Robotica (2014) volume 32, E1, pp. 1–24. © Cambridge University Press 2014
doi:10.1017/S0263574714002410

Adaptive backstepping control for parallel robot with uncertainties in dynamics and kinematics

Jing Zou and John K. Schueller*

Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611, USA

(Accepted August 28, 2014)

SUMMARY
It is common in robot tracking control that controllers are designed based on the exact kinematic model of the robot manipulator. However, because of measurement errors and changes of states in practice, the original kinematic model is often no longer accurate and will degrade the control result. An adaptive backstepping controller is designed here for parallel robot systems with kinematics and dynamics uncertainties. Backstepping control is used to manage the transformation between the errors in task space and joint space. Adaptive control is utilized to compensate for uncertainties in both dynamics and kinematics. The controller demonstrated good performance in simulation.

KEYWORDS: Parallel robots; Adaptive control; Backstepping control; Dynamics; Robot kinematics; Tracking control; Uncertainty.

1. Introduction

Robot manipulators are highly nonlinear in their dynamics and kinematics. More nonlinearity typically appears in parallel robot manipulators than serial robot manipulators. In order to have a good tracking performance of parallel robot manipulators, many try to compensate for the nonlinearities and use feedback PD control to minimize the tracking error. In refs. [1] and [2], a nonlinear PD controller was proposed by using the nonlinear terms in robot dynamics as nonlinear feedback to cancel those terms and PD feedback to control the tracking error. This controller is very sensitive to uncertainties in the robot model as it needs very accurate knowledge of the robot dynamics to cancel the nonlinear terms in the system. To make the controller robust to the dynamic uncertainties of the parallel robot manipulator, adaptive control, high gain control and high frequency control methods are introduced. In refs. [3] and [4], an adaptive controller was created with an estimator for the dynamic parameters of the robot to compensate for the uncertainties. And in ref.[5], sliding mode control method is applied to decentralize uncertain dynamic parameters of the robot manipulator to get a more robust performance. Those controllers work well with parallel robots having uncertainties in dynamics. However, since there are no estimators to predict the uncertain parameters in kinematic functions and the decentralization method is not applied to uncertain terms appearing in the kinematics, they are not robust to kinematic uncertainties. In refs. [6] and [7], adaptive controllers are proposed to make the whole system resistant to uncertainties in both dynamics and kinematics through the design of an estimator to predict and compensate the uncertain terms in both dynamic and kinematic functions. The controllers give good control results. Nevertheless, the researchers produce integrated controllers to compensate for both dynamics and kinematics uncertainties. As the kinematics uncertainties are decoupled from the control input, much more mathematical analysis and structure complexity are required for the controllers. A robust backstepping controller is proposed in ref. [8]. The design needs less effort, but its Lyapunov analysis is based on the slow varying assumption on some parameters, which means the robot is not influenced by potentially arbitrarily large and fast external torques, and this is a bad assumption for parallel robot manipulators, where arbitrarily large and fast external torques can appear due to geometric constraints on the links of the robot.

* Corresponding author. E-mail: schuejk@ufl.edu

In this paper, the mathematical analysis of the Jacobian matrix of a parallel robot helps to conclude that it is linear in physical parameters. And then through the implementation of backstepping control and adaptive control, a controller which is robust to uncertainties in dynamics and kinematics is constructed. With the application of backstepping control, massive mathematical analysis according to the decoupling of control input and kinematics uncertainties is avoided. And the adaptive control has a good performance for the parallel robot with arbitrarily large and fast dynamics caused by geometric constraints.

2. Kinematics and Dynamics Analysis of Parallel Robot

2.1. Kinematics analysis

Here, a kinematic structure that has a rigid base connected to a rigid end-effector by means of n serial kinematic chains in parallel is discussed. Each set of a serial kinematic chain is defined as a "leg". The ith "leg" has n_{i} degrees of freedom, $q_{i,1}, \ldots, q_{i,n_i}$, collected in a $n_{\mathrm{i}} \times 1$ vector q_i. Let N be the total number of joints: $N = \sum_{i=1}^{n} n_{\mathrm{i}}$, and Q be the $N \times 1$ vector of all joint angles: $Q = (q_1^T \ldots q_n^T)^T$.

