PAPER • OPEN ACCESS

The Combination Trajectory Planning of Serial Robot in Cartesian Space and Joint Space

To cite this article: Jun Ren and Qianghao Zhang 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **563** 052039

The Combination Trajectory Planning of Serial Robot in Cartesian Space and Joint Space

Jun Ren*, **Qianghao Zhang**

(School of Mechanical Engineering, Hubei University of Technology, Wuhan, 430068)

*Corresponding author's e-mail: renjun@mail.hbut.edu.cn

Abstract. In order to improve the working efficiency of serial robot, a new trajectory planning method is proposed. Taking 6-DOF robot as an example, the 6-DOF open-chain robot is transformed into a 12-DOF closed-chain robot by creating a virtual robot at the end-effecter. Then, the virtual joint variable of the virtual robot is used to represent the end position of the robot, and the direct relationship between the joint variable and the position attitude is obtained. The B-spline Curve is used to plan the trajectory in the Cartesian space, and the joint motion trajectories of the robot are controlled indirectly to meet the requirements of the joint space speed and acceleration. Finally, the time-optimal solution of the trajectory programming is solved by the Genetic Algorithm under the condition of satisfying the joint space and the Cartesian space.

1. Introduction

Since the 1960s, serial robots have been used in more and more fields, especially in the field of automobile manufacturing, and instead of manual machining, welding, heat treatment, surface coating, polishing, materials, assembly, detection and warehouse operations, which greatly improve production efficiency and ensure the consistency of manufactured products[1].When the serial robot performs the work, reasonable space trajectory planning can improve the working efficiency of the robot and reduce mechanical vibration and joint wear. There are two kinds of trajectory planning. One is to plan the trajectory in the joint space, which requires the smoothing of the planned trajectory function to make the mechanical arm move smoothly. The other is to plan the trajectory in the Cartesian space, which requires the pose of the hand, speed meets constraints[2]. According to these two trajectory planning, domestic and foreign scholars have done a lot of researches: Alessandro Gasparetto et al.[3] developed a method by taking the sum of the running time and the squared integral of the joint acceleration derivative (jerk) as the objective function, in case the maximum joint velocity is not exceeded, the minimum value of the objective function is solved by trajectory planning in the joint space; Xu et al. [4] developed a method to perform trajectory planning in joint space, and solved the optimal solution of the sum of time and energy by using gene double cloning deductive method under the condition of meeting joint speed, acceleration and quadratic acceleration constraints; Zhu et al. [5] developed a method by using 7-time B-spline curve as the interpolation curve of joint space trajectory planning, and solved the time-optimal solution meeting kinematic constraints by sequential quadratic programming method; Ren et al. [6] developed a method to use the virtual joint to plan the trajectory of the robot's last pose in Cartesian space, which ensures the speed and acceleration of the last pose is continuous and controllable. Li et al. [7] developed a method by using multiple quaternion convert the pose of Cartesian space robot to a four-dimensional space, and the posture of the robot is linearly interpolated by the rotation of the hypersphere, the computational complexity is small, and the real-time is good.

IOP Conf. Series: Materials Science and Engineering **563** (2019) 052039 doi:10.1088/1757-899X/563/5/052039

Although domestic and foreign scholars have done a lot of research on trajectory planning in these two coordinate spaces, they are limited to only one kind of coordinate space. The trajectory planning in joint space can continuously smooth the joint motion, velocity and acceleration trajectory curve of robots, and reduce mechanical vibration and joint wear; The trajectory planning of Cartesian coordinate space is more visual and convenient for the robot's pose control. Therefore, the trajectory planning of the two coordinate spaces is combined, so that the advantages of the two kinds of trajectory planning will greatly improve the production efficiency of the serial robot.

In this paper, the virtual robot is firstly established to obtain the mapping relationship between the joints and the robot poses. Then, the trajectory planning is performed in Cartesian space, and the B-spline curve is used as the interpolation curve. Using the mapping relationship between each joint and the robot pose to control the pose of robot and satisfy the constraints. Finally, the genetic algorithm is used to solve the time optimal solution under the condition of joint space and Cartesian space constraints.

