In this paper, a combined polar-Cartesian approach to generate a smooth trajectory of a robotic arm along priori defined via-points is presented. Due to the characteristics/geometry of the robotic arm, cylindrical coordinates are associated with the trajectory of motion. Possible trajectories representing the system dynamics are generated by mix-matching higher order polar piecewise polynomials used to devise the radial trajectory and Cartesian piecewise polynomials used to calculate the related height in a normal plane unfolded along the radial trajectory of the motion. To describe the kinematic properties of the end effector, a moving noninertial orthonormal Frenet frame is considered. Using the Frenet frame, the components of the velocity and acceleration along the frame unit vectors are calculated. Numerical simulations are performed for two different configurations in order to validate the approach.

**KEYWORDS**
- dynamics
- kinematics
- path planning
- piecewise interpolation

## 1 | INTRODUCTION

Trajectory planning of robotic manipulators is considered a fundamental factor in industry and automation with important consequences in improving production life cycle and minimizing costs.\textsuperscript{1} The capacity to plan smooth trajectories involves taking into consideration kinematic constraints,\textsuperscript{2-4} execution time,\textsuperscript{5} and jerk.\textsuperscript{6}

The last decade has seen important research into the assessment of novel lightweight robotic devices specifically designed for rehabilitation.\textsuperscript{7} This includes a novel system\textsuperscript{8} to measure and analyze the kinetic data as a way to develop and improve robotic rehabilitation systems. An adaptive trajectory generation approach for a bilateral upper arm rehabilitation training has been considered in the work of Miao et al,\textsuperscript{9} and a novel filtered kinematic matrix adaptive control was examined in the work of Brahim et al.\textsuperscript{10} A robotic platform for upper arm neurorehabilitation was considered in the work of Fraile et al.\textsuperscript{11} In the work of Liu et al,\textsuperscript{12} the movements of a human arm are considered and analyzed in order to describe the kinematic of an upper arm exoskeleton rehabilitation robot with two actuators. The kinematics and dynamics of a pantograph-based rehabilitation robot is considered in the work of Mancisidor et al\textsuperscript{13} as a way to create a robust control that allow stroke patients\textsuperscript{11} to complete rehabilitation exercises of their upper arm, elbow, or shoulder.

Piecewise interpolating functions with high continuity and/or geometrically continuous splines\textsuperscript{14-16} are adequate tools in generating smooth motion of the robotic manipulators when the manipulator kinematics (velocity, acceleration,\textsuperscript{17} and/or jerk) or dynamics (force and/or torque) is considered.\textsuperscript{18} Such approach\textsuperscript{14} should reduce resonant frequency excitation and generate smoother trajectory profile. The interpolation of smooth curves (twice differentiable and cubic in the parametrized coordinates) invariant with respect to the fixed/moving frame represents an excellent approach to minimize
angular acceleration. A new planning approach of a manipulator along a set of nodal points for a collection of established kinematical requirements is presented in the work of du Plessis and Snyman. Kinematic variables and joint-space trajectories can be easily calculated/planned through a sequence of specified joints for smooth and continuous motion while preserving the C^k continuity.

In this study, the modeling and simulation of 3D smooth trajectories of a related robotic arm using piecewise interpolants is addressed. Path planning of the robotic arm is devised using a given number of via-points the end effector should reach. Due to the geometry constraints, i.e., a trajectory does not exist outside the working envelope, the relation between the geometry of the robotic arm and its base location is examined. Possible trajectories are generated using Hermite polar piecewise interpolants for the projected radial trajectory on the Oxy-plane combined with a linear approximation of the trajectory height. Two sets of numerical results to highlight the correlation between the geometry and the working envelope are presented.

