

Smooth trajectory generation for rotating extensible manipulators

M. Dupac

Department of Design and Engineering, Faculty of Science and Technology,
Bournemouth University, Poole, United Kingdom
E-mail: mdupac@bournemouth.ac.uk

Abstract

In this study the generation of smooth trajectories of the end-effector of a rotating extensible manipulator arm is considered. Possible trajectories are modelled using Cartesian and polar piecewise cubic interpolants expressed as polynomial Hermite-type functions. The use of polar piecewise cubic interpolants devises continuous first and - in some cases - second order derivatives and allows easy calculation of kinematics variables such as velocity and acceleration. Moreover, the manipulator equations of motion can be easily handled, and the constrained trajectory of the non-active end of the manipulator derived directly from the position of the end-effector. To verify the proposed approach, numerical simulations are conducted for two different configurations.

Keywords: Mathematical modelling, smooth trajectories, manipulator, robotic arm

1 Introduction

One of the important problems in robotics is the generation of smooth trajectories that minimise system energy, jerk and deflection caused by flexibility, clearance and components deformation. The problem can be approached by "interpolating" or "approximating" the desired path by piecewise interpolating curves through a set of a priori defined points that control the manipulator trajectory.

Various approaches to generate trajectories have been considered so far. Among these, piecewise interpolating curves with slope continuity, geometrically continuous Catmull-Rom splines [4], parametric and/or geometric continuous splines [9, 15] or uniform cubic B-Spline with parametric and geometric continuity, proved to be adequate in generating a smooth motion especially when manipulator dynamics [13] is considered. Lately, cartesian univariate and bivariate splines have been successfully used in [14,15] for preserving the shape of multiscale data.

Interpolation in curvilinear or polar coordinate systems - which have received a considerable attention lately - may represent an excellent alternative to smooth trajectory generation especially when rotating robotic systems are involved. A general approach to polar coordinate interpolation has been considered in [18, 20]. Polar coordinate interpolation including polynomial splines approximation has been discussed in [16] while single values splines and splines focales in [19, 21]. A survey on trigonometric splines has been presented in [23] while trigonometric B-splines have been considered in [17].

In this paper a mathematical approach for generating the smooth trajectory of the end effector of a rotating extensible robotic arm is presented. Cartesian and polar piecewise polynomial interpolating curves are considered for the generation of the geometric path of the end-effector. To verify the proposed approach, the trajectory and the velocity profile for the end effector and non-active end of the constrained trajectory are computed for two different configurations.

2 Mathematical Modelling of Manipulator Trajectory

2.1 System Model

The rotating extensible robotic arm shown in Fig. 1 is composed of a rigid guide OE and a sliding rod SP which is constrained to a curved trajectory by the end S . The non-active end of the sliding part SP is denoted by S and its active end-effector is denoted by P .