2. Establishment of Virtual Robot

To obtain the mapping relationship between the joint variables of the robot and the spatial pose of the robot, take the typical UR5 arm in a 6-DOF serial robot as an example. A virtual robot is constructed at the end of the robot arm, and the virtual joint variable of the virtual robot is used to represent the spatial pose changes of the robot. The structural model of the virtual robot is shown in figure 1.

As shown in the figure 1, the base coordinate of the virtual robot coincides with the base coordinate of the original robot, the virtual joints 1, 2, 3 are moving joints, the virtual joints 4, 5, 6 are rotating joints, the joint axis directions of the virtual joints 1, 2, and 3 are respectively along the x, y, and z directions of the base coordinate, and the activity space determined by the length of the connecting rod includes the working space of the original robot; The virtual joints 4, 5, and 6 have their axial directions along the x, y and z direction of the base coordinate, the link length is set to 0.

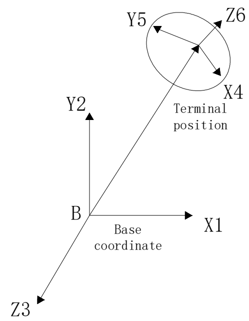

Figure 1. Virtual joint coordinate system.

It can be seen that according to control the joint variables of the virtual joints 1, 2, 3, the terminal position of the connecting rod can be controlled. Since the length of the connecting rod is 0, the terminal position of the connecting rod is consistent with the terminal position of the welding torch, and similarly the virtual joint 4, 5, 6 can control the terminal pose of the robot. Therefore, the terminal pose of the robot can be represented by the joint variable of the virtual joint.

3. Combination trajectory planning in Cartesian space and joint space

3.1. Robotic kinematic solution

The robot of joint kinematic solutions 12-joint robot, because the joints are more and the length of some links is 0, the traditional D-H parameter method is more complicated. Therefore, the spiral theory and POE formula are used to carry out the inverse kinematics of the robot.

According to the screw theory, the motion of each joint of the robot is generated by the motion spiral $\hat{\xi}$ located at the joint axis, and obtain the geometric description of kinematics. If $g(0)$ is used to indicate the initial configuration of the rigid body and $g(a)$ is the final configuration, the rigid body motion described by the motion spiral can be expressed as [8]:

$$g(a) = e^{\hat{\xi}_1\theta_1}e^{\hat{\xi}_2\theta_2}e^{\hat{\xi}_3\theta_3}\cdots e^{\hat{\xi}_n\theta_n}g(0) \tag{1}$$

For the 12-joint robot established above, kinematics can be expressed as:

$$g(a) = e^{\hat{\xi}_1\theta_1}e^{\hat{\xi}_2\theta_2}e^{\hat{\xi}_3\theta_3}\cdots e^{\hat{\xi}_{12}\theta_{12}}g(0) \tag{2}$$

Since the base coordinate of the virtual robot coincides with the base coordinate of the physical robot,$g(a) = g(0)$, the formula (2) can be turned into:

$$E = e^{\hat{\xi}_1\theta_1}e^{\hat{\xi}_2\theta_2}e^{\hat{\xi}_3\theta_3}\cdots e^{\hat{\xi}_{12}\theta_{12}} \tag{3}$$

Knowing the spatial pose of the robot, the joint variables θ_1, θ_2, θ_3, θ_4, θ_5, θ_6 of each virtual joint can be obtained, and the motion expression similar to the 6-joint robot can be obtained, according to use the kinematic solution method of 6-joint robot, the corresponding joint sequence can be obtained.

3.2. Combination trajectory planning

3.2.1. Robot Cartesian space trajectory planning. Let B and A be two sets: $B = (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \in R^6$ is a combination of virtual joint variables, $A = (\theta_7, \theta_8, \theta_9, \theta_{10}, \theta_{11}, \theta_{12}) \in R^6$is a combination of real joint variables, by the formula (3), get the mapping relationship f from B to A as follow:

$$f: B \to A, f(b) = A$$

In the virtual joint, 3 non-rational B-spline curves are used to interpolate the pose trajectory between the key points, which can realize the velocity and the acceleration of the robot Cartesian space is continuous. The definition of the three non-rational B-spline curves is:

$$C(u) = \sum_{i=0}^{q} N_{i,p}(u)P_i, \qquad 0 \le u \le 1 \tag{4}$$

Where $\{P_i\}$ is the control point and $\{N_{i,p}(u)\}$ is the 3 times B-spline basis function defined on the non-periodic node vector $U = \{0,0,0,0, u_4, u_5, u_6 \cdots u_{q+2}, 1,1,1,1\}$. Normalize t_i by the method of cumulative duration, then find the node vector of the corresponding B-spline curve

$$u_0 = u_1 = \ldots u_3 = 0, \quad u_{q+3} = u_{q+4} = \ldots u_{q+6} = 1, \quad u_i = u_{i-1} + \frac{|\Delta t_{i-4}|}{\sum_{j=0}^{n-1}|\Delta t_i|} \tag{5}$$

It shows that the known control point and the node vector can be used to obtain the corresponding B-spline curve, and the control point can be obtained from the key point and the node vector which the curve passes, since the key point is known. B-spline curve can be controlled by the node vector. As u grows from 0 to 1, the joints move from the starting position to the terminal position. The 6-dimensional space B-spline curve composed of the motion curves of the 6 joints is recorded as $G(A,u)$, and the 6-dimensional pose curve is recorded as$H(B,u)$, then:

$$H(B,u) \xrightarrow{f} G(A,u)$$

Set *di* as the control vertex of H, $di^1$and di^2 are the control vertices of the first-order second-order curve of curve H. By using the convex hull of B-spline curve, the trajectory constraints of Cartesian space can be satisfied by controlling *di*, di^1, and di^2.

$$|di| \le l_e, \ |di^1| \le v_e, \ |di^2| \le a_e \tag{6}$$

Where l_e, v_e, a_e is the path, velocity and acceleration constraints of Cartesian space, di^1, di^2can be obtained by Debord recursion formula:

$$p^{(c)}(u) = \sum_{j=i-3+c}^{i} d_j^{(c)} N_{j,k-c}(u), \qquad c = 1,2,3, \quad j = i-3+l, \ldots i, \quad l = 1,2, \ldots c \tag{7}$$

$$d_j^{(c)} = (4-c)\frac{d_j^{l-1} - d_{j-1}^{l-1}}{u_{j+4-l} - u_j}, \quad j = i-3+l, \ldots i, l = 1,2, \ldots c \tag{8}$$

3.2.2. Robot joint space trajectory planning. Let the mapping relationship between Cartesian space velocity and joint space velocity beg, then:

$$g: \dot{B} \to \dot{A}, \quad g(\dot{B}) = \dot{A}$$

It is known that the relationship between Cartesian space velocity and joint space velocity can be represented by a Jacobian matrix:

$$\dot{B} = J(A)\dot{A}, \quad \dot{A} = J(A)^{-1}\dot{B}$$

The velocity of the Cartesian space is linear with the velocity of the joint space, so g is a linear mapping; Since each set of joint variables of the solid robot always corresponds to a corresponding set of poses, f is the mapping to the top, that is surjection, and f is a continuous mapping, using affine invariance and strong convex husking of B-spline combined with mapping relationship f and g in the joint space trajectory planning, can indirectly control the robot's motion trajectory in the joint space.

$$\dot{H}(B,u) \xrightarrow{g} \dot{G}(A,u)$$

It can be known from the affine invariance of the B-spline curve that when $\dot{H}$ changes $J(B)^{-1}$ to $\dot{G}$, performing the affine transformation on the control point di of the original curve, and the transformed control point di' is obtained as follow:

$$di \xrightarrow{J^{-1} \circ B(u)} di', \quad i = 1,2 \cdots n; \quad 0 \leq u \leq 1 \tag{9}$$

Where i is the number of control points. When u increases from 0 to 1, the curve formed by A(u) is a spatial 6-dimensional B-spline curve, since the original control point is unchanged, the Jacobian inverse transform is a linear transformation, the curve formed by the transformed control point di' in the process of u from 0 to 1 is a space 6-dimensional B-spline curve $\dot{H}$. The new B-spline curve formed by affine transformation is recorded as $di(u)'$.