2 | MANIPULATOR MODEL AND TRAJECTORY GENERATION

The robotic arm is represented by a z-guide (link 0) denoted by zG, a rigid sliding guide (link 1) denoted by RG, and a sliding link (link 2) denoted by SL, as shown in Figure 1A. The z-guide (link 0) of the robotic arm represented in a fixed Cartesian reference frame Oxyz can rotate about the Oz-axis. The rigid guide (link 1) of the robotic arm can slide up and down on the z-guide (link 0) to reach a desired height while rotating (with the z-guide) about the Oz-axis. Link 2 and the rigid guide are joined by the means of a slider joint, that is, the link can slide in and out of the rigid guide, as shown in Figure 1A. The length of the z-guide is l_z, the length of the rigid guide is l_RG, and the length of the sliding link is l_SL.

The interpolated trajectory T_P between the via-points P_i, i = 1, n (Figure 1A) is represented using

\[ r_{P_k} = r_{P_k} \cos \theta_{P_k} i_0 + r_{P_k} \sin \theta_{P_k} j_0 + z_{P_k} k_0. \]  

(1)

where T_P represents the 3D piecewise trajectory followed by the end effector, P_k represents the interpolating points along the piecewise curve defined by the via-points P_i, and P_{i+1}, r_{P_k} = d(O_1, P_k) is the radial distance/radius from point O_1 (of the mobile reference frame O_1x_1y_1z_1 attached to the rigid guide (link 1)) to point P_k, z_{P_k} is the associated height (distance from P_k to O_0x_0y_0), and \( \theta_{P_k} \) is the azimutal coordinate given in an anticlockwise direction. When the end effector describes the 3D piecewise trajectory T_P given by the via-points P_i, i = 1, n, its projection on the O_0x_0y_0-plane is the planar polar trajectory described by \( Q_r(\theta_{P_k} \cos \theta_{P_k}, r_{P_k} \sin \theta_{P_k}) \).

To interpolate between the via-points P_i, i = 0, N, specified by the data \( \{r_i, \theta_i, z_i\}_{i=0, N} \) (Figure 1A), the combination between a piecewise polar interpolation (that approximates the projected radial interpolation shown in Figure 1C) and a Cartesian interpolation (Figure 1B) was considered. For each interval \( [\theta_i, \theta_{i+1}] \subset \mathbb{R} \subset \mathbb{R} \) and lengths \( r_i \) and \( r_{i+1} \) of the consecutive points Q_i and Q_{i+1}, a polar piecewise interpolation (Figure 1C) can be devised as a Hermite-type polynomial:

\[ r(\theta) = \sum_{k=0}^{q} c_k(\theta - \theta_i) \]

(2)

where \( c_0 = r_i, c_1 = \dot{r}_i, c_2 = \frac{1}{h_i}[(2\dot{r}_i + \ddot{r}_{i+1}) + 3\Delta r_i], c_3 = \frac{1}{h_i^2} [\ddot{r}_i + \dddot{r}_{i+1} - 2\Delta \ddot{r}_i], r_i = r(\theta_i), r_{i+1} = r(\theta_{i+1}), \Delta h_i = \theta_{i+1} - \theta_i, \Delta \dot{r}_i = \frac{r_{i+1} - r_i}{h_i}, \ddot{r}_i, \dddot{r}_i, and \dot{r}_{i+1} = \frac{dr(\theta_{i+1})}{d\theta} = \dot{r}_{i+1}. \)

Trajectory height, which relates the change in height with the piecewise polar trajectory of motion (Figure 1B), is computed in the unfolded normal plane (Figure 1A) tracking the radial trajectory (Figure 1C). The computation is performed by Cartesian piecewise interpolation with Hermite polynomials defined by

\[ z(x) = \sum_{k=0}^{q} c_k(x - x_i)^k. \]  