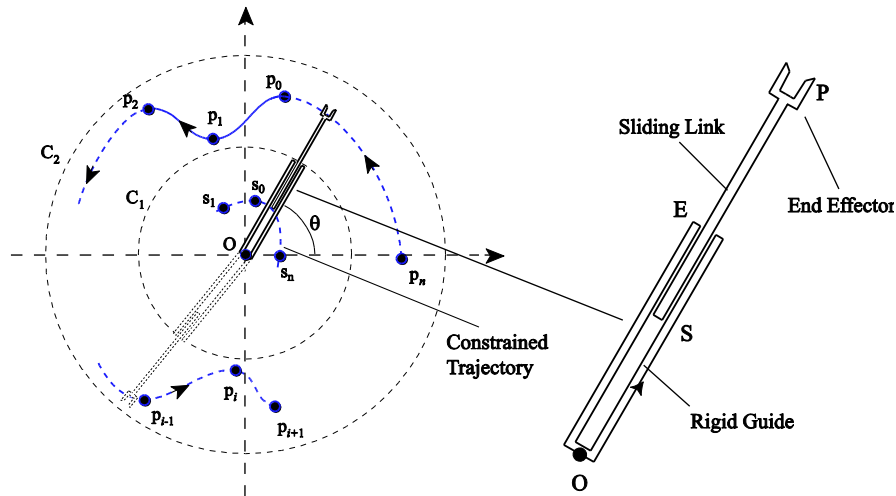


Figure 1: Rotating extensible robotic arm system model

The rigid guide of the robotic arm has length d_{OE} , the sliding part has the length d_{SP} . The total length of the robotic arm denoted by $r=d_{OP}$ (distance between the manipulator base location O and its end-effector P) varies due to the rotation of the rigid guide of the extensible manipulator. The end effector of the robotic arm OP should reach the points $p_i(x_i, y_i)_{i=1,2,\dots,n}$. The path followed by the non-active end S of the link SP controls the position of the active end of the robotic arm defined by the points $p_i(x_i, y_i)_{i=1,2,\dots,n}$ shown in Fig. 2.

2.2 Piecewise Trajectory Generation

2.2.1 Cartesian interpolating curves

In order to achieve some desired properties of the manipulator trajectory piecewise polynomial interpolating curves may be considered. Such interpolating curves [2, 3, 8, 9, 11, 12, 13] guarantee slope continuity, and/or minimal data storage, and/or local control and smoothness (no abrupt changes in displacement and velocity). For the general case, Hermite interpolation is given based on derivative values at data points $y(x_i) = y_i$, $\dot{y}(x_i) = \dot{y}_i$, $\ddot{y}(x_i) = \ddot{y}_i$, ..., on the finite real interval $[x_0, x_N]$, where $x_0 < x_1 < x_2 < \dots < x_{N-1} < x_N$. In each interval $\{x_i, x_{i+1}\}_{i=0, N-1}$ defined by the two consecutive control points p_i , p_{i+1} , a piecewise cubic interpolant can be expressed as a Hermite-type function [15,22] (continuous first derivative) by

$$y(x) = \sum_{k=0}^3 m_k^i (x - x_i)^k \quad (1)$$

where $h_i = x_{i+1} - x_i$, $y_i = y(x_i)$, $y_{i+1} = y(x_{i+1})$, $\Delta y_i = \frac{y_{i+1} - y_i}{h_i}$, $m_0^i = y_i$, $m_1^i = \dot{y}_i$, $m_2^i = \frac{1}{h_i} [-(2\dot{y}_i + \dot{y}_{i+1}) + 3\Delta y_i]$, $m_3^i = \frac{1}{h_i^2} [\dot{y}_i + \dot{y}_{i+1} - 2\Delta y_i]$, and where the derivatives at the endpoints p_i , p_{i+1} are $\dot{y}(x_i) = \frac{dy(x_i)}{dx} = \dot{y}_i$ and $\dot{y}(x_{i+1}) = \frac{dy(x_{i+1})}{dx} = \dot{y}_{i+1}$ respectively.

Hermite cubic interpolation has also a continuous second derivative if the given first derivative is exact. A more complex representation generally having continuous first and second derivatives - which guarantee a smooth trajectory of the manipulator end-effector - is given by quintic Hermite interpolation in Eq.(1) with $m_0^i = y_i$, $m_1^i = \dot{y}_i$,

$$m_2^i = \ddot{y}_i, \quad m_3^i = \frac{\ddot{y}_{i+1} - 3\ddot{y}_i}{2h_i} - \frac{4\dot{y}_{i+1} + 6\dot{y}_i - 10\Delta y_i}{h_i^2}, \quad m_4^i = \frac{3\ddot{y}_i - 2\ddot{y}_{i+1}}{2h_i^2} + \frac{7\dot{y}_{i+1} + 8\dot{y}_i - 15\Delta y_i}{h_i^3},$$

$$m_5^i = \frac{\ddot{y}_{i+1} - \ddot{y}_i}{2h_i^3} - \frac{\dot{y}_{i+1} + \dot{y}_i - 6\Delta y_i}{h_i^4}$$

