MedCrave
Step into the World of Research

International Robotics & Automation Journal

Research Article

Open Access

CrossMark

Adaptive back–stepping robust control of a 3–[P2(US)] parallel robot on optimal trajectory

Abstract

This paper presents a robust control technique combining adaptive sliding mode control and back–stepping methods to control a 3–DOF translational parallel robot. Dynamic modeling of the robot is accomplished using the Lagrange method. Since the derived dynamic model suffers from uncertainties and disturbances, a combination of robust techniques namely back–stepping and sliding mode controllers are applied together with adaptive control. Determination of uncertainty bounds is intransitive in the design of the proposed controller. To prove the system stability, the system model is divided into several subsystems and stability is illustrated step by step using Lyapunov criteria, then an adaptive law is obtained to estimate the uncertainties. This controller is also robust in presence of time–varying disturbances. Performance of the controller is evaluated by simulations on optimal trajectories. Trajectory planning is performed by optimization of some accuracy points using Harmony Search Algorithm (HSA) and cubic spline interpolation. Simulation results show improved performance of the proposed control technique compared to conventional feedback linearization and sliding mode techniques.

Keywords: 3–[P–2(US)] parallel robot, harmony search algorithm, adaptive back–stepping control, trajectory planning

Volume 5 Issue 3 - 2019

Mahmood Mazare, Mostafa Taghizadeh

School of Mechanical and Energy Engineering, Shahid Beheshti University, Iran

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

Received: January 06, 2019 | **Published:** May 13, 2019

Introduction

Parallel robots are widely used in various fields of engineering and industrial applications such as, machine tools, flight simulators, earthquake simulators, medical equipment and etc. However, they have some limitations such as small and complex workspace, combination of translational and rotational motions and clearance in joints. They are also restricted by other limitations such as existence of singular points which should be studied and analysed for effective and efficient use. Therefore, their complex dynamics and forward kinematics require a very sophisticated real time control.[1–3]

Since this type of robotic manipulators have a complex dynamic model, many different approaches have been applied to control them which can be classified based on the controller design methods. In this regard, classical controllers such as simple.[4] PID controllers which are the most commonly used in industrial applications, do not need complete dynamic information of the robot. Although, these controllers usually show desirable performance in tracking of plants with linear time–invariant models, however, for robots with nonlinear dynamics, more advanced techniques such as Lyapunov based methods[5,6] or computed torque control (CTC)[7,8] improve the control performance. These methods require complete dynamic model of the robot, while the exact model is unreachable due to uncertainties. This problem restricts the performance of these nonlinear controllers. The deficiency caused by uncertainties can be compensated by adaptive back–stepping sliding mode controller which is proposed in this paper.

Sliding mode control is one of the most applicable nonlinear control strategies[9,10] which counteracts the effects of the dynamic model uncertainties. The main reason why sliding mode controller is so widely used in control of nonlinear systems is the capability of solving the two most challenging issues, stability and robustness.[11,12] In order to reduce the effect of uncertainties which exist in dynamic model of parallel manipulator, a sliding mode controller is designed in.[13] Gracia et al.[14] proposed an integrated solution based on sliding mode for trajectory tracking of a 6R robot model PUMA–560. They presented three sliding–mode algorithms for speed auto–regulation, path conditioning and redundancy resolution and finally evaluated their approach by simulations. Recently, many researchers have been characterized on designing and improving control of the parallel robots with uncertainties. Robust control based on Lyapunov's control is the principle of most of robust and nonlinear control methods, adaptive control, sliding mode control, and combination of sliding mode and adaptive controls.[15,16] Based on the standard nonlinear adaptive methods, the controller tries to obtain specific structurally developed dynamic variable parameters; this might achieve an acceptable tracking performance and cover structuralized uncertainties and limited disturbances.[17] Consequently, these factors might affect nonlinear adaptive controllers when dynamic model of the robot is not clearly known or when fast real–time control is needed.[3–5]

Combined controllers of sliding mode and adaptive controls have been studied as a solution to overcome the problem of adaptive control and sliding mode. The main idea is to use adaptive control to estimate unknown parameters of the dynamic system and sliding mode control is used to overcome non–modeled dynamics and external disturbances.[18,19] However, adaptive combined control needs a parameterized liner model of the system that is analyzed and prior knowledge about the bound of uncertainty. In addition, a large number of parameters and an adaptation gain (e.g. designing parameter) corresponding with each parameter demonstrate the extent of complicacy. The problem of a robust and adaptive controller without knowledge of the bounds of uncertainties is addressed in,[20] where a parameterized linear model of the system is needed to design the controller. To control robot tracking in presence of uncertainties of the model and time–variable external disturbances, Notash et al.[21] introduced a robust adaptive controller based on fuzzy logic model using neural network as a regulator tool. The controller was designed based on theoretical sliding mode control and stability of the controller was analyzed based on Lyapunov's theory. Their findings supported very high performance and acceptable robustness in presence of disturbances. Back–stepping technique is only used to control

strict feedback systems without considering the uncertain factors in practice,[22] and its application is severely restricted. To gain adequate performances of control methods, many researchers have interested to the sliding mode control based on back–stepping and applied in different fields, such as chaos synchronization,[22,23] quadrotor,[24,25] motor drive,[26] tracking control[27] and etc.

In this paper, kinematic and dynamic modeling of a 3–[P–2(US)] parallel mechanism[28,29] is investigated and equations of motion are derived via Lagrange formulation. Then, optimal trajectory planning is done based on Harmony Search Algorithm (HSA) and two trajectories are obtained by spline interpolation of optimal accuracy points in an area containing obstacles. An adaptive back–stepping sliding mode control combining both the advantages of adaptive back–stepping control and sliding mode control is used to achieve accurate control of parallel robot. In order to design of this controller, a back–stepping sliding mode controller is derived supposing that the bound of the uncertainty is well–known. Then an adaptive approach is used to estimate the uncertainties and disturbances of the derived dynamic model of the parallel robot. A Lyapunov function is used to prove the stability of the proposed controller. The proposed controller guarantees closed loop stability in spite of nonlinearities of the parallel robot. Compared with the back–stepping control scheme,[30] the proposed controller has the advantages of both adaptive method and robust control, which in presence of uncertainty, nonlinearity and external disturbances have a good performance. In order to overcome the existing uncertainty in this model, adaptive method is used. The main idea of back–stepping approach is to choose recursively some proper functions of state variables as fictional control inputs for lower dimension subsystems of the total system.[31]

The rest of this paper is organized as follows. In section 2, the configuration of the robot is investigated. Kinematic and dynamic modeling is depicted in section 3. In section 4, adaptive back–stepping sliding mode controller is designed. Section 5 deals with optimal path planning. Simulation results of controller implementation are reported in sections 6 and finally, conclusion is drawn in section 7.

Robot configuration

The studied mechanism constitutes of a bottom mobile platform named end–effector, a fixed platform and three limbs. Each limb is connected to the fixed base by a prismatic joint. Prismatic joints, which are employed as actuators in this mechanism, are connected to the base from one side and connected to end–effector from other side by parallel connecting rods. In Figure 1, the assembled robot is demonstrated. Schematic presentation of the robot mechanism is depicted in Figure 2. On each limb, two universal joints are connected to two spherical joints by two connecting rods. Joints of this robot are designed in a way that only three translational motions are possible for the end–effector. Consequently, in order to prevent rotational motion of end–effector, parallelogram linkages are constructed using universal and spherical joints, thus, this robot is called 3–[P2–(US)].[28,29] The result was three translational DOF for the mechanism.

Kinematic and dynamic modeling

Inverse Kinematics

In the inverse kinematics, the goal is obtaining the actuators positions, given the end–effector position with respect to the fixed platform frame. Figure 3 demonstrates kinematic variables of a robot limb in which the end–effector position P is described in fixed platform (Frame O).

