Research Article

Kinematic and dynamic analysis of a schönflies parallel manipulator with horizontal rotation axis

Abstract

This paper addresses the kinematic and dynamic analysis of a 4-DOF parallel manipulator with Schönflies motion, designed for pick-and-place applications. Its low inertia makes it a suitable choice for pick-and-place applications, which demands high velocity and acceleration. However, the constrained structure of this robot, with its translational and rotational degrees of freedom makes its dynamic analysis a challenging task. Kinematic analysis is performed via geometric method. After extracting the constraint equations, velocity analysis is performed and Jacobian matrices are evaluated. Using the constraint equations and joint limits, reachable workspace is determined applying point-to-point search algorithm and the existing singularities are identified by the inverse and direct Jacobian matrices. For dynamic modeling, Euler-Lagrange formulation and Virtual work principle are applied. To verify the validity of the extracted kinematic and dynamic model, first the mechanism is designed in CATIA and then imported in ADAMS software. Simulation results of the extracted models are compared to ADAMS model over a smooth trajectory formed by B-spline.

Keywords: high-speed parallel robot, schönflies motion generator, kinematics, dynamics

Pegah Ghaf Ghanbari, Mostafa Taghizadeh, Mahmood Mazare

School of Mechanical Engineering, Shahid Beheshti University, Iran

Correspondence: Mahmood Mazare, School of Mechanical Engineering, Shahid Beheshti University, Tehran, Iran, Email M_mazar@sbu.ac.ir

Introduction

Parallel manipulators have proven their superiority over the conventional serial ones, in terms of high accuracy, velocity, stiffness, payload capacity and great dynamic performance.[1,2] Although early generation benefited from 6-DOF, nowadays mechanisms with less degrees of freedom are preferred, thanks to their simpler structure, less complex kinematics and dynamics and lower manufacturing cost.[3] Pick-and-place which accounts for a great portion of industrial robotic applications require three translational and one rotational motion (3T1R), called Schönflies or SCARA[1] motion. 4-DOF parallel robots utilizing parallelograms in their limbs are a good choice for this application as their light-weight; low-inertia structure is compatible with its requirements.

Two eminent high-speed and high acceleration Schönflies parallel robot in the market are Delta robot and Adept Quattro. The former[4] consists of three identical R-(SS)2 limbs, generating translational motion of the end-effector in three spatial directions, and a forth telescopic limb, providing the rotational motion about the vertical axis. Evolved from H4,[5] I4,[6] and Par4[7] designs, Adept Quattro[8] employs four identical R-(SS)2 limbs with an articulated traveling plate to generate the rotation about the vertical axis by an angular amplification device. Recently developed X4,[9] uses a single platform structure and can produce ±90° rotation. Four identical arms are connected to the end-effector via four revolute joints whose rotation axes are parallel to a common vertical axis, forming R-(SS)²-R arms. These robots are able to generate rotational motion about the vertical axis. In this paper, a schönflies-motion parallel robot, which is a modification on Veloce redundantly actuated pure 3-DOF translational parallel robot[10] is studied. This manipulator is able to produce rotational motion around a horizontal axis and can be an option to fill the existing gap in the industry for such a robot.

Aside from all positive points of parallel robots, their design, analysis, and manufacturing are complicated due to the presence of constraints imposed by their closed kinematic chains. For this reason, their kinematic and dynamic analysis has been the subject of numerous research works. Song,[11] Li,[12] and Mazare[13] focused on kinematic analysis of parallel robots. For dynamic modeling of these robots, various approaches have been applied including Newton-Euler method,[14,15,16] Euler-Lagrange formulation,[17,18] principle of virtual work,[19,20,21] Kane method,[22,23] Gibbs-Appel formulation, and Hamilton method.[24]

Mazare, et al[25] derived the dynamic model of a 3-DOF parallel robot by Euler-Lagrange formulation and principle of virtual work and investigated the computation time for solving the inverse dynamic problems. They qualitatively validated the results by means of work-energy and impulse-momentum methods. Li and Xu[26] investigated the dynamics of a 3-PRS parallel mechanism using these two approaches. By introducing a simplifying hypothesis, a simplified dynamic model was set up based on principle of virtual work, and it was demonstrated that simplified dynamic model is reasonable under such kind of assumptions. Brinker, et al[27] compared the complete and simplified dynamic models of delta robot in terms of computation times and accuracy. Three approaches were applied to derive dynamic formulations, including Newton-Euler method, principle of virtual work, and Euler-Lagrange formulation. In spite of the fact that capability and efficiency of high-speed pick-and-place parallel robots is directly related to their dynamic characteristics and controller, a proper trajectory planning also plays a crucial role in smooth torques, low energy consumption, low residual vibrations and short cycle times.[28,29]

Gallant, et al[30] showed that payload capacity of a robot is affected by the trajectory it follows. Khoukhi, et al[31] solved the trajectory planning problem of a robot using a variational calculus framework. They Considered robot kinematic and dynamic models, while optimizing execution time and required, avoiding singularities and satisfying several constraints related to the robot, task and workspace. Gasparetto, et al[32] suggested a new approach to optimize the trajectory of robot manipulators for minimum integral of the squared jerk and

---

[1]Selective Compliance Assembly Robot Arm

execution time. To compose the overall trajectory, quantic B-spline curves were utilized which produces continuous position, velocity, and acceleration profiles. Similarly, making use of quantic B-spline curves and exploiting the symmetric properties of the path, Li, et al[33] proposed a new approach for optimal smooth trajectory planning of high-speed pick-and-place parallel robots and compared the resulting input torques and residual vibrations of the manipulator with the trajectories obtained from Lamé curves and piecewise polynomials.

The rest of the paper is organized as follows. In section 2 the parallel manipulator is presented. Section 3 deals with the kinematic problem, while in section 4 the dynamic models of the robot are derived based on Euler-Lagrange formulation and principle of virtual work. Section 5 describes trajectory the generated for pick-and-place application, employing quintic B-splines. Section 6 is devoted to designing a PD controller. Section 7 presents the simulation results for both dynamic models and compares them to ADAMS output to validate the models and present the superior performance of the chosen trajectory. Conclusions are derided in section 8.

Manipulator description

The mechanism under review is a modification on redundantly actuated Veloce manipulator, comprising of four identical R-(SS)[2] arms, connecting the base to the end-effector. By shifting the connection of the two opposite arms to the end-effector along opposite directions and adding two revolute joints as shown in Figure 1, the mechanism will be able to generate Schönflies motion, with its rotation around horizontal axis.[34]
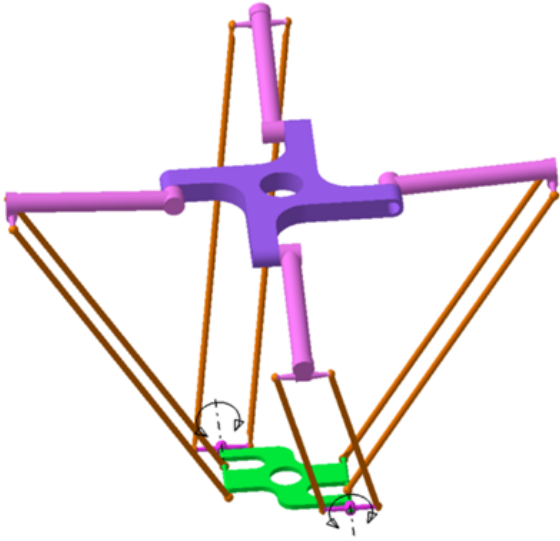

Figure 1 CAD model of Schönflies motion robot.