2.2.2 Polar interpolating curves

To interpolate the periodic data $\{r_k, \theta_k\}_{k=0, N}$ which define the rotating extending link with $r_k > 0 \quad \forall k$ and $r_N = r_0$, one can consider the strictly monotonic angles $\theta_0 < \theta_1 < \theta_2 < \dots < \theta_{N-1} < \theta_N = \theta_0 + 2\pi$ defined on the real interval $[\theta_0, \theta_0 + 2\pi]$. In each interval $\{\theta_i, \theta_{i+1}\}_{i=0, N-1}$ defined by the two consecutive control points p_i , p_{i+1} , a piecewise cubic interpolant (Fig. 2) can be expressed as a Hermite-type function [15,22] by

$$r(\theta) = \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \quad (2)$$

where $h_i = \theta_{i+1} - \theta_i$, $\Delta y_i = \frac{r_{i+1} - r_i}{h_i}$, $m_0^i = r_i$, $m_1^i = \dot{r}_i$, $m_2^i = \frac{1}{h_i} [-(2\dot{r}_i + \dot{r}_{i+1}) + 3\Delta r_i]$, $m_3^i = \frac{1}{h_i^2} [\dot{r}_i + \dot{r}_{i+1} - 2\Delta r_i]$, $r_i = r(\theta_i)$, $r_{i+1} = r(\theta_{i+1})$, and where the derivatives at the endpoints p_i , p_{i+1} are $\dot{r}(\theta_i) = \frac{dr(\theta_i)}{d\theta} = \dot{r}_i$ and $\dot{r}(\theta_{i+1}) = \frac{dr(\theta_{i+1})}{d\theta} = \dot{r}_{i+1}$ respectively.

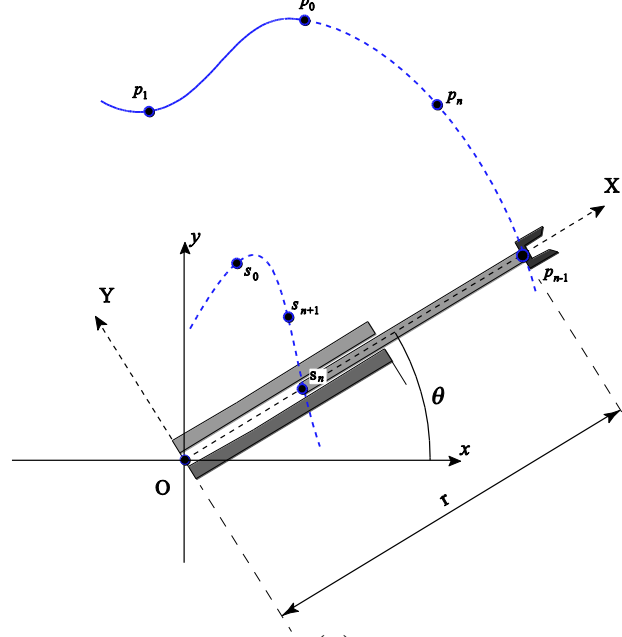


Figure 2: Trajectory of the end effector $r(\theta)$ expressed as a Hermite-type function, and trajectory of the non-active end of the link

One can parametrize the Cartesian coordinates using the polar coordinate $q(r, \theta)$ by

$$\begin{aligned}
 q &= \begin{Bmatrix} r \cos \theta \\ r \sin \theta \end{Bmatrix} = r \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} \\
 &= \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} \\
 &= m_0^i \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + m_1^i (\theta - \theta_i) \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + m_2^i (\theta - \theta_i)^2 \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + m_3^i (\theta - \theta_i)^3 \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix}
 \end{aligned} \tag{3}$$

