Adaptive control of a parallel robot via backstepping technique

Li Wang
Department of Electrical Engineering, Lakehead University
E-mail: lwang2@lakeheadu.ca

Zongtao Lu
Department of Electrical Engineering and Computer Science, Case Western Reserve University
E-mail: zongtao.lu@case.edu

Xiaoping Liu
Department of Electrical Engineering, Lakehead University
E-mail: xliu2@lakeheadu.ca

Kefu Liu
Department of Mechanical Engineering, Lakehead University
E-mail: kliu@lakeheadu.ca

Dan Zhang
Faculty of Engineering and Applied Science, University of Ontario Institute of Technology
E-mail: dan.zhang@uoit.ca

Abstract: The adaptive backstepping technique is applied to set point control of a planar parallel robot. The dynamic model of the robot is characterised by a set of differential algebraic equations. The model is reduced into a set of Ordinary Differential Equations (ODEs). A model-based adaptive backstepping controller is designed. The proposed controller is tested by experiments. Comparison among the adaptive backstepping controller, backstepping controller, adaptive PD controller and PD controller is made based on experimental results. Experimental results show that the adaptive backstepping controller outperforms all the other controllers in terms of steady-state errors.

Keywords: parallel robot; adaptive backstepping; nonlinear control; DAE; differential algebraic equation; systems.

1 Introduction

Recently, parallel robots have been receiving growing attentions from both academia and industries due to their advantages over serial robots, such as high stiffness, high speeds, low inertia and large payload capacity (Stewart, 1965; Hunt, 1978; Fichter, 1986; Gosselin, 1988; Boer et al., 1999; Zhang, 2000). In general, the dynamic models of parallel robots are much more complex than those of serial robots. Serial robots are described by a set of Ordinary Differential Equations (ODEs). Unlike serial robots, parallel robots are often modelled by a set of Differential Algebraic Equations (DAEs), so dynamic equations for parallel robots are not in an explicit form of the independent generalised coordinates or actuated joints. Calculation of these implicit relations in real time imposes a severe constraint on application of many well-established control methods for serial robots to parallel robots. Therefore, some early attempt in control of parallel robots focused on the use of non-model based control methods, such as Proportional Integral Derivative (PID) control (Amiral et al., 1996; Kokkinis and Stoughton, 1991) and artificial intelligence-based algorithms (Begon et al., 1995; Geng and Haynes, 1993). However, as pointed out in Ghorbel et al. (2000), these
methods have no guarantee of stability and performance. Some efforts have been made to extend model-based control algorithms for serial robots to parallel robots. The study reported in Gosselin (1996) proposed a parallel computational algorithm to speed up on-line computation. In Codourey (1998), the mass and inertia of the links were neglected in the dynamic model in order to implement the computed-torque control. A PD plus simple gravity compensation control law was proposed in Ghorbel et al. (2000) for set point control for a planar 2-DOF parallel robot. With the proved skew symmetry property, this controller guarantees a local asymptotic stability. The simple gravity compensation is a constant term which can be computed offline to any degree of accuracy. In Li and Wu (2004), the design for control approach was employed in the design stage of a parallel robot to find an appropriate mechanical structure with a simple dynamic model. This way, a simple control algorithm is sufficient to achieve a satisfactory control performance. In Vivas and Poignet (2005), a predictive functional control strategy is implemented for tracking control of a H4 parallel robot. The dynamic model is simplified by neglecting the effect of arm mass, which greatly facilitates the implementation of the controller.

Backstepping is a recursive design method for constructing both feedback control laws and associated Lyapunov functions in a systematic manner. During the last ten years, backstepping-based designs have emerged as powerful tools for stabilising nonlinear systems (Kanellakopoulos et al., 1991; Kristic et al., 1995; Marino and Tomei, 1995). Adaptive backstepping is able to handle nonlinear systems with unknown parameters (Kristic et al., 1995) whereas robust backstepping design is used to control nonlinear systems with uncertainties (Freeman and Kokotovic, 1996). The backstepping design method has been applied to control of serial robots (Lotfazar et al., 2003; Su and Stepanenko, 1997) and trajectory planning for mobile robots (Ma et al., 2005). To the authors’ best knowledge, there has been no report on application of the backstepping technique to control of parallel robots. In general, the model governing a parallel robot is highly nonlinear and a precise knowledge of its parameters is not readily available. Adaptive backstepping appears to be a suitable control design methodology for parallel robots.