Kinematics

The structure of one arm of the manipulator is illustrated in Figure 2. The base frame $\mathfrak{F}_O : O - XYZ$, is attached to the geometric center of the fixed platform defined by revolute joints ($A_i, i = 1,\ldots,4$), with its X direction pointing toward A_1 and its Z direction normal to the plane of the fixed platform. Similarly, the moving frame $\mathfrak{F}_P : P - xyz$, is set at the geometric center of the moving platform, with its x direction crossing point C_1 and its z direction perpendicular to the plane formed by C_i, ($i = 1,\ldots,4$), representing the attaching points of the arms to the end effector. y is the rotation axis of the end effector and it always remains parallel to Y.
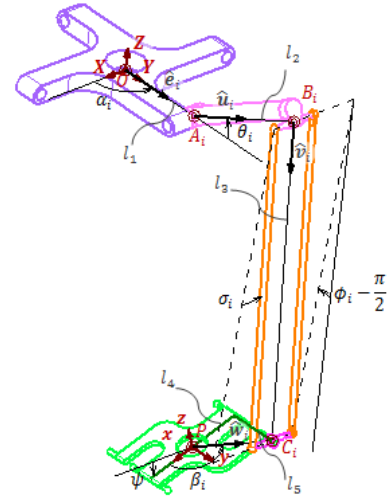

Figure 2 Coordinate frames and variables of the arm.

Position kinematics

The loop-closure equation for the *i*th arm is written as

$$\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i = \mathbf{r}_{Ai} + l_2 \hat{\boldsymbol{u}}_i + l_3 \hat{\boldsymbol{v}}_i \tag{1}$$

$$\varepsilon_i = \begin{cases} \sec \beta_i & i = 1, 4 \\ -\sec \beta_i & i = 2, 3 \end{cases} \tag{2}$$

Rearranging this equation and squaring both sides results in

$$\left(\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i - \mathbf{r}_{Ai} \right)^2 + l_2^2 - 2l_2 \left(\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i - \mathbf{r}_{Ai} \right).\hat{\boldsymbol{u}}_i = l_3^2 \tag{3}$$

which yields the constraint equations of the manipulator as

$$\boldsymbol{f}\left(\boldsymbol{\mathcal{X}},\boldsymbol{\Theta}\right) = 0 \tag{4}$$

where $\boldsymbol{\mathcal{X}} = \begin{bmatrix} \boldsymbol{r}_P^T & \psi \end{bmatrix}^T$ is the end effector pose, and $\boldsymbol{\Theta} = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \end{bmatrix}^T$ is the active joint variables.

$\hat{\boldsymbol{u}}_i$ is the only vector dependent on θ_i and is defined by

$$\hat{\boldsymbol{u}}_i = \begin{bmatrix} c\alpha_i c\theta_i & s\alpha_i c\theta_i & -s\theta_i \end{bmatrix}^T \tag{5}$$

Introducing (5) to equations (3) yields

$$\left(C_{i3} - C_{i2} \right) k_i^2 + 2C_{i1}k_i + \left(C_{i3} + C_{i2} \right) = 0 \tag{6}$$

$$k_i = \tan \frac{\theta_i}{2}$$

$$C_{i1} = 2l_2 \left(\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i - \mathbf{r}_{Ai} \right).\hat{\xi}_3$$

$$C_{i2} = -2l_2 \left(\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i - \mathbf{r}_{Ai} \right).\left\{ \hat{\xi}_1 c \, \alpha_i + \hat{\xi}_2 s \, \alpha_i \right\}$$

$$C_{i3} = \left(\mathbf{r}_P + \varepsilon_i l_4 \hat{\boldsymbol{w}}_i - \mathbf{r}_{Ai} \right)^2 + l_2^2 - l_3^2$$

$$\hat{\xi}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T, \quad \hat{\xi}_2 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T, \quad \hat{\xi}_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T \tag{7}$$

The solution to the second-degree equation (6) is the inverse kinematic equation of the robot, written as

$$\theta_i = 2\arctan\left(\frac{-C_{i1} \pm \sqrt{C_{i1}^2 - \left(C_{i3}^2 - C_{i2}^2\right)}}{C_{i3} - C_{i2}}\right) \tag{8}$$

Velocity analysis

The key element of velocity analysis is Jacobian matrix, which is a mapping between the joint velocities to the velocity of the end-effector. To derive this matrix, eq.(1) is differentiated with respect to time.

$$\dot{\mathbf{r}}_P = l_2\left(\omega_{Ui} \times \hat{\mathbf{u}}_i\right) + l_3\left(\omega_{Li} \times \hat{\mathbf{v}}_i\right) - \dot{\psi} \times \mathbf{r}_{PCi}^O \tag{9}$$

where ω_{Ui} and ω_{Li} are the angular velocity of the upper and lower links of the ith arm, respectively. Taking the dot product of both sides of eq.(9) with $\mathbf{v}_i$ and considering $\omega_{Ui} = \dot{\theta}_i\,\mathbf{n}_i$, we obtain

$$\dot{\mathbf{r}}_P.\hat{\mathbf{v}}_i = l_2\dot{\theta}_i\left(\hat{\mathbf{n}}_i \times \hat{\mathbf{u}}_i\right).\hat{\mathbf{v}}_i - \dot{\psi}\left(\hat{\xi}_2 \times \mathbf{r}_{PCi}^O\right).\hat{\mathbf{v}}_i \tag{10}$$

This gives the mapping between the velocity of the ith active joint and the end effector, which can be written in matrix form.

$$\dot{\theta}_i = \mathbf{J}_i\dot{\mathcal{X}} \tag{11}$$

$$\dot{x} = [\dot{r}_P\ \dot{\psi}]^T\ ,\, j_i = [j_{vi}\ j_\psi] \tag{12}$$

$$\mathbf{j}_{vi} = \frac{\hat{v}_i^T}{l_2(\hat{n}_i \times \hat{u}_i).\hat{v}_i}\ ,\, j_\psi = \frac{\varepsilon_i l_4(\hat{\xi}_2 \times \hat{w}_i).\hat{v}_i}{l_2(\hat{n}_i \times \hat{u}_i).\hat{v}_i}$$

Obviously, the velocity equation for the whole manipulator will be

$$\dot{\Theta} = \mathbf{J}\dot{\mathcal{X}}\ ,\quad \mathbf{J} = \begin{bmatrix} \mathbf{J}_1^T & \mathbf{J}_2^T & \mathbf{J}_3^T & \mathbf{J}_4^T \end{bmatrix}^T \tag{13}$$

where $\dot{\Theta} = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 & \dot{\theta}_4 \end{bmatrix}^T$ represents the vector of active joints angular velocity.