Once can calculate the velocity using $v = \dot{q} = \dot{r} \frac{\partial q}{\partial r} + \dot{\theta} \frac{\partial q}{\partial \theta}$, or equivalent

$$\begin{aligned}
v &= \dot{q} \\
&= \dot{r} \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \dot{\theta} \begin{Bmatrix} -r \sin \theta \\ r \cos \theta \end{Bmatrix} \\
&= \dot{r} \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \dot{\theta} r \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \\
&= \dot{\theta} \sum_{k=1}^3 m_k^i k (\theta - \theta_i)^{k-1} \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \dot{\theta} \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \\
&= m_0^i \dot{\theta} \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} + m_1^i \dot{\theta} \left[\begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + (\theta - \theta_k) \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \right] \\
&\quad + m_2^i (\theta - \theta_i) \dot{\theta} \left[\begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + (\theta - \theta_i) \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \right] + m_3^i (\theta - \theta_i)^3 \dot{\theta} \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix}
\end{aligned} \tag{4}$$

where the derivative of $r(\theta)$ was calculated using $\dot{r}(\theta) = \dot{\theta} \sum_{k=1}^3 m_k^i k (\theta - \theta_i)^{k-1}$.

Since $\frac{\partial^2 q}{\partial r^2} = 0$, $\frac{\partial^2 q}{\partial r \partial \theta} = \frac{\partial^2 q}{\partial \theta \partial r} = \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} = \frac{1}{r} \frac{\partial q}{\partial \theta}$ and $\frac{\partial^2 q}{\partial \theta^2} = -r \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} = -r \frac{\partial q}{\partial r}$ the acceleration can be calculated as

$$\begin{aligned}
a &= \ddot{q} \\
&= \ddot{r} \frac{\partial q}{\partial r} + \ddot{\theta} \frac{\partial q}{\partial \theta} + \dot{r} \frac{d}{dt} \frac{\partial q}{\partial r} + \dot{\theta} \frac{d}{dt} \frac{\partial q}{\partial \theta} \\
&= \ddot{r} \frac{\partial q}{\partial r} + \ddot{\theta} \frac{\partial q}{\partial \theta} + \dot{r} \left(\frac{d}{dt} \frac{\partial}{\partial r} + \frac{d\theta}{dt} \frac{\partial}{\partial \theta} \right) \frac{\partial q}{\partial r} + \dot{\theta} \left(\frac{d}{dt} \frac{\partial}{\partial r} + \frac{d\theta}{dt} \frac{\partial}{\partial \theta} \right) \frac{\partial q}{\partial \theta} \\
&= \ddot{r} \frac{\partial q}{\partial r} + \ddot{\theta} \frac{\partial q}{\partial \theta} + 2\dot{r}\dot{\theta} \frac{1}{r} \frac{\partial q}{\partial \theta} + \dot{\theta}^2 r \frac{\partial q}{\partial r} \\
&= (\ddot{r} - \dot{\theta}^2 r) \frac{\partial q}{\partial r} + \left(\ddot{\theta} + 2\dot{r}\dot{\theta} \frac{1}{r} \right) \frac{\partial q}{\partial \theta} \\
&= (\ddot{r} - \dot{\theta}^2 r) \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \left(\ddot{\theta} + 2\dot{r}\dot{\theta} \frac{1}{r} \right) \begin{Bmatrix} -r \sin \theta \\ r \cos \theta \end{Bmatrix}
\end{aligned} \tag{5}$$

