An Approach for Smooth Trajectory Planning of High-Speed Pick-and-Place Parallel Robots using Quintic B-Splines

Yuhang Li [a] , Tian Huang [a,b,*] and Derek G. Chetwynd[b]

[a] *Key Laboratory of Mechanism Theory and Equipment Design of State Ministry of Education, Tianjin University, Tianjin 300072, China*

[b] *School of Engineering, The University of Warwick, Coventry CV4 7AL, UK*

Abstract: This paper presents a new, highly effective approach for optimal smooth trajectory planning of high-speed pick-and-place parallel robots. The pick-and-place path is decomposed into two orthogonal coordinate axes in the Cartesian space and quintic B-spline curves are used to generate the motion profile along each axis for achieving C^4-continuity. By using symmetrical properties of the geometric path defined, the proposed motion profile becomes essentially dominated by two key factors, representing the ratios of the time intervals for the end-effector to move from the initial point to the adjacent virtual and/or the via-points on the path. These two factors can then be determined by maximizing a weighted sum of two normalized single-objective functions and expressed by curve fitting as functions of the width/height ratio of the pick-and-place path, so allowing them to be stored in a look-up table to enable real-time implementation. Experimental results on a 4-DOF SCARA type parallel robot show that the residual vibration of the end-effector can be substantially reduced thanks to the very continuous and smooth joint torques obtained.

Keywords: Pick-and-place parallel robot; Trajectory planning; B-splines

1 Introduction

Recent years have seen ever increasing demands from food, pharmaceutical, packaging and many other light industries for high-speed pick-and-place parallel robots using limbs containing proximal revolute actuated joints and parallelograms. This is exemplified by the many very successful applications of Delta robots and similar types [1-5]. From a system viewpoint, the capability and efficiency of high-speed pick-and-place parallel robots clearly depends on desirable dynamic characteristics and good quality computer control, but also requires sound trajectory planning for achieving superior performance in terms of smoother joint torques, lower residual vibrations and shorter cycle times [6-9]. Large amounts of effort have been devoted to trajectory planning of pick-and-place operations over the last few decades. The currently available approaches can be broadly classified into two categories: path-based trajectory planning and coordinate-based trajectory planning.

Path-based trajectory planning is concerned with first generating an appropriate geometric path parameterized with respect to the arc length and then designing a proper motion profile along that path. For example, Gauthier *et al*. [10] took Lamé curves with G^2-continuity at square corners linking the vertical and horizontal segments, and employed a '4-5-6-7 polynomial' with C^3-continuity as the motion profile. The trajectory was then generated by minimizing the root-mean-square value of the time-derivative of the kinetic energy per unit mass of the payload. However, although the Lamé curve parameters can be optimized off-line, the point coordinates of the curve cannot be expressed explicitly in terms of the arc length. Therefore, a nonlinear algebraic equation has to be solved in the on-line coarse interpolation to determine the point coordinates associated with an interpolated arc length. In order to reduce the computational burdens in real time implementation, Masey *et al*. [11] suggested taking a bisected ellipse as the geometric path such that the arc length along the path could be approximated as a linear function of two normalized parameters so as to include any specified durations of constant velocity or degree of asymmetry. They concluded that the use of an asymmetry factor would be helpful to reduce peak joint torques. The transition linking the vertical and horizontal segments could also be shaped as a Bézier curve, clothoid spline or quintic polynomial for achieving at least G^2-continuity [12-16].

* Corresponding author
E-mail address: tianhuang@tju.edu.cn

Compared to path-based trajectory planning, it can be easier to use coordinate-based trajectory planning in either the Cartesian or the joint space. Planning in the joint space involves first transforming the coordinates of a set of via-points on a geometric path into a sequence of joint displacements *via* inverse kinematics, and then generating the motion profiles for each joint by interpolation subject to a set of specific constraints provided by the design requirements [17-21]. For example, Gosselin *et al*. [18] employed a ninth-order polynomial for a single motion profile to guarantee C^3-continuity by inserting a lift-off and a set-down point. They concluded that undesirable joint torque fluctuations arising from the use of higher order polynomials could be improved by adjusting the time intervals when the lift-off and set-down points are reached or by using a combination of a number of piecewise lower order polynomials. There are many other well-developed piecewise motion profiles available for this purpose, achieving at most C^3-continuity throughout the entire trajectory, for instance 3-4-5 polynomial, 4-5-6-7 polynomial, modified sine, modified trapezoid, and many others [6].

In order to achieve C^4-continuity for a smooth motion profile throughout the entire trajectory, there can be potential advantages in using a fifth-order B-spline as the interpolation function. The trajectory planning problem can then be stated as the determination of the sequence of time intervals necessary for a spline to connect two adjacent knots by minimizing a weighted performance index subject to a set of specific constraints. In this context, Constantinescu *et al*. [22] proposed a method for minimum time trajectory planning subject to the limits imposed upon the joint torques and their first derivatives. Gasparetto *et al*. [23-25] presented an algorithm for optimal smooth trajectory planning by minimizing, subject to limits on joint velocity, acceleration and jerk, a weighted sum of the integral of joint jerk squared and the total cycle time. Similarly, trajectory planning in the Cartesian space involves first generating motion profiles along two orthogonal axes of a path lying in a plane and passing through a set of via-points, and then transforming the interpolated point coordinates into the corresponding joint variables *via* inverse kinematics [26-29]. Optimization problems can be formulated in the Cartesian space similar to those in the joint space.

With the B-spline interpolation ensuring C^4-continuity, an obvious advantage of optimal trajectory planning in the joint space is that it should be easier to impose a set of specified limits upon the actuated joints. However, a problem encountered in practice is that the optimized motion profiles are configuration dependent; it is difficult, if not impossible, to solve the resulting complicated nonlinear programming problems online. Note that a smooth motion profile planned in the Cartesian space ensures that in the joint space at nonsingular configurations, and the joint torques are closely related to the joint accelerations. Therefore, the development of a widely-effective, architecture and configuration free approach to generate the smooth motion profiles in the Cartesian space would be highly beneficial for real-time implementation.

Responding to the many practical needs and inspired by the method proposed in [23], this paper presents a new approach for smooth trajectory planning of high-speed pick-and-place parallel robots using quintic B-splines. The method features an initial offline determination of two key factors dominating the normalized motion profiles along a path defined in a local frame of the Cartesian space, followed by the online generation of the smooth joint trajectories using a look-up table. The remainder of this paper is organized as follows. Section 2 defines a pick-and-place path in a local frame, employs fifth-order B-splines to generate the motion profile along each axis of the path with C^4-continuity, and so identifies two non-dimensional control factors. In Section 3, the influence of these two factors on the maximum acceleration and jerk over the path is investigated, leading to an optimization problem by maximizing a weighted performance index. This allows the relationship between the two factors and the width/height ratio of the path to be established offline by curve fitting, which, in turn, enables ready online generation of the smooth joint trajectory for real-time implementation. Section 4 presents results from both simulations and experiments using a 4-DOF SCARA type parallel robot [5] to demonstrate the good performance of the proposed approach for achieving smooth joint torques and thereby reducing residual vibrations of the end-effector. Conclusions are drawn in Section 5.

