International Journal of Automation and Computing
www.ijac.net

18(6), December 2021, 926-934
DOI: 10.1007/s11633-021-1311-2

# Mechanical Design and Dynamic Compliance Control of Lightweight Manipulator

Shao-Lin Zhang[1] Yue-Guang Ge[1,2] Hai-Tao Wang[1,2] Shuo Wang[1,2]

[1] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

[2] University of Chinese Academy of Sciences, Beijing 100049, China

**Abstract:** In the existing modular joint design and control methods of collaborative robots, the inertia of the manipulator link is large, the dynamic trajectory planning ability is weak, the collision stop safety strategy is dependent, and the adaptability and safety to the changing environment are limited. This paper develops a six-degree-of-freedom lightweight collaborative manipulator with real-time dynamic trajectory planning and active compliance control. Firstly, a novel motor installation, joint transmission, and link design method is put forward to reduce the inertia of the links and improve intrinsic safety. At the same time, to enhance the dynamic operation capability and quick response of the manipulator, a smooth planning of position and orientation under initial/end pose and velocity constraints is proposed. The adaptability to the environment is improved by the active compliance control. Finally, experiments are carried out to verify the effectiveness of the proposed design, planning, and control methods.

**Keywords:** Lightweight manipulator, mechanical design, dynamic trajectory planning, compliance control, robot control.

**Citation:** S. L. Zhang, Y. G. Ge, H. T. Wang, S. Wang. Mechanical design and dynamic compliance control of lightweight manipulator. *International Journal of Automation and Computing*, vol.18, no.6, pp.926–934, 2021. http://doi.org/10.1007/s11633-021-1311-2



## 1 Introduction

Collaborative robots can adapt to new tasks through manual guiding and teaching and work together with humans to achieve task goals. The design and control of collaborative robots is a hot topic of current research. Different from other industrial robots, collaborative robots usually work in an unstructured environment. It is necessary to consider issues such as the safety of the shared workspace with people, work efficiency, and adaptability to the environment. There are high requirements for mechanical design, trajectory planning, and control.

At present, most of the mechanical designs of collaborative robots are similar. The motors, encoders, brakes, and reducers are packaged into drive modules, which are connected in sequence through links to form a multi-degree-of-freedom manipulator, such as light-weight robot (LWR)[1] and universal robot(UR)[2]. In this design, the utilization efficiency of motors is not high, and the ratio of self-weight of the manipulator to the load is large. When the power of the motor is increased, the load of link is also increased so that the increases in velocity and acceleration are limited. Moreover, the installation method of the motor increases the inertia of the manipulator, which reduces the safety of collisions. A reduction in the inertia of the links is usually achieved by optimizing the size of the manipulator and choosing lighter materials. However, the improvement of this kind of methods is limited. One method is to use only one motor to provide the driving torque for all joints, transmit the power to each joint through transmission mechanisms such as gears and worm gears, and use pulse-width modulation (PWM) speed control clutches to control the motion of each joint[3]. This method can reduce the weight and volume of the manipulator and is often used in aviation and other situations where the weight of the manipulator is strictly limited. However, this method only uses one motor to drive, and the joint torque cannot be calculated from the motor current. The bandwidth of the clutch is low and reduces the control performance of the manipulator. Another method is to install the motors near the base and drive the joints through ropes[4]. This method designs a complex tension amplifying mechanism so that the joints have high stiffness. However, the joint movement near the base affects the rope length of the end joints. A complicated mechanical design of the joint routing is required, which increases the difficulty of the system design

Research Article
Manuscript received June 9, 2021; accepted September 26, 2021; published online November 6, 2021
Recommended by Associate Editor Qing-Long Han
Colored figures are available in the online version at https://link.springer.com/journal/11633

and manufacture. Therefore, it is necessary to design a lightweight collaborative manipulator to improve the safety of collaborative operations by reducing the inertia of links.

At present, collaborative robots are not only used for tasks with fixed trajectories such as handling, welding, spraying, etc., but also combined with reinforcement learning, knowledge graphs, and other methods to improve the competence for other dynamic and flexible tasks[5]. This kind of methods usually gives a changing target state and requires high flexibility and environmental adaptability for robot trajectory planning. Generally, the trajectory planning of collaborative robots includes commands such as lines and circles and cannot plan from any initial pose and velocity to a target pose and velocity. Although there are already some methods to transit between adjacent lines, circles, and orientations[6, 7], they can only achieve a smooth transition in predetermined trajectories. This greatly limits the efficiency of collaborative robots in dynamic and flexible tasks. Therefore, some scholars have proposed methods of planning based on polynomial curves[8, 9]. The target trajectory can be obtained by solving the polynomial parameters. This method is used in the trajectory planning of the table tennis robot system. When the target state of the robot changes, the robot trajectory is adjusted to adapt to the new state. This method is easy to calculate and can meet the state constraints of the beginning and end points, but ignores the velocity and acceleration constraints of the intermediate points, which may exceed the range of robot motion capabilities. Another method sets optimization goals based on smoothness and trajectory time and solves the smooth planning problem through optimization search[10]. However, this method takes a long time to solve and may not be able to get a suitable solution in time. In addition, the non-uniform rational basis spline (NURBS) curve[11] can also be used to fit the trajectory and interpolate within the range of velocity and acceleration constraints. This method smoothly connects discrete points in three-dimensional space with NURBS curves according to the set maximum deviation. The trajectory can be adjusted by the number and weight of control points. The trajectory has good adjustability and strong adaptability to discrete point sequences. However, this method requires a large amount of calculation for trajectory solving and interpolation. It usually requires offline calculations. Using the spline curve for real-time dynamic planning needs further research to make each point on the trajectory meet the kinematic constraints.