Now assuming that the center of mass of the upper link is located in the middle of $\overline{A_l B_l}$, the linear velocity of the point D_i can be determined from

$$\mathbf{v}_{Di} = \mathbf{J}_{Di}\dot{\mathcal{X}}\ ,\quad \mathbf{J}_{Di} = \frac{l_2}{2}\left(\hat{\mathbf{n}}_i \times \hat{\mathbf{u}}_i\right)\mathbf{J}_i \tag{14}$$

Workspace analysis

Reachable workspace refers to areas of space that the end-effector central point can access regardless of orientation. The constraints for determining the robot workspace embody the connection between the arms, the performance range of the joints, the length of the links, and the internal correlations. In this research, numerical method is used to extract the workspace of the studied robot and the following constraints are applied to the joints.

$$-\pi/2 \le \psi \le \pi/2$$
$$-\pi/3 \le \theta_i \le \pi/2 \tag{15}$$
$$-\pi/3 \le \sigma_i \le \pi/3$$
$$\pi/6 \le \gamma_i \le 5\pi/6$$

Where $\gamma_i = \pi - \left(\phi_i - \theta_i\right)$ is the angle of upper and lower links in each chain. Figure 3– depicts the workspace of the robot in three distinct rotations.
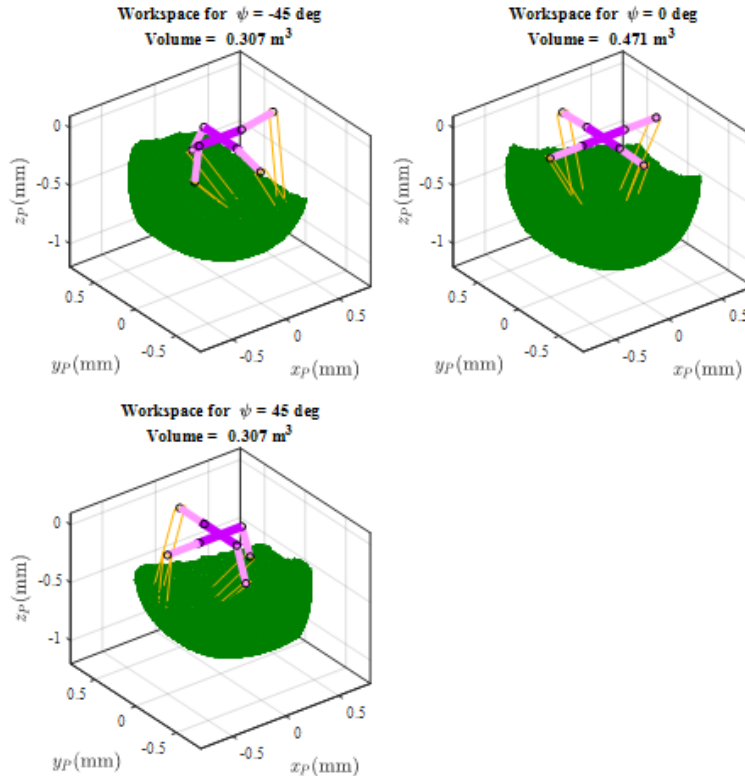

Figure 3 Constant orientation workspace.

Dynamics

For dynamic modeling of the manipulator, both the principle of virtual work and Euler-Lagrange formulation are used. As the rotational inertia of the lower links is negligible, in both approaches, they are considered as point masses on their both extreme ends. (17)

$$= \begin{bmatrix} -(m_U + m_L)\boldsymbol{J}_{Di} \\ -\left(\mathbf{I}^O_{Ui} + \frac{1}{2}m_L l_2^2\right)\hat{\boldsymbol{n}}_i \boldsymbol{J}_i \end{bmatrix}\ddot{\boldsymbol{\mathcal{X}}} + \begin{bmatrix} \left(m_U + \frac{1}{2}m_L\right)\mathbf{g} \\ 0 \end{bmatrix} + \begin{bmatrix} -(m_U + m_L)\boldsymbol{a}'_{Di} \\ -\mathbf{I}^O_{Ui}\hat{\boldsymbol{n}}_i\ddot{\theta}_i - \omega_{Ui}\times(\mathbf{I}^O_{Ui}\omega_{Ui}) \end{bmatrix}$$

In which $\mathbf{g}$ is the gravity vector, $\mathbf{I}^O_{Ui}$ is the moment of inertia of the ith upper link about its center of mass with respect to base coordinate frame, $\mathfrak{F}_O$. Force system of the end-effector is deduced as

$$\boldsymbol{\mathcal{F}}_p = \boldsymbol{\mathcal{F}}_{pI} + \boldsymbol{\mathcal{F}}_{pG} + \boldsymbol{\mathcal{F}}_{pE} + \boldsymbol{\mathcal{F}}_{pC} = \begin{bmatrix} -m_e\boldsymbol{a}_p \\ -\mathbf{I}^O_P\ddot{\psi}.\xi_2 \end{bmatrix} + \begin{bmatrix} m_e\mathbf{g} \\ 0 \end{bmatrix} + \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{T}.\xi_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \left(-\dot{\psi}\times(\mathbf{I}^O_P\dot{\psi})\right).\xi_2 \end{bmatrix}$$

(18)

Where I^o_p is the moment of inertia of the end-effector and F_{pE} is the external wrench applied to it.

Extending the principle of the virtual work from the static to the dynamic case with the D'Alembert's principle, we can get

$$\sum_{i=1}^4 \delta \boldsymbol{t}^T_{ai}\boldsymbol{\mathcal{F}}_{Ui} + \delta\boldsymbol{\Theta}^T\tau + \delta\boldsymbol{\mathcal{X}}^T\boldsymbol{\mathcal{F}}_p = 0 \qquad (19)$$

where τ is the torque applied by the actuators, and $\delta\boldsymbol{t}_{ai}$ is the virtual displacement of the ith arm.

Referring to the kinematic analysis, the virtual displacements can be obtained as

$$\delta\boldsymbol{t}_{ai} = \begin{bmatrix} \delta\mathbf{r}_{Di} \\ \delta\boldsymbol{e}_i \end{bmatrix} = \begin{bmatrix} \boldsymbol{J}_{Di} \\ \boldsymbol{n}_i\boldsymbol{J}_i \end{bmatrix}\delta\boldsymbol{\mathcal{X}} = \boldsymbol{J}_{ai}\delta\boldsymbol{\mathcal{X}} \qquad (20)$$

$$\delta\boldsymbol{\Theta} = \boldsymbol{J}\delta\boldsymbol{\mathcal{X}} \qquad (21)$$

Substituting the deduced forces and virtual displacements into eq.(21) yields the inverse dynamics of the manipulator.

$$\tau = \boldsymbol{\mathcal{M}}\ddot{\boldsymbol{\mathcal{X}}} + \boldsymbol{\mathcal{V}} + \boldsymbol{\mathcal{G}} + \boldsymbol{\mathcal{F}} \qquad (22)$$

$$\boldsymbol{\mathcal{M}} = \boldsymbol{J}^{-T}\begin{bmatrix} m_e\boldsymbol{I}_{3\times3} & 0 \\ 0 & I^P_{P_{yy}} \end{bmatrix} + \boldsymbol{J}^{-T}\sum_{i=1}^4\boldsymbol{J}^T_{Ui}\begin{bmatrix} (m_U + m_L)\boldsymbol{J}_{Di} \\ \left(\mathbf{I}^O_{Ui} + \frac{1}{2}m_L l_2^2\right)\hat{\boldsymbol{n}}_i\boldsymbol{J}_i \end{bmatrix} \qquad (23)$$

Virtual work principle

Force system of the ith arm can be written as

$$\boldsymbol{\mathcal{F}}_{ai} = \boldsymbol{F}_{aIi} + \boldsymbol{F}_{aGi} + \boldsymbol{F}_{aCi} \qquad (16)$$

$$\boldsymbol{\mathcal{V}} = -\boldsymbol{J}^{-T}\left\{\boldsymbol{\mathcal{F}}_{pC} + \sum_{i=1}^4\boldsymbol{J}^T_{ai}\boldsymbol{\mathcal{F}}_{aCi}\right\} \qquad (24)$$

$$\boldsymbol{\mathcal{G}} = -\boldsymbol{J}^{-T}\left\{\boldsymbol{\mathcal{F}}_{pG} + \sum_{i=1}^4\boldsymbol{J}^T_{ai}\boldsymbol{\mathcal{F}}_{aGi}\right\} \qquad (25)$$

$$\boldsymbol{\mathcal{F}} = -\boldsymbol{J}^{-T}\boldsymbol{\mathcal{F}}_{pE} \qquad (26)$$