2 Generation of Geometric Path and Motion Profile in the Cartesian Space

2.1 Description of the geometric path

As shown in Fig. 1, let $P_1(x_1, y_1, z_1)$ and $P_f(x_f, y_f, z_f)$ be the initial and final points of the end effector in a pick-and-place operation, where x_1, y_1, z_1 and x_f, y_f, z_f are the coordinates of P_1 and P_f with respect to the reference frame $O-xyz$. When the pick-and-place operation is planned in an environment free of obstacles, it is not in principle

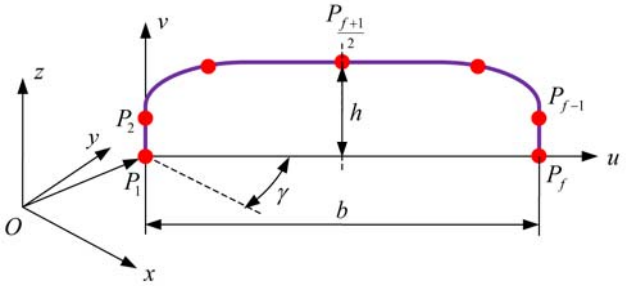


Fig. 1. A pick-and-place path defined in a local frame

necessary to specify the Cartesian trajectory that will be followed by the end-effector because only the initial and final points are relevant to the task to be performed. However, it is preferable to insert a lift-off point P_2 and a set-down point P_{f-1} on the path so that acceptable departure and arrival motions can be provided. In addition, a middle point $P_{\frac{f+1}{2}}$ is also considered. Thus, at least $f = 5$ via-points in total need to be set along the path to fulfill these requirements.

Without loss of generality, let the pick-and-place path be located in a plane $u - v$ which is normal to the $x - y$ plane and let $z_1 = z_f$. Then, the coordinates of an arbitrary point $P(x, y, z)$ on the path with respect to the global frame $O - xyz$ can be expressed in terms of those with respect to the local frame $P_1 - uv$ shown in Fig. 1 by the coordinate transformation

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} u \\ 0 \\ v \end{pmatrix} + \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} \tag{1}$$

where $\gamma = \arctan\dfrac{y_f - y_1}{x_f - x_1}$. In addition, the width/height ratio of the path is defined as

$$\lambda = \frac{b}{h} \text{ with } b = \sqrt{\left(x_f - x_1\right)^2 + \left(y_f - y_1\right)^2} \text{ and } h = z_{\frac{f+1}{2}} - z_{1(f)} \tag{2}$$

2.2 Generation of a motion profile along an axis using quintic B-splines

A brief review of quintic B-splines [30] will lead to the formulation of the normalized motion profile along an axis of the geometric path. A B-spline of degree p is a linear combination of polynomials $N_{i,p}(\tau)$, called base or blending functions, weighted by coefficients Q_i named control points. τ is a normalized independent parameter here taken to represent time. The normalized motion profile $s(\tau)$ is then constructed as

$$s(\tau) = \sum_{i=1}^{n} Q_i \cdot N_{i,p}(\tau), \ 0 \le \tau \le 1 \tag{3}$$

with $N_{i,p}(\tau)$ satisfying the De Boor formula [30]

$$\begin{cases} N_{i,0}(\tau) = \begin{cases} 1, & \tau_i \le \tau \le \tau_{i+1} \\ 0, & \text{elsewhere} \end{cases} \\ N_{i,p}(\tau) = \dfrac{\tau - \tau_i}{\tau_{i+p} - \tau_i} N_{i,p-1}(\tau) + \dfrac{\tau_{i+p+1} - \tau}{\tau_{i+p+1} - \tau_{i+1}} N_{i+1,p-1}(\tau) \end{cases} \tag{4}$$

where τ_i ($i = 1, 2, \cdots, m = n + p + 1$) define a sequence of nodes necessary for building $s(\tau)$ by interpolation.

It can be proved that the rth derivative of $s(\tau)$ with respect to τ can be expressed as

$$\frac{\mathrm{d}^r s}{\mathrm{d}\tau^r} = \sum_{i=1}^{n-r} Q_{i,r} \cdot N_{i,p-r}(\tau) \tag{5}$$

where $Q_{i,r}$ ($i = 1,2,\cdots,n-r$) is the control point of the rth derivative of $s(\tau)$ and can be obtained recursively by

$$Q_{i,r} = \begin{cases} Q_i & r = 0 \\ \dfrac{p-r+1}{\tau_{i+p+1} - \tau_{i+r}}\left(Q_{i+1,r-1} - Q_{i,r-1}\right) & 0 < r \le p \end{cases} \tag{6}$$

The nodes at both ends of the path must be repeated $p+1$ times so that the control points coincide with the initial and final via-points, i.e. $s(0) = Q_1$ and $s(1) = Q_n$. Furthermore, two virtual points need to be introduced at the second and the second-last positions of the node sequence in order to obtain a trajectory with no jerk at either end [23]. These considerations result in $m = 2(p+1) + f$ and $n = p + f + 1$, with the required sequence of nodes being

$$\left\{ \underbrace{0,0,\cdots,0}_{p+1}, \tau_{p+2}, \underbrace{\tau_{p+3}, \tau_{p+4}, \cdots, \tau_{p+f}}_{f-2}, \tau_{p+f+1}, \underbrace{1,1,\cdots,1}_{p+1} \right\} \tag{7}$$

Since a B-spline of degree p is of C^{p-1}-continuity, it is necessary to take $p = 5$ to ensure a smooth trajectory in terms of jerk. This leads to $n = f + 6$ and $m = 12 + f$. Consequently, f constraint equations can be formulated using Eq. (4) and sequence (7) to ensure that the trajectory passes through f via-points,

$$\begin{cases} s(0) = s_1 = Q_1 \\ s\left(\tau_{p+j+1}\right) = s_j = \sum_{i=1}^{n} N_{i,p}\left(\tau_{p+j+1}\right) \cdot Q_i \quad j = 2,3,\cdots,f-1 \\ s(1) = s_f = Q_n \end{cases} \tag{8}$$

Also, six constraint equations are obtained by imposing the boundary conditions in terms of velocity, acceleration and jerk at both ends of the trajectory,

$$\begin{cases} Q_{1,r} = \left.\dfrac{\mathrm{d}^r s}{\mathrm{d}\tau^r}\right|_{\tau=0} = \sum_{i=1}^{r+1} C_{i,r} \cdot Q_i = 0 \\ Q_{n-r,r} = \left.\dfrac{\mathrm{d}^r s}{\mathrm{d}\tau^r}\right|_{\tau=1} = \sum_{i=n-r}^{n} C_{i,r} \cdot Q_i = 0 \end{cases} \quad r = 1,2,3 \tag{9}$$

where the coefficients $C_{i,r}$ ($i = 1,\cdots,r+1$ and $i = n-r,\cdots,n$) can be determined using Eqs. (5) and (6) in a recursive manner. Rewriting Eqs. (8) and (9) in matrix form then yields the solution to the control points

$$\boldsymbol{q} = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{10}$$

where

$$\boldsymbol{q} = \begin{pmatrix} Q_1 & Q_2 & \cdots & Q_n \end{pmatrix}^{\mathrm{T}}$$