In this paper, a backstepping-based control scheme is applied to set point control for a planar 2-DOF parallel robot. By assuming that inertia parameters and some geometric dimensions of the robot are not known precisely, an adaptive backstepping controller is designed. For the purpose of comparison, an adaptive Proportional and Derivative (PD) controller is designed as well. The performance of each controller is tested by experiments.

The rest of the paper is organised as follows. In Section 2, the dynamic model of the planar 2-DOF parallel robot under study is presented. In Section 3, an adaptive backstepping-based controller and an adaptive PD controller are designed, respectively. In Section 4, the experimental results with the controllers are presented. Finally the study conclusions are drawn in Section 5.

2 Dynamic model

A schematic of a planar 2-DOF parallel robot is shown in Figure 1 where $m_i$, $a_i$, and $l_i$ are the mass, length and distance to the centre of mass from the lower joint of link $i$, respectively, $I_i$ denotes the mass moment of inertia of link $i$. Joints $q_1$ and
$q_2$ are actuated while joints $q_3$ and $q_4$ are passive. The dynamical model of the robot, presented in Ghorbel et al. (2000), is described as follows:

$$ D'(q')q' + C'(q', q')q' + g'(q') = u' $$

$$ \phi(q') = 0 $$

where $q' = [q_1, q_2, q_3, q_4]^T$ is the vector of dependent generalised coordinates, $u' = [u_1, u_2, 0, 0]^T$ with $u_1$ and $u_2$ are the torque applied on joints $q_1$ and $q_2$, respectively, $D'(q') \in \mathbb{R}^{4 \times 4}$ is the inertia matrix, $C'(q', q')q' \in \mathbb{R}^4$ represents the centrifugal and Coriolis terms and $g'(q') \in \mathbb{R}^4$ is the gravity vector, and $\phi(q')$ represents the constraints defined by

$$ \phi(q') = \begin{bmatrix} \phi_1(q') \\ \phi_2(q') \end{bmatrix} = \begin{bmatrix} a_1 \cos(q_1) + a_3 \cos(q_1 + q_3) - c - a_2 \cos(q_2) \\ -a_4 \cos(q_2 + q_4) \\ a_1 \sin(q_1) + a_3 \sin(q_1 + q_3) - a_2 \sin(q_2) - a_4 \sin(q_2 + q_4) \end{bmatrix}. $$ (3)

Figure 1 The schematic of a 2DOF parallel robot

Assume that the parameters $m_i$, $l_i$ and $I_i$ are unknown and define $\Theta = [\theta_1, \theta_2, \ldots, \theta_{10}]^T$ with $\theta_1 = m_1l_1^2 + m_3a_3^2 + I_1$, $\theta_2 = m_2l_2^2 + m_4a_4^2 + I_2$, $\theta_3 = m_3l_3^2 + I_3, \theta_4 = m_4l_4^2 + I_4, \theta_5 = m_5a_5l_5, \theta_6 = m_6a_6l_6, \theta_7 = 9.81(m_1l_1 + m_3a_3)$, $\theta_8 = 9.81(m_2l_2 + m_4a_4), \theta_9 = 9.81m_3l_3$ and $\theta_{10} = 9.81m_4l_4$ as unknown parameters. Then, $D'(q'), C'(q', q')$ and $g'(q')$ can be expressed in terms of the unknown parameters as follows:

$$ D'(q') = \begin{bmatrix} \theta_1 + 2\theta_5 \cos(q_4) & 0 & \theta_3 + \theta_5 \cos(q_3) & 0 \\ 0 & \theta_2 + \theta_4 + 2\theta_6 \cos(q_4) & 0 & \theta_4 + \theta_6 \cos(q_3) \\ \theta_3 + \theta_5 \cos(q_3) & 0 & \theta_3 & 0 \\ 0 & \theta_4 + \theta_6 \cos(q_3) & 0 & \theta_4 \end{bmatrix} $$

$$ C'(q', q') = \begin{bmatrix} -\theta_5 \sin(q_3)q_3 & 0 & -\theta_5 \sin(q_3)(q_1 + q_3) & 0 \\ 0 & -\theta_6 \sin(q_4)q_4 & 0 & -\theta_6 \sin(q_4)(q_2 + q_4) \\ \theta_5 \sin(q_3)q_1 & 0 & 0 & 0 \\ 0 & \theta_6 \sin(q_4)q_2 & 0 & 0 \end{bmatrix}.$$
3 Controller design