Euler-lagrange formulation

For modeling a dynamic system using Euler-Lagrange formulation, the first step is determining the Lagrangian, which is defined as the difference between kinetic energy and potential energy of the system. In this section, we neglect the inertia of the lower links and consider them as point masses on their both extreme ends. So, the variables will be active joints variables $\boldsymbol{\Theta}$ and end effector pose $\boldsymbol{\mathcal{X}}$.

$$\mathcal{L}(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\boldsymbol{\mathcal{X}},\dot{\boldsymbol{\mathcal{X}}}) = T(\boldsymbol{\Theta},\dot{\boldsymbol{\Theta}},\boldsymbol{\mathcal{X}},\dot{\boldsymbol{\mathcal{X}}}) - U(\boldsymbol{\Theta},\boldsymbol{\mathcal{X}}) \quad (27)$$

Potential energy is caused by conservative forces acting on the system. Here, the only conservative force is gravity. So, the potential energy of the arms is written as

$$U_a(\boldsymbol{\Theta}) = -\mathbf{g}^T\sum_{i=1}^4\left(m_U\mathbf{r}_{Di} + \frac{1}{2}m_L\mathbf{r}_{Bi}\right) = -\frac{1}{2}\mathbf{g}^T\left\{l_1(2m_U + m_L)\sum_{i=1}^4\boldsymbol{e}_i + l_2(m_U + m_L)\sum_{i=1}^4\boldsymbol{u}_i\right\} \qquad (28)$$

And the potential energy of the end-effector is obtained from

$$U_P = -(m_e + 2m_L)\begin{bmatrix} \mathbf{g}^T & 0 \end{bmatrix}^T\boldsymbol{\mathcal{X}} \qquad (29)$$

The potential energy of the manipulator is deduced from adding the potential energy of its subsystems.

$$U(\boldsymbol{\Theta},\boldsymbol{\mathcal{X}}) = U_a(\boldsymbol{\Theta}) + U_P(\boldsymbol{\mathcal{X}}) \qquad (30)$$

The kinetic energy of the arms is a function of active joints variable, and is determined by

$$T_a(\dot{\boldsymbol{\Theta}}) = \frac{1}{12}l_2^2(2m_U + 3m_L)\dot{\boldsymbol{\Theta}}^T\dot{\boldsymbol{\Theta}} \qquad (31)$$

The kinetic energy of the end effector is obtained from

$$T_P = \frac{1}{2}\dot{\boldsymbol{\mathcal{X}}}^T\boldsymbol{M}_P\dot{\boldsymbol{\mathcal{X}}} \qquad (32)$$

Where

$$\boldsymbol{M}_P = \text{diag}\begin{bmatrix} (m_e + 2m_L)\boldsymbol{I}_{3\times3} & I^P_{P_{yy}} + 2m_L l_4^2 \end{bmatrix} \qquad (33)$$

The kinetic energy of the manipulator is obtained from adding the kinetic energy of the arms and the end-effector.

$$T(\dot{\boldsymbol{\Theta}},\dot{\boldsymbol{\mathcal{X}}}) = T_a(\dot{\boldsymbol{\Theta}}) + T_P(\dot{\boldsymbol{\mathcal{X}}}) \qquad (34)$$

As is obvious, the Lagrangian can be written as the contribution of the two subsystems, each one depending on its own variables, though connected together via constraint equations. This helps us with deriving two sets of dynamic equations using 1st type of Lagrange formulation.

The dynamic model of the arms is calculated from

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}_a}{\partial \dot{\boldsymbol{E}}}\right) - \frac{\partial \mathcal{L}_a}{\partial \boldsymbol{E}} = \tau + \left(\frac{\partial \boldsymbol{f}}{\partial \dot{\boldsymbol{E}}}\right)^T \lambda \quad (35)$$

where, $\lambda_{4 \times 1}$ is the vector of Lagrange multipliers, and $\boldsymbol{f}$ represents the constraint equations of the manipulator. This yields the following equations

$$\boldsymbol{M}_a \ddot{\boldsymbol{\Theta}} + \boldsymbol{G}_a = \tau + \left(\frac{\partial \boldsymbol{f}}{\partial \dot{\boldsymbol{E}}}\right)^T \lambda \quad (36)$$

$$\boldsymbol{M}_a = \frac{1}{6} l_2^2 \left(2m_U + 3m_L\right) \boldsymbol{I}_{4 \times 4} \quad (37)$$

$$\boldsymbol{G}_a = -\frac{1}{2} l_2 \left(m_U + m_L\right) g \cos \boldsymbol{\Theta} \quad (38)$$

Similarly, the dynamic model of the end-effector is calculated by

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}_P}{\partial \dot{\boldsymbol{X}}}\right) - \frac{\partial \mathcal{L}_P}{\partial \boldsymbol{X}} = \mathcal{F}_{PE} + \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{X}}\right)^T \lambda \quad (39)$$

This results in the following dynamic equations.

$$M_p \ddot{X} + G_p = F_{PE} + \left(\frac{\partial f}{\partial X}\right)^T \lambda \quad (40)$$

$$\boldsymbol{G}_P = -\left(m_e + 2m_L\right) \begin{bmatrix} \mathbf{g} \\ 0 \end{bmatrix} \quad (41)$$

In order to calculate the dynamic model of the whole manipulator, these two models need to be combined by cancelling the Lagrange multipliers. To do so, eq.(35) is solved for λ and the result is introduced to eq.(39).

$$\tau = \boldsymbol{M}_a \ddot{\boldsymbol{\Theta}} + \boldsymbol{G}_a - \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\Theta}}\right)^T \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{X}}\right)^{-T} \left(M_P \ddot{X} + \boldsymbol{G}_P - \mathcal{F}_{PE}\right) \quad (42)$$

Differentiating eq.(3) with respect to time, we obtain

$$\frac{\partial f}{\partial \Theta} \dot{\Theta} + \frac{\partial f}{\partial X} \dot{X} = 0 \quad (43)$$

Solving this equation for $\boldsymbol{X}$ results in

$$\dot{\boldsymbol{X}} = -\left(\frac{\partial f}{\partial \boldsymbol{X}}\right)^{-1} \frac{\partial f}{\partial \Theta} \dot{\Theta} \quad (44)$$

Comparing eq.(13) and eq.(43), we can infer that

$$-\left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\Theta}}\right)^T \left(\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{X}}\right)^{-T} = \boldsymbol{J}^{-T} \quad (45)$$

Substituting eq.(44) into (41) gives the dynamic model of the manipulator in workspace.

$$\tau = \mathcal{M} \ddot{\boldsymbol{X}} + \mathcal{C} + \mathcal{G} + \mathcal{F} \quad (46)$$

$$\mathcal{M} = \boldsymbol{M}_a \boldsymbol{J} + \boldsymbol{J}^{-T} \boldsymbol{M}_P \quad (47)$$

$$C = M_a \ddot{\theta} \quad (48)$$

$$\mathcal{G} = \boldsymbol{G}_a + \boldsymbol{J}^{-T} \boldsymbol{G}_P \quad (49)$$

$$\mathcal{F} = -\boldsymbol{J}^{-T} \mathcal{F}_{PE} \quad (50)$$

Trajectory planning

PnP operations are usually performed in an environment free of obstacles, as a result just initial and end points of the path are important. However to have more control on departure and arrival, we define some via points as shown in Figure 4. Here, v-axis is parallel to Z-axis of the $\mathfrak{F}_O : O - XYZ$ coordinate frame; and v-axis is located in $X - Y$ plane.
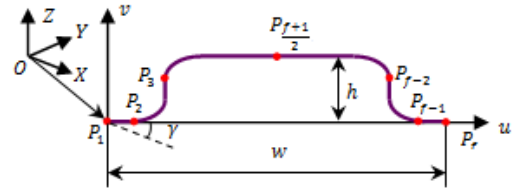
Figure 4 PnP path defined in a local frame.