In terms of safety, collaborative robots rely on collision detection and response strategies in unstructured environments, require high sensor detection accuracy, and have weak active adaptability to the environment[12]. When there is an obstacle in the trajectory or the pose of the operation object changes, the manipulator is stopped, and a new teaching trajectory is required. Passive flexibility based on flexible components[13] and active flexibility based on impedance control[14] are widely used to solve this problem. However, this kind of methods ignores trajectory tracking while achieving compliance control. In addition, there are trajectories obtained based on reinforcement learning[15, 16], which have specific requirements for the manipulator, operation object, and environment. It is difficult to adapt to changes in the manipulator and environmental parameters. Therefore, it is necessary to study the trajectory following and compliant control method under environmental constraints to improve efficiency. The target trajectory is accurately followed when there is no obstacle. Otherwise, it ensures that the force of the manipulator does not exceed the permitted range.

Therefore, this paper aims to develop a lightweight collaborative manipulator with the ability to operate efficiently and safely. The manipulator has a real-time dynamic and smooth trajectory planning and can adapt to obstacles in the changing environment. Specifically, the manipulator has the following characteristics:

1) A novel design of the motor installation and transmission system to reduce the inertia of the links and make the mechanical system intrinsically safe;

2) Real-time position and orientation planning method based on spline curve, to realize efficient trajectory planning from arbitrary initial pose and velocity to target pose and velocity;

3) The six-degree-of-freedom force position hybrid control method to realize trajectory following and external force limitation so that the manipulator can approach the target point in a changing environment and improve the success rate of task execution.

The following chapters are organized as follows. Section 2 introduces the mechanical design of the lightweight manipulator. Section 3 analyzes the real-time smooth trajectory planning method of the manipulator. Section 4 introduces the active compliance control method of the manipulator. Section 5 verifies the effectiveness of the proposed design, planning, and control methods through experiments. Finally, Section 6 gives the conclusions of the paper.

## 2 Lightweight mechanical design

The mechanical design of the lightweight manipulator is shown in Fig. 1. The manipulator has six degrees of freedom, all of which are rotary joints. In order to reduce the inertia of the links and achieve larger velocity and acceleration, high-power motors are required. If the motors are installed in each joint, the joint mass increases, and links with better rigidity are required, making the mechanical design bulky and limiting the velocity and acceleration. To effectively utilize the driving power, the motors of all joints are installed on the base and the first joint, and the joints are driven through the transmission mech-

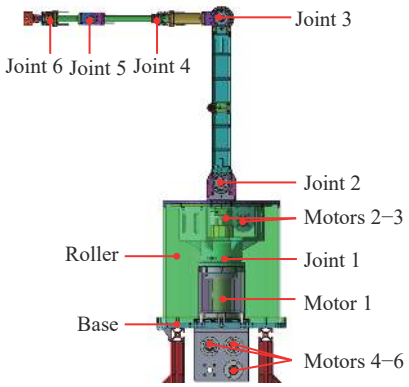



Fig. 1 Lightweight mechanical design of the manipulator

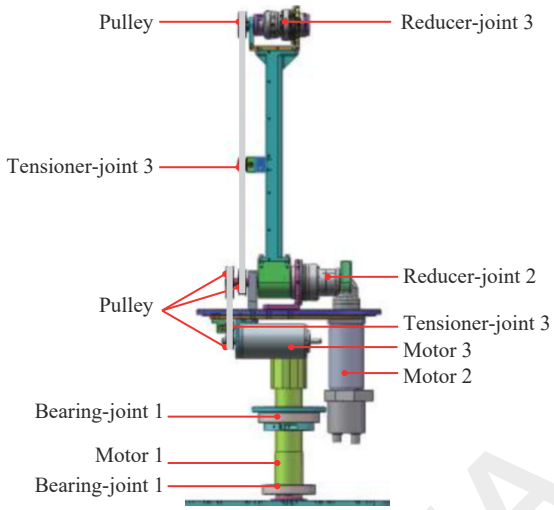



Fig. 2 Mechanical transmission structure of joints 1–3

anism.

A compact transmission system is designed to reduce the inertia of the links. The transmission mechanism of joints 1–3 are shown in Fig. 2. The motors 1–3 are installed in the roller on the base. The rotor of motor 1 is locked on the base with a tensioning mechanism, and the stator is equipped with two bearings and rotates with the first joint. Motor 2 drives the second joint through a 90-degree angle reducer. Motor 3 decelerates through two synchronous belts, transmits power to the third rotation axis, and then drives the third joint through a harmonic reducer. The third joint has two synchronous pulleys coaxially installed with the second joint and is designed as a nested structure. Thus, the movement of the second joint does not affect the third joint.

Joints 4–6 are far from the base, which is inconvenient to drive through gears and links. Therefore, the motors and joints are directly connected by wire ropes, as shown in Fig. 3. The motors 4–6 are installed under the base. A wire wheel is installed on the motor shaft. There are multiple holes on the wheel, and some of the holes are threaded to fasten the wire rope. Joints 4 and 6 change the direction of wire rope through guide wheels perpendicular to the axis of joint 5. The axes of joints 4–6 intersect at one point and can get Pieper's solution of the inverse kinematics for the multi-joint robot, making it convenient for calculating the analytical solution.