(3)

where \( q \) is the order of the polynomial, \( h_i = x_{i+1} - x_i, z_i = z(x_i), z_{i+1} = z(x_{i+1}), \Delta z_i = \frac{z_{i+1} - z_i}{h_i} C_0 = z_i, C_1 = \dot{z}_i, C_2 = \frac{1}{h_i} [(2\ddot{z}_i + \dddot{z}_{i+1}) + 3\Delta z_i], C_3 = \frac{1}{h_i^2} [\dddot{z}_i + \dddot{z}_{i+1} - 2\Delta \dddot{z}_i], \) and where the derivatives at the endpoints P_i and P_{i+1} are calculated as \( \ddot{z}(x_i) = \frac{dz(x_i)}{dx} = \ddot{z}_i \) and \( \dddot{z}(x_{i+1}) = \frac{d^2z(x_{i+1})}{dx^2} = \dddot{z}_{i+1}, \) respectively.
The variable $x$ in Equation (3) is the length of the curve shown in Figure 1C and represents the polar trajectory of motion. The length of the trajectory shown in Figure 1C through the points $Q_i$ and $Q_{i+1}$ is calculated using

$$l(Q_i, Q_{i+1}) = \int_{\theta_i}^{\theta_{i+1}} \sqrt{r^2 + \left(\frac{dr}{d\theta}\right)^2} \, d\theta,$$

(4)

where $r = r(\theta)$ is given by Equation (2). The distance $d(O, P_{ik})$ is calculated using

$$d(O, P_{ik}) = \sqrt{r_{ik}^2 + z_{ik}^2} \, dx,$$

(5)

where $z_{ik}$ is the height of each trajectory point, $r_{ik}$ is given by Equation (1), and the maximal and minimal distance from the robotic arm base to the end-effector trajectory $T_{P_i}$ is calculated as in the work of Dupac and Sewell using

$$d_{\text{min}} = \inf_{P_k \in T_{P_i}} d(O, P_k), \quad d_{\text{max}} = \sup_{P_k \in T_{P_i}} d(O, P_k).$$

(6)
Since the geometric path $T_p$ of the robotic arm should be reachable by the end effector, i.e., the trajectory does not exist outside the envelope of the robotic arm, the length/geometry of the extensible arm conveys the existence of a solution. That is, a trajectory exists if and only if the system

$$\begin{align*}
    r_{\text{max}} & \leq l_{\text{LG}} + l_{\text{SL}} \\
    \max(l_{\text{LG}}, l_{\text{SL}}) & \leq r_{\text{min}} \\
    l_{z} & \leq \sup_{i=1,n-1,k=1,N} z_i
\end{align*}$$

(7)

has a solution (for more details, see, other related works$^{25-28}$) where

$$r_{\text{min}} = \inf_{i_x \in [i, i+1]} d(O_2, P_i), \quad r_{\text{max}} = \sup_{i_x \in [i, i+1]} d(O_2, P_i)$$

(8)

for any $i = 1, n - 1$.

## 3 | END-EFFECTOR PATH PLANNING

The path can be parameterized$^{23,25,26}$ using the cylindrical coordinate $r, \theta$ and $z$ by $r = r(\theta) \cos \theta i_0 + r(\theta) \sin \theta j_0 + z k_0$, that is,

$$r = \begin{bmatrix} r \cos \theta \\
    r \sin \theta \\
    z \end{bmatrix} = \begin{bmatrix}
    \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k \cos \theta \\
    \sum_{k=0}^{3} c_k^j (\theta - \theta_i)^k \sin \theta \\
    \sum_{k=0}^{3} c_k^z (x - x_i)^k
\end{bmatrix}.$$  

(9)

The unit vectors $i_0, j_0, k_0$ can be expressed in a Frenet frame by $i_0 = \cos \theta e_r - \sin \theta e_\theta, j_0 = \sin \theta e_r + \cos \theta e_\theta$, $k_0 = e_z$. The velocity vector $v = \dot{r} = i e_r + \dot{\theta} e_\theta + \dot{z} e_z$ is written with

$$v = \begin{bmatrix}
    \dot{r} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^{k-1} \cos \theta - \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k \sin \theta \\
    \dot{r} \sum_{k=0}^{3} c_k^j (\theta - \theta_i)^{k-1} \cos \theta + \dot{\theta} \sum_{k=0}^{3} c_k^j (\theta - \theta_i)^k \cos \theta \\
    \dot{r} \sum_{k=0}^{3} c_k^z (x - x_i)^{k-1}
\end{bmatrix},$$