Any point on the path in defined in the local coordinate can be transformed to the base coordinate frame as follows.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ 0 \\ v \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \quad (51)$$

Where, $\gamma = \arctan \frac{y_f - y_1}{x_f - x_1}$ is the rotation angle of local frame with respect to the fixed frame. Parameters defining the path are

$$\lambda = \frac{w}{h} \quad (52)$$

$$w = \sqrt{\left(x_f - x_1\right)^2 + \left(y_f - y_1\right)^2} \quad (53)$$

$$h = z_{\frac{f+1}{2}} - z_{1(f)} \quad (54)$$

A B-spline of degree p and order $k = p + 1$ is a linear combination of polynomials $N_{i,p}(\tau)$ of degree p, called *base* or *blending functions*, weighted by control points, Q_i. τ represents a normalized independent parameter here taken as the time. Then, the normalized motion profile $s(\tau)$ is then formed as

$$s(\tau) = \sum_{i=1}^{n} Q_i \cdot N_{i,p}(\tau) \quad , \ 0 \leq \tau \leq 1 \quad (55)$$

The base function is defined recursively by means of the De Boor formula.[35]

$$\begin{cases} N_{i,0}(\tau) = \begin{cases} 1, & \tau_i \leq \tau < \tau_{i+1} \\ 0, & \text{elsewhere} \end{cases} \\ N_{i,p}(\tau) = \frac{\tau - \tau_i}{\tau_{i+p} - \tau_i} N_{i,p-1}(\tau) \\ \qquad\qquad + \frac{\tau_{i+p+1} - \tau}{\tau_{i+p+1} - \tau_{i+1}} N_{i+1,p-1}(\tau) \end{cases} \quad (56)$$

$\tau_i\left(i=1,2,\ldots,m=n+p+1\right)$ is a sequence of nodes on which $s(\tau)$ is built by interpolation. The rth derivative of B-spline is calculated from

$$\frac{d^r s}{d\tau^r} = \sum_{i=1}^{n-r} Q_{i,r} . N_{i,p-r}(\tau) \tag{57}$$

Where $Q_{i,r}$ represents the control points of the rth derivative of $s(\tau)$ and can be deduced recursively by

$$Q_{i,r} = \begin{cases} Q_i & ,r=0 \\ \dfrac{p-r+1}{\tau_{i+p+1}-\tau_{i+r}}\left(Q_{i+1,r-1}-Q_{i,r-1}\right) & ,0<r\le p \end{cases} \tag{58}$$

In order for the control points to coincide with the initial and final via-points, the nodes at both ends of the trajectory must be repeated $p+1$ times. Moreover, to obtain trajectories with no jerk at both ends, two virtual points is introduced at the second and second-last position of the node sequence. With these considerations, the number of nodes and the number of control points will be $m=2(p+1)+f$ and $n=p+f+1$ respectively.

A B-spline of degree p has C^{p-1}-continuity, so for the trajectory to be smooth in terms of jerk, it is necessary to choose $p=5$. Therefore, $n=f+6$ and $m=12+f$. In order to guarantee that the trajectory passes through the via-points, the following equations must be satisfied.

$$\begin{cases} s(0)=s_1=Q_1 \\ s\left(\tau_{p+j+1}\right)=s_j & j=2,3,\ldots,f-1 \\ \quad = \sum_{i=1}^{n} N_{i,p}\left(\tau_{p+j+1}\right).Q_i \\ s(1)=s_f=Q_n \end{cases} \tag{59}$$

Imposing boundary conditions for velocity, acceleration and jerk at both ends of the trajectory results in six constraint equations as

$$\begin{cases} Q_{1,r} = \dfrac{d^r s}{d\tau^r}\bigg|_{\tau=0} & ,r=1,2,3 \\ \quad = \sum_{i=1}^{r+1} C_{i,r}.Q_i = 0 \\ Q_{n-r,r} = \dfrac{d^r s}{d\tau^r}\bigg|_{\tau=1} & ,r=1,2,3 \\ \quad = \sum_{i=n-r}^{n} C_{i,r}.Q_i = 0 \end{cases} \tag{60}$$

$C_{i,r}$ can be deduced from (59) and (60) recursively. To calculate the control points of the desired trajectory, equations (61) and (62) are written in matrix form, yielding

$$\boldsymbol{Q} = \boldsymbol{A}^{-1}\boldsymbol{b} \tag{61}$$

Where

$$\boldsymbol{Q} = \begin{bmatrix} Q_1 & Q_2 & \cdots & Q_n \end{bmatrix}^T \tag{62}$$

$$\boldsymbol{A} = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{63}$$

$$\boldsymbol{b}_1 = \begin{bmatrix} s_1 & s_2 & \cdots & s_f \end{bmatrix}^T$$
$$\boldsymbol{b}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{64}$$

$$A_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ N_{1,p}(\tau_{p+3}) & N_{2,p}(\tau_{p+3}) & \cdots & N_{n,p}(\tau_{p+3}) \\ N_{1,p}(\tau_{p+4}) & N_{2,p}(\tau_{p+4}) & \cdots & N_{n,p}(\tau_{p+4}) \\ \vdots & \vdots & \vdots & \vdots \\ N_{1,p}(\tau_{p+f}) & N_{2,p}(\tau_{p+f}) & \cdots & N_{n,p}(\tau_{p+f}) \\ 0 & 0 & \cdots & 1 \end{bmatrix}_{fxn} \tag{65}$$

$$A_2 = \begin{bmatrix} C_{1,1} & C_{2,1} & 0 & 0 \cdots & & & & \\ C_{1,2} & C_{2,2} & C_{3,2} & 0 \cdots & & & 0 & \\ C_{1,3} & C_{2,3} & C_{3,3} & C_{4,3}\cdots & & & & \\ & & & \cdots & 0 & 0 & C_{n-1,1} & C_{n,1} \\ & 0 & & \cdots & 0 & C_{n-2,2} & C_{n-1,2} & C_{n,2} \\ & & & \cdots C_{n-3,3} & C_{n-2,3} & C_{n-1,3} & C_{n,3} \end{bmatrix}_{6xn} \tag{66}$$

For this problem, we choose $f=7$ via-points as follows

$$P_1(0,0) \rightarrow P_{v1} \rightarrow P_2(\eta_1 w,0) \rightarrow P_3(\eta_2 w,\eta_3 h) \rightarrow P_4\left(\frac{1}{2},h\right) \rightarrow P_5\left((1-\eta_2)w,\eta_3 h\right) \rightarrow P_6\left((1-\eta_1)w,0\right) \rightarrow P_{v2} \rightarrow P_7(w,0) \tag{67}$$

where $0<\eta_1<1/2$ and $0<\eta_2<1/2$ are the ratio of the u coordinate of p_2 and p_3 to the corresponding coordinate of the final point, respectively, and obviously, $\eta_1<\eta_2$. Moreover, $0<\eta_3<1$ is the ratio of the v coordinate of p_3 to the corresponding coordinate of the mid-point. Therefore, the sequence of nodes will be

$$\left\{ \underbrace{0,0,\ldots,0}_{6}, \tau_7,\tau_8, \underbrace{\tau_9, \tau_{10}, \tau_{11}}_{5}, \tau_{12}, \underbrace{\tau_{13},1,1,\ldots,1}_{6} \right\} \tag{67}$$