In the following, two adaptive controllers: adaptive backstepping controller and adaptive PD controller, will be designed to achieve the set-point control. Each controller consists of a control law and an update law for the parameter estimation. For brevity, we denote \( D(q') \) as \( D \), \( C(q', q) \) as \( C \) and \( g(q') \) as \( g \).
3.1 Adaptive controller based on backstepping

In order to formulate equation (5) into a form suitable for set point control using the adaptive backstepping technique, assign \( x_1 = q_1 - q_1^d, \ x_2 = q_2 - q_2^d, \ x_3 = \dot{q}_1, \ x_4 = \dot{q}_2 \) with \( q_1^d \) and \( q_2^d \) being the desired angles for \( q_1 \) and \( q_2 \), respectively. Let \( \hat{\Theta} \) be the estimation of \( \Theta \). A lower triangular form is obtained as:

\[
\begin{align*}
\dot{x}_1 &= x_3 \\
\dot{x}_2 &= x_4 \\
D \begin{bmatrix} \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= u - C\dot{q} - g.
\end{align*}
\]

Following the backstepping design procedure, first, choose the Lyapunov function candidate:

\[
V_1 = \frac{1}{2} x_1^2 + \frac{1}{2} x_2^2.
\]

By introducing virtual controllers: \( \alpha_1 = -c_1 x_1, \ \alpha_2 = -c_2 x_2 \), where \( c_1 \) and \( c_2 \) are positive numbers, \( \dot{V}_1 \) can be rewritten into:

\[
\dot{V}_1 = -c_1 x_1^2 - c_2 x_2^2 + x_1 (x_3 - \alpha_1) + x_2 (x_4 - \alpha_2).
\]

Now, choose the Lyapunov function candidate:

\[
V_2 = V_1 + \frac{1}{2} \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix}^T D \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix} + \frac{1}{2} (\Theta - \hat{\Theta})^T \Gamma (\Theta - \hat{\Theta})
\]

where \( \Gamma = \text{diag}[\gamma_1, \gamma_2, \ldots, \gamma_{10}] \) is a positive definite matrix. Note that \( D \) is positive definite. Differentiating \( V_2 \) with respect to time yields:

\[
\dot{V}_2 = -c_1 x_1^2 - c_2 x_2^2 + x_1 (x_3 - \alpha_1) + x_2 (x_4 - \alpha_2) + \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix}^T D \begin{bmatrix} \dot{x}_3 - \dot{\alpha}_1 \\ \dot{x}_4 - \dot{\alpha}_2 \end{bmatrix}
\]

\[
+ \frac{1}{2} \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix}^T \dot{D} \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix} - \dot{\hat{\Theta}}^T \Gamma (\Theta - \hat{\Theta}).
\]

As pointed out in Ghorbel et al. (2000), the matrix \( \dot{D} - 2C \) is skew symmetric. As a result, we can get

\[
\frac{1}{2} \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix}^T (\dot{D} - 2C) \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix} = 0.
\]

With the parameterisations in equations (8)–(10), substituting equations (13) and (18) into (17) yields:

\[
\dot{V}_2 = -c_1 x_1^2 - c_2 x_2^2 + \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix}^T \left( u + \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \Lambda \right) - \dot{\hat{\Theta}}^T \Gamma (\Theta - \hat{\Theta})
\]
Adaptive control of a parallel robot via backstepping technique

where:

\[
\Lambda = \Lambda_0 \Theta = - \left[ \dot{\alpha}_1 D_{o11} + \dot{\alpha}_2 D_{o12} + \alpha_1 C_{o11} + \alpha_2 C_{o12} + g_{o1} \right] \Theta. \tag{20}
\]