(10)

where the derivative of $r = r(\theta)$ was calculated using $\dot{r}(\theta) = \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^{k-1}$, and the derivative of $z = z(x)$ was calculated using $\dot{z}(x) = \dot{x} \sum_{k=0}^{3} c_k^z (x - x_i)^{k-1}$. It results

$$v = \begin{bmatrix}
    \left( \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^{k-1} \cos \theta - \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k \sin \theta \right)^2 \\
    + \left( \dot{\theta} \sum_{k=0}^{3} c_k^j (\theta - \theta_i)^{k-1} \cos \theta + \dot{\theta} \sum_{k=0}^{3} c_k^j (\theta - \theta_i)^k \cos \theta \right)^2 \\
    + \left( \dot{x} \sum_{k=0}^{3} c_k^z (x - x_i)^{k-1} \right)^2
\end{bmatrix}^{1/2}.$$  

(11)
The acceleration vector can be written as \( \mathbf{a} = \ddot{\mathbf{r}} = \ddot{r} \hat{r} + \ddot{\theta} \hat{\theta} \times \hat{r} + \dddot{\mathbf{z}} \), by

\[
\mathbf{a} = \left[ \begin{array}{c}
\left( \ddot{r} - \dot{\theta}^2 \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k \right) \cos \theta - \left( \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k + 2 \dot{r} \dot{\theta} \right) \sin \theta \\
\left( \ddot{r} - \dot{\theta}^2 \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k \right) \sin \theta - \left( \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k + 2 \dot{r} \dot{\theta} \right) \cos \theta \\
\dddot{z} \sum_{k=1}^{3} C_k^i k(x - x_i)^{k-1} + \dot{x}^2 \sum_{k=2}^{3} C_k^i (k - 1)k(x - x_i)^{k-2}
\end{array} \right]
\]

\[
= \left[ \begin{array}{c}
\left( -\dot{\theta}^2 \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k + \dot{\theta} \sum_{k=1}^{3} c_k^i (\theta - \theta_i)^{k-1} + \ddot{\theta} \sum_{k=2}^{3} c_k^i (k - 1)(\theta - \theta_i)^{k-2} \right) \cos \theta \\
\left( -\dot{\theta}^2 \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k + \dot{\theta} \sum_{k=1}^{3} c_k^i (\theta - \theta_i)^{k-1} + \ddot{\theta} \sum_{k=2}^{3} c_k^i (k - 1)(\theta - \theta_i)^{k-2} \right) \sin \theta \\
\dot{x} \sum_{k=1}^{3} C_k^i k(x - x_i)^{k-1}
\end{array} \right]
\]

\[
+ \left[ \begin{array}{c}
\left( -\dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k - 2 \dot{\theta} \sum_{k=1}^{3} c_k^i (\theta - \theta_i)^{k-1} \right) \sin \theta \\
\left( \dot{\theta} \sum_{k=0}^{3} c_k^i (\theta - \theta_i)^k + 2 \ddot{\theta} \sum_{k=1}^{3} c_k^i (\theta - \theta_i)^{k-1} \right) \cos \theta \\
\ddot{x} \sum_{k=2}^{3} C_k^i (k - 1)k(x - x_i)^{k-2}
\end{array} \right],
\]

where \( \ddot{r}(\theta) = \dot{\theta} \sum_{k=0}^{3} c_k^i k(\theta - \theta_i)^{k-1} + \ddot{\theta} \sum_{k=2}^{3} c_k^i (k - 1)(\theta - \theta_i)^{k-2} \), and the second derivative of \( z \) was calculated using

\[
\dddot{z} = \dot{x} \sum_{k=1}^{3} C_k^i k(x - x_i)^{k-1} + \dot{x}^2 \sum_{k=2}^{3} C_k^i (k - 1)k(x - x_i)^{k-2},
\]