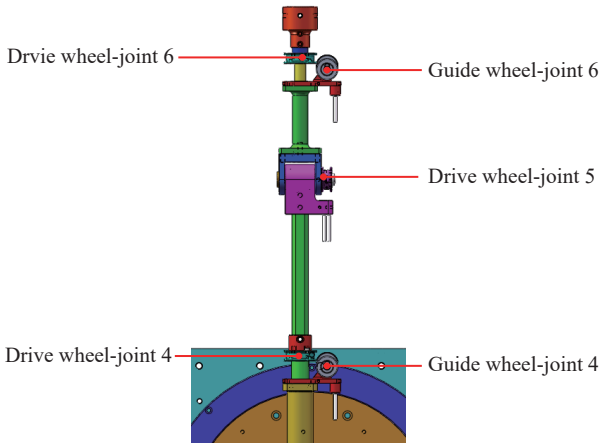



Fig. 3 Mechanical transmission structure of joints 4–6

## 3 Real-time dynamic trajectory planning

During the robot's movement, specifically when the target pose and velocity change, replanning is required immediately to adapt to the new task. This section explains separately from the position planning and orientation planning.

### 3.1 Position planning

Consider the movement shown in Fig. 4. The current position of the robot's end effector is $P_i$, and the velocity is $V_i$. The target task changes immediately, and the next point becomes $P_{i+1}$ with a velocity of $V_{i+1}$.

Bezier curve is used for planning. For calculation efficiency, a single cubic Bezier curve is preferred for planning. To satisfy the continuity at points $P_i$ and $P_{i+1}$, points $P_i$ and $P_{i+1}$ are selected as control points. At the same time, add points $A$ and $B$ as intermediate control points. In order to ensure that the velocity direction does not change suddenly at the points $P_i$ and $P_{i+1}$, the direction of $P_iA$ is set the same as the velocity $V_i$, and the direction of $BP_{i+1}$ is set the same as the velocity $V_{i+1}$:

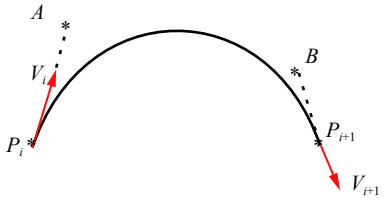



Fig. 4 Schematic diagram of position planning

$$\begin{aligned} P_iA &= kV_i \\ BP_{i+1} &= kV_{i+1}. \end{aligned} \tag{1}$$

The parameter $k$ is used to set points $A$ and $B$. To improve the smoothness of the movement, a constraint[17] is added as

$$J = \int_0^1 C''(\mu) \cdot C''(\mu) \mathrm{d}\mu. \tag{2}$$

The minimum value of $J$ is solved to make the curve as smooth as possible. Let $\mathrm{d}J/\mathrm{d}k = 0$, then

$$k = \frac{(a_i+a_{i+1})(x_{i+1}-x_i)+(b_i+b_{i+1})(y_{i+1}-y_i)+(c_i+c_{i+1})(z_{i+1}-z_i)}{2(a_i^2+a_ia_{i+1}+a_{i+1}^2+b_i^2+b_ib_{i+1}+b_{i+1}^2+c_i^2+c_ic_{i+1}+c_{i+1}^2)} \tag{3}$$

where

$$\begin{aligned} P_i &= (x_i, y_i, z_i) \\ P_{i+1} &= (x_{i+1}, y_{i+1}, z_{i+1}) \\ V_i &= (a_i, b_i, c_i) \\ V_{i+1} &= (a_{i+1}, b_{i+1}, c_{i+1}). \end{aligned} \tag{4}$$

In applications, to maintain the velocity direction, $k$ should be larger than 0. Set the minimum value of $k$ to $k_{\min}$:

$$\mathrm{k} = \begin{cases} k, & \text{if } k > k_{\min} \\ k_{\min}, & \text{if } k \le k_{\min}. \end{cases} \tag{5}$$

Then the Bezier curve given by points $P_i$, $A$, $B$, and $P_{i+1}$ has good smoothness and meets the requirements of continuous position and consistent velocity direction.

### 3.2 Orientation planning

In addition to position planning, orientation planning is also required in the Cartesian space. The robot's current orientation is expressed by quaternion as $o_i$, the rotation axis is $z_i$, and the angular velocity is $\omega_i$. The target orientation is expressed by quaternion as $o_{i+1}$, the rotation axis is $z_{i+1}$, and the angular velocity is $\omega_{i+1}$. The relationship between the orientations is shown in Fig. 5. $o_A$ is obtained by turning the posture $o_i$ around the axis $z_i$ by angle $k\omega_i$. $o_B$ is obtained by turning the posture $o_{i+1}$ around the axis $z_{i+1}$ by angle $-k\omega_{i+1}$. $k$ is similar to the $k$ in position planning.

$$\begin{aligned} o_{i,A} &= [\cos(k\omega_i/2), z_i] \\ o_{i+1,B} &= [\cos(-k\omega_{i+1}/2), z_{i+1}] \\ o_A &= o_{i,A}\mathrm{o_i o_{i,A}^{-1}} \\ o_B &= o_{i+1,\mathrm{B}}o_{i+1}o_{i+1,B}^{-1}. \end{aligned} \tag{6}$$

The spherical Bezier curve[18] is used for planning:

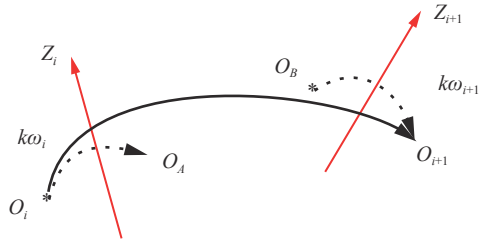



Fig. 5 Schematic diagram of orientation planning

$$C(\mu) = o_i^{(1-\mu)^3} o_A^{3\mu(1-\mu)^2} o_B^{3\mu^2(1-\mu)} o_{i+1}^{\mu^3}. \tag{7}$$

Similar to position planning, the optimal value of $k$ can also be obtained by means of minimum curvature. However, it is difficult to obtain an analytical solution[19] and a numerical iteration method is required. The numerical iteration method requires a large amount of calculation, which affects the real-time performance of dynamic operations. Therefore, the following method is used to calculate $o_{i,A}$ and $o_{B,i+1}$:

$$\begin{aligned} o_{i,A} &= [\cos(\alpha\theta_1/2), z_i] \\ o_{B,\mathrm{i+1}} &= [\cos(\alpha\theta_2/2), z_{i+1}] \end{aligned} \tag{8}$$

where $\theta_1$ and $\theta_2$ are the deceleration angles that reduce the angular velocity $\omega_1$ and $\omega_2$ to 0 under the constraints of the maximum deceleration and jerk, $\alpha$ is a positive real number.

The spherical Bezier curve determined by points $o_i$, $A$, $B$ and $o_{i+1}$ can be obtained by (7) and (8). The interpolation method refers to our previous work[7].

## 4 Active compliance control

The lightweight manipulator has a certain degree of safety due to the small link inertia. To further improve the adaptability and exploration ability in the unstructured environment, active flexibility is applied according to the force of the end effector. Consider the force of the manipulator as follows:

$$M(q)\ddot{q} + V(q,\dot{q}) + G(q) + \tau_{\mathrm{f}} - \tau_d = \tau_{ext} \tag{9}$$

where $q, \dot{q}, \ddot{q}$ are the angle, angular velocity, and angular acceleration of the joints, $M(q)\ddot{q}$ is the inertial force of the links, $V(q,\dot{q})$ is the centrifugal force and Coriolis force, $G(q)$ is gravity, $\tau_f$, $\tau_d$, and $\tau_{ext}$ are the joint friction torque, joint driving force, and external torque on the joint respectively.

The dynamic parameters in (9) can be obtained by dynamic and friction parameter calibration[20]. The joint torque $\tau_d$ can be calculated from the motor current. Then the external torque on the joint $\tau_{ext}$ can be calculated when the robot is in contact with the environment. The relationship between the force in Cartesian space $F_{ext}$ and joint torque $\tau_{ext}$ is

$$F_{ext} = J^{-\mathrm{T}}\tau_{ext} \tag{10}$$

where $J$ is the Jacobian matrix.

Based on the force of the end effector in Cartesian space $F_{ext}$, the active flexibility is applied as

$$F_{ext} = F_d + K_p X_e + K_b \dot{X}_e \tag{11}$$

where $X_e$ is the trajectory following error in the Cartesian space, $F_d$ is the expected environmental forces, $K_p$ and $K_b$ are real numbers, representing stiffness and damping coefficients.

From (9)–(11), we can get:

$$K_p X_e + K_b \dot{X}_e = J^{-\mathrm{T}}(M(q)\,\ddot{q} + V(q,\dot{q}) + G(q) + \tau_{\mathrm{f}} - \tau_d) - F_d. \tag{12}$$

Equation (12) is calculated in the six dimensions of position and posture, respectively. Usually, $F_{ext}$ calculated based on motor current has obvious noise. Therefore, a threshold $F_{\min,i}$ is introduced. When $F_{ext,i} \le F_{\min,i}$, $X_{e,i} = 0$. In this case, the manipulator follows the planned trajectory. When $F_{ext,i} > F_{\min,i}$, $X_e$ is calculated by (12). The stiffness coefficient is set as shown in Fig. 6. When $F_{ext}$ becomes larger, $K_p$ gradually decreases, which helps to suppress noise.

The block diagram of the control system is shown in Fig. 7. The target pose $P_{i+1}, o_{i+1}$ and velocity $V_{i+1}, \omega_{i+1}$ are input to the trajectory planning module. $X_r = \{P_r, o_r\}$ is obtained by the position and orientation planning under the velocity and acceleration constraints according to the method in Section 3. The external force detection module calculates the external joint torque based on the motor current and the dynamic model of the manipulator. $X_e = \{P_e, o_e\}$ is obtained by (12), added with $X_r$, and output to the inverse kinematics calculation module.

The following analyzes the contact characteristics of the system and the environment. For simplicity, let $K_b = 0$. When the robot has no contact with the external environment, the impedance control does not work, and the trajectory following module is used to follow the planned trajectory. When the robot is in contact with the environment, there are $F_{ext,i} > F_{\min,i}$ in one or more directions, and the compliance control module has output $X_{e,i}$. Suppose the maximum value of $F_{ext,i}$ is $F_{\max,i}$. The impedance output satisfies

$$X_{e,i} \le (F_{\max,i} - F_{d,i})/K_{p,i}. \tag{13}$$

If the output of the trajectory plan satisfies

$$\Delta X_{r,i} < \Delta X_{e,i} \tag{14}$$

the manipulator moves in the direction of $F_{ext,i}$ to reduce $F_{ext,i}$. Therefore, $\Delta X_{r,i}$ is limited by the maximum velocity and acceleration to satisfy (14), or the trajectory following is stopped when $F_{ext,i} > F_{\min,i}$.