Apparently, if the controller is chosen to be:

\[
u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = - \begin{bmatrix} c_3 (x_3 - \alpha_1) + x_1 \\ c_4 (x_4 - \alpha_2) + x_2 \end{bmatrix} - \Lambda_0 \hat{\Theta} \tag{21}
\]

and the unknown parameters’ updating law is chosen to be:

\[
\dot{\hat{\Theta}} = \Gamma^{-1} \Lambda_0^T \begin{bmatrix} x_3 - \alpha_1 \\ x_4 - \alpha_2 \end{bmatrix} \tag{22}
\]

where \(c_3\) and \(c_4\) are positive numbers, the derivative of \(V_2\) is negative, i.e.,

\[
\dot{V}_2 = -c_1 x_1^2 - c_2 x_2^2 - c_3 (x_3 - \alpha_1)^2 - c_4 (x_4 - \alpha_2)^2 \tag{23}
\]

which means that the corresponding closed-loop system is stable.

If \(\Theta\) is known, the control effort of a corresponding non-adaptive backstepping controller is given as:

\[
u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = - \begin{bmatrix} c_3 (x_3 - \alpha_1) + x_1 \\ c_4 (x_4 - \alpha_2) + x_2 \end{bmatrix} - \Lambda_0 \Theta. \tag{24}
\]

### 3.2 Adaptive PD controller

It is worthwhile comparing the controller of equation (21) with an adaptive PD controller. Choose the Lyapunov function candidate:

\[
V = \frac{1}{2} (q - q^d)^T K_p (q - q^d) + \frac{1}{2} \dot{q}^T D \dot{q} + \frac{1}{2} \left( \Theta - \hat{\Theta} \right)^T \Gamma_{pd} \left( \Theta - \hat{\Theta} \right) \tag{25}
\]

where \(\Gamma_{pd} = \text{diag}[\gamma_{pd1}, \gamma_{pd2}, \ldots, \gamma_{pd8}]\) and \(K_p = \text{diag}[k_{p1}, k_{p2}]\) with \(\gamma_{pd1} > 0\) for \(1 \leq i \leq 8\) and \(k_{pi} > 0\) for \(i = 1, 2\). Differentiating \(V\) with respect to time yields:

\[
\dot{V} = (q - q^d)^T K_p \ddot{q} + \frac{1}{2} \dot{q}^T \dot{D} \ddot{q} + \dot{q}^T D \ddot{q} - \dot{\hat{\Theta}}^T \Gamma_{pd} \left( \Theta - \hat{\Theta} \right). \tag{26}
\]

According to Ghorbel et al. (2000) \(\dot{D} = 2C\) is skew symmetric and \(\dot{D}\) is symmetric. So we can get:

\[
\dot{D} = C + C^T. \tag{27}
\]

With the parameterisations in equations (8)–(10), substituting equations (13) and (27) into (26) gives

\[
\dot{V} = (q - q^d)^T K_p \ddot{q} + \dot{q}^T [u + \Lambda_{pd}] - \dot{\Theta}^T \Gamma_{pd} \left( \Theta - \hat{\Theta} \right) \tag{28}
\]
where
\[ \Lambda_{pd} \Delta \Theta = \frac{1}{2} \left[ \hat{\dot{q}}_2 (C_{o21} - C_{o12}) - g_{o1} \right] \Theta, \]

Apparently, if the controller is set to be:
\[ u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = -K_p (q - q_d) - K_v \dot{q} - \Lambda_{pd} \hat{\Theta} \] (29)

and the unknown parameters’ updating law is selected as
\[ \dot{\hat{\Theta}} = \Gamma_{pd}^{-1} \Lambda_{pdo} \hat{\dot{q}} \] (30)

where \( K_v = \text{diag}[k_{v1}, k_{v2}] \) is positive definite matrix, the derivative of \( V_2 \) is
\[ \dot{V}_2 = -\dot{\dot{q}}^T K_v \dot{\dot{q}} \] (31)

which means that the corresponding closed-loop system is stable.

