360 |
| B | 0 | $a_2$=-425 | 0 | $\theta_2(0)$ | -360∼ 360 |
| C | 0 | $a_3$=-392 | 0 | $\theta_3(0)$ | -360∼ 360 |
| D | $\alpha_4$=90 | 0 | $d_4$=109.3 | $\theta_4(0)$ | -360∼ 360 |
| E | $\alpha_5$=-90 | 0 | $d_5$=94.75 | $\theta_5(0)$ | -360∼ 360 |
| F | 0 | 0 | $d_6$=82.5 | $\theta_6(0)$ | -360∼ 360 |

### 2.5 MoveIt! configuration setup

Through MoveIt! (ROS toolkit), setup assistant was installed the sorting robot arm as shown in Figure 2. The configuration steps include loading the sorting robot arm URDF model, configuring the self-collision matrix, creating a planning group, defining the sorting robot posture (initial position home and front position forward), and configuring the end-effector (Gripper), generate configuration files, etc. For three-dimensional model of the manipulator in an emulation environment and based on the use of mechanical unity arm described format URDF(Unified Description Robot the format) model, three-dimensional modelling Solidworks software created manipulator model exported as URDF robot arm model file using sw2urdf conversion tool in the interface. In this framework, MoveIt! is executed to avoid the collision of the robot arm. By uploading the collision matrix of the robot arm containing the rule set into the integrated toolbox, the motion planning of the robot arm can avoid singular trajectories. The loading robot arm collide toolbox integrated into the matrix so that the mechanical arm motion avoid special tracks and the purpose of avoiding the singular position in ROS toolkit is achieved.

When creating a planning group, select the Random Extended Tree Algorithm (RRT) algorithm in the Open Motion Planning Library (OMPL) to analyse the motion planning of the sorting robot arm. In Robotic arm solution algorithm, starting position $P_{start}$, targeting object position $P_{goal}$, randomly select the next object position $P_{rand}$, and the sorting robot explores one step at a time, finds the random position $P_{rand}$ every time, and finds the closest goal for object $P_{near}$ – move $P_{near}$ along the direction of $P_{rand}$ by the step length $\rho$, and explore the new position $P_{new}$. If the point is closer to the end point than the original point, the new point is selected until the search for the target fruit position ends after approaching a straight path.

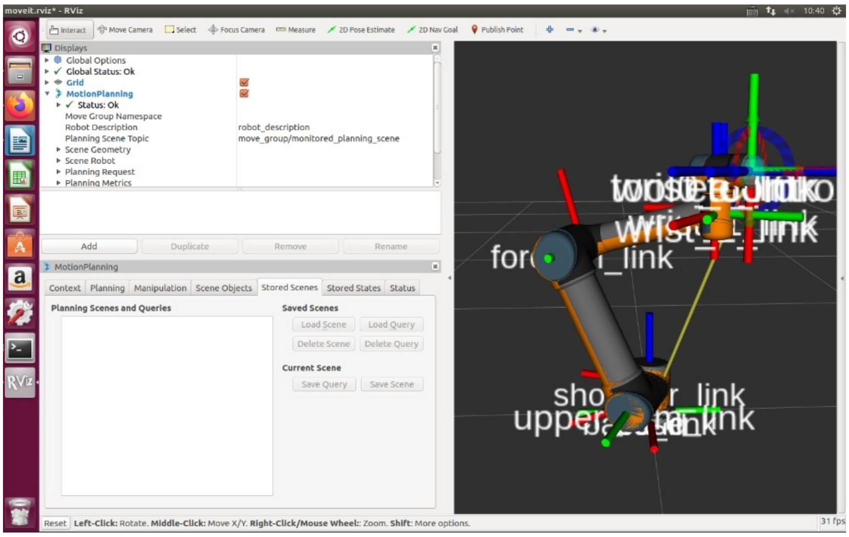


**Fig. 2** MoveIt! plugin of ROS.

To mark the emulator (Cartesian space) manipulator terminal (target position by tool_link) to a target position – click operation Planning instruction, then the mechanical actuator arm path planning instruction, set RRT path planning for 5 times, and obtain optimal operation path in MoveIt! toolkit. By clicking the Execute command, the robotic arm solves the inverse kinematics solution and runs the shortest path planning result.