Referring to Figure 3, the following vector relation can be written

$$L_i = q_i e_{A_iB_i} + l e_{B_iC_i} \quad (i = 1,2,3) \tag{1}$$

Where $L_i = \overrightarrow{A_iC_i}$, $\mathbf{e}_{A_iB_i}$ and $\mathbf{e}_{B_iC_i}$ are the unit vectors along A_iB_i and B_iC_i, respectively, and q_i represents the linear displacement of the $i–th$ actuator.

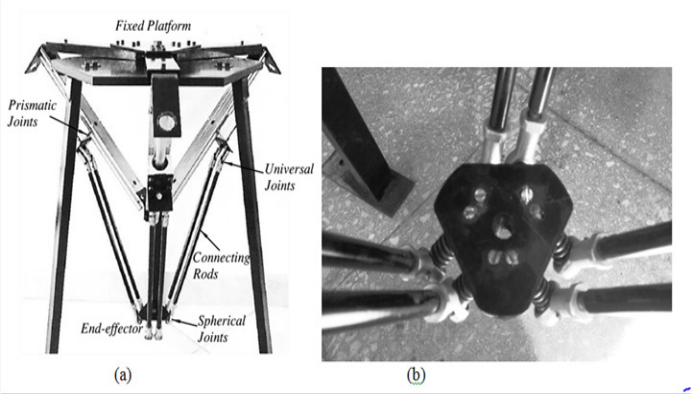

Figure 1 (a) The assembled mechanism and (b) End–effector.

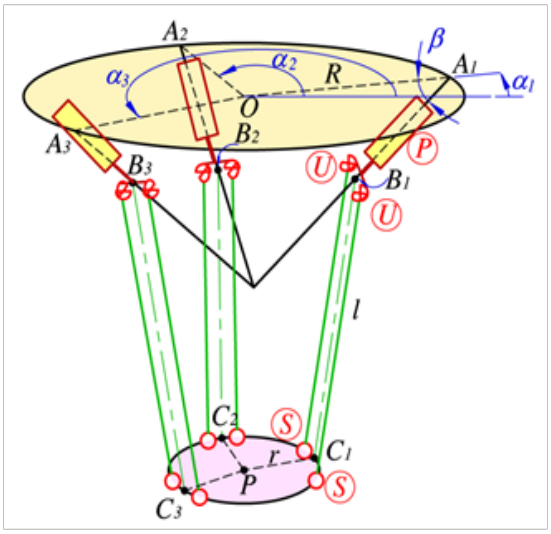

Figure 2 Geometry the 3–[P2(US)] mechanism.

Through rewriting Eq. (1),

$$\left|L_i - q_i e_{A_iB_i}\right| = \left|l e_{B_iC_i}\right| \tag{2}$$

After some mathematic simplification, the constraint equation is derived as:

$$\Gamma_i = (x - R\sin\alpha_i + r\sin\alpha_i + q_i\cos\beta\sin\alpha_i)^2$$
$$+(y + R\cos\alpha_i - r\cos\alpha_i - q_i\cos\beta\cos\alpha_i)^2 \tag{3}$$
$$+(z + q_i\sin\beta)^2 - l^2 = 0 \quad (i=1,2,3)$$

Lagrange equations

In this section, the dynamics of the robot is investigated and the dynamic model of the mechanism is derived based on the kinematic modeling which was presented in previous section. In dynamic modelling, the mass of connecting link is considered as a mass point and so waiver the joint inertia, energy dissipation and link flexibility. It is supposed that the un–modeled dynamic is evident as a disturbance

in exact model. To drive the dynamic model of the robot, the Lagrange method is used. Since there are constraints on generalized coordinates in the parallel mechanism the general formulation of constrained Lagrange equations with Lagrange multipliers is considered as follows

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}_j}\right)-\frac{\partial L}{\partial \theta_j}=Q_j+\sum_{i=1}^{k}\lambda_i\frac{\partial \Gamma_i}{\partial \theta_j}\left(j=1,\ldots,6\right) \tag{4}$$

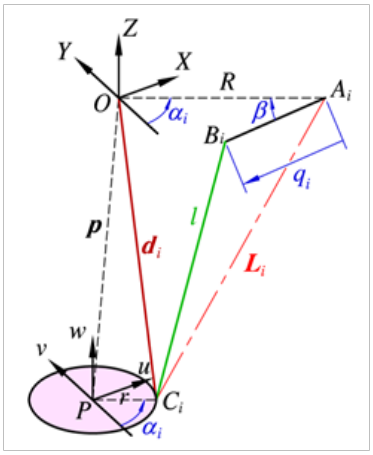

Figure 3 Kinematic variables of one of the limbs.

Where θ_j the *j–th* generalized coordinate and Q_j is the corresponding generalized force. Also, λ_i, Γ_i and k indicate the Lagrange multipliers, constraint functions and the number of system constraints, respectively. Generalized coordinates to describe the system are

$$\theta=\begin{bmatrix}\mathrm{q}\\ \mathrm{p}\end{bmatrix} \tag{5}$$

Where q includes the positions of actuators (prismatic joint variables) and p includes end–effector position components. The general formulation of Lagrangian function is written as

$$\mathrm{L}(\theta,\dot{\theta})=\mathrm{K}(\theta,\dot{\theta})-\mathrm{U}(\theta) \tag{6}$$

Where, K and U are the kinetic and potential energies of the robot, respectively. The kinetic energy term of the mechanism can be written as

$$K=\frac{1}{2}m_1(\dot{q}_1^{\,2}+\dot{q}_2^{\,2}+\dot{q}_3^{\,2})+\frac{1}{2}m_2(\dot{x}^2+\dot{y}^2+\dot{z}^2)$$

$$m_1=m_p+\frac{m_l}{2}\quad m_2=m_e+3\left(\frac{m_l}{2}\right) \tag{7}$$

where, m_e, m_p and m_p are the masses of end–effector, two connecting rods, and actuators piston, respectively. Thus four concentrated moving masses constitute the kinetic energy (three masses of m_1 associated to sliding parts of actuators and one mass of m_2 associated to moving platform). According to the Figure 4, it is supposed that the masses of six connecting rods are equal and concentrated at rod ends, neglecting their rotational inertia. (Table 1)

Now, the potential energy of the mechanism can be written as

$$U=-m_1 g\sin\beta(q_1+q_2+q_3)+m_2 gz \tag{8}$$

Where, β is angle of prismatic joint direction angle. Substituting Eq. (7), and (8) in Eq. (6) will result Lagrangian function as follows

$$L=\frac{1}{2}m_1(\dot{q}_1^{\,2}+\dot{q}_2^{\,2}+\dot{q}_3^{\,2})+\frac{1}{2}m_2(\dot{x}^2+\dot{y}^2+\dot{z}^2)$$
$$-\left[m_2 gz-m_1 g\sin\beta(q_1+q_2+q_3)\right] \tag{9}$$

Table 1 Geometric parameters

Value	Parameters
α_i	$(\mathrm{i}-1)\times 120^\circ$
β	40°
r	$28mm$
R	$325mm$
l	$340mm$
m_p , m_e	0.5 kg
m_l	0.2 kg

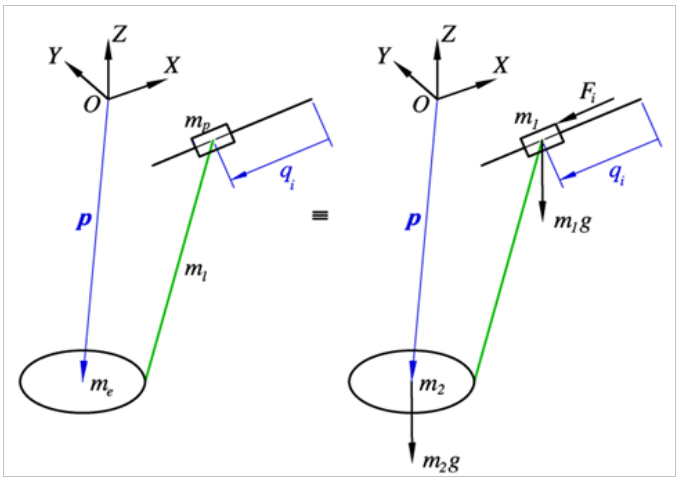

Figure 4 Mass elements of one of the limbs.