Although curve $\dot{G}$ is a curve composed of different affine transformations at each point on $\dot{H}$, the strong convexity of B-spline curve shows that the maximum value of curve $\dot{G}$ is always within the control polygon formed by each maximum new control point, then:

$$\dot{G}(B,u) \leq Max(di(u)') \tag{10}$$

For the B-spline curve $di(u)'$, the strong convexity shows that the maximum value of $di(u)'$ is within the control polygon contained in each control vertex. Let $di(u)_l'$ be control vertex of $di(u)'$:

$$Pi(u)^1 \leq Max(di(u)_l') \tag{11}$$

Similarly, the minimum value for curve $\dot{G}$:

$$Pi(u)^1 \geq Min(di(u)_l') \tag{12}$$

Arranging formula(10) and formula (11)results in:

$$\dot{G}(B,u) \leq Max(di(u)_l') \tag{13}$$

Similarly, the minimum value for curve $\dot{G}$ is:

$$\dot{G}(B,u) \geq Min(di(u)_l') \qquad i = 1,2 \cdots n, \quad l = 1,2 \cdots n; \quad 0 \leq u \leq 1 \quad (14)$$

Let the acceleration curves of the Cartesian space and the acceleration curves of the joint space are $\ddot{H}$ and $\ddot{G}$, and the two sides of the equation are respectively derived from time:

$$\ddot{A} = J^{-1} \circ B(u) \bullet \ddot{B} + J^{-1} \circ \dot{B}(u) \bullet \dot{B}(u) \qquad 0 \leq u \leq 1 \tag{15}$$

Let:

$$Q_1(u) = J^{-1} \circ B(u) \bullet \ddot{B}(u); \quad 0 \leq u \leq 1 \tag{16}$$

$$Q_2(u) = J^{-1} \circ \dot{B}(u) \bullet \dot{B}(u); \quad 0 \leq u \leq 1 \tag{17}$$

The acceleration curve $\ddot{G}(u)$ of the joint space can be synthesized from curves $Q_1(u)$ and $Q_2(u)$:

$$\ddot{G}(u) = Q_1(u) + Q_2(u); \quad 0 \leq u \leq 1 \tag{18}$$

It can be seen from the continuous differentiability of the B-spline curve that $\dot{B}(u)$ and $\ddot{B}(u)$ are both B-spline curves, and similar results can be obtained from Eq.(10)~(12):

$$Q_1(B,u) \leq Max(Q_1 di(u)_l') \tag{19}$$

$$Q_2(B,u) \leq Max(Q_2 di(u)_l') \tag{20}$$

$$Q_1(B,u) \geq Min(Q_1 di(u)_l') \tag{21}$$

$$Q_1(B,u) \geq Min(Q_1 di(u)_l') \tag{22}$$

Combined equations:

$$\ddot{G}(B,u) \leq Max(Q_1 di(u)_l') + Max(Q_2 di(u)_l') \tag{23}$$

$$\ddot{G}(B,u) \geq Min(Q_1 di(u)_l') + Min(Q_2 di(u)_l') \tag{24}$$

Let the speed constraint and the acceleration constraint of the joint space are v_e^l anda_e^l, respectively. The simultaneous connection can indirectly control the speed and acceleration of the joint space:

$$\begin{cases} Max(di(u)_l') \leq v^1{}_e \\ Max(Q_1 di(u)_l') + Max(Q_2 di(u)_l') \leq a^1{}_e \end{cases} \quad i = 1,2 \cdots n, l = 1,2 \cdots n; \quad 0 \leq u \leq 1 \tag{25}$$

IOP Conf. Series: Materials Science and Engineering **563** (2019) 052039 doi:10.1088/1757-899X/563/5/052039