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_1 \\ \boldsymbol{A}_2 \end{bmatrix}, \ \boldsymbol{b} = \begin{pmatrix} \boldsymbol{b}_1 \\ \boldsymbol{b}_2 \end{pmatrix}$$

$$\boldsymbol{b}_1 = \begin{pmatrix} s_1 & s_2 & \cdots & s_f \end{pmatrix}^{\mathrm{T}}, \ \boldsymbol{b}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^{\mathrm{T}}$$

$$\boldsymbol{A}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ N_{1,p}\left(\tau_{p+3}\right) & N_{2,p}\left(\tau_{p+3}\right) & \cdots & N_{n,p}\left(\tau_{p+3}\right) \\ N_{1,p}\left(\tau_{p+4}\right) & N_{2,p}\left(\tau_{p+4}\right) & \cdots & N_{n,p}\left(\tau_{p+4}\right) \\ \vdots & \vdots & \vdots & \vdots \\ N_{1,p}\left(\tau_{p+f}\right) & N_{2,p}\left(\tau_{p+f}\right) & \cdots & N_{n,p}\left(\tau_{p+f}\right) \\ 0 & 0 & \cdots & 1 \end{bmatrix}_{f\times n}$$

$$\boldsymbol{A}_2 = \begin{bmatrix} C_{1,1} & C_{2,1} & 0 & 0 & \cdots & & & & \\ C_{1,2} & C_{2,2} & C_{3,2} & 0 & \cdots & & & \mathbf{0} & \\ C_{1,3} & C_{2,3} & C_{3,3} & C_{4,3} & \cdots & & & & \\ & & & & \cdots & 0 & 0 & C_{n-1,1} & C_{n,1} \\ & & \mathbf{0} & & \cdots & 0 & C_{n-2,2} & C_{n-1,2} & C_{n,2} \\ & & & & \cdots & C_{n-3,3} & C_{n-2,3} & C_{n-1,3} & C_{n,3} \end{bmatrix}_{6\times n}$$

2.3 Generation of a motion profile along a path

The motion profile along the path will be generated from the profiles along the two orthogonal axes of $P_1 - uv$. Using the path description in Section 2.1, take $f = 5$ such that $n = f + 6 = 11$ and $m = 12 + f = 17$. This leads to five via-points, sequentially denoted by $P_i\left(u_i, v_i\right)$ ($i = 1, 2, \cdots, 5$), where P_2 is the lift-off point, P_4 is the set-down point and P_3 is the middle point. As discussed above, introduce two virtual points, P_{v1} between P_1 and P_2, and P_{v2} between P_4 and P_5. The whole set of relevant points is assumed to be symmetrical with respect to P_3. Hence, the sequence of via-points (including the virtual points) can be described by

$$P_1\left(0,0\right) \to P_{v1} \to P_2\left(0,\eta h\right) \to P_3\left(b/2, h\right) \to P_4\left(b,\eta h\right) \to P_{v2} \to P_5\left(b,0\right) \tag{11}$$

where $0 < \eta < 1$ is defined as the lift-off/set-down ratio, namely, the ratio of the *v*-coordinate of $P_{2(4)}$ to that of P_3. Corresponding to sequence (7), the sequence of nodes becomes

$$\left\{ \underbrace{0, 0, \cdots, 0}_{6}, \tau_7, \underbrace{\tau_8, \tau_9, \tau_{10}}_{3}, \tau_{11}, \underbrace{1, 1, \cdots, 1}_{6} \right\} \tag{12}$$

Two time ratio factors are now defined by

$$k_1 = \tau_8/\tau_9, \quad k_2 = \tau_7/\tau_8 \tag{13}$$

where $0 < k_1 < 1$ denotes the ratio between the time for the end-effector to move from P_1 to P_2 and the time from P_1 to P_3, whereas $0 < k_2 < 1$ represents the ratio of the time from P_1 to P_{v1} to that from P_1 to P_2. Keeping in mind the geometric symmetry of the point locations, the nodes can then be expressed in terms of k_1 and k_2 as

$$\tau_1 \cdots \tau_6 = 0, \ \tau_7 = \frac{k_1 k_2}{2}, \ \tau_8 = \frac{k_1}{2}, \ \tau_9 = \frac{1}{2}, \ \tau_{10} = 1 - \frac{k_1}{2}, \ \tau_{11} = 1 - \frac{k_1 k_2}{2}, \ \tau_{12} \cdots \tau_{17} = 1 \tag{14}$$

Adding a subscript '*u*' or '*v*' to identify the motion profile (see Eq. (3)) associated with an axis, the position vector, $\boldsymbol{r}\left(t\right)$, of an arbitrary point on the path can be expressed as

$$\boldsymbol{r}\left(t\right) = \begin{pmatrix} u\left(t\right) & v\left(t\right) \end{pmatrix}^{\mathrm{T}} = \frac{b}{T^0}\begin{pmatrix} s_u\left(\tau\right) & \lambda^{-1} s_v\left(\tau\right) \end{pmatrix}^{\mathrm{T}}, \ \tau = t/T \tag{15}$$

where T is the time taken for the end-effector to move from P_1 to P_5 by passing through all the via-points in between. In using Eq. (10) to determine the control points of $s_u(\tau)$ and $s_v(\tau)$, note that $\boldsymbol{A}_u = \boldsymbol{A}_v$ and $\boldsymbol{b}_{u,2} = \boldsymbol{b}_{v,2}$, while

$$\boldsymbol{b}_{u,1} = \begin{pmatrix} 0 & 0 & 1/2 & 1 & 1 \end{pmatrix}^{\mathrm{T}},\ \boldsymbol{b}_{v,1} = \begin{pmatrix} 0 & \eta & 1 & \eta & 0 \end{pmatrix}^{\mathrm{T}} \tag{16}$$

Taking the rth derivative of Eq. (15) with respect to time and normalizing the outcomes by b/T^r, yields the dimensionless position, velocity, acceleration and jerk of a point on the path,

$$\begin{aligned} \boldsymbol{s}(\tau) &= \begin{pmatrix} s_u(\tau) & \lambda^{-1} s_v(\tau) \end{pmatrix}^{\mathrm{T}},\ \boldsymbol{v}(\tau) = \begin{pmatrix} \dfrac{\mathrm{d}s_u}{\mathrm{d}\tau} & \lambda^{-1}\dfrac{\mathrm{d}s_v}{\mathrm{d}\tau} \end{pmatrix}^{\mathrm{T}} \\ \boldsymbol{a}(\tau) &= \begin{pmatrix} \dfrac{\mathrm{d}^2 s_u}{\mathrm{d}\tau^2} & \lambda^{-1}\dfrac{\mathrm{d}^2 s_v}{\mathrm{d}\tau^2} \end{pmatrix}^{\mathrm{T}},\ \boldsymbol{j}(\tau) = \begin{pmatrix} \dfrac{\mathrm{d}^3 s_u}{\mathrm{d}\tau^3} & \lambda^{-1}\dfrac{\mathrm{d}^3 s_v}{\mathrm{d}\tau^3} \end{pmatrix}^{\mathrm{T}} \end{aligned} \tag{17}$$

Thus, given a reasonable value of η for a pick-and-place operation, $\boldsymbol{s}(\tau)$, $\boldsymbol{v}(\tau)$, $\boldsymbol{a}(\tau)$ and $\boldsymbol{j}(\tau)$ are dominated by k_1 and k_2 once λ is specified. The method for determining k_1 and k_2 will now be addressed.