By taking partial derivatives of constraint equations used in Lagrange equations and substituting in Lagrange equation and then some simplification,[32] matrix form of equations of motion is extracted as follows:

$$\mathrm{M}(\theta)\ddot{\theta}+\mathrm{C}(\theta,\dot{\theta})\dot{\theta}+\mathrm{G}(\theta)=\bar{\mathrm{F}} \tag{10}$$

Partitioned form of the equations is written as

$$\begin{bmatrix}\mathrm{M_{qq}} & \mathrm{M_{pq}}\\ \mathrm{M_{qp}} & \mathrm{M_{pp}}\end{bmatrix}\begin{bmatrix}\ddot{\mathrm{q}}\\ \ddot{\mathrm{p}}\end{bmatrix}+\begin{bmatrix}\mathrm{C_{qq}} & \mathrm{C_{pq}}\\ \mathrm{C_{qp}} & \mathrm{C_{pp}}\end{bmatrix}\begin{bmatrix}\dot{\mathrm{q}}\\ \dot{\mathrm{p}}\end{bmatrix}+\begin{bmatrix}\mathrm{G_q}\\ \mathrm{G_p}\end{bmatrix}=\begin{bmatrix}\mathrm{F}\\ 0\end{bmatrix} \tag{11}$$

So,

$$\mathrm{M_{qq}}\ddot{\mathrm{q}}+\mathrm{M_{pq}}\ddot{\mathrm{p}}+\mathrm{C_{qq}}\dot{\mathrm{q}}+\mathrm{C_{pq}}\dot{\mathrm{p}}+\mathrm{G_q}=\mathrm{F}$$

$$\mathrm{M_{qp}}\ddot{\mathrm{q}}+\mathrm{M_{pp}}\ddot{\mathrm{p}}+\mathrm{C_{qp}}\dot{\mathrm{q}}+\mathrm{C_{pp}}\dot{\mathrm{p}}+\mathrm{G_p}=0 \tag{12}$$

Eq. (12) is actually forward kinematics of acceleration and can be used in inverse dynamics module of control loop as follows

$$\ddot{\mathrm{p}}=-\mathrm{M_{pp}}^{-1}(\mathrm{M_{qp}}\ddot{\mathrm{q}}+\mathrm{C_{qp}}\dot{\mathrm{q}}+\mathrm{C_{pp}}\dot{\mathrm{p}}+\mathrm{G_p}) \tag{13}$$

Controller design

In this section three controller is designed which is introduced in the following.

Adaptive back–stepping sliding mode control

Referring to the derived dynamic model of the proposed mechanism and defining $X_1 = \theta, X_2 = \dot{\theta}, X = \left[X_1^T, X_2^T\right]^T$, state equations of the robot can be written as:

$$\begin{cases} \dot{X}_1 = X_2 \\ \dot{X}_2 = M^{-1}\left[F - CX_2 - G - D\right] \end{cases} \quad (14)$$

Considering q_d as the desired trajectory of the parallel robot, the trajectory tracking error is

$$e_p = q - q_d \quad (15)$$

Virtual control variables for the parallel robot are defined as follows:

$$\sigma = \varepsilon e_p \quad (16)$$

where ε is a positive definite diagonal coefficient matrix. The velocity error of the system is defined as

$$e_v = \dot{e}_p + \sigma = \dot{q} - \dot{q}_d + \varepsilon e_p \quad (17)$$

In order to show the stability of the method, a Lyapunov function is found as follows:

$$V_1(t) = \frac{1}{2} e_p{}^T e_p \quad (18)$$

Differentiating Eq. (18) with respect to time

$$\dot{V}_1(t) = e_p{}^T \dot{e}_p = e_p{}^T (e_v - \varepsilon e_p) = e_p{}^T e_v - e_p{}^T \varepsilon e_p \quad (19)$$

If $e_v = 0$, then $\dot{V}_1(t) \le 0$, therefore the first subsystem of the parallel robot is stable. Differentiating Eq. (17) with respect to time gives:

$$\dot{e}_v = \ddot{e}_p + \dot{\sigma} = \ddot{q} - \ddot{q}_d + \dot{\sigma} \quad (20)$$

To indicate stability of the system, a second Lyapunov function is chosen as follows:

$$V_2(t) = \frac{1}{2}\left(e_p{}^T e_p + S^T S\right) \quad (21)$$

Where

$$S = \Lambda e_p + e_v \quad (22)$$

Where Λ is a positive definite diagonal coefficient matrix. So, the derivation of Eq. (21) is:

$$\dot{V}_2(t) = e_p{}^T \dot{e}_v - e_p{}^T \varepsilon e_p + S^T\left[\Lambda(e_v - \varepsilon e_p) + M_{qq}{}^{-1}(F - M_{pq}\ddot{p} - C_{qq}\dot{q} - C_{pq}\dot{p} - G_q - D) - \ddot{q}_d + \varepsilon \dot{e}_p\right] \quad (23)$$

To satisfy $\dot{V}_2(t) \le 0$, and stabilize the second sub–system, the controller can be designed as:

$$F = M_{qq}\left[-\Lambda(e_v - \varepsilon e_p) + \ddot{q}_d - \varepsilon \dot{e}_p - bS\right] + M_{pq}\ddot{p} + C_{qq}\dot{q} + C_{pq}\dot{p} + G_q + \hat{D} - \gamma M_{qq}\, b\tanh\left(S/\tau\right) \quad (24)$$

In a real–time control system, it is difficult to predict the general uncertainty **D**. To avoid adopting the upper bounds of **D**, an adaptive algorithm is applied based on back–stepping sliding mode control. By defining:

$$\tilde{D} = D - \hat{D} \quad (25)$$

Where $\hat{D}$ is the estimation of the general uncertainty D and $\tilde{D}$ is the estimation error. Then

$$\dot{\tilde{D}} = \dot{D} - \dot{\hat{D}} = -\dot{\hat{D}} \quad (26)$$

The third Lyapunov function is defined as

$$V_3(t) = \frac{1}{2}\left(S^T S + \zeta^{-1}\tilde{D}^T \tilde{D}\right) \quad (27)$$

Where ζ is a positive constant. By taking time derivative of $V_3(t)$ and substituting Eq.(24) in it, $\dot{V}_3(t)$ is as follows:

$$\dot{V}_3(t) = e_p{}^T \dot{e}_v - e_p{}^T \varepsilon e_p + S^T\left[\Lambda(e_v - \varepsilon e_p) + M_{qq}{}^{-1}(F - M_{pq}\ddot{p} - C_{qq}\dot{q} - C_{pq}\dot{p} - G_q - D) - \ddot{q}_d + \varepsilon \dot{e}_p\right] - \zeta^{-1}\tilde{D}^T\left(\dot{\hat{D}} + \zeta M_{qq}{}^{-T} S\right) \quad (28)$$