3.2.3. Combination trajectory planning time optimal solution. The solution to the time-optimal problem of combination trajectory planning is to solve the smallest time series of total time corresponding to the normalized time node u while satisfying the joint space and Cartesian space constraints. The genetic algorithm is used to solve the problem. Since the problem is an optimization problem with a constraint problem, it is generally processed by a penalty function. The cost function is:

$$f(x) = min \sum_0^{n-1} x_i + r_g \phi(x_i) \tag{26}$$

The combination trajectory planning needs to consider the obstacle avoidance problem of the robot. At the same time, the speed and acceleration of the actuator at the end of the operating space and the joint speed and acceleration of the joint space meet the requirements. Therefore, the constraint of the problem can be simplified as:

$$s.t. \begin{cases} Max(di(u)_l') \le v_e^1 \\ Min(di(u)_l') \ge -v_e^1 \\ Max(Q_1 di(u)_l') + Max(Q_2 di(u)_l') \le a_e^1 \\ Min(Q_1 di(u)_l') + Min(Q_2 di(u)_l') \ge -a_e^1 \end{cases} \quad |di| \le l_e,\ |di^1| \le v_e,\ |di^2| \le a_e \tag{27}$$

Where$x_i = \Delta t_i = t_{i+1} - t_i$, $i = 1,2 \cdots n, l = 1,2 \cdots n$, r_g is the penalty coefficients, $\phi(x_i)$ is the penalty function, and the size is:

$$\phi(x_i) = \begin{cases} 1, & infeasible\ solution \\ 0, & feasble\ solution \end{cases}$$

The genetic algorithm can be used to solve the minimum value of *f(x)* and obtain the corresponding time series. The corresponding time node vector can be obtained by the formula, according to the simultaneous equation, the motion curve of the end effector of the operation space can be solved. From the mapping relationship, the corresponding joint motion curve can be obtained, and complete the combination planning of the trajectory.

4. Experiment and Results

The development of the trajectory servo controller for this experiment is based on the development software UR+ of Shanghai Youao (UR) Co., Ltd., which is carried out on the virtual machine UR Sim 3.3.0.145 Starter, the experimental development was carried out by writing a running program, and the experiment was carried out using the UR5 robot as shown in figure 2.

Figure 2. UR5 manipulator.

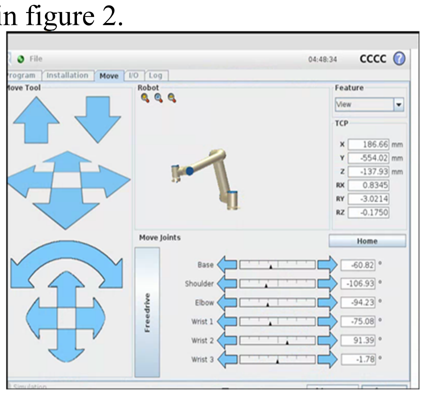

Figure 3. Robot joint variables and position and attitude data on virtual machines.

The motion curve of the position and pose of UR5 is obtained by theoretical derivation and optimization algorithm according to the key points given, which belongs to the reservation, and its image is generated directly by matlab. The indirect measurement method is used in this paper. Specifically, by detecting the posture data of the end effector of the robot on the virtual machine URSim 3.3.0.145 Starter (as shown in figure 3), it is verified by comparing with the original set trajectory.

Select a series of key points of the spatial trajectory, the corresponding pose sequence is as follows:

Table 1. Manipulator positions.

	X	Y	Z	Rx	Ry	Rz
P1	-5.92	-288.21	599.73	0.9249	1.6025	-1.6289
P2	276.82	-344.39	498.68	0.4188	2.2235	-1.8402
P3	372.02	-405.50	359.49	0.2190	2.7168	-1.6888
P4	400.35	-450.51	197.90	0.1190	3.2508	-1.3855
P5	393.68	-447.20	52.95	0.2588	3.6985	-0.7094
P6	424.11	-408.11	34.33	0.7590	3.4899	-0.2978

Let the time series corresponding to the 6 key points in Table 1 be $T=(t_1, t_2, t_3, t_4, t_5, t_6)$, and use the formula to find the corresponding time node sequence to solve each control point of the motion curve. The joint space kinematic constraints of the robot are:

Table 2. Joint space constraints.

	P1	P2	P3	P4	P5	P6
$v_e^1/(°/s)$	100	100	100	100	100	100
$a_e^1/(°/s^2)$	40	50	75	75	90	80

Cartesian space kinematic constraints are:

Table 3. Cartesian space constraints.