One can calculate now

$$\begin{aligned}
a &= \ddot{q} \\
&= \left(\ddot{r} - \dot{\theta}^2 r \right) \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \left(\ddot{\theta} + 2\dot{r}\dot{\theta} \frac{1}{r} \right) r \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \\
&= \left(\ddot{r} - \dot{\theta}^2 \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \right) \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} + \left(\ddot{\theta} + \frac{2\dot{r}\dot{\theta}}{\sum_{k=0}^3 m_k^i (\theta - \theta_i)^k} \right) \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \\
&= \left(\ddot{\theta} \sum_{k=1}^3 m_k^i k (\theta - \theta_i)^{k-1} + \dot{\theta}^2 \sum_{k=2}^3 m_k^i (k-1)k (\theta - \theta_i)^{k-2} - \dot{\theta}^2 \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \right) \begin{Bmatrix} \cos \theta \\ \sin \theta \end{Bmatrix} \\
&\quad + \left(\ddot{\theta} + 2\dot{\theta}^2 \frac{\sum_{k=1}^3 m_k^i k (\theta - \theta_i)^{k-1}}{\sum_{k=0}^3 m_k^i (\theta - \theta_i)^k} \right) \sum_{k=0}^3 m_k^i (\theta - \theta_i)^k \begin{Bmatrix} -\sin \theta \\ \cos \theta \end{Bmatrix} \tag{6}
\end{aligned}$$

where the second derivative of $r(\theta)$ was calculated as

$$\ddot{r}(\theta) = \ddot{\theta} \sum_{k=1}^3 m_k^i k (\theta - \theta_i)^{k-1} + \dot{\theta}^2 \sum_{k=2}^3 m_k^i (k-1)k (\theta - \theta_i)^{k-2}$$

2.3 Equations of Motion

The Lagrange formulation [6] to derive equations of motion can be expressed as

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{s}_i} \right) + \frac{\partial T}{\partial s_i} = R_i, \quad i = 1, 2$$

where R_i are the generalized forces, $s_1 = \theta, s_2 = r$ are the time dependent generalized coordinates and the subscript $i = 1, 2$ represents the number of the generalized forces/coordinates. The total kinetic energy of the system can be expressed as

$$T = \sum_{i=1}^N T_i = \frac{1}{2} \sum_{i=1}^N (m_i v_{C_i}^2 + I_{C_i} \omega_i^2)$$

where T_i is the kinetic energy, I_{C_i} is the mass moment of inertia, and ω_i is the angular velocity of each rod. The displacement of the end effector along the motion path - when the sliding rod SP is flexible (Fig. 3) - can be calculated as in [7]. Then, the equations of motion (based on the mode shapes of a clamped-free beam) can be expressed by

$$[M] \begin{Bmatrix} \ddot{s} \\ \ddot{u} \\ \ddot{\theta} \end{Bmatrix} + [H] \begin{Bmatrix} \dot{s} \\ \dot{u} \\ \dot{\theta} \end{Bmatrix} + [G] \begin{Bmatrix} s \\ u \\ \theta \end{Bmatrix} = \begin{Bmatrix} 0 \\ F \\ T \end{Bmatrix}$$

where F is the axial force that slide the flexible rod in the rigid guide OE , T is the torque that rotate the guide, and u is the un-deformed part of the sliding rod (Fig. 3).

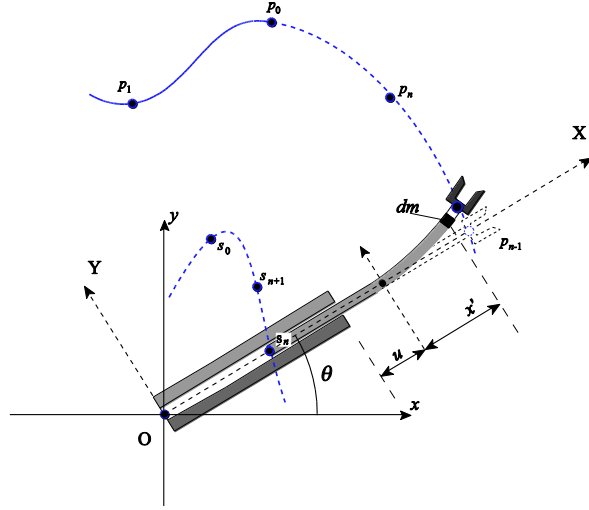


Figure 3: Trajectory of the end effector for a flexible sliding rod

The matrices $[M]$, $[H]$, and $[G]$ are defined as in [7] by

$$\begin{aligned} [M] &= f(l_0, \rho, A, M_e, u, [I], [C_i], [C_i]), \\ [H] &= f(l_0, \rho, A, M_e, u, \dot{u}, [I], [C_i], [C_i]), \\ [G] &= f(l_0, \rho, A, E, M_e, u, \dot{u}, \ddot{u}, [I], [C_i], [C_i]), \end{aligned}$$