Due to $\dot{V}_3(t)$ should be negative, consider the adaptation law as follows:

$$\dot{\hat{D}} = -\zeta M_{qq}{}^{-T} S \quad (29)$$

Then, the adaptive back–stepping sliding mode control law is derived as:

$$F = M_{qq}\left[-\Lambda(e_v - \varepsilon e_p) + \ddot{q}_d - \varepsilon \dot{e}_p - bS\right] + M_{pq}\ddot{p} + C_{qq}\dot{q} + C_{pq}\dot{p} + G_q + \hat{D} - \gamma M_{qq}\, b\tanh\left(S/\tau\right) \quad (30)$$

Where γ is a positive definite constant and b is a positive definite diagonal coefficient matrix. To avoid the enhanced chattering phenomenon, a hyperbolic tangent function is chosen in which τ is an adjustable parameter. Substituting (29) and (30) in (28), the derivative of $V_3(t)$ is as follows:

$$\dot{V}_3(t) = e_p{}^T e_v - e_p{}^T \varepsilon e_p - S^T bS - \gamma S^T b\tanh\left(S/\tau\right) \le e_p{}^T e_v - e_p{}^T \varepsilon e_p - S^T bS \quad (31)$$

Where

$$S^T \tanh\left(S/\hat{o}\right) \text{ ³ } 0 \ , \ b > 0 \ , \ \gamma > 0 \ , \ \tau > 0 \tag{32}$$

Introducing a matrix as

$$\varphi = \begin{bmatrix} \varepsilon + \Lambda^T b\Lambda & b\Lambda - 0.5I \\ b\Lambda - 0.5I & b \end{bmatrix} \tag{33}$$

Then,

$$\left\| \ddot{o} \right\| = b(\mathring{a} + \ddot{E}) - 0.25 \tag{34}$$

By selecting the suitable parameters b, ε and $\ddot{E}$ to satisfy $\|\varphi\| > 0$, matrix φ is regard as the positive definite matrix. Due to

$$e^T \varphi e = \begin{bmatrix} e_p & e_v \end{bmatrix} \begin{bmatrix} \varepsilon + \Lambda^T b\Lambda & b\Lambda - 0.5I \\ b\Lambda - 0.5I & b \end{bmatrix} \begin{bmatrix} e_p \\ e_v \end{bmatrix}$$

$$= e_p{}^T \varepsilon e_p - e_p{}^T e_v + (\Lambda e_p + e_v)^T b(\Lambda e_p + e_v) = e_p{}^T \varepsilon e_p - e_p{}^T e_v + S^T bS \tag{35}$$

The Eq. (31) is rewritten as

$$\dot{V}_3(t) \le -e^T \varphi e \tag{36}$$

Considering suitable values for b, ε and $\ddot{E}$ then φ can be a positive definite matrix and Eq. (36) satisfies that $\dot{V}_3(t) \le 0$

Therefore, the globally exponential asymptotic stability of the proposed parallel robot is guaranteed. Block diagram of the proposed controller is depicted in Figure 5.

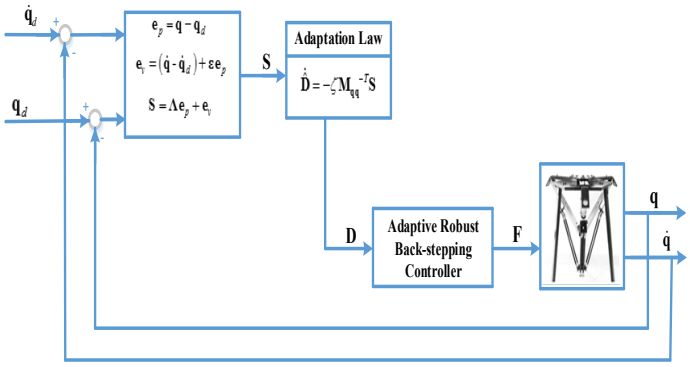

Figure 5 Block–diagram of adaptive back–stepping sliding mode control.

Inverse dynamics feedback linearization controller

Feedback linearization technique using inverse dynamics is implemented in contouring control of the proposed mechanism. Joint space variables q and $\dot{q}$ are measured and taken as feedback signals to the controller. The following control law is applied in the controller.

$$u = \ddot{q}_d - k_d \dot{e} - k_p e \tag{37}$$

Substituting $\ddot{q} = u$ in the inverse dynamics equations, the computed actuator forces can be determined. Block diagram of the applied control loop is illustrated in Figure 5. This control action leads to error dynamics in the form of

$$\ddot{e} + k_d \dot{e} + k_p e = 0 \tag{38}$$

Elements of the two matrices k_p and k_d should be chosen that the linear time–invariant error dynamics system is stable. There exists exponential convergence in the system if k_p and k_d re positive definite, symmetric matrices. Tuned values of the control parameters k_d and k_d is required for the close loop system to have desired performance in transient and steady–state responses. (Figure 6)

Sliding mode controller

A single input–single output nonlinear system can be defined as

$$q^{(n)} = f(q, t) + b(q, t)F(t) \tag{39}$$

Where, $q(t)$ is the state vector, $F(t)$ is the control input (in our case braking torque or pressure on the pedal) and q is the output state of the interest, $f(q,t)$ and $b(q,t)$ are generally nonlinear functions of time and states. The control problem is to get the state q to track a specific time–varying state $q_d(t)$ in the presence of model imprecision on $f(q,t)$ and $b(q,t)$.Considering the sliding surface equation of the form

$$S(x,t) = \left(\frac{d}{dt} + \Lambda\right)^2 \left(\int_0^t \tilde{q} dt\right) = \left(\frac{d^2}{dt^2} + 2\Lambda \frac{d}{dt} + \Lambda^2\right) \left(\int_0^t e dt\right) = \dot{e} + 2\Lambda e + \Lambda^2 \int_0^t e dt \tag{40}$$

Differentiation of the sliding variable yields

$$\dot{S} = \left(\ddot{q} - \ddot{q}_d\right) + 2\Lambda\left(\dot{q} - \dot{q}_d\right) + \Lambda^2\left(q - q_d\right) \tag{41}$$

Consider a simple second order system

$$\ddot{q} = f(q,t) + F(t) \rightarrow \dot{S} = f + F - \ddot{q}_d + 2\Lambda\left(\dot{q} - \dot{q}_d\right) + \Lambda^2\left(q - q_d\right) \tag{42}$$

The approximation of control law $\hat{F}(t)$ to achieve $\dot{S} = 0$ is

$$\hat{F}(t) = -\hat{f} + \ddot{q}_d - 2\Lambda\left(\dot{q} - \dot{q}_d\right) - \Lambda^2\left(q - q_d\right) \tag{43}$$

Which a simple solution for determining the sliding surface conditions, when the system parameters are uncertain, is the switching control law

$$F(t) = \hat{F}(t) - b\tanh\left(S/\tau\right) \tag{44}$$

Where K is a positive definite diagonal matrix. After some simplification, the Sliding mode control law for parallel robot is obtained as follow:

$$F_{sliding} = M_{qq}(\ddot{q}_d - 2\Lambda\dot{e} - \Lambda^2 e) + C_{qq}\dot{q} + G_q - M_{qq}.b\tanh\left(S/\tau\right) \tag{45}$$

Optimal path planning

In this section, the objective of optimal path planning is to design a trajectory for the end–effector path in an area containing some obstacles so that the trajectory is of minimum length and avoiding any collision to the obstacles. A predetermined margin is considered around the obstacles that the end–effector would keep a minimum distance with the borders of the obstacles.