4 | KINEMATICS

The generalized coordinates \( \theta, z \), and \( r \) (Figure 2) relate to the center of the mass of link 0 (z-guide zG), link 1 (the rigid sliding guide RG), and respectively link 2 (sliding hand support SL). The frames \( O_0 x_0 y_0 z_0 \) and \( O_1 x_1 y_1 z_1 \) are defined by \( i_0 = i, j_0 = j, k_0 = k \), and respectively by \( i_1, j_1, k_1 \) (Figure 3). The Euler angles \( \alpha \) relate \( i_1, j_1, k_1 \) and \( i_0, j_0, k_0 \) by

\[
\begin{bmatrix}
\begin{array}{c}
i_1 \\
j_1 \\
k_1
\end{array}
\end{bmatrix} = \begin{bmatrix}
\begin{array}{ccc}
cos \theta & -\sin \theta & 0 \\
\sin \theta & \cos \theta & 0 \\
0 & 0 & 1
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
i_0 \\
j_0 \\
k_0
\end{array}
\end{bmatrix}.
\]

No other reference frames are needed since the mobile reference frame defined by \( i_1, j_1, k_1 \) can be properly used to express all the link’s position, velocity (respectively angular velocity), and acceleration (respectively, angular acceleration). The angular velocity/acceleration of links 1, 2, and 3 can be expressed by

\[
\omega_1 = \omega_2 = \omega_3 = \dot{\theta} k_1 = \ddot{\theta} k_0 \\
\alpha_1 = \alpha_2 = \alpha_3 = \ddot{\theta} k_1 = \dddot{\theta} k_0.
\]
The position of the mass center of link 1, link 2, and link 3 can be calculated with

\[
\mathbf{r}_{c_1} = \frac{l_{CG}}{2} \mathbf{k}_1 = \frac{l_{CG}}{2} \mathbf{k}_0
\]

\[
\mathbf{r}_{c_2} = z_c \mathbf{k}_1 + \frac{l_{RG}}{2} \mathbf{j}_1
\]

\[
= \frac{l_{RG}}{2} \sin \theta_i + \frac{l_{RG}}{2} \cos \theta_j + z_c \mathbf{k}_0
\]

\[
\mathbf{r}_{c_3} = z_c \mathbf{k}_1 + r_c \mathbf{j}_1
\]

\[
= r_c \sin \theta_i + r_c \cos \theta_j + z_c \mathbf{k}_0.
\]
The velocity of the mass center of link 1, link 2, and link 3 can be calculated with

\[ \mathbf{v}_{c_i} = \frac{d}{dt} \mathbf{r}_{c_i} = \dot{\mathbf{r}}_{c_i} = \mathbf{0} \]

\[ \mathbf{v}_{c_2} = \frac{d}{dt} \mathbf{r}_{c_2} = \frac{d}{dt} \left( \frac{l_{RG}}{2} \sin \theta \mathbf{i}_0 + \frac{l_{RG}}{2} \cos \theta \mathbf{j}_0 + z_{C_2} \mathbf{k}_0 \right) \]

\[ = \frac{1}{2} l_{RG} \dot{\theta} \cos \theta \mathbf{i}_0 - \frac{1}{2} l_{RG} \dot{\theta} \sin \theta \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0 \]

\[ \mathbf{v}_{c_3} = \frac{d}{dt} \mathbf{r}_{c_3} = \frac{d}{dt} \left( r_{c_3} \sin \theta \mathbf{i}_0 + r_{c_3} \cos \theta \mathbf{j}_0 + z_{C_2} \mathbf{k}_0 \right) \]

\[ = \cos \theta \left( \dot{r}_{C_3} \tan \theta + r_{c_3} \dot{\theta} \right) \mathbf{i}_0 + \cos \theta \left( \dot{r}_{C_3} - r_{c_3} \dot{\theta} \tan \theta \right) \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0. \] (16)