## 2.6 Basic principles of motion planning

(1) Positive kinematics solution

Table 1 in UR5 robot arm D-H parameters may establish its coordinates *i* sitting standard system *i-1* homogeneous transformation matrix, the matrix rotation or translational motion relative to the current coordinate system transformation, the robot arm. The D-H parameters are substituted into Equation 1, and the transformation matrix of all adjacent coordinate systems is obtained.

With the transformation matrix of coordinates between the TCP of the sorting robot arm and the base and input the joint angle $\theta_i$ of the sorting robot arm, and we calculate the transformation matrix $T$ of the tool end coordinates of the sorting robot arm in the base coordinate system $T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T$.The coordinates of the TCP are expressed on the teach pendant as $[x, y, z, r_x, r_y, r_z]$, where $[x, y, z]$ and $[r_x, r_y, r_z]$ are three-dimensional Cartesian coordinates (spatial position) and coordinate rotation vector (spatial attitude), respectively. Turn the transformation matrix $T$ into 6-valued coordinates $[x, y, z, r_x, r_y, r_z]$, as in formula expressed in the form of matrix (1):

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

In the formula, the upper left corner of $T$ is the rotation matrix $R = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$; the upper right corner of $T$ $3 \times 1$ is $P= [r_x, r_y, r_z]$ is the translation matrix of the end joint.

Transform the matrix $T$ to obtain the 6-value coordinates of the manipulator's pose and complete the positive kinematics solution of the manipulator. Solving the $T$ matrix of the positive kinematics solution of the sorting robot can give the TCP pose of the sorting robot in its base coordinates.

(2) Solving inverse kinematics

The joint angle is calculated from the space position and posture of the end-effector. The analytical method can obtain all the solutions of the sorting manipulator. The UR5 6-degree-of-freedom manipulator has finite solutions. The intermediate matrix for the process of solving the transformation matrix $T$ is calculated. The Jacobian matrix in the inverse kinematics of the sorting manipulator is also derived from the $T$ matrix.

Since the corresponding rows and columns of the matrices $T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T$ on both sides of the equation ${}^0_1T^{-1}\,{}^5_6T^{-1} = {}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T = {}^1_5T$ are equal and the following equation is obtained:

$$-(s_1 a_x - c_1 a_y) + s_1 p_x - c_1 p_y = d_4 \tag{2}$$

Solving $\theta_1 = \pm\arccos\left(\frac{d_4}{\sqrt{(d_6 a_y - p_y)^2 + (p_x - d_6 a_x)^2}}\right) + \arctan\left(\frac{p_x - d_6 a_x}{d_6 a_y - p_y}\right)$, two solutions are obtained.

Solving the arctangent function (*arctan*), the return value is $(-\pi, +\pi]$. The return value of the arc cosine is $[0, \pi]$ which ensures that the two solutions are different in the range of $2\pi$. From the third row and third column, we get $s_1 a_x - c_1 a_y = c_5$, and two solutions can be obtained $\theta_5 = \pm\arccos(s_1 a_x - c_1 a_y)$.

From the third row and second column, we obtain $(s_1 n_x - c_1 n_y)s_6 + (s_1 o_x - c_1 o_y) = 0$. The solutions of joints $\theta_1$, $\theta_2$, $\theta_3$, $\theta_4$, $\theta_5$, $\theta_6$ are obtained by using the following equation: $\theta_6 = \arctan\left(-\frac{s_1 o_x - c_1 o_y}{s_1 n_x - c_1 n_y}\right)$.