To create a 2–D trajectory from starting point $P_0 = (X_0, Y_0)$ to final point $P_f = (X_f, Y_f)$ in the time interval of $[T_0, T_f]$, a number of accuracy points are considered as:

$$P_i = (X_i, Y_i) at\, T_i \ (i = 1, 2, ..., n) \quad (46)$$

Where T_i is the time instant at which the end–effector is planned to be located at X_i. All accuracy points will be checked to be placed in the reachable workspace and far enough from singular points. End–effector trajectory is derived by cubic spline interpolation of accuracy points coordinates X_i and Y_i with respect to time that guarantees the continuity of velocity and acceleration. So, the trajectory curve can be obtained as,

$$x_j = spline(X, T, t_j) y_j = spline(Y, T, t_j) \ (j = 1, 2, ..., N) \quad (47)$$

Where, N is the number of interpolated points along the curve. Spatial and time vectors are defined as,

$$X = \left[X_0 X_1 X_2 ... X_n X_f \right] Y = \left[Y_0 Y_1 Y_2 ... Y_n Y_f \right] T = \left[T_0 T_1 T_2 ... T_n T_f \right] \quad (48)$$

The optimization problem is to obtain 2n variables consisting of X_i and Y_i such that the interpolated trajectory has minimum length from P_0 to P_f and keeping the predefined margin from the obstacles. So, an objective function is considered for the optimization problem as the following form to be minimized:

$$ObjFunc = \alpha C + \beta L \quad (49)$$

Where L and C denote the trajectory length and the index of collision constraints, respectively, and the coefficients α and β are the corresponding weights. The length of the trajectory is calculated by:

$$L = \sum_{j=2}^{N} \sqrt{(x_j - x_{j-1})^2 + (y_j - y_{j-1})^2} \quad (50)$$

In this research, the obstacles are considered as some circular objects. So, the index of collision constraints can be obtained as follows:

$$d_{j,k} = \sqrt{(x_j - x_{C_k})^2 + (y_j - y_{C_k})^2}\, j = 1, ..., N k = 1, ...n_c$$

$$\Delta_{j,k} = \max\left(1 - \frac{d_{j,k}}{R_k + m}, 0\right) C_k = mean(\Delta_{j,k}) C = \sum_{k=1}^{n_c} C_k \quad (51)$$

Where n_c is the number of circular obstacles, (X_{C_k}, Y_{C_k}) and R_k are the center coordinates and the radius of the *k–th* obstacle, and m is the margin, respectively. The margin is considered in order to avoid the tool attached to the end–effector to collide with the obstacles. In this paper, we assume that the obstacles have been placed lower than parallel mechanism and there is no possibility for collisions of robot links with the obstacles. (Figure 7)

Same problem definition can be done for 3–D trajectory planning by introducing third coordinate *Z*, and solving the optimization problem with *3n* variables to obtain the shortest collision–free trajectory with, for example, spherical obstacles. A harmony search algorithm is utilized for solving the optimization problem which has advantages over some other algorithms such as GA.

Simulation result

In this section, simulations are performed to illustrate the effectiveness of the proposed adaptive back–stepping sliding mode controller (ABS) in tracking the end–effector position. For this purpose, two trajectories are designed as a reference path and are optimized using HSA in presence of obstacles. The controller is then implemented to the robot over the designed trajectories and simulation results are obtained. Also, to indicate the robustness of the designed ABS controller, parametric uncertainties and time–varying disturbances are considered in all simulations. In order to show the outperforming of the proposed controller, their results are compared with two other controllers, a feedback linearization controller (FL) and a conventional sliding mode controller (SM). Optimal reference trajectories are tracked by the robot end–effector and discussed separately in the following. Control parameters are obtained based on harmony search algorithm. To do this, cost function which is considered is as follows:

$$Obj = ITAE + |\dot{u}| \quad (52)$$

Optimal parameters are depicted in Table 2.

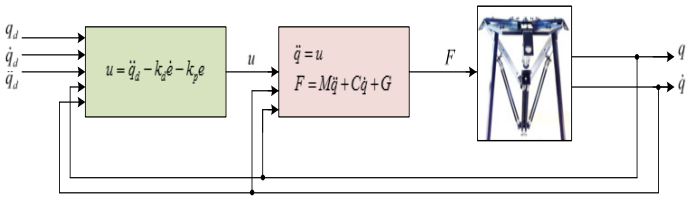

Figure 6 Block–diagram of feedback linearization scheme.

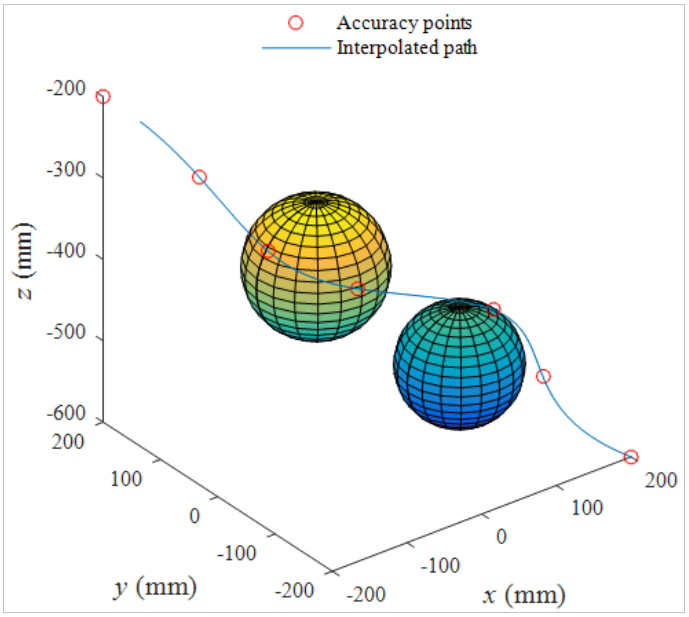

Figure 7 Optimal trajectory generated by HAS.

Test case 1: The first trajectory is depicted in Figure 8 and tracking performances of ABS, SM, and FL controllers are shown in Figure 9, Figure 10.

As it can be seen in above figures, the proposed ABS controller provides a more reasonable tracking performance compared to the SM and FL controllers. The tracked trajectory of the ABS controller converges to the reference trajectory with a faster rate and less fluctuations than that of the SM and FL controllers. Figure 6 compares the control signal of the SM and ABS controllers which are more robust than the FL controller. As is obvious, the range of control effort in both controllers are approximately equal, however the control signal of the SM controller has larger values and intense fluctuations. Tracking errors are also indicated in Figure 10 which shows the unacceptable performance of the FL controller. The RMS

of error in the SM is 2.354mm, while in the ABS method, this value is reducing to 0.7007mm.

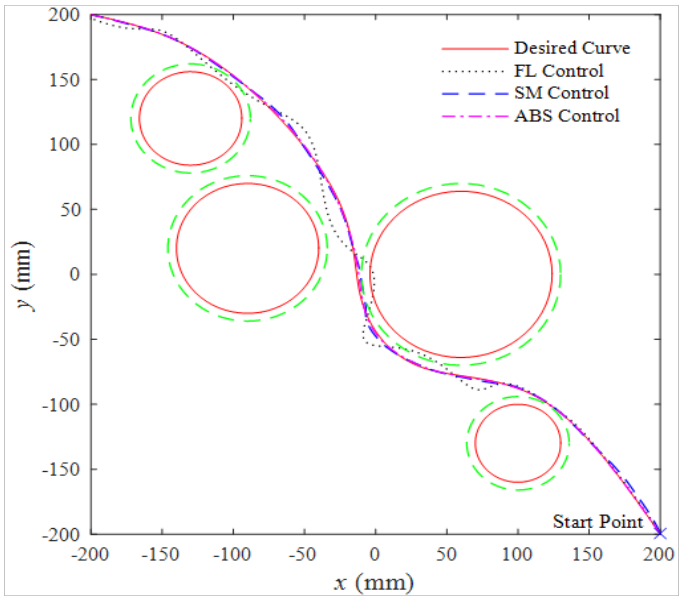

Figure 8 End–effector trajectory in the maneuver 1.