3 Smooth Trajectory Planning in the Cartesian Space

3.1 The performance indices

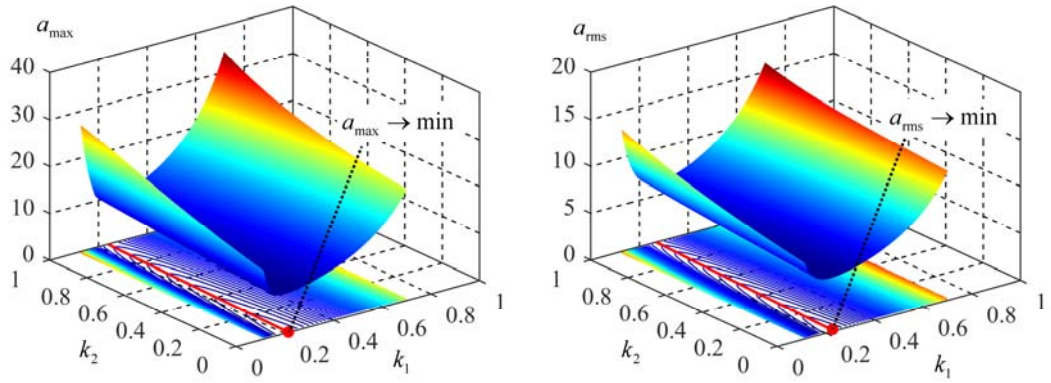
Just as for trajectory planning in the joint space, the quality of a trajectory planned in the Cartesian space can be evaluated using appropriate performance indices based upon the second and third time derivatives of the motion profile (acceleration and jerk) of the end-effector. This is because the peak values and smoothness of the time histories of the joint torques are closely related *via* inverse kinematics and dynamics to those of acceleration and jerk of the end-effector at non-singular configurations [31]. Given a specified width/height ratio λ, two pairs of performance indices are proposed for evaluating acceleration and jerk of the end-effector over the entire normalized path within the normalized time duration, that is effectively taking $b = 1$ and $T = 1$. They are defined as functions of k_1 and k_2,

$$a_{\max}(k_1, k_2) = \max_{0 \le \tau \le 1} \left\| \boldsymbol{a}(\tau) \right\|,\ a_{\mathrm{rms}}(k_1, k_2) = \sqrt{\int_0^1 \left\| \boldsymbol{a}(\tau) \right\|^2 \mathrm{d}\tau} \tag{18}$$

$$j_{\max}(k_1, k_2) = \max_{0 \le \tau \le 1} \left\| \boldsymbol{j}(\tau) \right\|,\ j_{\mathrm{rms}}(k_1, k_2) = \sqrt{\int_0^1 \left\| \boldsymbol{j}(\tau) \right\|^2 \mathrm{d}\tau} \tag{19}$$

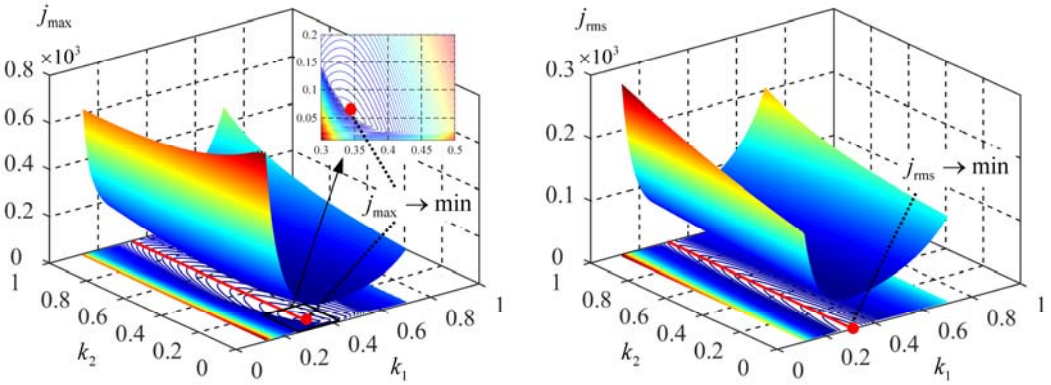
The two indices in the first pair (Eq. (18)) represent the maximum and root mean square acceleration, whereas those in the second are related to the jerk.

An example suffices to assess these indices without loss of generality. Taking $\lambda = 28$ and $\eta = 0.5$ for the Extended Adept Cycle (EAC) [3], Fig. 2(a) and (b) show distributions of $a_{\max}$ and a_{rms} *vs.* k_1 and k_2. They look extremely similar to each other in shape. The local minimum of $a_{\max}$ (a_{rms}) lies along a valley represented by a nearly straight line (see the red line in the contour diagram) and monotonically increases with increasing k_2 such that it takes a global minimum at $k_1 = 0.21$ ($k_1 = 0.23$) when $k_2 \to 0$. Also, $a_{\max}(a_{\mathrm{rms}}) \to \infty$ when $k_1 \to 0$ or $k_1 \to 1$. The distributions of $j_{\max}$ and j_{rms} *vs.* k_1 and k_2 shown in Fig. 3(a) and (b) behave in a similar way. The global minimum of $j_{\max}$ occurs at $k_1 = 0.34$ and $k_2 = 0.07$, while j_{rms} takes a minimum value at $k_1 = 0.31$ when $k_2 \to 0$. Since $k_2 \to 0$ leads to violation of the jerk boundary conditions, it is rational to use $a_{\max}$ and $j_{\max}$ as the performance indices of a trajectory.



(a) (b)

Fig. 2. Distributions of $a_{\max}$ and a_{rms} versus k_1 and k_2 with $\lambda = 28$ and $\eta = 0.5$



(a) (b)

Fig. 3. Distributions of $j_{\max}$ and j_{rms} versus k_1 and k_2 with $\lambda = 28$ and $\eta = 0.5$

3.2 The optimization problem

Considering that $a_{\max}$ and $j_{\max}$ are functions of k_1 and k_2, two normalized single objective functions can be formulated for maximization, i.e.,

$$F_a(k_1, k_2) = \frac{\min\limits_{0<k_1<1} \min\limits_{0<k_2<1} a_{\max}(k_1, k_2)}{a_{\max}(k_1, k_2)} \to \max \tag{20}$$

$$F_j(k_1, k_2) = \frac{\min\limits_{0<k_1<1} \min\limits_{0<k_2<1} j_{\max}(k_1, k_2)}{j_{\max}(k_1, k_2)} \to \max \tag{21}$$

Obviously, the maximum acceleration (jerk) takes its minimum value when $F_a(F_j) \to 1$. Hence, a weighed objective function is created using the linear scalarization method,

$$\begin{aligned} &\max \quad F = w_a F_a + w_j F_j \\ &\text{subject to:} \\ &\qquad 0 < k_1 < 1,\ 0 < k_2 < 1 \end{aligned} \tag{22}$$

where the weights $w_a > 0$ and $w_j > 0$ with $w_a + w_j = 1$ can be suitably adjusted to trade off the significance between two normalized single-objective functions F_a and F_j to best match particular applications. For example, a larger weight of F_a will lead to lower joint torques, while a larger weight of F_j will lead to lower rates of change of joint torques. Concentrating here on the most likely, general-purpose case where $w_a = w_j = 0.5$, Fig. 4 shows that the maximum value of

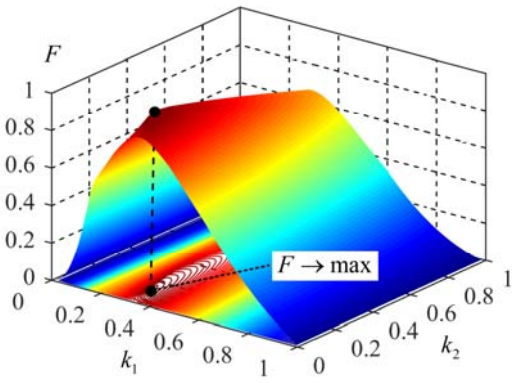


Fig. 4. Variation of F versus k_1 and k_2 ($w_a = w_j = 0.5$)

F is achieved at $k_1^* = 0.31$ and $k_2^* = 0.12$. Global optimality exists because F and its gradient are continuous and F is convex in the feasible domain of k_1 and k_2 (see Fig. 4). Thus, the optimization problem can be solved with ease by, for example, calling the fmincon function in the Matlab® Optimization Toolbox.