## 5 Experiments

The developed manipulator is shown in Fig. 8. The corresponding control system is also designed, and the key parameters are shown in Table 1. Except for the base and first joint, the mass of other links is about 4 kg. Other manipulators with a similar operating radius, such as JAKA ZU5 and UR5, have a mass of more than 20 kg. It can be seen that the mechanical design can effectively reduce the mass of links and reduce the impact of a collision.

In the control system of the manipulator, the Cartesian space trajectory planning, interpolation, and transition method[6, 7] are realized, the dynamic parameters are calibrated, and the external force is detected[20, 21]. The following experiments are carried out separately from

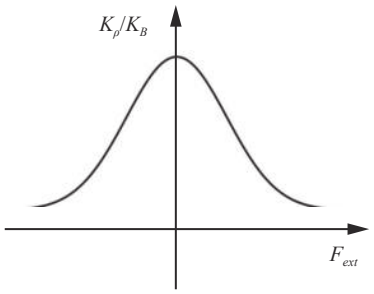



Fig. 6 Setting of $K_p/K_b$

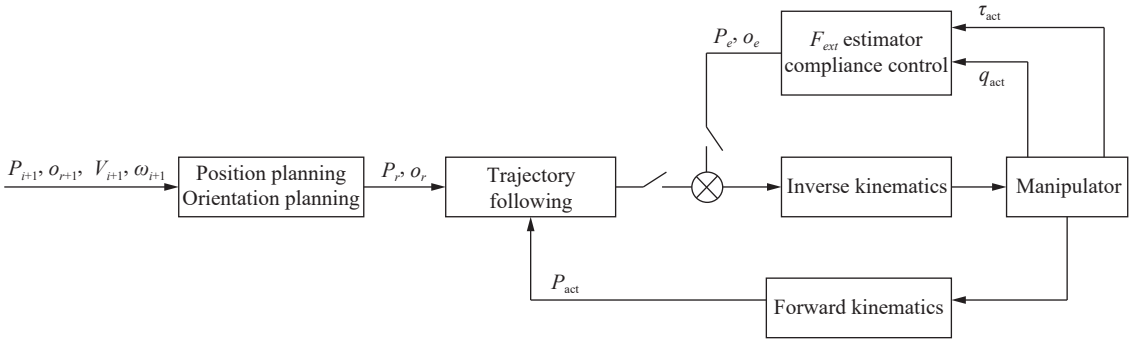



Fig. 7 Block diagram of robot control system

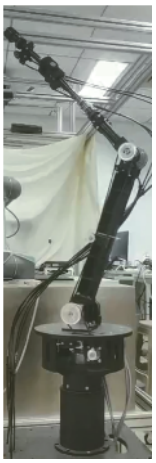


Fig. 8 Lightweight manipulator

Table 1 Main parameters of manipulator

| Item | Parameter |
|---|---|
| Controller | Beckhoff CX5130 |
| Driver | Maxon EPOS2 70/10 |
| Motor | Maxon RE50, RE65 |
| Bus | CANopen |
| Operating radius | About 1.1 m |
| Mass (No base and joint 1) | About 4 kg |
| Max velocity (End effector) | 5 m/s |

real-time dynamic trajectory planning and compliance control.

## 5.1 Real-time dynamic trajectory planning

In order to test the feasibility of the proposed position and orientation planning method, the initial point of the manipulator is set to $A$ and moves along the $Y$-axis. When the end effector moves to point $B$, the target point is set to $C$, and point $C$ has velocity and angular velocity constraints. When the manipulator reaches point $C$, the target is set to stop at point $D$. The poses $A$, $B$, $C$ and $D$ in the Cartesian space are set as shown in Table 2.

Points $A$, $B$, $C$ and $D$ can be obtained by teaching in the Cartesian space. The velocity settings of the four points are set as shown in Table 3. It can be seen that the movement from point $A$ to point $D$ includes three trajectories, which cover: 1) The initial velocity and angular velocity are 0, the end velocity and angular velocity are not 0. 2) The initial velocity and angular velocity are not 0, the end velocity and the angular velocity are also not 0. 3) The initial velocity and angular velocity are not 0, and the end velocity and angular velocity are 0.

Table 2 Pose of each point

| Point | $X$ | $Y$ | $Z$ | Roll | Pitch | Yaw |
|---|---|---|---|---|---|---|
| $A$ | 0.343 3 | −0.226 | 0.158 3 | −2.121 | 1.495 | 2.527 |
| $B$ | 0.343 3 | 0.171 8 | 0.158 3 | −0.061 | 1.112 | −1.712 |
| $C$ | 0.343 3 | 0.172 1 | 0.390 5 | −2.958 | 1.108 | 1.704 |
| $D$ | 0.343 3 | −0.123 9 | 0.390 5 | −2.121 | 1.492 | 2.527 |

Table 3 Velocity of each point

| Point | $V_x$ (m/s) | $V_y$ (m/s) | $V_z$ (m/s) | $\omega_x$ (rad/s) | $\omega_y$ (rad/s) | $\omega_z$ (rad/s) |
|---|---|---|---|---|---|---|
| $A$ | 0 | 0 | 0 | 0 | 0 | 0 |
| $B$ | 0 | 0.2 | 0 | −0.2 | 0 | 0 |
| $C$ | 0 | −0.2 | 0 | 0 | 0.2 | 0 |
| $D$ | 0 | 0 | 0 | 0 | 0 | 0 |