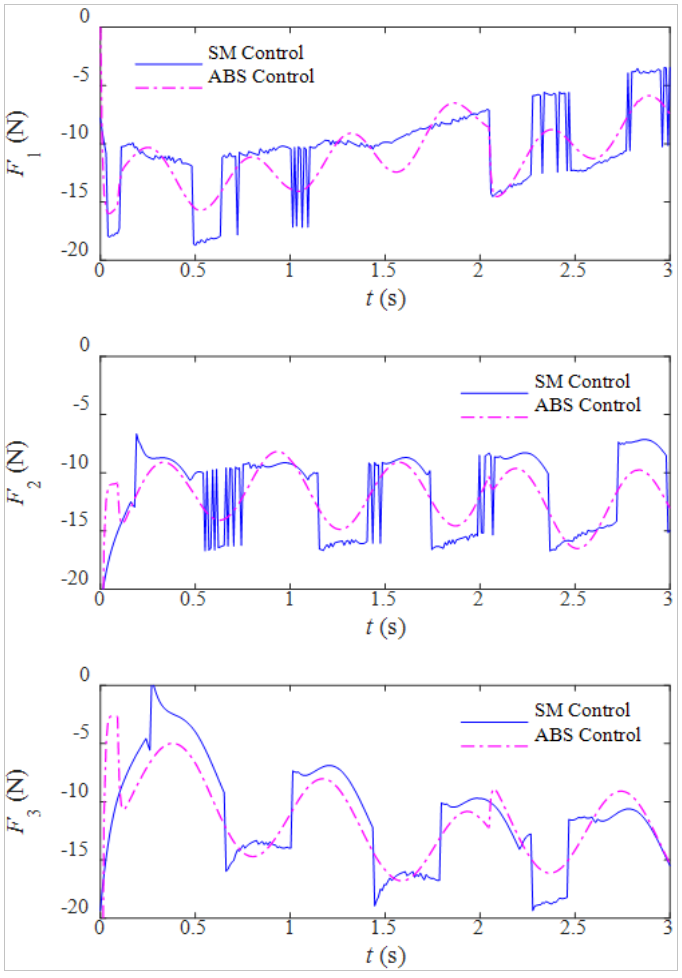

Figure 9 Control signals of proposed controller.

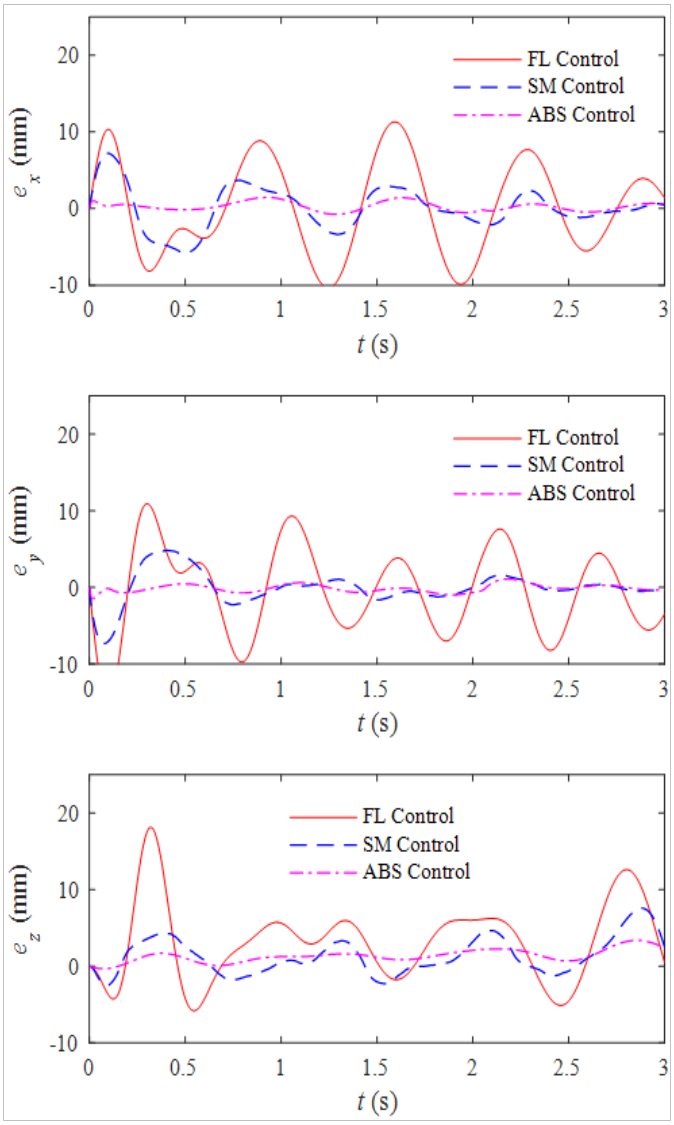

Figure 10 Error components of end–effector position in the maneuver 1.

To demonstrate the performance of the proposed controller, three criteria named as the Integral of the Time multiplied by the Absolute value of the Error (ITAE), the Integral of the Time multiplied by the Absolute value of the Squared of the Error (ITASE) and the Integral of the Absolute value of the Error (IAE) for different controller have been investigated. Three error criteria including *ITAE*, *ITASE* and *IASE* are considered which are defined in the following equations:

$$ITAE = \int \left| e(t).t \right| dt$$

$$ITASE = \int t.\left| e(t) \right|^2 dt \qquad (53)$$

$$IASE = \int \left| e(t) \right|^2 dt$$

Cost function may be chosen one of the defined error criteria in the paper as Eq. (53). Table 3 compares the results with different criteria. As can be seen, the results of ABS are significantly better than others.

In order to show robustness of the proposed controllers, standard deviation (STD) of error is extracted in presence of different uncertainties and depicted in Table 4.

Citation: Mazare M, Taghizadeh M. Adaptive back–stepping robust control of a 3–[P2(US)] parallel robot on optimal trajectory. *Int Rob Auto J.* 2019;5(3):101–110. DOI: 10.15406/iratj.2019.05.00181

Table 2 optimal parameters of controllers

k_p	k_d	ζ	Λ	b	b	γ
127.3752	4.9863	0.0512	3.2843	2.9827	1.9789	1.1258

Table 3 Different criteria for comparing the performance of the controllers

Controller	ITAE	IASE	ITASE
FL	0.075386	0.000957	0.001339
SM	0.048203	0.000408	0.000559
ABS	0.008611	0.00001	0.000019

Table 4 Altitude and error STD in presence of uncertainty

Controller	5%	15%	25%	35%
FL	0.00548	0.02907	0.06569	0.09652
SM	0.000847	0.000849	0.000871	0.000951
ABS	0.000075	0.000075	0.000075	0.000075

As it can be seen, the robustness of proposed ABS controller is more than two other controllers. It should be noted that in each simulation of the controllers, external disturbances were applied.

Test case 2: A 3D spatial trajectory is considered for the second test case which is indicated from two different views in Figure 11, Figure 12. Again the tracking performances of ABS, SM, and FL controllers are simulated and shown in Figure 11, Figure 12.

Figure 13 indicates the performance of the proposed ABS controller which has faster convergence compared to the SM and FL controllers. Also, less fluctuations are observed in response of the ABS controller than that of the SM and FL controllers. Tracking errors are also indicated in Figure 13 which shows the unacceptable performance of the FL controller and the 20% error reduction achieved by the ABS controller compared to the SM controller. The RMS of error in the SM is 2.652mm, while it has been reduced to 1.520mm in ABS control.

Figure 14 compares the control signal of the SM and ABS controllers which are more robust than the FL controller. As is obvious, the range of control effort in both controllers are approximately equal, however the control signal of the SM controller has larger values and intense fluctuations. According to Figure 14 the control signal of the ABS controller has a relatively smooth behavior and no signal saturation is occurred.