From the above expression, it can be concluded that the UR5 manipulator has 8 sets of inverse solutions. However, for gripping operations, make sure that the tool end of the robotic arm (installation of the end-effector) is vertically downward, approaching the target workpiece from the top. Therefore, set the initial position (position) of the robotic arm to ensure that the wrist joints (E, F) of the robotic arm are perpendicular to the base plane (horizontal plane).

## 3 Results and discussion

### 3.1 Preset target motion planning

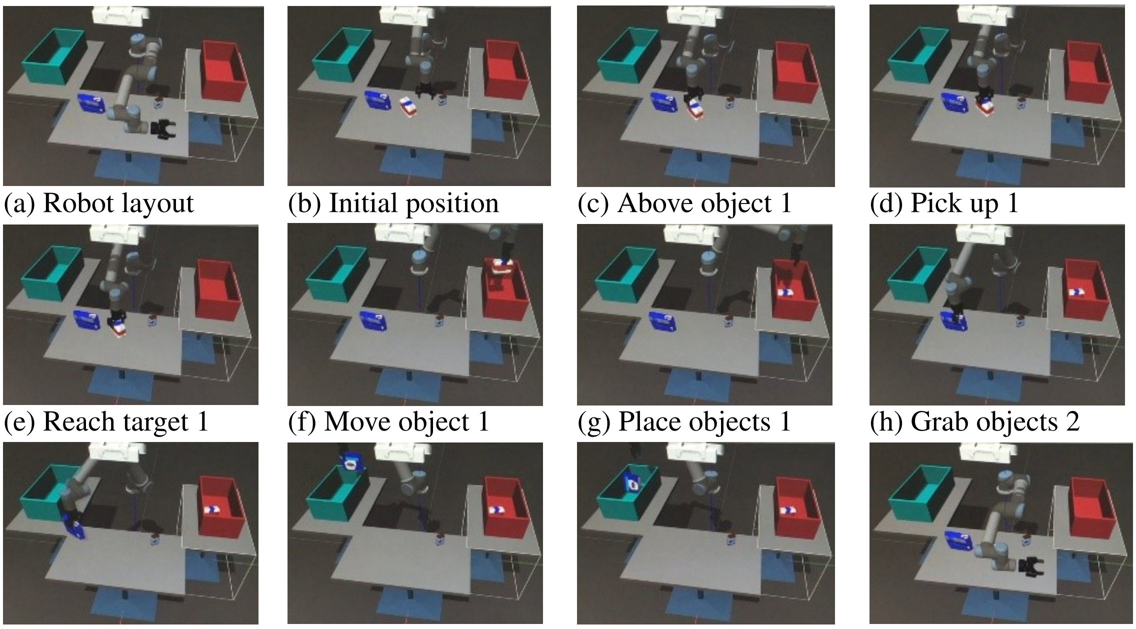
Further, a series of positions (points) are generated between the target position points to ensure that the start and end of the movement are the constraints of the vertical pose. The above positions represent the time optimal path based on the speed and acceleration of each joint, and the time step is 0.1 s. The path is parameterised using the time-optimised path, and the generated path makes the robotic arm to move smoothly and reduces the moving time of the sorting robotic arm.

### 3.2 Sorting obstacle avoidance movement planning

The ROS simulator RViz performs path planning for the UR5 robotic arm sorting operation. After the sorting robot arm reaches the target position, the robot arm moves from the initial position to the top position of the sorting target, and then to the joint position where it reaches the target position, to simulate the UR5 robot arm object grasping movement process. Select the random extended tree algorithm (RRT) algorithm in the OMPL for the motion planning of the sorting manipulator. The ROS MoveIt! toolkit executes the path planning results (Run Planning and Execute in turn), and finally completes the motion path planning of the robotic arm sorting process, driving the real robotic arm to approach the target fruit from the top and grab the target workpiece. The sorting action of UR5 robotic arm and end-effector approach vertically upwards from the top, as shown in Figures 3 and 4.