	X	Y	Z	Rx	Ry	Rz
l_e/mm	1000	1000	750	--	--	--
$v_e/mm \cdot s^{-1}$	500	500	300	0.5	0.5	0.5
$a_e/mm \cdot s^{-2}$	800	800	500	0.8	0.8	0.8

According to the genetic algorithm, set group M=40, terminate algebra T=300, select tournament method as select operator, cyclic crossover as loop operator, PC=0.8, mutation operator takes value 0.02, penalty coefficient take value 50. The program is written in MATLAB software, and the optimal solution is [2.2541, 1.6614, 1.9075, 1.6605, 1.6165]. The total time is T=9.1s.

Knowing the time series, the time node vector of the B-spline curve can be obtained from Eq.(5). Combined with the position sequence pi, the running trajectory curve of the terminal of the serial robot in Cartesian space can be drawn as shown in figure 4~5. Velocity curve is shown in figure 6~7, the acceleration curve is shown in figure 8~9.

Meanwhile, the mapping relationship *f* from the Cartesian space to the joint space can be used to find the joint position, joint velocity curve and joint acceleration curve of the serial robot in the joint space, as shown in figures 10~12.

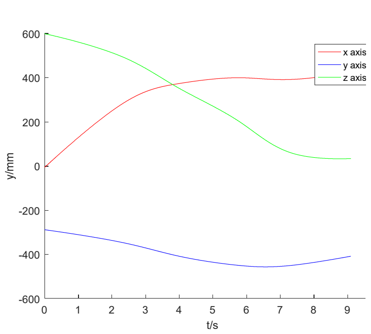

Figure 4. Robot end-effecter position.

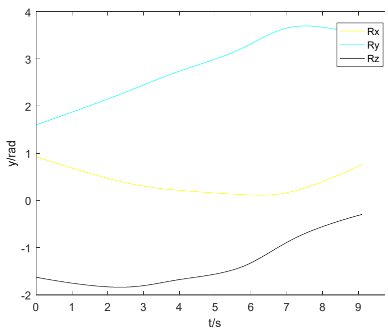

Figure 5. Robot end-effecter Euler angle.

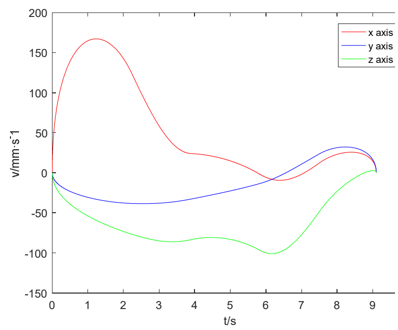

Figure 6. Robot end-effecter speed.

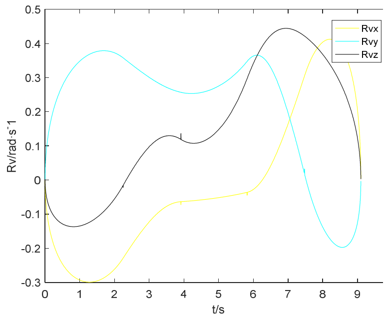

Figure 7. Robot end-effecter angular velocity.

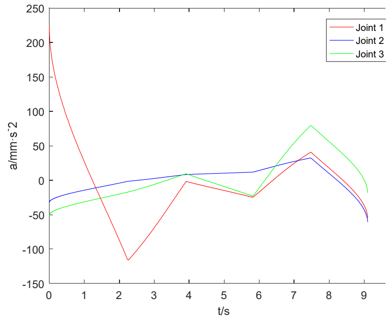

Figure 8. Robot end-effecter acceleration.

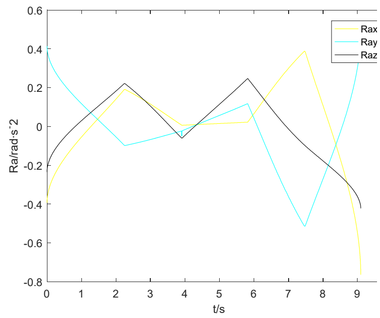

Figure 9. Robot end-effecter angular acceleration.