In order to resemble some actual conditions in laboratory setup of the mechanism, the simulations are performed in presence of disturbing forces and uncertainties considered in the dynamic model. Three sinusoidal disturbing forces with amplitudes of 2, 3, and 4 N and frequencies of 12, 10, and 8 rad/s are considered to be applied on prismatic joints. Also, a 30% deviation is parameters considered to account for uncertainties in the model. Small error components appeared in both simulations which show outperforming of the control technique in spite of the applied disturbances and uncertainties. This performance could be improved by parameter optimization of the controller. Result of the error criteria for this case is depicted in Table 5.

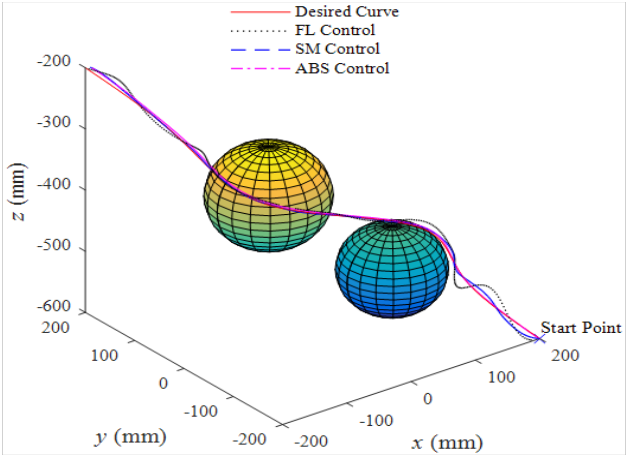

Figure 11 3D end–effector trajectory in the second maneuver.

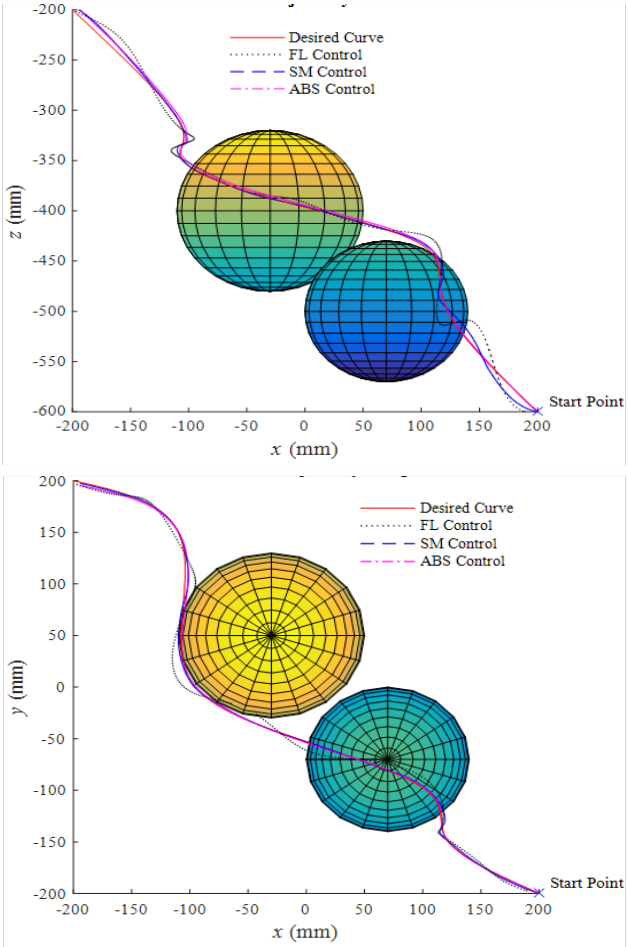

Figure 12 End–effector trajectory in the second maneuver Top view.

Citation: Mazare M, Taghizadeh M. Adaptive back–stepping robust control of a 3–[P2(US)] parallel robot on optimal trajectory. *Int Rob Auto J.* 2019;5(3):101–110. DOI: 10.15406/iratj.2019.05.00181

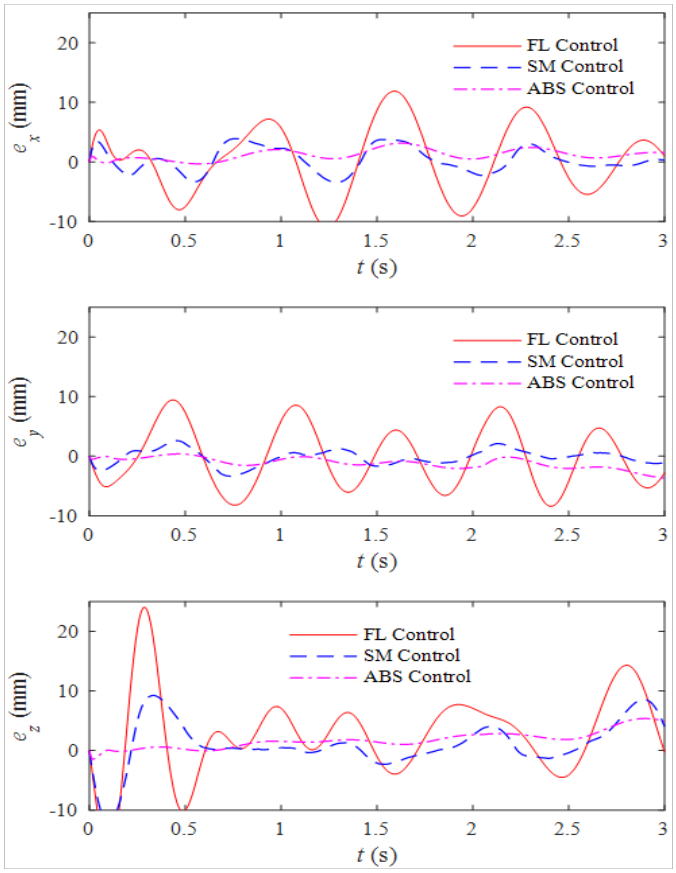

Figure 13 Error components of end–effector position in the maneuver 2.

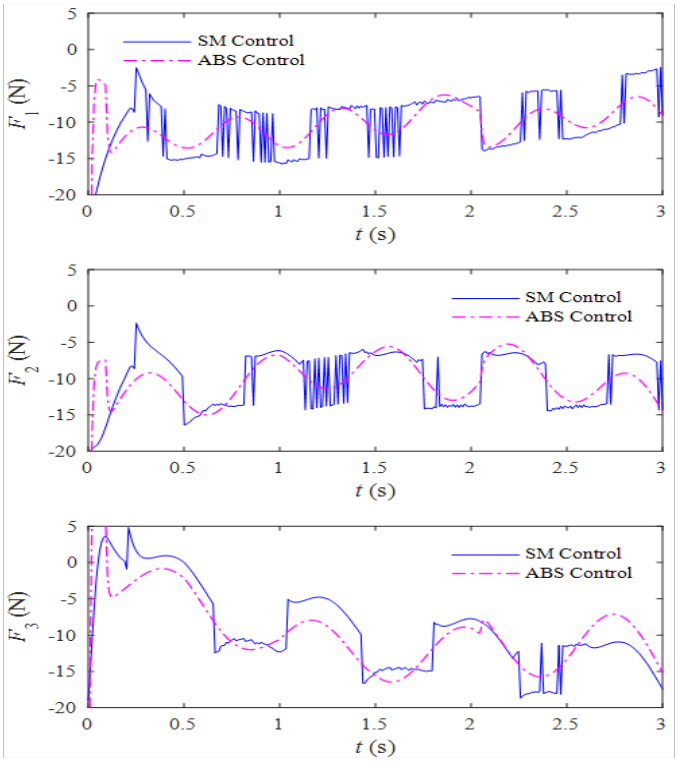

Figure 14 Control signals of proposed controller.