### 3.3 Discussion

Realise topic (rostopic information) through ROS hardware connection of main computer vision processing computer and robotic arm controller (router connected to 3 controllers), and the software program operated in the debugging process. By rqt_plot change View manipulator joint position, velocity, force and the mechanical actuator arm during movement of the joint position change curve as in Figure 4. Randomly robot gripping position of the workpiece by RViz view manipulator torque of each joint (the simulator the Torque) value, are shown in Table 2.



(a) Robot layout (b) Initial position (c) Above object 1 (d) Pick up 1
(e) Reach target 1 (f) Move object 1 (g) Place objects 1 (h) Grab objects 2
(i) Pick up object 2 (j) Reach the target 2 (k) Place the object 2 (l) Reset

**Fig. 3** UR5 robotic arm grasping simulation experiment.

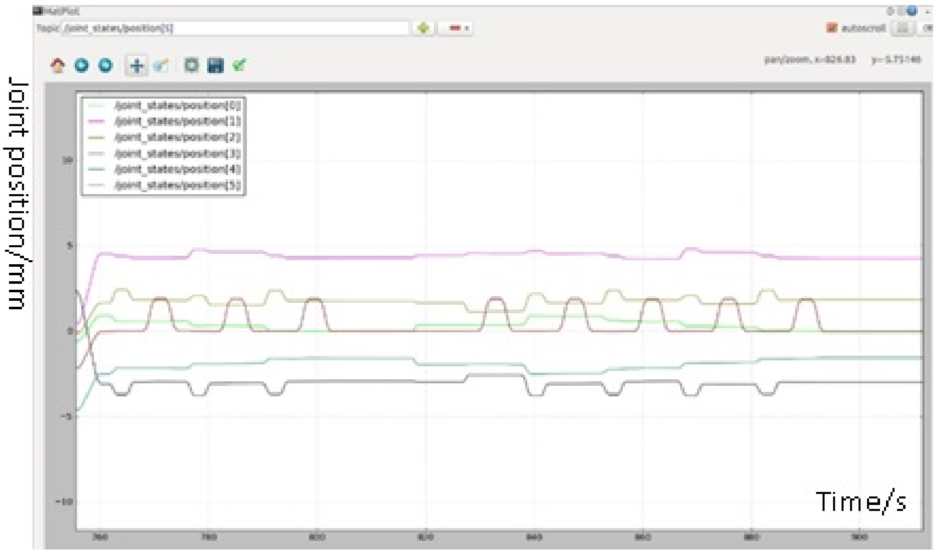



**Fig. 4** The change curve of the joint position of the robotic arm.
Note: Joint F is green line (joint_states/position[0]), Joint E is red line (joint_states/position[1]), Joint D is yellow line (joint_states/position[2]), Joint C is brown line (joint_states/position[3]), Joint B is blue line (joint_states/position[4]), Joint A is orange line (joint_states/position[5]).

Through the ROS plug-in, we can view the running situation of the real-time robot, as shown in Figures 3 and 4, and we can get the position changes and speed changes of the six DOF running robots. It can be concluded

**Table 2** Torque change value of UR5 robotic arm picking action (unit Nm).

| **Joint torque** | Joint *F* | Joint *E* | Joint *D* | Joint C | Joint *B* | Joint *A* |
|---|---|---|---|---|---|---|
| Initial position | -0.00 | -3.51 | 18.75 | 0.81 | -0.00 | 0.00 |
| Joint position | 0.00 | 26.21 | 20.10 | 0.81 | -0.00 | 0.00 |
| Target location | -0.00 | 26.66 | 18.85 | 0.76 | -0.00 | 0.00 |

from Figure 4 and Table 2 that the tool end position of the sorting manipulator remains vertical, which ensures that the end-effector can approach the target workpiece from the top and carry out the grabbing and handling actions of the target workpiece. Select the motion path of the manipulator according to the distance of the handling position, execute the path planning result through the ROS MoveIt! toolkit, and finally complete the motion path planning of the robot arm sorting process. In addition, a small joint_state position is defined in the motion control and task decision-making layer of the ROS simulation software in the process of grasping the manipulator, which can select different data according to different task types, object positions and object types and send them to the following motion planning layer.