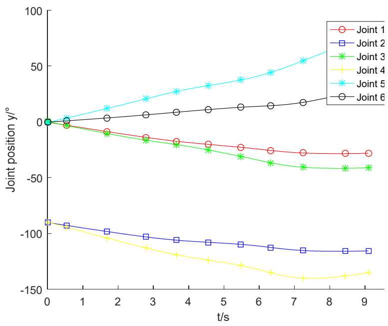

Figure 10. Joint position.

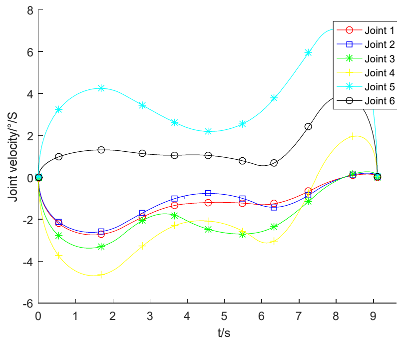

Figure 11. Joint velocity curve.

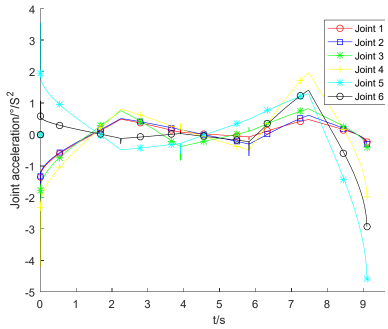

Figure 12. Joint acceleration curve.

In figures 10~11, the joint motion curve is smooth and continuous, and both within the range of the robot's joint motion, meeting the set conditions; In figure 12, the joint acceleration curve is continuous but limited with the order of the B-spline curve, in the form of a broken line, but the curves are within the constraints of the robot's joint motion, and the set conditions are also met.

5. Conclusion

(1) According to the kinematics inverse, obtain robot joint variables, adds a virtual joint at the end of the open-chain serial robot actuator, through the mapping relationship between joint variables and pose variables, the terminal posture of robot can be expressed by the virtual joint variables.

(2) The trajectory planning is carried out in the Cartesian space of the robot. The affine invariance of the B-spline curve and the strong convex hull are used to indirectly control the joint motion trajectory of the robot and realize the combination planning of the robot joint space and Cartesian space. The robot motion trajectory meets the constraints of Cartesian space and joint space at the same time, so that the robot joint wear and mechanical vibration can be reduced under the condition of stable robot motion. Experiments on the UR5 robot show that the method is effective and feasible.

Acknowledgments

This work was supported by Hubei Provincial Natural Science Foundation of China (2017CFB128).

References

[1] Craig, J.J. (2004) Introduction to Robotics Mechanics and Control 3rd edition. Prentice Hall, Englewood.

[2] FU, X.L., HE, J.P., WANG, F.X. (2015) Research Status on Trajectory Planning of Welding Robot. Light Industry Machinery, 33(02):110-113.

[3] Gasparetto, A., Zanotto, V.A.(2008) Technique For Time-jerk Optimal Planning of Robot Trajectories. Robotics & Computer Integrated Manufacturing, 24(3):415-426.

[4] XU, H.L., XIE X.R., ZHUANG J., WANG S.A. (2010) Global Time-energy Optimal Planning of Industrial Robot Trajectories. Chinese Journal of Mechanical Engineering, 46(09):19-25.
[5] ZHU, S.Q., LIU, S.G., WANG, X.Y., WANG H.F. (2010) Time-optimal and Jerk-continuous Trajectory Planning Algorithm for Manipulators. Chinese Journal of Mechanical Engineering,46(03):47-52.
[6] REN, J.Y., SUN, H.X. (2002) A Novel Method of Trajectory Planning in Cartesian Space. ROBOT, (03):217-221.
[7] LI, H.S., WANG, Y.H., HUANG J.C., SHI, X.X. (2016) Research on Trajectory Planning of Industrial Robots with Double Quaternion. China Mechanical Engineering, 27(20): 2711-2716.
[8] Suleiman W., Yoshida E., Laumond J.P, et al. (2008) Optimizing Humanoid Motions Using Recursive Dynamics and Lie Groups. In: International Conference on Information and Communication Technologies: From Theory to Applications. Damascus. IEEE, 1-6.