Table 5 Different criteria for comparing the performance of the controllers

Controller	ITAE	IASE	ITASE
FL	0.079069	0.001067	0.001483
SM	0.045551	0.000067	0.000076
ABS	0.015567	0.000023	0.000043

Conclusion

A three DOF translational parallel mechanism is investigated and controlled in this research. Kinematic equations are extracted based on the robot geometry and dynamic equations of motion are derived via Lagrange formulation in order to be used in robot control. To achieve minimum length and collision free trajectories for the robot motion, optimum path planning is performed using harmony search algorithm. Cubic spline interpolation is used through the obtained accuracy points in order to preserve continuity of velocity and acceleration. In order to compensate nonlinearity and uncertainties in the dynamic model, an adaptive back–stepping sliding mode control scheme is used as a robust control technique. Sliding surface equation with second order differentiation coupled with error integration is considered. Performance of the control technique is represented through two proposed maneuvers for the robot motion. Results show good achievements of the proposed controller in tracking the two desired trajectories which leads to acceptable errors even in presence of disturbances and uncertainties. In order to show the performance of proposed controller, two controllers are designed and result of the proposed controller is compared with sliding mode and feedback linearization controllers. The computed control signals are relatively smooth and saturations are not observed. According to the RMS of error in both maneuvers, the proposed controller has a better performance than the SM control.

Acknowledgements

None.

Conflict of interest

The author declares there is no conflict of interest.

References

1. Merlet JP. *Parallel robots*. 2nd ed. Netherlands: Springer; 2006. 471 p.
2. Sai LW. *Robot analysis: The mechanics of serial and parallel manipulators*. USA: John Wiley & Sons; 1999. 520 p.
3. Staicu Stefan. Dynamics modelling of a Stewart–based hybrid parallel robot. *Advanced Robotics*. 2015;29:929–938.
4. Tang L, Landers RG. Multi axis contour control the state of the art. IEEE Trans. *Control Syst Technol*. 2013;21(6):1997–2010.
5. Khoshdarregi MR, Tappe S, Altintas YY. Integrated five–axis trajectory shaping and contour error compensation for high–speed CNC machine tools. *IEEE/ASME Trans. Mechatronics*. 2014;19(6):1859–1871.
6. Wu J, Xiong Z, Ding H. Integral design of contour error model and control for biaxial system. *Int J Mach Tools Manuf*. 2015;89:159–169.
7. Chen SL, Tsai YC. Contouring control for multi–axis motion systems with holonomic constraints: Theory and application to a parallel system. *Asian J Control*. 2016;18(3): 888–898.

8. Yang J, Chen Y, Chen Y, et al. A tool path generation and contour error estimation method for four–axis serial machines. *Mechatronics*. 2015;31:78–88.
9. Hu C, Yao B, Wang Q. Coordinated adaptive robust contouring controller design for an industrial biaxial precision gantry. *IEEE/ASME Trans Mechatronics*. 2010;15(5):728–735.
10. Chen SL, Chou CY. Contouring control of robot manipulators based on equivalent errors. *Asian J Control*. 2014;16(3):904–914.
11. Liu S, Sun D, Zhu C. Coordinated motion planning for multiple mobile robots along designed paths with formation requirement. *IEEE/ASME Trans. Mechatronics*. 2011;16(6):1021–1031.
12. Lou Y, Li Z, Zhong Y, et al. Dynamics and contouring control of a 3–DoF parallel kinematics machine. *Mechatronics*. 2011;21(1):215–26.
13. Garagi C D, Srinivasan K. Contouring control of Stewart platform based machine tools. *Proceedings of the 2004 American Control Conference*. 2004;4:3831–3838.
14. Lou Y, Chen N, Li Z. Task space based contouring control of parallel machining systems. *International Conference on Intelligent Robots and Systems*. 2006. p. 2047–2052.
15. Craig JJ, Hsu P, Sastry SS. Adaptive control of mechanical manipulators. *The International Journal of Robotics Research*. 1987;6(2):16–28.
16. Slotine JJE, Li W. *Applied Nonlinear Control*. USA: Prentice–Hall Inc; 1991. 259 p.
17. Seraji H. A new approach to adaptive control of manipulators. *Journal of dynamic systems, measurement, and control*. 1987;109(3):193–202.
18. Yao B, Bu F, Chiu GT. Non–linear adaptive robust control of electro–hydraulic systems driven by double–rod actuators. *International Journal of Control*. 2001;74(8):761–775.
19. Navabi M, Mirzaei H. Robust Optimal Adaptive Trajectory Tracking Control of Quadrotor Helicopter. *Latin American Journal of Solids and Structures*. 2017;14(6):1040–1063.
20. Gong JQ, Yao B. Adaptive robust control without knowing bounds of parameter variations. *Proceedings of the 38th IEEE Decision and Control*. 1999;4:3334–3339.
21. Zeinali M, Notash L. Robust adaptive neural fuzzy controller with model uncertainty estimator for manipulators. *Transactions of the Canadian Society for Mechanical Engineering*. 2004;28(2):197–219.
22. Yu H, Wang J, Deng B, et al. Adaptive back–stepping sliding mode control for chaos synchronization of two coupled neurons in the external electrical stimulation. *Communications in Nonlinear Science and Numerical Simulation*. 2012;17(3):1344–1354.
23. Li HY, Hu YA. Robust sliding–mode backstepping design for synchronization control of cross–strict feedback hyperchaotic systems with unmatched uncertainties. *Communications in Nonlinear Science and Numerical Simulation*. 2011;16(10):3904–3913.
24. Bouabdallah S, Siegwart R. Backstepping and sliding–mode techniques applied to an indoor micro quadrotor. *Proceedings of the 2005 IEEE international conference on robotics and automation*. 2005. p. 2247–2252.
25. Bouadi H, Bouchoucha M, Tadjine M. Sliding mode control based on backstepping approach for an uav type–quadrotor. *Int J Appl Math Comput Sci*. 2007;4(1):22–27.
26. Lin FJ, Chang CK, Huang PK. FPGA–based adaptive back–stepping sliding–mode control for linear induction motor drive. *IEEE transactions on power electronics*. 2007;22(4):1222–1231.
27. Choi J, Han S, Kim J. Development of a novel dynamic friction model and precise tracking control using adaptive back–stepping sliding mode controller. *Mechatronics*. 2006;16(2):97–104.
28. Mazare M, Taghizadeh, M, Najafi MR. Kinematic analysis and design of a 3–DOF translational parallel robot. *International Journal of Automation and Computing*. 2016;14(4): 432–441.
29. Mazare M, Taghizadeh, Najafi MR. Contouring control of a 3–[P2 (US)] parallel manipulator. *Advanced Robotics*. 2017;31(9): 496–508.
30. Faieghi MR, Delavari H, Baleanu D. A novel adaptive controller for two–degree of freedom polar robot with unknown perturbations. *Communications in Nonlinear Science and Numerical Simulation*. 2012;17(2):1021–1030.
31. Chiu CH, Peng YF, Lin YW. Intelligent back–stepping control for wheeled inverted pendulum. *Expert Systems with Applications*. 2011;38(4):3364–3371.
32. Mazare M, Taghizadeh, Najafi MR. Inverse dynamics of a 3–[P2 (US)] parallel robot. *Robotica*. 2019;37(4): 708–728.

Citation: Mazare M, Taghizadeh M. Adaptive back–stepping robust control of a 3–[P2(US)] parallel robot on optimal trajectory. *Int Rob Auto J.* 2019;5(3):101–110. DOI: 10.15406/iratj.2019.05.00181