If \( \Theta \) is known, then the adaptive PD controller becomes a non-adaptive PD controller with Coriolis and centrifugal terms and gravity compensation:
\[ u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = -K_p (q - q_d) - K_v \dot{q} - \Lambda_{pd} \Theta. \] (32)

4 Experiment

Figure 2 shows a photo of the planar 2-DOF parallel robot developed for the purpose of this study. Links 1 and 2 are driven by two Direct Current (DC) gearhead motors, respectively. The motors are from Kollmorgen Motion Technologies Group. The gear ratio is 99 to 1 and the peak torque is 17.1 N-m. The optical encoders are built in the motors with the resolution of 1000 pulses per revolution. The values of the link parameters are given in Table 1. The distance between the shafts of the motors is \( c = 0.4240 \text{ m} \).

Figure 2  The photo of a 2-DOF robot
Table 1  Link parameters

<table>
<thead>
<tr>
<th>Link i</th>
<th>m_i (kg)</th>
<th>a_i (m)</th>
<th>l_i (m)</th>
<th>I_i (kg·m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1950</td>
<td>0.4600</td>
<td>0.3367</td>
<td>4.567 × 10⁻³</td>
</tr>
<tr>
<td>2</td>
<td>0.1950</td>
<td>0.4600</td>
<td>0.3367</td>
<td>4.567 × 10⁻³</td>
</tr>
<tr>
<td>3</td>
<td>0.2538</td>
<td>0.4600</td>
<td>0.2400</td>
<td>8.626 × 10⁻³</td>
</tr>
<tr>
<td>4</td>
<td>0.2538</td>
<td>0.4600</td>
<td>0.2400</td>
<td>8.626 × 10⁻³</td>
</tr>
</tbody>
</table>

The computer control system has three main parts: a Pentium III Personal Computer (PC) is used in which two Data Acquisition (DAQ) boards (PCI-6024E and PCI-MIO-16E, National Instruments) are installed; these two DAQ boards are connected with two SCB-68 connector blocks through shielded cables; the motors are driven by two H-Bridge circuits, which are controlled by Pulse Width Modulation (PWM) signals from the computer. The controller is implemented by using Visual C++. Angular positions of Links 1 and 2 are provided by the optical encoders. The encoder readings are passed through an analog low pass filters before sampled by the DAQ boards. Angular velocities of Links 1 and 2 are calculated digitally based on the position measurements. A digital low pass filter is used for the velocity calculation, which is given by:

\[ v_{k+1} = \frac{(p_{k+1} - p_k + \tau v_k)}{(\tau + T)} \]  

where \( v_k \) and \( v_{k+1} \) are the angular velocity at the sampling instants \( k \) and \( k + 1 \), \( p_k \) and \( p_{k+1} \) are the position measurements of the links at the sampling instants \( k \) and \( k + 1 \), respectively, \( T \) is the sampling period and \( \tau \) is the time constant set as 0.1.

As for experiments, the control inputs are not the torques applied to the joints. The direct control inputs are the armature voltages of the DC motors. Therefore, in order to implement the designed controllers in terms of torque of the motors, the computed torque is converted into the armature voltages of the DC motors. The conversion formula is given as follows:

\[ u_i = \frac{G K}{R} (V_{ai} - K_c G \omega_i), \quad i = 1, 2 \]  

where \( u_i \) is the torque applied at the motor, \( V_{ai} \) is the armature voltage supplied, \( G = 99 \) is the gear ratio of the motor, \( K_t \) is the torque constant with the value of 2.28 N·cm/Amp, \( K_c \) is the back Electromotive Force (EMF) constant with the value of 0.02282 volts/rad/s, \( R \) is the armature resistance with the value of 0.640 Ohms and \( \omega_i \) is the angular velocity of the gear shaft. The maximum voltage of the driver board is 15 volts.

In the experiment, a sampling period of 1 ms was used. In each sampling period, the computer obtains the current positions and velocities of Links 1 and 2, calculates the armature voltages in terms of duty cycles of the PWM signals and sends the PWM signals to the driver boards to control the DC motors.