Again taking the EAC path with $\lambda = 28$ and $\eta = 0.5$, Fig.5 shows the norms of acceleration and jerk generated by three pick-and-place trajectories planned in the Cartesian space. The first uses two symmetric Lamé curves to link the lift-off and set-down points with a 4-5-6-7 polynomial providing the motion profile along the entire path [10]. The second employs piecewise 4-5-6-7 polynomials [4] for the motion profile along two coordinate axes with the optimized $\tau_v / \tau_u = 0.35$, where τ_v and τ_u are the normalized time spent from P_1 to P_2 and from P_2 to P_3, respectively. The third adopts the fifth-order B-spline as the motion profile along two coordinate axes with the optimized $k_1^* = 0.31$ and $k_2^* = 0.12$ as proposed in this paper. We name three trajectories as 'Lamé curve', 'Piecewise poly' and 'B-spline', for short. The reasons why this comparison study considers only the Lamé curve and the 4-5-6-7 polynomial motion profile are: (1) for path-based trajectory planning, the Lamé curve exhibits superior smoothness over many other paths, the clothoid for instance [15]; (2) for coordinate-based trajectory planning, the 4-5-6-7 polynomial exhibits better performance than the cycloid and modified trapezoid, *etc*. Fig. 5 shows that the 'B-spline' exhibits overall superior behavior in terms of both peak value and smoothness compared with the other two trajectories. Table 1 shows their performance indices.

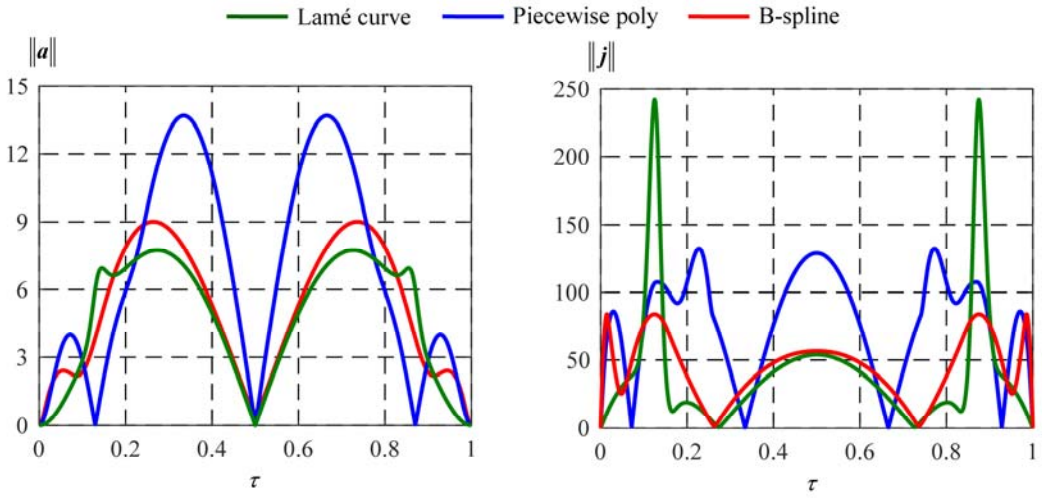


Fig. 5. Behaviors of three trajectories planned in the Cartesian space

Table 1 Performance indices of three motion profiles

	Lamé curve	Piecewise poly	B-spline
$a_{\max}$	7.8	13.7	9.0
$j_{\max}$	242.1	132.5	84.0

Finally, an analytical expression of the optimized k_1^* and k_2^* *vs.* λ can be developed by means of curve fitting using a third-order polynomial

$$k_i^* = \sum_{j=0}^{3} \lambda^j g_{i,j} \ , \ i = 1,2 \tag{23}$$

where the coefficients $g_{i,j}$ are given in Table 2 and the fitted curves are shown in Fig. 6. This treatment allows a look-up table to be developed to determine k_1^* and k_2^* for any λ assigned within a reasonable range. Obviously, k_1^* and k_2^* are independent of the horizontal width b and the cycle time T.

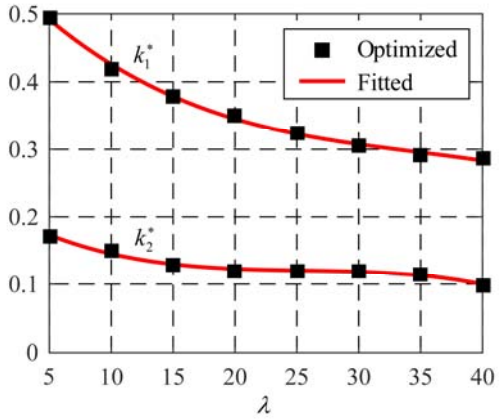


Fig. 6. The optimized k_1^* and k_2^* versus λ

Table 2 Coefficients of the third-order polynomial, Eq. (23)

	$j=0$	$j=1$	$j=2$	$j=3$
$g_{1,j}$	0.579	-1.99×10^{-2}	5.09×10^{-4}	-4.91×10^{-6}
$g_{2,j}$	0.218	-1.09×10^{-2}	4.15×10^{-4}	-5.41×10^{-6}

3.3 Determination of T

In real-time implementation, b can directly be determined by the distance between the initial and final via-points, while the cycle time T must be subject to the limits of motor's speed and torque constraints. These limits can be set in accordance with the specifications of the motor and gear reducer used. Then, the velocity and acceleration limits of the end-effector, defined as the movement capacity and denoted by $[v]$ and $[a]$, over the entire task workspace can be determined by means of the singular decomposition technique proposed in [5], leading to the following constraints

$$\max_{0\le t\le T} \left\| \frac{\mathrm{d}\boldsymbol{r}}{\mathrm{d}t} \right\| = \frac{b}{T} v_{\max} \le [v] \tag{24}$$

$$\max_{0\le t\le T} \left\| \frac{\mathrm{d}^2\boldsymbol{r}}{\mathrm{d}t^2} \right\| = \frac{b}{T^2} a_{\max} \le [a] \tag{25}$$

where $v_{\max} = \max_{0\le\tau\le 1} \|\boldsymbol{v}(\tau)\|$ and $a_{\max} = \max_{0\le\tau\le 1} \|\boldsymbol{a}(\tau)\|$. Hence, the cycle time T must be subject to

$$T \ge \max\left(\frac{v_{\max} b}{[v]} \quad \sqrt{\frac{a_{\max} b}{[a]}} \right) \tag{26}$$

Consequently, given T and b, the optimal smooth motion profiles $u(t)$ and $v(t)$ given in Eq. (15) can be interpolated, transformed into the coordinates $x(t)$, $y(t)$ and $z(t)$ with respect to $O-xyz$ by Eq. (1), and converted into joint variables *via* inverse kinematics for real-time implementation. Finally, it is worthwhile pointing out that the proposed trajectory is not time-minimum because the limits $[v]$ and $[a]$ represent the upper bounds of the maximum allowable velocity and acceleration of the end-effector according to the motor specifications.

4 Verification

Both simulation and physical experiments have been carried out on a 4-DOF SCARA type parallel robot [5] to verify the effectiveness of the proposed approach. Fig. 7 shows a 3D view of the robot used. It has two identical closed-loop sub-chains, each consisting of two identical R-$(SS)^2$ limbs connected with the base at one end and with either subpart 1 or 2 of the travelling plate at the other. Subparts 1 and 2 are articulated by prismatic joints to subpart 3 (the end-effector). The required rotation of $\pm\pi$ about the z axis is generated from relative translation between subparts 1 and 2 *via* a rack-and-pinion assembly centered on subpart 3. For details about the dimensional and structural parameters of the robot, please refer to [5]. Note that the joint axes of the four forearms are symmetrically placed with respect to the $u-v$ plane, as shown in Fig. 8, such that the motion of joint 1(3) is the same as that of joint 2(4). Thus, only the motions of joints 1 and 3 will be analyzed.