Three-time ratio are defined as

$$k_1 = \tau_8/\tau_{10}, \quad k_2 = \tau_7/\tau_8, \quad k_3 = \tau_9/\tau_{10} \tag{68}$$

where $0 < k_1 < 1$ is the ratio between the time the end-effector spends to move from p_1 to p_2 and the time from p_1 to p_4. $0 < k_2 < 1$ represents the ratio between the time the end-effector spends to move from p_1 to p_{v1} and the time from p_1 to $0 < k_3 < 1$ p_2. denotes the ratio between the time the end-effector spends to move from p_1 to p_3 and the time from p_1 to p_4. To maintain the correct order of the nodes, the condition $k_1 < k_3$ must be met. The symmetry of the path allows expressing the nodes in terms of k_1, K_2 and K_3.

$$\tau_1 \ldots \tau_6 = 0, \quad \tau_7 = \frac{k_1 k_2}{2}, \quad \tau_8 = \frac{k_1}{2}, \quad \tau_9 = \frac{k_3}{2}, \quad \tau_{10} = \frac{1}{2}, \quad \tau_{11} = 1 - \frac{k_3}{2},$$
(69)

Now, position vector of any point on the path can be deduced from

$$\boldsymbol{r}(t) = \begin{bmatrix} u(t) & v(t) \end{bmatrix}^T = \frac{w}{T^0} \begin{bmatrix} s_u(t) & \lambda^{-1} s_v(t) \end{bmatrix}^T, \quad \tau = t/T$$
(70)

Where subscripts 'u' and 'v' stands for motion profile along the horizontal and vertical axis in local frame. T is the time taken by the end-effector to move from the initial to the final point. Considering $A_u = A_v$ and $b_{u,2} = b_{v,2}$, while

$$\boldsymbol{b}_{u,1} = \begin{bmatrix} 0 & \eta_1 & \eta_2 & 1/2 & 1-\eta_2 & 1-\eta_1 & 1 \end{bmatrix}^T$$

$$\boldsymbol{b}_{v,1} = \begin{bmatrix} 0 & 0 & \eta_3 & 1 & \eta_3 & 0 & 0 \end{bmatrix}^T$$
(71)

We can determine the control points from eq.(63).

Dimensionless position, velocity, acceleration and jerk of any point on the path can be determined by normalizing the position and its derivatives by w/T^r.

$$\boldsymbol{s}(\tau) = \begin{bmatrix} s_u(\tau) & \lambda^{-1} s_v(t) \end{bmatrix}^T$$

$$\boldsymbol{v}(\tau) = \begin{bmatrix} \frac{ds_u}{d\tau} & \lambda^{-1} \frac{ds_v}{d\tau} \end{bmatrix}^T$$

$$\boldsymbol{a}(\tau) = \begin{bmatrix} \frac{d^2 s_u}{d\tau^2} & \lambda^{-1} \frac{d^2 s_v}{d\tau^2} \end{bmatrix}^T$$
(72)

$$\boldsymbol{j}(\tau) = \begin{bmatrix} \frac{d^3 s_u}{d\tau^3} & \lambda^{-1} \frac{d^3 s_v}{d\tau^3} \end{bmatrix}^T$$

Cycle time, T, is limited by the speed and torque constraint of the motors. According to Li, et al[33] it can be proved that the cycle time must be subject to

$$T \geq \max \left(\frac{v_{\max} w}{[v]} \quad \sqrt{\frac{a_{\max} w}{[a]}} \right)$$
(73)

Trajectory optimization problem

As the peak values and smoothness of the time histories of the active joint torques are directly related to those of acceleration and jerk of the end-effector at non-singular configurations, we can optimize the trajectory in the Cartesian space using appropriate performance indices based on acceleration and jerk of the end-effector.[36] Li, et al[33]

proposed a pair of performance indices for evaluating acceleration and jerk of the end effector over the entire normalized path within the normalized time duration as

$$a_{\max}(k_1, k_2, k_3) = \max_{0 \leq \tau \leq 1} \boldsymbol{a}(\tau)$$

$$a_{\mathrm{rms}}(k_1, k_2, k_3) = \sqrt{\int_0^1 \boldsymbol{a}(\tau)^2 \, d\tau}$$

$$j_{\max}(k_1, k_2, k_3) = \max_{0 \leq \tau \leq 1} \boldsymbol{j}(\tau)$$
(74)

$$j_{\mathrm{rms}}(k_1, k_2, k_3) = \sqrt{\int_0^1 \boldsymbol{j}(\tau)^2 \, d\tau}$$

The multi-objective optimization problem for the smooth trajectory is formulated as:

Minimize	$f_1(\mathbf{x}) = a_{\max}$
Minimize	$f_2(\mathbf{x}) = a_{\mathrm{rms}}$
Minimize	$f_3(\mathbf{x}) = j_{\max}$
Minimize	$f_4(\mathbf{x}) = a_{\mathrm{rms}}$
Over	$\mathbf{x} = \begin{bmatrix} k_1 & k_2 & k_3 \end{bmatrix}$
	$g_1 : 0 < k_1 < k_3$
Subject to	$g_2 : 0 < k_2 < 1$
	$g_3 : 0 < k_3 < 1$

(75)

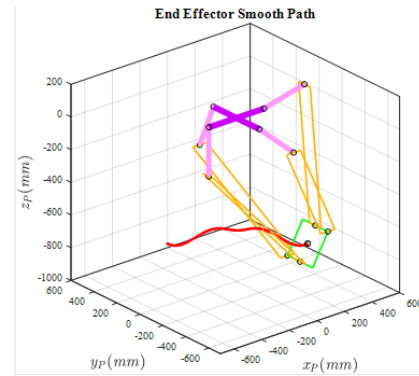Finally, the PnP trajectory in the workspace of the robot is depicted in Figure 5.

Figure 5 Robot reference PnP path.