The acceleration the mass center of link 1, link 2, and link 3 can be calculated with

\[ \mathbf{a}_{c_1} = \frac{d}{dt} \mathbf{v}_{c_1} = \dot{\mathbf{r}}_{c_1} = \mathbf{0} \]

\[ \mathbf{a}_{c_2} = \frac{d}{dt} \mathbf{v}_{c_2} = \frac{d}{dt} \left( \frac{1}{2} l_{RG} \dot{\theta} \cos \theta \mathbf{i}_0 - \frac{1}{2} l_{RG} \dot{\theta} \sin \theta \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0 \right) \]

\[ = \frac{1}{2} l_{RG} \cos \theta (\dot{\theta} - \dot{\theta}^2 \tan \theta) \mathbf{i}_0 - \frac{1}{2} l_{RG} \cos \theta (\dot{\theta} \tan \theta + \dot{\theta}^2) \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0 \]

\[ \mathbf{a}_{c_3} = \frac{d}{dt} \mathbf{v}_{c_3} \]

\[ = \frac{d}{dt} \left( \left( \dot{r}_{C_3} \sin \theta + r_{c_3} \dot{\theta} \cos \theta \right) \mathbf{i}_0 + \left( \dot{r}_{C_3} \cos \theta - r_{c_3} \dot{\theta} \sin \theta \right) \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0 \right) \]

\[ = \cos \theta \left( \dot{r}_{C_3} \tan \theta + 2 \dot{r}_{C_3} \dot{\theta} - r_{c_3} \dot{\theta}^2 \tan \theta \right) \mathbf{i}_0 \]

\[ + \cos \theta \left( \dot{r}_{C_3} - 2 \dot{r}_{C_3} \dot{\theta} \tan \theta - r_{c_3} \dot{\theta}^2 \right) \mathbf{j}_0 + \dot{z}_{C_2} \mathbf{k}_0. \] (17)

### 5 RESULTS

Two numerical examples\textsuperscript{29,30} are presented to illustrate trajectory generation of a z-guide of maximal height \( l_z = 1.2 \) m, rigid guide with \( l_G = 0.55 \) m and a sliding link with \( l_{SL} = 0.55 \) m. The trajectory generation in which the end effector moves smoothly\textsuperscript{28} mix-match polar and Cartesian piecewise polynomials. The numerical values of the via-points coordinates are presented in Table 1.

Path planning for the configuration in Table 1 obtained by mix-matching polar and Cartesian piecewise interpolating curves is shown in Figure 3. Figures 3A, 3B, and 3C represent the projection of end-effector trajectory on \( O_0x_0y_0 \)- (radial trajectory of the end effector), \( O_0x_0z_0 \)-, and \( O_0y_0z_0 \)-planes, respectively. For this simulation, although a working trajectory is obtained, it can be seen that the robotic end effector is in the proximity but still outside the working envelope, that is, it cannot handle all the desired via-points shown in Table 1 when following the computed path (Figure 3A).

The velocity projection on the \( O_0x_0y_0 \)-, \( O_0x_0z_0 \)-, and \( O_0y_0z_0 \)-planes denoted by \( v_{xy} \), \( v_{xz} \), and \( v_{yz} \) is shown in Figure 4A, Figure 4B, and respectively in Figure 4C.

| Parameter | Parameter value | Parameter value |
|-----------|----------------|----------------|
| \( l_i \) | 20 50 130 150 | 210 270 310 350 360+20 |
| \( \theta_i \) | 12 9 2 10 | 5 7 10 12 |
| \( r_i \) | 7 12 3 14 | 5 9 7 10 7 |

**TABLE 1** Via-points for the first configuration of the robotic arm
To better understand the system behavior, the trajectory (position and velocity) of the robotic arm is shown in the 3D space in Figure 5A and Figure 5B. The smoothness of the continuous position and velocity curves shown in Figure 5 proves the effectiveness of the trajectory planning of the end effector of the 3D mechanism. As a result, the forces needed to guide the arm of the robot along the prescribed path are also continuous. A second numerical example with the associated data shown in Table 2 is then considered. For this second configuration, Equation 7 is verified, that is, the robotic arm is placed inside the working envelope thus all the via-points in Table 2 and Figure 7 can be reached.