where $[C_i]$ denote a square matrix [7].

3 Numerical Results

Two numerical examples to generate the piecewise polynomial trajectory of the end-effector as well as the trajectory of the non-active link end which constrains the motion are considered.

The possible locations of the base as well as the phase diagram (position vs. velocity) of the active and non-active end of the robotic manipulator are also presented. Simulations have been performed [10] for a rigid guide of length $d_{OE} = 0.35$ m and a sliding part of length $d_{SP} = 30$ cm. The end of the guide - located on the origin O of the Cartesian reference frame Oxy - is rotating with an angular velocity $\omega = 1$ rad/s. The material property for the robotic arm has the density 7850 kg/m^3 , Young's modulus $2 \cdot 10^{11} \text{ Pa}$ and Poisson's ratio 0.3.

Using the points $p_i(x_i, y_i)_{i=1,2,\dots,10}$ shown in Fig. 4a (convex configuration) a new manipulator trajectory in which the end effector can move smoothly [1,4] along the path as well as the possible locations of the manipulator base [5] have been obtained.

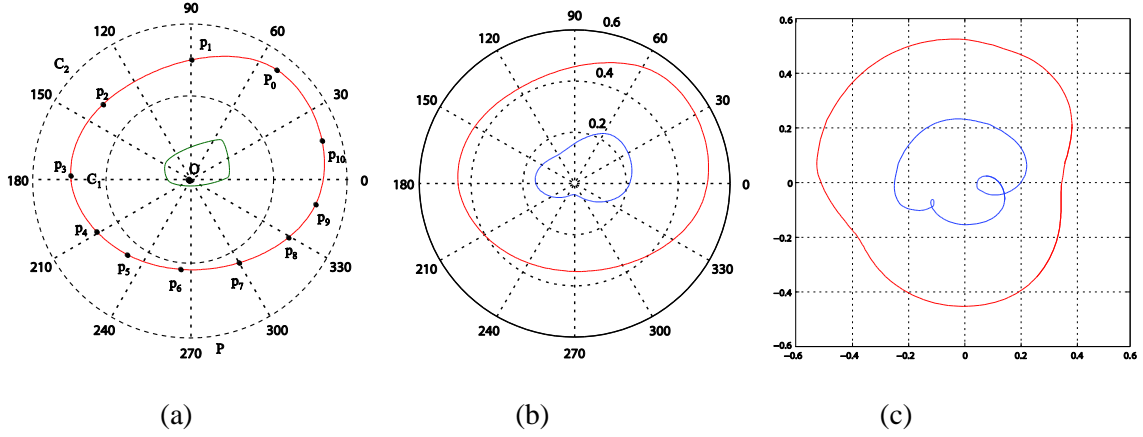


Figure 4: (a) end effector trajectory (red) generated using Hermite interpolation and base location (green), (b) end effector trajectory (red) and non-active link end trajectory (blue), and (d) phase diagram (position vs. velocity) of the end effector (red) and non-active end (blue) of the extensible link

The end effector trajectory of the active end and the resulting trajectory of the non-active end of the manipulator are plotted in Fig. 4b. The phase diagram (position vs. velocity) of the end effector and the non-active end (Ox trajectories) of the link is shown in Fig. 4c.