To compensate the effect of back-lash between gears in the two motors, a voltage compensation is applied in the experiment. When the computed armature voltage is larger than 0.01 volts, the armature voltage used in experiment is increased by 0.05 for motor 1 and 0.35 for motor 2; when the computed armature voltage is
less than −0.01 volts, the armature voltage used in experiment is increased by −0.65 for both motors.

Four controllers, namely backstepping, adaptive backstepping, PD and adaptive PD, were tested on the parallel robot. The controller gains are $c_1 = 3.18$, $c_2 = 3.7$, $c_3 = 14$ and $c_4 = 14$ for the backstepping; $c_1 = 3.18$, $c_2 = 3.7$, $c_3 = 14$, $c_4 = 14$, $\gamma_1 = 0.6$, $\gamma_2 = 1.5$, $\gamma_3 = 0.8$, $\gamma_4 = 2$, $\gamma_5 = 50.0$, $\gamma_6 = 60.0$, $\gamma_7 = 60.0$, $\gamma_8 = 10.0$, $\gamma_9 = 60.0$ and $\gamma_{10} = 110.0$ for the adaptive backstepping; $k_{p1} = 45$, $k_{p2} = 14$, $k_{v1} = 52$ and $k_{v2} = 14$ for PD; and $k_{p1} = 45$, $k_{p2} = 14$, $k_{v1} = 52$, $k_{v2} = 14$, $\gamma_1 = 0.1$, $\gamma_2 = 1.5$, $\gamma_3 = 0.13$, $\gamma_4 = 2$, $\gamma_5 = 50.0$, $\gamma_6 = 60.0$, $\gamma_7 = 60.0$ and $\gamma_8 = 10.0$ for the adaptive PD, which are tuned by trial and error.

Experiments were conducted on the 2DOF parallel robot as shown in Figure 2. In the experiments, the initial angles for the four links are chosen as $q_{1d}(0) = 90^\circ$, $q_{2d}(0) = 90^\circ$, $q_{3d}(0) = -27^\circ$ and $q_{4d}(0) = 27^\circ$, while the desired angles are set to be $q_{1d} = 150^\circ$, $q_{2d} = 160^\circ$, $q_{3d} = -96^\circ$ and $q_{4d} = 55^\circ$.

Figure 3 shows the errors for no-load tests while Figure 4 displays the errors for a load of 0.4869 kg. It is observed that the transient processes from the four controllers are similar for both no-load tests and load-tests. However, the adaptive backstepping controller outperforms all the other controllers in terms of steady-state errors.

Figure 3  Experimental results: errors with no load
Adaptive control of a parallel robot via backstepping technique

Figure 4  Experimental results: errors with a load of 486.9 g

![Graph showing errors with a load of 486.9 g](image)

(a)

Figure 5  Experimental results: errors against loads

![Graph showing errors against loads](image)
Figure 5 demonstrates curves of the steady-state errors vs. the load. It is easily seen that the adaptive backstepping controller is least sensitive to the load changes. It can be also concluded that the adaptive backstepping controllers are more robust to the load changes than their non-adaptive counterparts.

5 Conclusion

An adaptive backstepping controller has been developed for set point control of a planar 2-DOF parallel robot. The proposed controller guarantees the stability of the closed-loop system and is able to handle parameter uncertainties. For comparison, an adaptive PD controller has been also developed. The backstepping controller is nonlinear while the PD controller is a linear controller with Coriolis and centrifugal terms and gravity compensation. Compared with the adaptive PD controller, the control algorithm is more complicated and the update law contains more unknown parameters for the adaptive backstepping controller, so the adaptive backstepping controller demands more computation time. The experiments have been conducted to compare four controllers: adaptive backstepping, backstepping, adaptive PD and PD. The results have shown that all the controllers perform similarly in terms of transient responses. However, the adaptive controllers are more robust to load changes than their non-adaptive counterparts. The adaptive backstepping controller outperforms all the other controllers in terms of steady-state errors and sensitivity to load changes.

References


Adaptive control of a parallel robot via backstepping technique


Query

AQ1: AUTHOR PLEASE SUPPLY DETAILED AFFILIATION FOR EACH AUTHOR AND INDICATE WHO THE CORRESPONDING AUTHOR IS.