The end-effector trajectories of the robotic arm representing the projections on the \(O_0x_0y_0\) (radial trajectory of the end effector), \(O_0x_0z_0\), and \(O_0y_0z_0\) planes are shown Figure 6A, Figure 6B, and Figure 6C. The end-effector trajectory position and velocity of the robotic arm is shown in the 3D space in Figure 7A and Figure 7B. The smoothness of the continuous position and velocity proves again the performance of the method. Therefore, the forces acting on the end effector of the robotic arm along the generated path are also continuous.
FIGURE 6  End-effector trajectory projections. A, Oxy-plane; B, Oxz-plane; C, Oyz-plane

FIGURE 7  Three-dimensional end-effector trajectory position and velocity of the robotic arm. A, End-effector position; B, End-effector velocity

6 | CONCLUSION

In this study, the modeling and simulation of a robotic arm and the associated 3D trajectory planning of its end effector is presented. The robotic arm trajectory, expressed in cylindrical coordinates, is generated using a mix-matched polar and Cartesian piecewise Hermite–type polynomials in order to approximate the radial path and associated height respectively. Due to the system geometry which constrains the trajectory inside the working envelope, the existence of a solution in relation with the base of the robotic arm is addressed. Two numerical simulations are performed for two different configurations to validate the solution in relation with the working envelope.

ORCID

Mihai Dupac  https://orcid.org/0000-0002-3922-6299

REFERENCES

1. Gasparetto A, Boscariol P, Lanzutti A, Vidoni R. Trajectory planning in robotics. Math Comput Sci. 2012;6:269-279.
2. Hsu D, Kindel R, Latombe JC, Rock S. Randomized kinodynamic motion planning with moving obstacles. Int J Robot Res. 2002;21(3):233-255.
3. Dupac M. Kinematics and dynamics motion planning by polar piecewise interpolation and geometric considerations. Electron Notes Discrete Math. 2018;67:19-24.
4. Dupac M, Noroozi S. Dynamic modeling and simulation of a rotating single link flexible robotic manipulator subject to quick stops. Strojinski Vestnik – J Mech Eng. 2014;60(7-8):475-482.
5. Bobrow JE, Dubowsky S, Gibson JS. Time-optimal control of robotic manipulators along specified paths. Int J Robot Res. 1985;4(3):3-17.
6. Dong J, Ferreira PM, Stori JA. Feed-rate optimization with jerk constraints for generating minimum-time trajectories. *Int J Mach Tools Manuf.* 2007;47(12):1941-1955.

7. Lo HS, Xie SQ. Exoskeleton robots for upper-limb rehabilitation: state of the art and future prospects. *Med Eng Phys.* 2012;34:261-268.

8. Culmer PR, Jackson AE, Makower SG, et al. A novel robotic system for quantifying arm kinematics and kinetics: description and evaluation in therapist-assisted passive arm movements post-stroke. *J Neurosci Methods.* 2011;197:259-269.

9. Miao Q, McDaid A, Zhang M, Kebria P, Li H. A three-stage trajectory generation method for robot-assisted bilateral upper limb training with subject-specific adaptation. *Robot Auton Syst.* 2018;105:38-46.

10. Brahmi B, Saad M, Lam JTAT, Luna CO, Archambault PS, Rahman MH. Adaptive control of a 7-DOF exoskeleton robot with uncertainties on kinematics and dynamics. *J Neurosci Methods.* 2018;42:77-87.

11. Fraile JC, Perez-Turiel J, Baeyens E, et al. E2Rebot: a robotic platform for upper limb rehabilitation in patients with neuromotor disability. *Adv Mech Eng.* 2006;8(8):1-13.