The simulations assume that the end-effector undergoes pure translation along the specified EAC path by fixing the rotation about the vertical axis. Given $[v] = 10$ m/s and $[a] = 160$ m/s^2 according to the movement capacity of the robot, the cycle time for 'Lamé curve', 'Piecewise poly' and 'B-spline' is 0.185 s, 0.245 s and 0.198 s, respectively. Hence, $T = 0.25$ s is taken for the comparative study. Fig. 9 shows that the joint acceleration generated by 'B-spline' is similar in magnitude to that generated by 'Lamé curve' but much less than that generated by 'Piecewise poly'. Furthermore, the overall acceleration curves generated by 'B-spline' are smoother than those of the other two trajectories. Similar conclusions are drawn from Fig. 10 for the jerk analysis. These results clearly demonstrate that desirably small, continuous and smooth joint torques are achieved using a trajectory planned by the fifth-order B-splines in the Cartesian space.

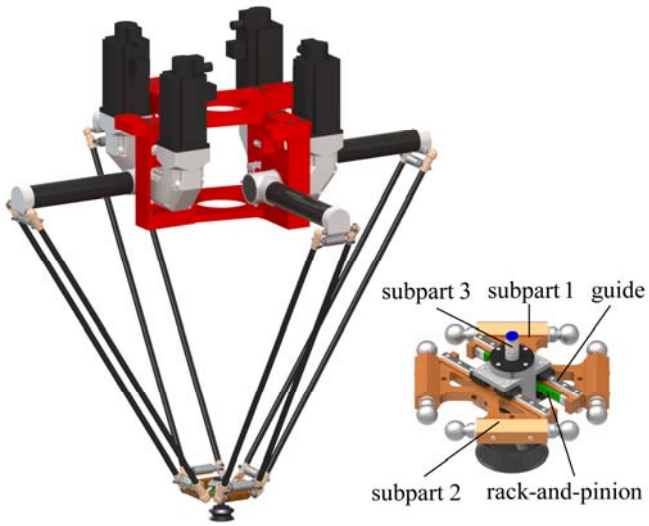


Fig. 7. 3D view of the 4-DOF SCARA parallel robot

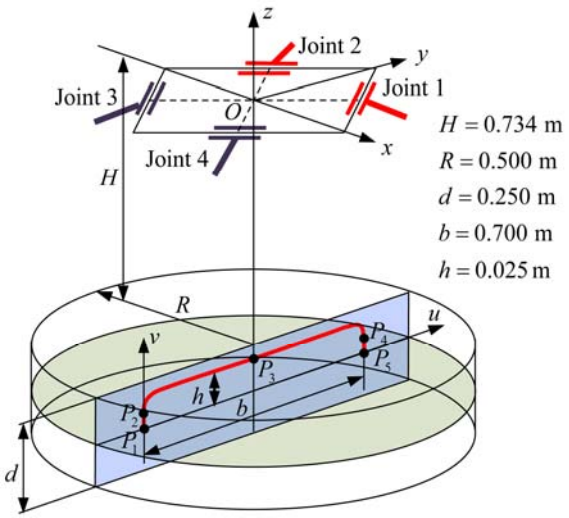


Fig. 8. Path of the Extended Adept Cycle in the cylindrical task workspace of the robot

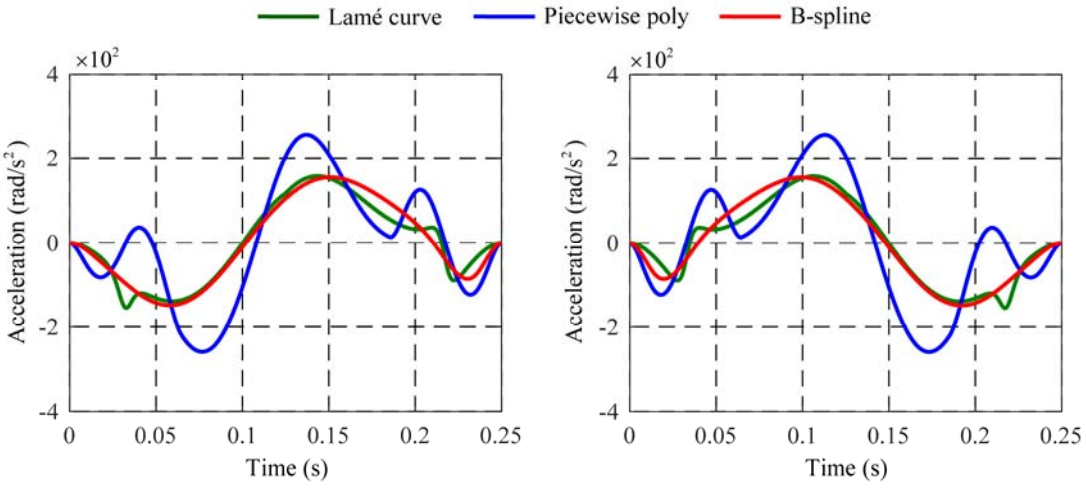


Fig. 9. Acceleration of (a) joint 1 and (b) joint 3 versus time

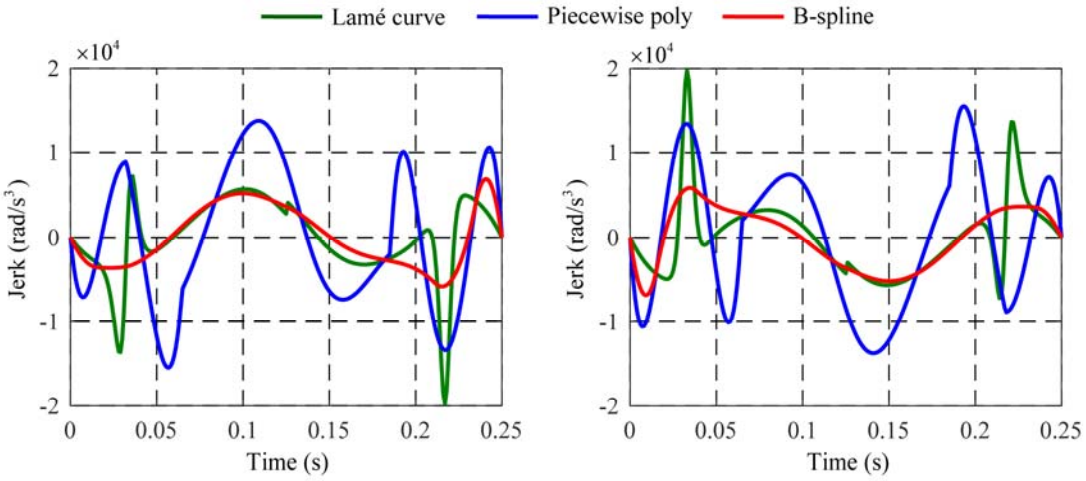


Fig. 10. Jerk of (a) joint 1 and (b) joint 3 versus time

With the simulations providing confidence in the method, experiments were carried out on the prototype robot shown in Fig. 11 to examine the realistic joint torques and the residual vibrations of the traveling plate. The joint torques were sensed by the servomotor currents, and the residual vibrations along the *u* and *v* axes measured by an IEPE MEAS 7131A-0500 3D accelerometer and analyzed by an LMS Test Lab-Signature Testing Processor. Fig. 12 shows the measured joint torque curves. It is easy to see that the toque curve of joint 1 (3) is similar in shape to that of joint 2(4) because of trajectory symmetry. The peak values of the joint torques generated by 'B-spline' are similar to those generated by 'Lamé curve' but up to 40% less than those generated by 'Piecewise poly'. The overall torque curves generated by 'B-spline' are smoother than those generated by the other two trajectories. These observations coincide exactly with the conclusions from acceleration analysis *via* the simulation. In addition, Fig. 12 clearly indicates that the residual fluctuations in joint torques associated with 'Lamé curve', 'Piecewise poly and 'B-spline' decrease successively. Consequently, the maximum magnitude of the residual vibration generated by 'B-spline' along the *u* axis is 39% and 53%, respectively, of that associated with 'Lamé curve' and 'Piecewise poly'. The corresponding reductions along the *v* axis are 46% and 55%. The results, summarized in Fig. 13 and Table 3, clearly show that the new approach enhances the dynamic positioning accuracy of the end-effector. The experiments for asymmetrical trajectories were also carried out to test the generality of the proposed method, resulting in the similar observations though they will not be reported here due to space limitations. Hence, the proposed trajectory planned in the Cartesian space is highly effective in residual vibration suppression thanks to the very continuous and smooth joint torques it generates.