Simulation results

To ensure the accuracy of the extracted kinematic and dynamic equations, the robot mechanism is simulated in the ADMS software, and its output is compared to the results obtained from the extracted equations. Reference trajectory components in task space are determined and illustrated in Figure 6. To verify the validity of the inverse dynamics model, the acceleration corresponding to the reference trajectory is applied to the mathematical model, and the torque of the actuators as the output of the dynamic model is compared with the ADAMS slightly different. simulation output. The results presented in Figure 7 demonstrate similar behavior, although the torque size is
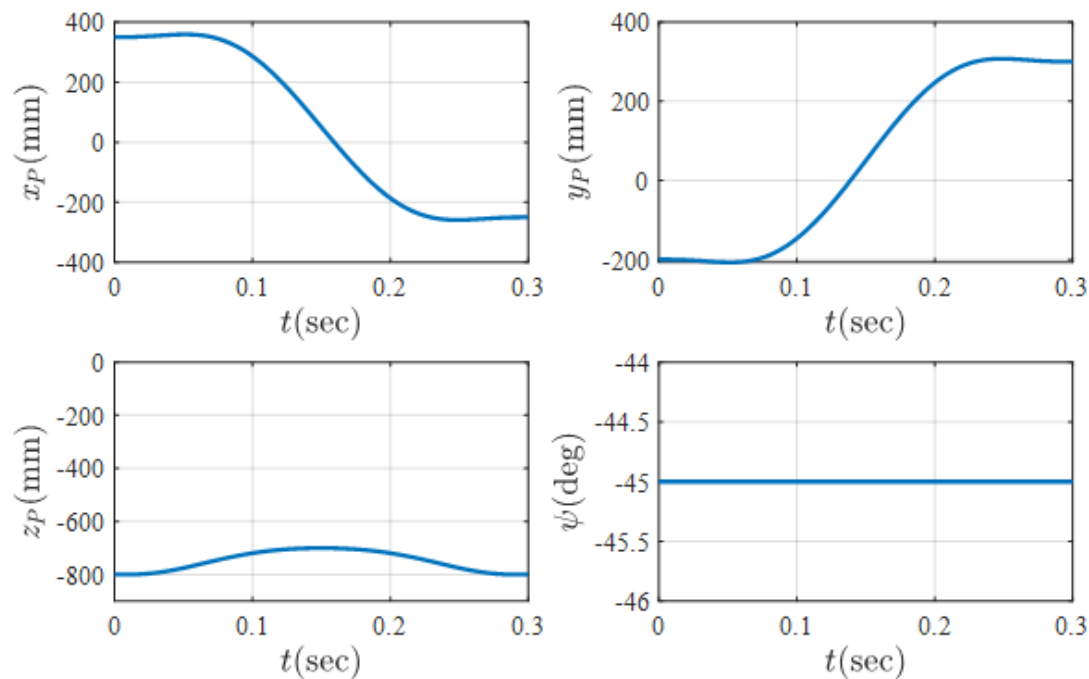
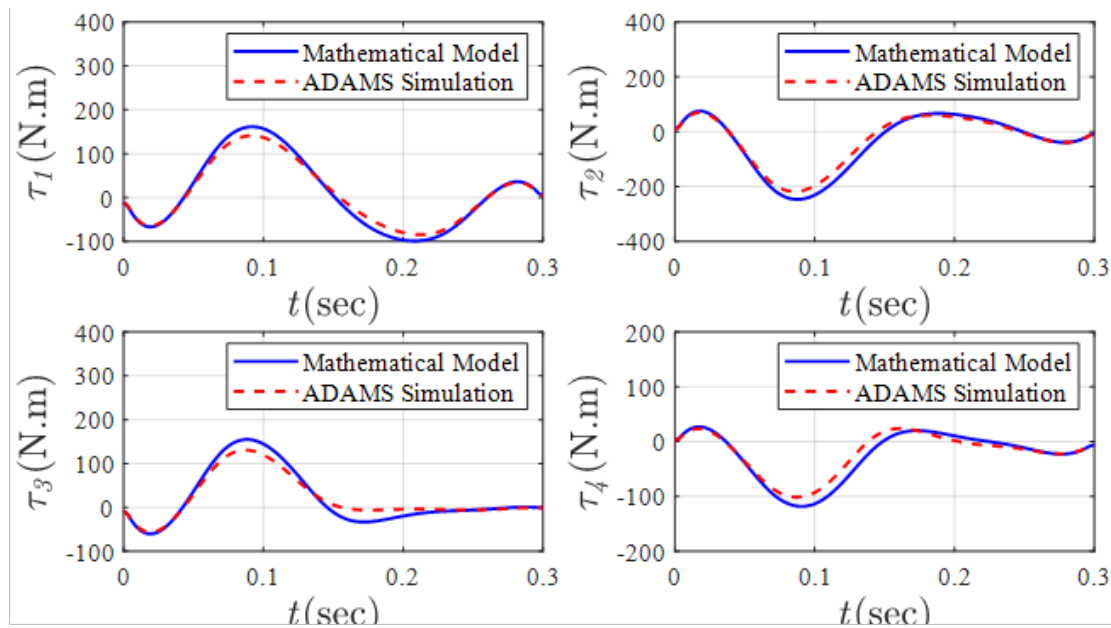**Figure 6** Robot reference PnP path.

Figure 7 Active joints torques.

The slight difference between the two distinct simulations arises from the assumptions that are included in system modeling. The first assumption was to place the center of gravity of the upper links in the center of the line along the center of their hinges on both ends. However, in the ADAMS model, the center of gravity deviates slightly from this point. The second hypothesis is ignoring the inertia of the lower links of the arms, which their effects are modeled as two identical lumped masses at the two ends of the links.

To solve the inverse kinematics of the robot, the input (reference trajectory) is applied to the end-effector and the angles of the active joints are calculated. Figure 8 shows the strict conformity of the results of the inverse kinematics analytically to the results of the mechanism simulated in the ADAMS.

In addition, in order to solve the forward kinematic problem, the angles of the active joints obtained from the inverse kinematic problem, are applied as inputs to the active joints. The results shown in Figure 9 which reveal a good agreement between the analytical method and the ADAMS model.
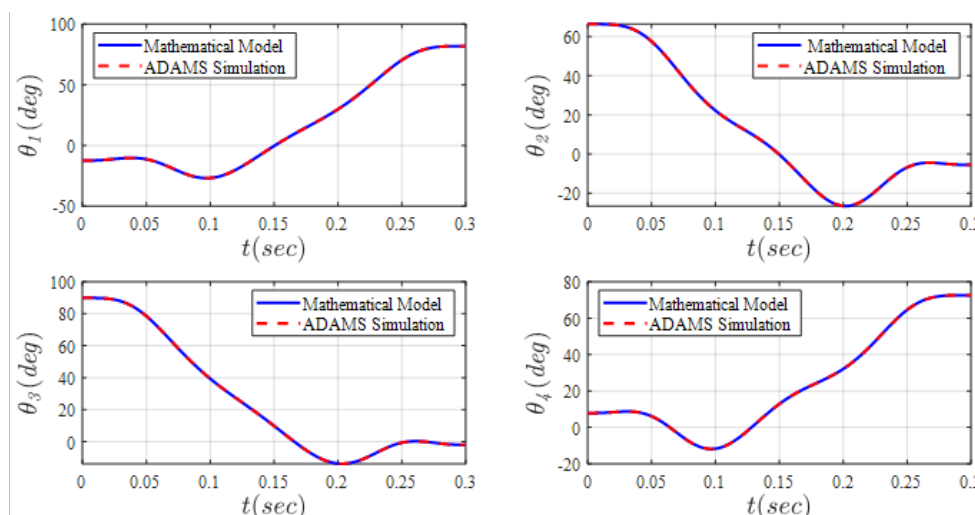
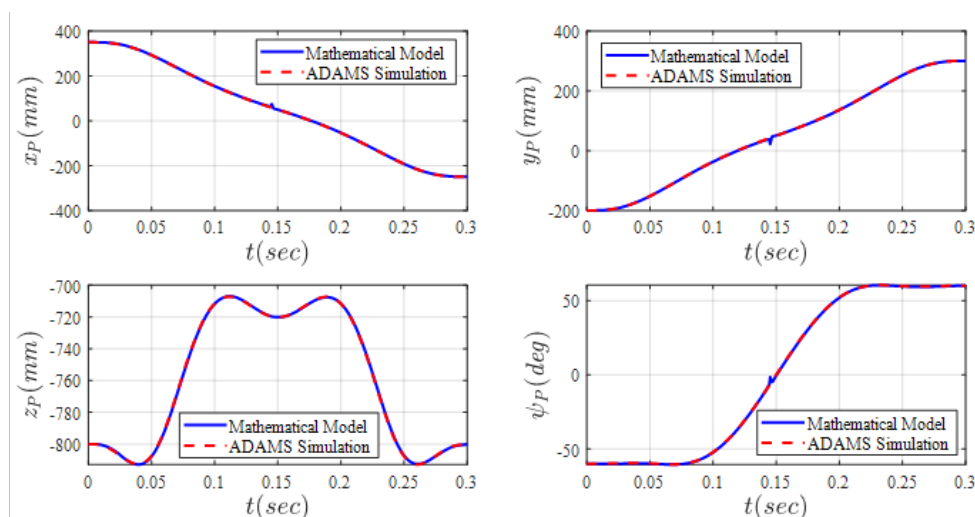**Figure 8** Joint space components by solving inverse kinematics.