The same simulations have been performed for the control points $p_i(x_i, y_i)_{i=1,2,\dots,8}$ shown in Fig. 5 for a non-convex configuration. The base location [5], end effector and constrained trajectory, and their associated velocity profile (Ox velocities) are shown in Fig. 5a, Fig. 5b, and Fig. 5c respectively. Both simulations in Fig. 4c and Fig. 5c have shown a continuous velocity profile, that is, the proof of a smooth trajectory.

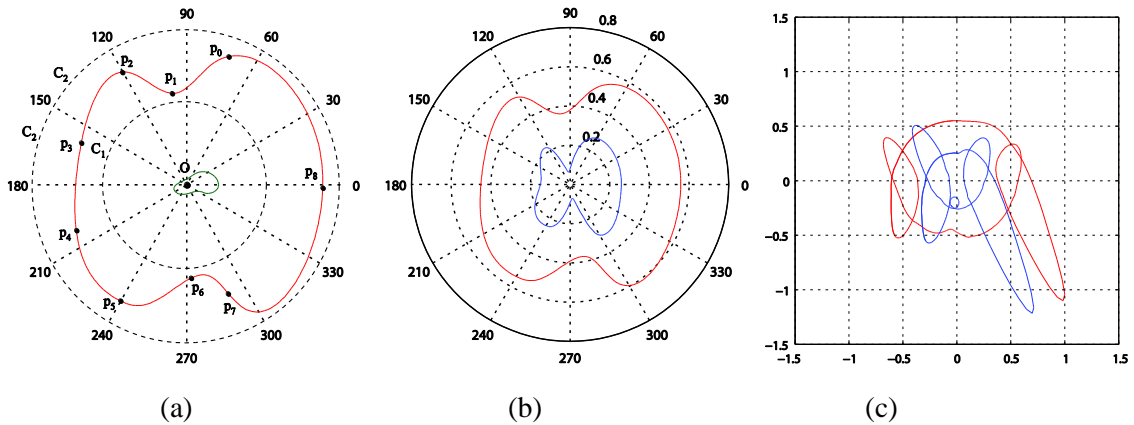


Figure 5: (a) end effector trajectory (red) generated using Hermite interpolation and base location (green), (b) end effector trajectory (red) and non-active link end trajectory (blue), and (d) phase diagram (position vs. velocity) of the end effector (red) and non-active end (blue) of the extensible link

4 Conclusions

In this paper the generation of polygonal trajectory of the end-effector of a rotating extensible robotic arm using Cartesian and polar piecewise cubic interpolants curves is presented. The use of polar piecewise cubic interpolants allows easy calculation of kinematics variables and smooth trajectories. Moreover, the manipulator equations of motion can be easily handled, and the constrained trajectory of the non-active end of the manipulator derived directly from the position of the end-effector. To verify the proposed approach numerical simulations have been performed for both the convex and non-convex configurations.