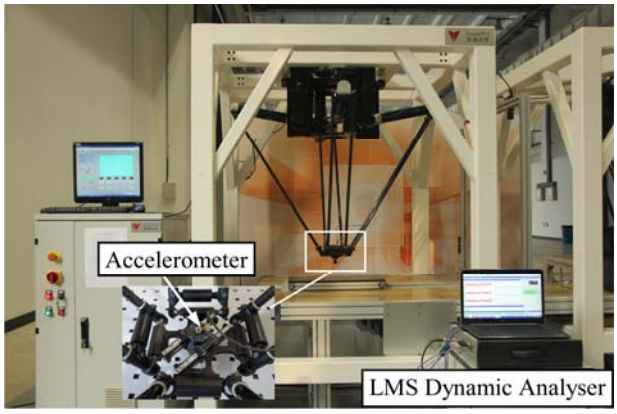


Fig. 11. Experimental set-up

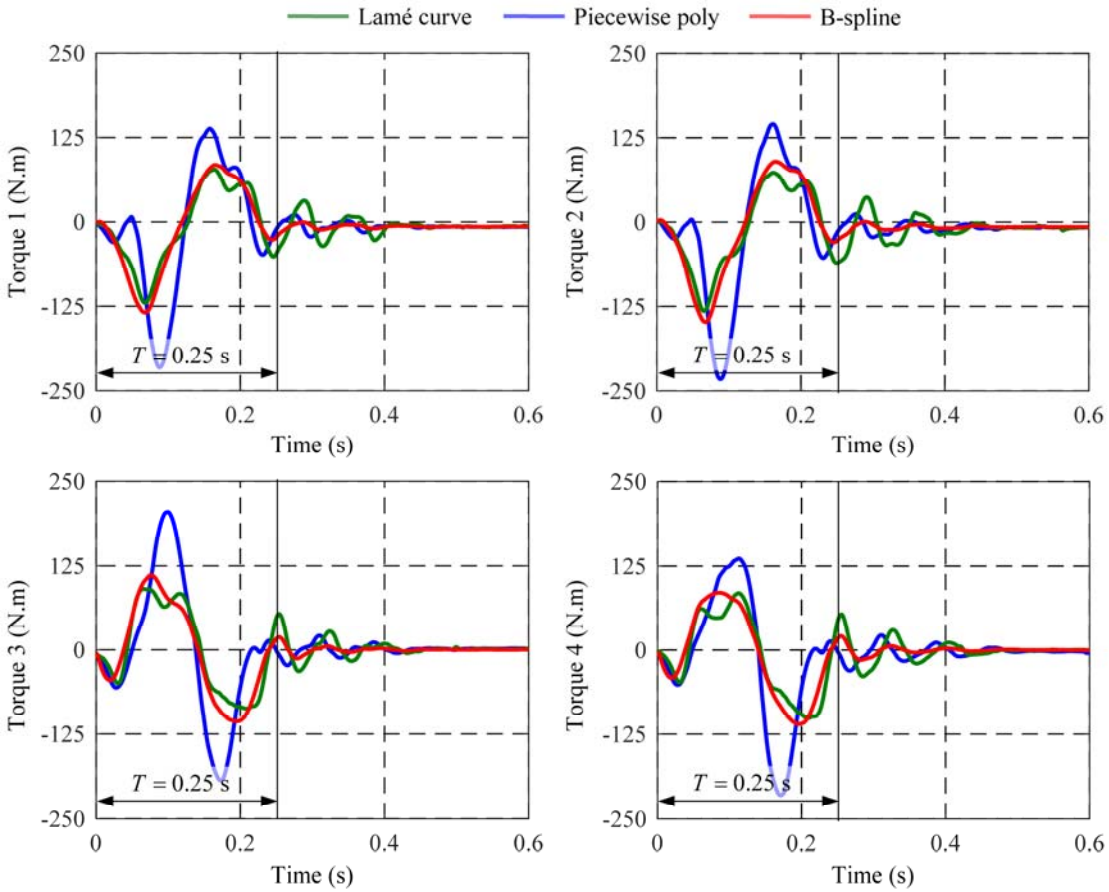


Fig. 12. Measured torque of four actuated joints

5 Conclusions

This paper proposes and demonstrates a new approach for optimal smooth trajectory planning of high-speed pick-and-place parallel robots using the fifth-order B-splines. It exploits the symmetric properties of the path defined in a local frame to propose a motion profile that is essentially dominated by two time ratio parameters. They can be determined in an off-line manner by maximizing a weighted sum of two single-objective functions and expressed as a function of the width/height ratio of the pick-and-place path in the Cartesian space. This then allows the online generation of the smooth joint trajectories using a look-up table. Compared with two existing trajectories planned in the Cartesian space, both simulation and experimental results on a 4-DOF SCARA type parallel robot show that the time profiles of the joint torques are relatively small and very continuous and smooth, with residual vibration of the end-effector substantially reduced.

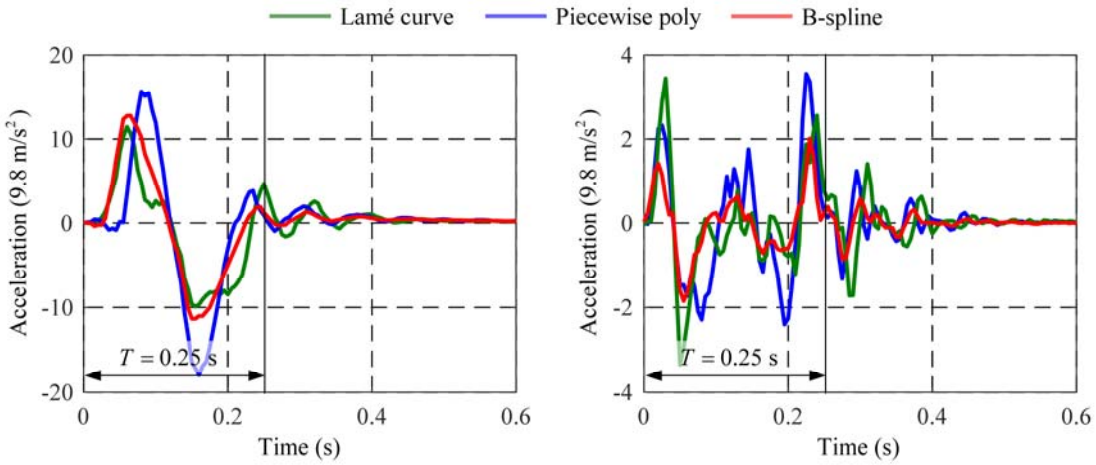


Fig. 13. Acceleration of the end-effector in the (a) *u* axis and (b) *v* axis

Table 3 Maximal amplitudes of residual vibration (unit: 9.8m/s^2)

	Lamé curve	Piecewise poly	B-spline
In the *u* axis	4.20	3.08	1.63
In the *v* axis	3.12	2.57	1.42

Acknowledgement

This work is partially supported by the National Natural Science Foundation of China (NSFC) under grants 51420105007 and 51135008.