Figure 9 Task space components by solving forward kinematics.

Conclusion

The purpose of this paper was to extract the kinematic and dynamic model of a Schoenefelis four-degree-of-freedom parallel robot. With three translational DoF and one DoF rotational, this robot is a good option for pick and place applications where high speed and acceleration are required. The studies began by extracting the kinematic equations based on the geometric method, and then velocity analysis. Using the obtained kinematic equations as well as joint constraints, the robot workspace was numerically determined. In the next step, the dynamic model was extracted using virtual work principle and Euler-Lagrange formulation for constrained systems, and the results showed the complete compliance. A smooth trajectory was designed for pick-and-place application by using B-spline. In order to validate the kinematic and dynamical equations, the robot was modeled in the ADAMS software, and the results of numerical solution of the aforementioned equations, were compared with the ADAMS output.

Conflicts of interest

The authors declare that there was no conflict of interest.

References

1. Yang G, Chen IM, Lin W, et al. Singularity analysis of three–legged parallel robots based on passive–joint velocities. *IEEE Transactions on Robotics and Automation*. 2001;17(4):413–422.

2. Y Wang. An Incremental Method for Forward Kinematics of Parallel Manipulators. Proceedings of IEEE International Conference on Robotics, Automation and Mechatronics; 2006 Jun 1-3; Thailand. Bangkok: IEEE; 2006.

3. Dai JS, Lipkin H, Huang Z. Mobility of overconstrained parallel mechanisms. *ASME Journal of Mechanical Design*. 2006;128(1):220–229.

4. R. Clavel. Une nouvelle structure de manipulateur parallèle pour la robotique légère. *APII*. 1989;23(6):501–519.

5. Pierrot F, Marquet F, Company O, et al. H4 parallel robot: modeling, design and preliminary experiments. Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation; 2001 May 21–26; South Korea: IEEE; 2001.

6. Krut S, Benoit M, Ota H, et al. I4: A new parallel mechanism for Scara motions. 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422); 2003 Sept 14-19; Taipei. Taiwan: IEEE; 2003.

7. Nabat V, Rodriguez O, Company O, et al. Par4: very high speed parallel robot for pick–and–place. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2005 Aug 2–6; Edmonton. Canada: IEEE; 2005.

8. Pierrot F, Nabat V, Company O, et al. Optimal Design of a 4–DOF Parallel Manipulator: From Academia to Industry. *IEEE Transactions on Robotics*. 2009;25(2):213–224.

9. Xie F, Liu X. Design and Development of a High–Speed and High–Rotation Robot With Four Identical Arms and a Single Platform. *Journal of Mechanisms and Robotics*. 2015;7(4):041015.

10. http://pentarobotics.com/products/#brochure.

11. Song Y, Gao H Sun T, et al. Kinematic analysis and optimal design of a novel 1T3R parallel manipulator with an articulated travelling plate. *Robotics and Computer–Integrated Manufacturing*. 2014;30(5):508–516.

12. A Novel Three–Loop Parallel Robot With Full Mobility: Kinematics, Singularity, Workspace, and Dexterity Analysis. *ASME Journal Mechanisms Robotics.* 2017;9(5):051003.

13. Mazare M, Taghizadeh M, Najafi M. Kinematic analysis and design of a 3–DOF translational parallel robot. *International Journal of Automation and Computing*. 2017;14(4):432–441.

14. Khalil W, Ibrahim O. General Solution for the Dynamic Modeling of Parallel Robots. *Journal of Intelligent and Robotic Systems*. 2007;49(1):19–37.

15. Shao ZF, Tang X, Wang L. Dynamics verification experiment of the Stewart parallel manipulator. International Journal of Advanced Robotic Systems. 2015;12(144):1–10.

16. Wang Z, Zhang N, Chai X, et al. Kinematic/dynamic analysis and optimization of a 2–URR–RRU parallel manipulator. *Nonlinear Dynamics*. 2017;88(1):503–519.

17. Abdellatif H, Heimann B. Computational efficient inverse dynamics of 6–DOF fully parallel manipulators by using the Lagrangian formalism. *Mechanism and Machine Theory*. 2009;44(1):192–207.

18. Tsai MS, Yuan WH. Dynamic Modeling and Decentralized Control of a 3 PRS Parallel Mechanism Based on Constrained Robotic Analysis. *Journal of Intelligent & Robotic Systems*. 2011;63(3–4):525–545.

19. Zhao Y, Gao F. Inverse dynamics of the 6–dof out–parallel manipulator by means of the principle of virtual work. *Robotica*. 2009;27(2):259–268.

20. Li Y, Xu Q. Dynamic modeling and robust control of a 3–PRC translational parallel kinematic machine. Robotics and Computer–Integrated Manufacturing. 2009;25(3):630–640.

21. Zhao Y, Zhang Z, Cheng G. Inverse rigid–body dynamic analysis for a 3UPS–PRU parallel robot. *Advances in Mechanical Engineering*. 2017;9(2):1–14.

22. Yang C, Huang Q, Jiang H,et al. PD control with gravity compensation for hydraulic 6–DOF parallel manipulator. *Mechanism and Machine Theory*. 2010;45(4):666–677.

23. Yang C, Huang Q, Han J, et al. Decoupling control for spatial six–degree–of–freedom electro–hydraulic parallel robot. *Robotics and Computer–Integrated Manufacturing*. 2012;28(1):14–23.

24. Miller K. Optimal Design and Modeling of Spatial Parallel Manipulators. *Journal of Robotics Research*. 2004;23(2):127–140.

25. Mazare M, Taghizadeh, Najafi MR. Inverse Dynamics of a 3–P[2(US)] Translational Parallel Robot. *Robotica*. 2018;1–21.

26. Li Y, Xu Q. Kinematics and inverse dynamics analysis for a general 3–PRS spatial parallel mechanism. *Robotica*. 2005;23(2):219–229.

27. Jadran L, Jean PM. Advances in Robot Kinematics 2016. Springer Proceedings in Advanced Robotics. 2016:119–128.

28. Pellicciari M, Berselli G, Leali F, et al. A method for reducing the energy consumption of pick–and–place industrial robots. *Mechatronics*. 2013;23(3):326–334.

29. Li H, Le MD, Gong ZM, et al. Motion profile design to reduce residual vibration of high–speed positioning stages. *IEEE/ASME Transactions on Mechatronics*. 2009;4(2):264–269.

30. Gallant A, Gosselin C. Extending the capabilities of robotic manipulators using trajectory optimization. *Mechanism and Machine Theory*. 2018;121:502–514.

31. Khoukhi A, Baron L, Balazinski M. Constrained multi–objective trajectory planning of parallel kinematic machines. *Robotics and Computer–Integrated Manufacturing*. 2009;25:756–769.

32. Gasparetto A, Zanotto V. A new method for smooth trajectory planning of robot manipulators. *Mechanism and Machine Theory*. 2007;42:455–471.

33. Li Y, Huang T, Chetwynd DG. An approach for smooth trajectory planning of high–speed pick–and–place parallel robots using quintic B–splines. *Mechanism and Machine Theory*. 2018;126:479–490.

34. Wu G. Kinematic analysis and optimal design of a wall–mounted four–limb parallel Schönflies–motion robot for pick–and–place operations. *Journal of Intelligent & Robotic Systems*. 2017;85:663–677.

35. Piegl W, Tiller W. The NURBS Book. Springer–Verlag: New York; 1997.

36. Barre PJ, Bearee R, Borne P, et al. Influence of a Jerk Controlled Movement Law on the Vibratory Behaviour of High–Dynamics Systems. *Journal of Intelligent and Robotic Systems*. 2005;42(3):275–293.