The trajectory of the manipulator is shown in Fig. 9. The small red circles represent points $A$, $B$, $C$ and $D$. Since the velocities of points $B$ and $C$ are not zero, the two points are connected by a Bezier curve to move with good smoothness. The velocity, rotation axis, and angular velocity in the Cartesian space are shown in Figs. 10–12, respectively. The trajectory first plans separately by the position and orientation planning method. The path with a shorter time is re-interpolated according to the time of the longer path. The $N$ values corresponding to points $B$ and $C$ are 185 and 511, respectively. At the two

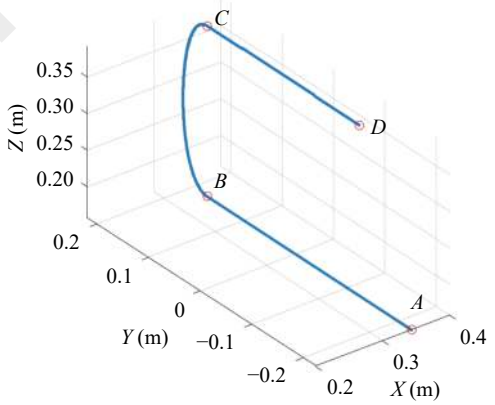



Fig. 9 Manipulator trajectory in the Cartesian space

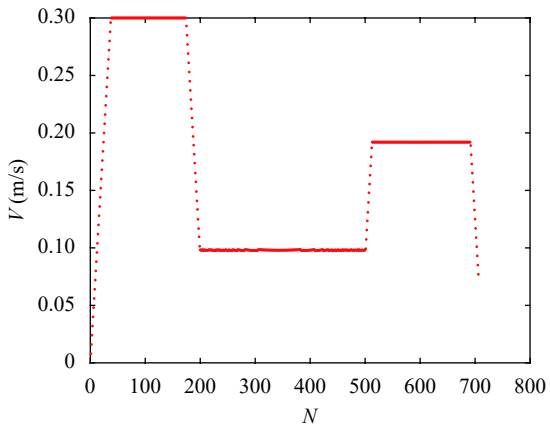



Fig. 10 Manipulator velocity in the Cartesian space

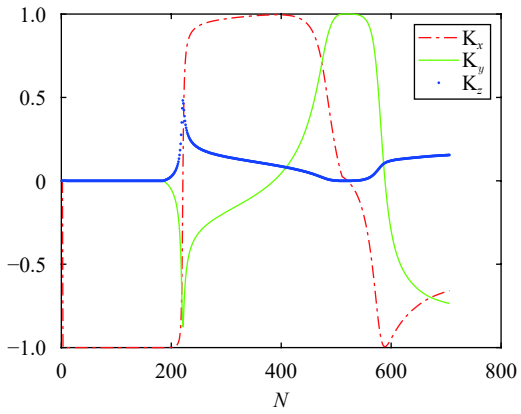



Fig. 11 Pose of the manipulator: rotation axes

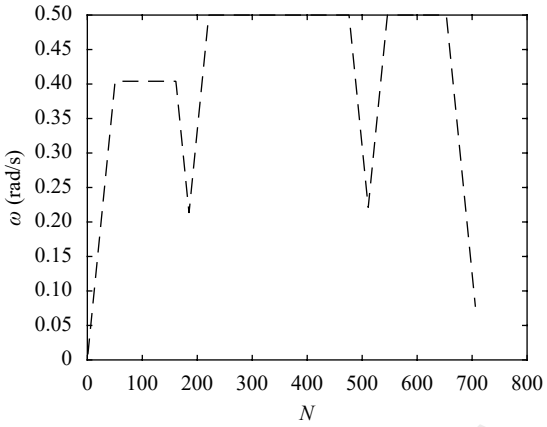



Fig. 12 Pose of the manipulator: angular velocity $\omega$

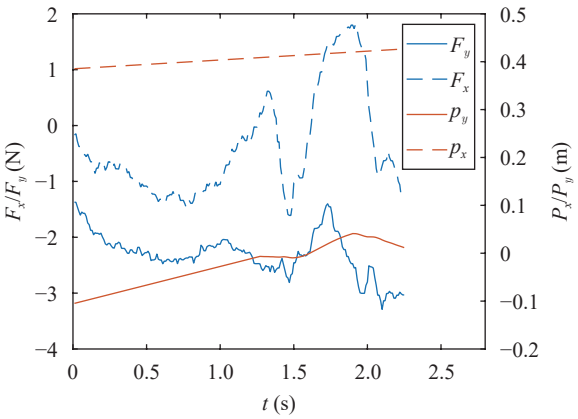



Fig. 13 Trajectory when the manipulator collides

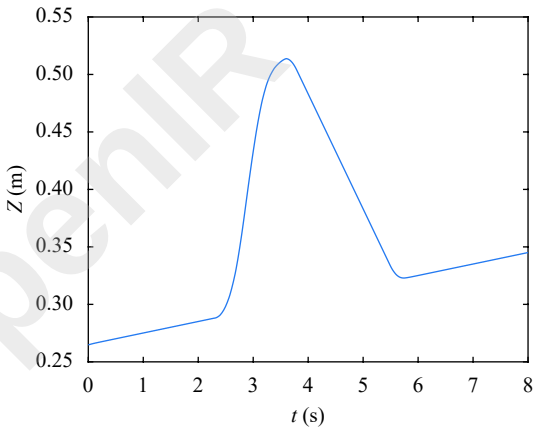



Fig. 14 Trajectory of manipulator under the influence of external force

points, the manipulator reaches the set velocities and angular velocities. The rotation axis also keeps changing continuously.