12. Liu K, Xiong C-H, He L, Chen W-B, Huang X-L. Postural synergy based design of exoskeleton robot replicating human arm reaching movements. *Robot Auton Syst.* 2018;99:84-96.

13. Mancisidor A, Zubizarreta A, Cabanes I, Bengoa P, Jung JH. Kinematical and dynamical modeling of a multipurpose upper limbs rehabilitation robot. *Robot Comput Integr Manuf.* 2018;49:374-387.

14. Lin CS, Chang PR, Luh JYS. Formulation and optimization of cubic polynomial joint trajectories for industrial robots. *IEEE Trans Autom Control.* 1983;28(12):1066-1073.

15. Sanchez-Reyes J. Single-valued spline curves in polar coordinates. *Comput Aided Des.* 1992;24:307-315.

16. Goodman TNT, Lee SL. B-splines on the circle and trigonometric B-splines. *Approx Theory Spline Funct.* 1984;136:297-325.

17. Kang IG, Park FC. Cubic spline algorithms for orientation interpolation. *Int J Numer Methods Eng.* 1999;46:65-64.

18. Adhami L, Coste E. Optimal planning for minimally invasive surgical robots. *IEEE Trans Robot Autom.* 2003;19(5):854-863.

19. du Plessis LJ, Snyman JA. Trajectory-planning through interpolation by overlapping cubic arcs and cubic splines. *Int J Numer Methods Eng.* 2003;57:1615-1641.

20. Su B, Zou L. Manipulator trajectory planning based on the algebraic trigonometric hermite blended interpolation spline. *Procedia Eng.* 2012;29:2093-2097.

21. Kalyoncu M. Mathematical modelling and dynamic response of a multistraight-line path tracing flexible robot manipulator with rotating-prismatic joint. *Appl Math Model.* 2008;32:1087-1098.

22. Dupac M. A virtual prototype of a constrained extensible crank mechanism: dynamic simulation and design. *Proc Inst Mech Eng K J Multi-body Dyn.* 2013;227(3):201-210.

23. Dupac M. A path planning approach of 3D manipulators using zenithal gnomic projection and polar piecewise interpolation. *Math Methods Appl Sci.* 2018;41(17):7202-7213.

24. Iwashita Y. Piecewise polynomial interpolation. *OpenGamma Quant Res.* 2014;15:1-22.

25. Dupac M, Sewell P. Quick 3D trajectory planning for rotating extensible manipulators using piecewise polynomial interpolation. In: Proceedings of the Congress on Numerical Methods in Engineering (CMN); 2017; Valencia, Spain.

26. Kohli D, Spanos J. Workspace analysis of mechanical manipulators using polynomial discriminants. *J Mech Transm Autom.* 1985;107(2):209-215.

27. Cao Y, Lu K, Li X, Zang Y. Accurate numerical methods for computing 2D and 3D robot workspace. *Int J Adv Robot Syst.* 2011;8(6):1-13.

28. Marghitu DB, Dupac M. *Advanced Dynamics: Analytical and Numerical Calculations With Matlab.* New York, NY: Springer; 2012.

29. Zlajpah L. Simulation in robotics. *Math Comput Simul.* 2008;79(4):879-897.

**AUTHOR BIOGRAPHY**

Mihai Dupac received a PhD in Mathematics from the West University of Timisoara, Romania, and a PhD in Mechanical Engineering from Auburn University, AL, USA. Since 2000, he has been with the Faculty of Science and Technology at Bournemouth University, UK, holding a senior lecturer position. His research interests are in the area of modeling and simulation concerned with the development of new computational methods that arise from the discretization of computational science and engineering problems. He published over 100 research papers and 4 books in the field of computational mathematics and engineering.

**How to cite this article:** Dupac M. A combined polar and Cartesian piecewise trajectory generation and analysis of a robotic arm. *Comp and Math Methods.* 2019;1:e1049. [https://doi.org/10.1002/cmm4.1049](https://doi.org/10.1002/cmm4.1049)