The robot decision-making module sends a grab status to the task decision-making layer of the manipulator. After receiving the status, the task decision-making layer will search in the state table stored in advance. If the matching and grasping status is successful, it will take out the state chain in the table and put it into the state queue. Each time the state controller analyses the data it needs according to the sub actions in the current state queue. The grabbing state can be divided into many sub actions: initial, preparation, grasping, paw opening, gripper clamping, holding object, etc. At this time, if the first action in the queue is initial one, the controller will take the corresponding position from the preset position pool of the robot, and take the required data from the Moveit! parameters table, and transfer the initial action to the motion planning layer for specification and execution. When the robot has completed the action, the task decision-making layer will compare the difference between the actual position of the manipulator and the expected position.

The Kinect sensor is used as a visual collector to study the object recognition and robot's grasping and handling. The visual recognition model is used to identify the object type in real-time. The internal and external parameters of Kinect V2 are obtained by calibration method, and the object position coordinates are calculated. Finally, the robot joints are controlled to complete the task of grasping and transporting objects. At the same time, the inverse kinematics solution and robot calibration technology are proposed. The joint angle can be obtained effectively by inverse kinematics solution, and the robot calibration technology can be used for error compensation. If the error is less than a certain prescribed value, it will return the execution result to the controller, and the controller will continue to execute the next state until the actions in the whole state queue are executed. If the error is too large, an error exit will be reported to prevent any uncontrollable accident in the manipulator.

## 4 Conclusion

In order to realise the industrial robot arm accurately grasp the target objects, the simulation analysis of the motion path planning of the robot arm was carried out. We arrive at following conclusions based on our analysis and work.

Using the ROS MoveIt! module, the robot arm's sorting position and motion path are planned, the inverse kinematics of the sorting robot arm is solved and the motion pose characteristics of the sorting robot arm are analysed to realise the robot arm sorting. The process of approaching and grabbing the target workpiece from the top of the job is achieved. Based on the motion control of the 6-DOF robot arm under ROS software, the shortest motion trajectory and sorting pose of the sorting robot arm are determined, and maximum radius of rotation of the TCP of the sorting robot arm and the target grab point position are obtained.

**Acknowledgements:** This research was conducted at the School of Mechanical Engineering, Shaanxi Polytechnic Institute, and was supported by research grants from the Shaanxi Polytechnic Institute doctoral research start-up fund project funding (Project number: 2020-05). The authors declare no conflicting financial interests.