## 5.2 Compliance control

In order to verify the compliance control of the manipulator to environmental obstacles, the end-effector moves in a straight line (including displacement in both $X$ and $Y$ directions), and foam obstacles are added to the trajectory. The threshold of the external force $F_{\min,i}$ is set to 2.5 N. The trajectory of the manipulator is shown in Fig. 13. It can be seen that when $t = 1.28$ s, the external force in the $Y$-axis direction reaches the threshold, and the position $P_y$ no longer increases. When the external force is larger than the threshold, $P_y$ starts to decrease. In the $X$-axis direction, since the external force is smaller than the threshold, it continues to move along the planned trajectory.

To verify the trajectory following of the manipulator to the operator′s interference with external force, let the manipulator move along the $Z$-axis direction. The trajectory generated with the operator guiding is shown in Fig. 14. It can be seen that the trajectory is changed to move along the positive $Z$-axis at 2.4 s when the manipulator is pushed along the $Z$-axis. When the external force disappears at 3.6 s, the manipulator follows the planned trajectory, and catches up with the planned linear trajectory at 5.7 s.

These two experiments verify that the manipulator has the corresponding ability of external force and improves collision safety. At the same time, it also shows the ability of the manipulator to follow the planned trajectory in a dynamic environment.

# 6 Conclusions

This paper develops a lightweight manipulator and proposes a real-time smooth trajectory planning method based on Bezier curves, including real-time position planning and orientation planning. The trajectory planning from any initial state to the target state within the kinematic constraints is achieved, which improves the adaptability of the manipulator to the dynamic target. At the same time, a compliance control method based on trajectory following and impedance control is designed so that the manipulator can respond to changes in external force while following the trajectory, protect the manipulator from violent collisions, and improve the operation safety.

Experiments on the manipulator show that the proposed design, planning, and control methods are feasible and effective.

## Acknowledgements

This work was supported by National Key Research and Development Program of China (No. 2018AAA 0103003), National Natural Science Foundation of China (No. 61773378), the Basic Research Program (No. JCKY*******B029), and the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDB 32050100).

## References

[1] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, G. Hirzinger. The DLR lightweight robot: Design and control concepts for robots in human environments. *Industrial Robot*, vol. 34, no. 5, pp. 376–385, 2007. DOI: 10.1108/01439910710774386.

[2] Q. Liu, D. G. Yang, W. D. Hao, Y. Wei. Research on kinematic modeling and analysis methods of UR robot. In *Proceedings of the 4th IEEE Information Technology and Mechatronics Engineering Conference*, IEEE, Chongqing, China, pp. 159−164, 2018. DOI: 10.1109/ITOEC.2018.8740681.

[3] X. L. Zhu, X. M. Dun, L. Shan, X. Y. Dun. Study and design of a light-weight multi-joint discretely-actuated manipulator. *Machine Tool & Hydraulics*, vol. 42, no. 21, pp. 1–5, 2014. DOI: 10.3969/j.issn.1001-3881.2014.21.001. (in Chinese)

[4] H. Song, Y. S. Kim, J. Yoon, S. H. Yun, J. Seo, Y. J. Kim. Development of low-inertia high-stiffness manipulator LIMS2 for high-speed manipulation of foldable objects. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, Madrid, Spain, pp. 4145−4151, 2018. DOI: 10.1109/IROS.2018.8594005.

[5] N. J. Liu, T. Lu, Y. H. Cai, S. Wang. A review of robot manipulation skills learning methods. *Acta Automatica Sinica*, vol. 45, no. 3, pp. 458–470, 2019. DOI: 10.16383/j.aas.c180076. (in Chinese)

[6] S. L. Zhang, F. S. Jing, S. Wang. A transition method based on Bezier curve for trajectory planning in Cartesian space. *High Technology Letters*, vol. 23, no. 2, pp. 141–148, 2017. DOI: 10.3772/j.issn.1006-6748.2017.02.004.

[7] S. L. Zhang, F. S. Jing, S. Wang. Orientation transition and interpolation method based on spherical Bezier. *Journal of Huazhong University of Science and Technology* (*Nature Science Edition*), vol. 45, no. 10, pp. 75–79, 2017. DOI: 10.13245/j.hust.171014. (in Chinese)

[8] F. Miyazaki, M. Matsushima, M. Takeuchi. Learning to dynamically manipulate: A table tennis robot controls a ball and rallies with a human being. *Advances in Robot Control: From Everyday Physics to Human-Like Movements*, S. Kawamura, M. Svinin, Eds., Berlin, Germany: Springer, pp. 317−341, 2006. DOI: 10.1007/978-3-540-37347-6_15.

[9] K. Mülling, J. Peters. A computational model of human table tennis for robot application. *Autonome Mobile Systeme 2009*, R. Dillmann, J. Beyerer, C. Stiller, J. M. Zöllner, T. Gindele, Eds., Berlin, Germany: Springer, pp. 57-64, 2009. DOI: 10.1007/978-3-642-10284-4_8.