References

- [1] B.B. Barsky, T.D. DeRose, “Three Characterizations of Geometric Continuity for Parametric Curves”, University of California at Berkeley Technical Report No. UCB/CSD 88/417, 1988
- [2] J.E. Bobrow, S. Dubowsky, J.S. Gibson, “Time-optimal control of robotic manipulators along specified paths”, *Int. J. Robot. Res.*, 4(3), 554–561, 1985
- [3] D. Constantinescu, E.A. Croft, “Smooth and time-optimal trajectory planning for industrial manipulators along specified paths”, *J. Robot. Syst.*, 17(5), 233–249, 2000
- [4] T.D. Derose, B. A. Barsky, “Geometric Continuity, Shape Parameters, and Geometric Constructions for Catmull-Rom Splines”, *ACM Transactions on Graphics*, 7(1), 1-41, 1988.
- [5] M. Dupac, “Trajectory Planning for a Rotating Extensible Robotic Manipulator: Design for Collision Avoidance and Accurate Position Placement”, "Proceedings of the Ninth International Conference on Engineering Computational Technology", Civil-Comp Press, Stirlingshire, UK, Paper 56, 2014.
- [6] M. Dupac, Dynamical analysis of a constrained flexible extensible link with rigid support and clearance, *Journal of Theoretical and Applied Mechanics* 52, 3, 665-676, 2014.
- [7] M. Kalyoncu, “Mathematical modelling and dynamic response of a multistraight-line path tracing flexible robot manipulator with rotating-prismatic joint”, *Applied Mathematical Modelling* 32, 1087–1098, 2008
- [8] S.-R. Chang, U.-Y. Huh, “A collision-free G^2 continuous path-smoothing algorithm using quadratic polynomial interpolation”, *International Journal of Advanced Robotic Systems*, 11:194, 2014
- [9] M. Krajnc, “Interpolation Scheme for Planar Cubic G^2 Spline Curves”, *Acta Applicandae Mathematicae*, 113(2), 129-143, 2011
- [10] D.B. Marghitu, M. Dupac, *Advanced Dynamics: Analytical and Numerical Calculations with MATLAB*, Springer, 2012
- [11] A. Nektarios, N.A. Aspragathos, “Optimal location of a general position and orientation end-effector’s path relative to manipulator’s base, considering velocity performance”, *Robotics and Computer-Integrated Manufacturing* 26 (2010) 162–173
- [12] A. Piazzzi, A. Visioli, “Global minimum-jerk trajectory planning of robot manipulators”, *IEEE Trans. Ind. Electron.* 47(1), 140–149, 2000
- [13] C. Zhengsheng, K. Minxiu, L. Ming, Y. Wei, “Dynamic modelling and trajectory

- tracking of parallel manipulator with flexible link”, International Journal of Advanced Robotic Systems, 10, 328:2013
- [14] Lavery, J.E., 2000. Univariate cubic L_p splines and shape-preserving, multiscale interpolation by univariate cubic L_1 splines. Computer Aided Geometric Design 17, 319–336.
 - [15] Lavery, J.E., 2001. Shape-preserving, multiscale interpolation by bi- and multivariate cubic L_1 splines. Computer Aided Geometric Design. To appear.
 - [16] Golomb, M., 1968. Approximation by periodic spline interpolants on uniform meshes. J. Approx. Theory 1, 26–65.
 - [17] Goodman, T.N.T., Lee, S.L., 1984. B-splines on the circle and trigonometric B-splines, in: Singh, S.P, Burry, J.H.W., Watson, B. (Eds.), Approximation Theory and Spline Functions. Reidel, Dordrecht, pp. 297–325.
 - [18] Alfeld, P., Neamtu, M., Schumaker, L.L., 1995. Circular Bernstein–Bézier polynomials, in: Dæhlen, M., Lyche, T., Schumaker, L.L. (Eds.), Mathematical Methods for Curves and Surfaces. Vanderbilt University Press, Nashville, TN, pp. 11–20.
 - [19] Sanchez-Reyes, J., 1992. Single-valued spline curves in polar coordinates. Computer Aided Design 24, 307–315.
 - [20] Casciola, G., Morigi, S., 1997. Spline curves in polar and Cartesian coordinates, in: Le Méhauté, A., Rabut, C., Schumaker, L.L. (Eds.), Curves and Surfaces with Applications in CAGD. Vanderbilt University Press, Nashville, TN, pp. 61–68.
 - [21] de Casteljaou, P., 1994. Splines focales, in: Laurent, P.-J., Le Méhauté, A., Schumaker, L.L. (Eds.), Curves and Surfaces in Geometric Design. A. K. Peters, Wellesley, MA, pp. 91–103.
 - [22] Yukinori Iwashita, Piecewise Polynomial interpolation, OpenGamma Quantitative Research, pp. 1-23, 2013 [<http://www.opengamma.com/research>]
 - [23] Lyche, T., Winther, R., 1979. A stable recurrence relation for trigonometric B-splines. J. Approx. Theory 25, 266–279.