## References

[1] Wang T. The development of robotics technology [J]. Robot, 2017, 39(04):385-386.
[2] Bechar A, Vigneault C. Agricultural robots for field operations: Concepts and components[J]. Biosystems Engineering, 2016, 149:94-111.
[3] Wang X, Tang B, Gu X. Research on obstacle avoidance strategy of welding robot [J]. Chinese Journal of Mechanical Engineering, 2019, 55(17):77-84
[4] Gao F, Guo W. Thoughts on the development strategy of Chinese robots [J]. Chinese Journal of Mechanical Engineering, 2016, 52(07):1-5.
[5] Yan Y, ROCKETS, Jiang C. Based on OpenGL study of the manipulator control system simulation platform [J]. Computer Simulation, 2006, (08): 252-257.
[6] Zhang S, Ming D, Wang X, et al. Based on Visual C++ of 6R design robot simulation platform program [J]. Shanghai Second Polytechnic University, 2011, 28 (01): 54-59.
[7] Bardinas J P, Aldrich C, Napier L F A. Predicting the Operating States of Grinding Circuits by Use of Recurrence Texture Analysis of Time Series Data[J]. Processes, 2018, 6(2):1-19.
[8] Cicek E, Dasdemir J, Zergeroglu E. Coordinated synchronization of multiple robot manipulators with dynamical uncertainty[J]. Transactions of the Institute of Measurement and Control, 2015, 37(5):672-683.
[9] Sanfilippo F, Helgerud E, Stadheim P A, et al. Serpens: A Highly Compliant Low-Cost ROS-Based Snake Robot with Series Elastic Actuators, Stereoscopic Vision and a Screw-Less Assembly Mechanism[J]. Applied Sciences-Basel, 2019, 9(3):1-20.
[10] Lavrenov R O, Magid E A, Matsuno F, et al. Development and Implementation of Spline-based Path Planning Algorithm in ROS/Gazebo Environment[J]. Trudy SPIIRAN, 2019, 18(1):57-84.
[11] Kollmitz M, Eitel A, Vasquez A, et al. Deep 3D perception of people and their mobility aids[J]. Robotics and Autonomous Systems, 2019, 114:29-40.
[12] Tardioli D, Parasuraman R, Ogren P. Pound: A multi-master ROS node for reducing delay and jitter in wireless multi-robot networks[J]. Robotics and Autonomous Systems, 2019, 111:73-87.
[13] Huang L, Wang B, Ni T, et al. Research on Robot Grasping System Based on Kinect [J]. Transactions of the Chinese Society of Agricultural Machinery, 2019, 50(01): 390-399.
[14] Krizhevsky A, Sutskever I, Hinton G E. ImageNet Classification with Deep Convolutional Neural Networks[J]. Communications of the Acm, 2017, 60(6):84-90.
[15] Meng S, Liang Y, Shi H. Motion planning of six-degree-of-freedom manipulator based on ROS platform [J]. Journal of Shanghai Jiaotong University, 2016, 50(S1): 94-97
[16] Grosu V, Grosu S, Vanderborght B, et al. Multi-Axis Force Sensor for Human-Robot Interaction Sensing in a Rehabilitation Robotic Device[J]. Sensors, 2017, 17(6):1-15.
[17] Shi D, Zhang W, Zhang W, et al. A Review on Lower Limb Rehabilitation Exoskeleton Robots[J]. Chinese Journal of Mechanical Engineering, 2019, 32(04):12-22.
[18] Tardioli D, Parasuraman R, Ogren P. Pound: A multi-master ROS node for reducing delay and jitter in wireless multi-robot networks [J]. Robotics and Autonomous Systems, 2019, 111:73-87.
[19] Yang K-M O, Han J-B, seob s h, et al. An Integrated Multi-robot Control System based on ROS for Autonomous Exploring Disaster Environment[J]. Journal of Institute of Control, Robotics and Systems, 2019, 25(6):534-540.
[20] Zenkevich S L, Nazarova A V, Hua Z. Simulation and Analysis of the Movement of a Group of Mobile Robots in ROS[J]. Mekhatronika, avtomatizatsiya, upravlenie, 2017, 18(5):317-320.
[21] Zenkevich S L, Zhu H, Huo J. Experimental Study of Motion of the Mobile Robots Moving in the Convoy Type Formation[J]. Mekhatronika, avtomatizatsiya, upravlenie, 2018, 19(5):331-335.
[22] Wei K, Ren B. A Method on Dynamic Path Planning for Robotic Manipulator Autonomous Obstacle Avoidance Based on an Improved RRT Algorithm[J]. Sensors, 2018, 18(2):1-18.
[23] Lopes A T, de Aguiar E, De Souza A F, et al. Facial expression recognition with Convolutional Neural Networks: Coping with few data and the training sample order[J]. Pattern Recognition, 2017, 61:610-628.