References

[1] R. Clavel, Delta, a fast robot with parallel geometry, 18th International Symposium on Industrial Robots (ISIR), Sydney, Australia, (1988) 91–100.

[2] T. Huang, Z.X. Li., M. Li, D.G. Chetwynd, C.M. Gosselin, Conceptual design and dimensional synthesis of a novel 2-DOF translational parallel robot for pick-and-place operations, AMSE Journal of Mechanical Design 126(3) (2004) 449-455.

[3] F. Pierrot, V. Nabat, O. Company, S. Krut, P. Poignet, Optimal design of a 4-DOF parallel manipulator: from academia to industry, IEEE Transactions on Robotics 25(2) (2009) 213-224.

[4] S.T. Liu, T. Huang, J.P. Mei, X.M. Zhao, P.F. Wang, D.G. Chetwynd, Optimal design of a 4-DOF SCARA type parallel robot using dynamic performance indices and angular constraints, ASME Journal of Mechanisms and Robotics 4(3) (2012) 031005.

[5] Y.H. Li, Y. Ma, S.T. Liu, Z.J. Luo, J.P. Mei, T. Huang, D.G. Chetwynd, Integrated design of a 4-DOF high-speed pick-and-place parallel robot, CIRP Annals - Manufacturing Technology 63(1) (2014) 185-188.

[6] L. Biagiotti, C. Melchiorri, Trajectory planning for automatic machines and robots, Springer-Verlag, Berlin Heidelberg (2008).

[7] H.Z. Li, M.D. Le, Z.M. Gong, W. Lin, Motion profile design to reduce residual vibration of high-speed positioning stages, IEEE/ASME Transactions on Mechatronics 14(2) (2009) 264-269.

[8] Y.Q. Zhang, R.N. Huang, Y.J. Lou, Z.X. Li, Dynamics based time-optimal smooth motion planning for the Delta robot, IEEE International Conference on Robotics and Biomimetics (ROBIO), Guangzhou, China, Dec. 11-14 (2012) 1789-1794.

[9] M. Pellicciari, G. Berselli, F. Leali, A. Vergnano, A method for reducing the energy consumption of pick-and-place industrial robots, Mechatronics 23(3) (2013) 326-334.

[10] J.-F. Gauthier, J. Angeles, S. Nokleby, Optimization of a test trajectory for SCARA systems, Advances in Robot Kinematics: Analysis and Design (2008) 225-234.

[11] R.J. M. Masey, J.O. Gray, T.J. Dodd, D.G. Caldwell, Elliptical point to point trajectory planning using electronic cam motion profiles for high speed industrial pick and place robots, IEEE Conference on Emerging Technologies and Factory Automation (ETFA), Mallorca, Spain, Sept. 22-25 (2009) 1-8.

[12] Z.W. Dai, X.J. Sheng, J. Hu, H. Wang, D.G. Zhang, Design and implementation of Bézier curve trajectory planning in DELTA parallel robots, International Conference on Intelligent Robotics and Applications (ICIRA), Portsmouth, UK, Aug. 24 (2015) 420-430.

[13] V. Nabat, M. de la O Rodriguez, O. Company, S. Krut, F. Pierrot, Par4: very high speed parallel robot for pick-and-place, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Alberta, Canada, Aug. 2-6 (2005) 1202-1207.

[14] T. Huang, P.F. Wang, J.P. Mei, X.M. Zhao, D.G. Chetwynd, Time minimum trajectory planning of a 2-DOF translational parallel robot for pick-and-place operations, CIRP Annals - Manufacturing Technology 56(1) (2007) 365-368.

[15] C.J. Barnard, S. Briot, S. Caro, Trajectory generation for high speed pick and place robots, ASME 2012 11th Biennial Conference on Engineering Systems Design and Analysis (ESDA 2012), Nantes, France, Jul. 2–4 (2012) pp. 165-174.

[16] Z.X. Xie, P.X. Wu, P. Ren, A comparative study on the pick-and-place trajectories for a Delta robot, ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE 2016), Charlotte, North Carolina, USA, Aug. 21–24 (2016) V05AT07A040.
[17] J. Angeles, A. Alivizatos, P.J. Zsombor-Murray, The synthesis of smooth trajectories for pick-and-place operations, IEEE Transactions on Systems, Man and Cybernetics 18(1) (1988) 173-178.
[18] C. M. Gosselin, A. Hadj-Messaoud, Automatic planning of smooth trajectories for pick-and-place operations, AMSE Journal of Mechanical Design 115(3) (1993) 450-456.
[19] A. Piazzi, A. Visioli, Global minimum-jerk trajectory planning of robot manipulators, IEEE Transactions on Industrial Electronics 47(1) (2000) 140-149.
[20] S. Macfarlane, E.A. Croft, Jerk-bounded manipulator trajectory planning: design for real-time applications, IEEE Transactions on Robotics and Automation 19**(**1) (2003) 42-52.
[21] H.S. Liu, X.B. Lai, W.X. Wu, Time-optimal and jerk-continuous trajectory planning for robot manipulators with kinematic constraints, Robotics and Computer-Integrated Manufacturing 29(2) (2013) 309-317.
[22] D. Constantinescu, E.A. Croft, Smooth and time-optimal trajectory planning for industrial manipulators along specified paths, Journal of Robotic Systems 17(5) (2000) 233-249.
[23] A. Gasparetto, V. Zanotto, A new method for smooth trajectory planning of robot manipulators, Mechanism and Machine Theory 42(4) (2007) 455–471.
[24] A. Gasparetto, V. Zanotto, A technique for time-jerk optimal planning of robot trajectories, Robotics and Computer-Integrated Manufacturing 24(3) (2008) 415–426.
[25] A. Gasparetto, A. Lanzutti, R. Vidoni, V. Zanotto, Validation of minimum time-jerk algorithms for trajectory planning of industrial robots, ASME Journal of Mechanisms and Robotics 3(3) (2011) 031003.
[26] S.A. Bazaz, B. Tondu, 3-cubic spline for on-line Cartesian space trajectory planning of an industrial manipulator, International Workshop on Advanced Motion Control (AMC 98), Coimbra, Portugal, Jun. 29-Jul. 1 (1998) 493-498.
[27] F. Valero, V. Mata, A. Besa, Trajectory planning in workspaces with obstacles taking into account the dynamic robot behaviour, Mechanism and Machine Theory 41(5) (2006) 525-536.
[28] D.S. Chen, B.G. Zhang, M. Wang, Cartesian space trajectory planning on 7-DOF manipulator, IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, China, Dec. 6-9 (2015) 940-945.
[29] M.R. Azizi, R. Khani, An algorithm for smooth trajectory planning optimization of isotropic translational parallel manipulators, Proceedings of the Institution of Mechanical Engineers Part C 230(12) (2016) 1987-2002.
[30] L. Piegl, W. Tiller, The NURBS book, 2nd ed., Springer-Verlag, New York (1997).
[31] P.J. Barre, R. Bearee, P. Borne, E. Dumetz, Influence of a jerk controlled movement law on the vibratory behaviour of high-dynamics systems, Journal of Intelligent and Robotic Systems 42(3) (2005) 275-293.