[10] A. Gasparetto, V. Zanotto. Optimal trajectory planning for industrial robots. *Advances in Engineering Software*, vol. 41, no. 4, pp. 548–556, 2010. DOI: 10.1016/j.advengsoft.2009.11.001.

[11] Y. Liu, L. Shi, X. C. Tian. Weld seam fitting and welding torch trajectory planning based on NURBS in intersecting curve welding. *The International Journal of Advanced Manufacturing Technology*, vol. 95, no. 5, pp. 2457–2471, 2018. DOI: 10.1007/s00170-017-1374-y.

[12] Z. J. Li, H. B. Wu, J. M. Yang, M. H. Wang, J. H. Ye. A position and torque switching control method for robot collision safety. *International Journal of Automation and Computing*, vol. 15, no. 2, pp. 156–168, 2018. DOI: 10.1007/S11633-017-1104-9.

[13] H. Matsumori, M. C. Deng, Y Noge. An operator-based nonlinear vibration control system using a flexible arm with shape memory alloy. *International Journal of Automation and Computing*, vol. 17, no. 1, pp. 139–150, 2020. DOI: 10.1007/s11633-018-1149-4.

[14] J. J. Duan, Y. H. Gan, M. Chen, X. Z. Dai. Adaptive variable impedance control for dynamic contact force tracking in uncertain environment. *Robotics and Autonomous Systems*, vol. 102, pp. 54–65, 2018. DOI: 10.1016/j.robot.2018.01.009.

[15] Z. J. Li, T. Zhao, F. Chen, Y. B. Hu, C. Y. Su, T. Fukuda. Reinforcement learning of manipulation and grasping using dynamical movement primitives for a Humanoidlike mobile manipulator. *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 121–131, 2017. DOI: 10.1109/TMECH.2017.2717461.

[16] B. Sangiovanni, A. Rendiniello, G. P. Incremona, A. Ferrara, M. Piastra. Deep reinforcement learning for collision avoidance of robotic manipulators. In *Proceedings of European Control Conference*, IEEE, Limassol, Cyprus, pp. 2063−2068, 2018. DOI: 10.23919/ECC.2018.8550363.

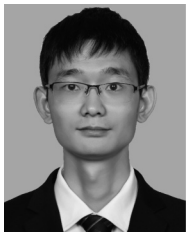
[17] A. H. Barr, B. Currin, S. Gabriel, J. F. Hughes. Smooth interpolation of orientations with angular velocity constraints using quaternions. *ACM SIGGRAPH Computer Graphics*, vol. 26, no. 2, pp. 313–320, 1992. DOI: 10.1145/142920.134086.

[18] M. Jafari, H. Molaei. Spherical linear interpolation and Bézier curves. *General Scientific Researches*, vol. 2, no. 1, pp. 13–17, 2014.

[19] G. M. Nielson. Smooth Interpolation of Orientations. *Models and Techniques in Computer Animation*, N. M. Thalmann, D. Thalmann, Eds., Tokyo, Japan: Springer, pp. 75−93, 1993. DOI: 10.1007/978-4-431-66911-1_8.

[20] S. L. Zhang, S. Wang, F. S. Jing, M. Tan. A Sensorless hand guiding scheme based on model identification and control for industrial robot. *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5204–5213, 2019. DOI: 10.1109/TII.2019.2900119.

[21] S. L. Zhang, S. Wang, F. S. Jing, M. Tan. Parameter estimation survey for multi-joint robot dynamic calibration case study. *Science China Information Sciences*, vol. 62, no. 10, Article number 202203, 2019. DOI: 10.1007/s11432-018-9726-3.



**Shao-Lin Zhang** received the B. Eng. and M. Eng. degrees in mechanical science and engineering from Huazhong University of Science and Technology, China in 2010 and 2013, and the Ph. D. degree in control theory and control engineering from Institute of Automation, Chinese Academy of Sciences, China in 2019. He is currently an assistant professor with State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China.

His research interests include intelligent robot system and automation systems.

E-mail: zhangshaolin2015@ia.ac.cn
ORCID iD: 0000-0002-1544-2552

**Yue-Guang Ge** received the B. Eng. degree in computer science and technology from Northeastern University, China in 2006, the M. Eng. degree in computer software and theory from North China Electric Power University, China in 2014. He is a Ph. D. degree candidate in control theory and control engineering at University of Chinese Academy of Sciences, China.

His research interests include intelligent robot system, knowledge representation and reasoning.

E-mail: yueguang.ge@ia.ac.cn
ORCID iD: 0000-0001-6850-2179

**Hai-Tao Wang** received the B. Eng. degree in communication engineering from University of Electronic Science and technology of China, China in 2014. He is a Ph. D. degree candidate in control theory and control engineering at University of Chinese Academy of Sciences, Institute of Automation, Chinese Academy of Sciences, China.

His research interests include robot task planning and knowledge reasoning.

E-mail: wanghaitao2019@ia.ac.cn
ORCID iD: 0000-0002-5297-9543

**Shuo Wang** received the B. Eng. degree in electrical engineering from the Shenyang Architecture and Civil Engineering Institute, China in 1995, the M. Eng. degree in industrial automation from Northeastern University, China in 1998, and the Ph. D. degree in control theory and control engineering from Institute of Automation, Chinese Academy of Sciences, China in 2001, respectively. He is currently a professor with State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, China.

His research interests include biomimetic robot, underwater robot, and multirobot systems.

E-mail: shuo.wang@ia.ac.cn (Corresponding author)
ORCID iD: 0000-0002-1390-9219