The definition of the corresponding terms in this section can be found in ref. [9]. According to ref. [9]

$$J_T = J_j S_j \left(\dot{Q}^1 \ldots \dot{Q}^{n_T}\right) = J_j S_j P, \tag{1}$$

where J_T is the Jacobian matrix for which $\dot{x} = J_T \dot{q}_a$, and $\dot{x}$ is the velocity of the end-effector, $\dot{q}_a$ is vector containing angular velocities of the active (or driving) joint. P is the dependency matrix defined as $\rho = \dot{q}_a - \dot{q}_d$, with $\dot{Q}^l$ the $N \times 1$ vector of all joint velocities when the ith driving joint is given a unit speed and all other $(n_T - 1)$ driving joints are kept motionless.

The rest of this section discussed the linearity property of the Jacobian matrix for the proposed parallel robot.

From the definition of selection matrices S_d, S_p and S_i, the dependency matrix P, A_d and A_p in ref. [1], Eq. (1) can be transformed into

$$\begin{aligned} J_T &= J_j S_j \left(\dot{Q}^1 \ldots \dot{Q}^{n_T}\right) = J_j S_j S_d^T \left(\left(S_d^1\right)^T \ldots \left(S_d^{n_T}\right)^T\right) - J_j S_j S_p^T A_p^+ A_d \left(\left(S_d^1\right)^T \ldots \left(S_d^{n_T}\right)^T\right) \\ &= J_j S_j S_d^T S_{n_T} - J_j S_j S_p^T A_p^+ A_d S_{n_T}, \end{aligned} \tag{2}$$

where A_p^+ is the Moore–Penrose pseudo-inverse of A_p, and S_{n_T} is a $n_T \times n_T$ matrix. Usually with proper arrangement of $\dot{q}_a$ in $\dot{Q}_a$ and $\dot{Q}$, $S_{n_T} = I_{n_T \times n_T}$.

It can be derived that

$$\dot{x} = J_T \dot{Q}_a = J_j S_j S_d^T S_{n_T} \dot{Q}_a - J_j S_j S_p^T A_p^+ A_d S_{n_T} \dot{Q}_a. \tag{3}$$

The following theoretical analysis is focused on exploring the linearity properties of Eq. (3).

The conclusion that S_d, S_p, S_i, S_{n_T}, A_p, A_d and J_j are all linear in physical parameters is obtained from their descriptions in ref. [13].

The first component of Eq. (3), i.e., $J_j S_j S_d^T S_{n_T} \dot{Q}_a$ is therefore linear in a set of physical parameters $\theta_2 = (\theta_{21}, \theta_{22}, \ldots, \theta_{2m_2})$

$$J_j S_j S_d^T S_{n_T} \dot{Q}_a = Y_2\left(q, \dot{Q}_a\right) \theta_2, \tag{4}$$

where $Y_2\left(q, \dot{Q}_a\right)$ is the regressor matrix.

The linearity exploration of the second part of Eq. (3), i.e., $J_j S_j S_p^T A_p^+ A_d S_{n_T} \dot{Q}_a$ requires more mathematical analysis.

According to the definition of the Moore–Penrose pseudo-inverse, $J_j S_j S_p^T A_p^+ A_d S_{n_T} \dot{Q}_a$ could be transformed as follows:

$$J_j S_j S_p^T A_p^+ A_d S_{n_T} \dot{Q}_a = J_j S_j S_p^T \left(A_p^T A_p\right)^{-1} A_p^T S_{n_T} \dot{Q}_a = \frac{J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a}{\left|A_p^T A_p\right|}, \tag{5}$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

where $A_{pp} = (A_p^T A_p)^{-1} |A_p^T A_p|$. $(J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a)$ is linear in a set of physical parameters $\theta_3 = (\theta_{31}, \theta_{32}, \ldots, \theta_{3m_3})$

$$J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a = Y_3 \left(q, \dot{Q}_a\right) \theta_3, \tag{6}$$

where $Y_3(q, \dot{Q}_a)$ is the regressor matrix.

Then the linearity of $\frac{J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a}{|A_p^T A_p|}$ is relevant with $|A_p^T A_p|$. If $|A_p^T A_p|$ is scalar linear with regard to only one physical parameter (or combination of physical parameters), $\frac{J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a}{|A_p^T A_p|}$ will be linear in a set of physical parameters.

For parallel robots with several "legs" connected by rotation joints, the relationship between the speed of the end-effector and the angular velocities of the joint angles in the ith "leg" could be treated like a serial robot with the same number of linkages, then

$$\dot{x} = \dot{q}_{i1} (p_{i1} \times r_{i1}) + \ldots + \dot{q}_{ij} \left(p_{ij} \times r_{ij}\right) + \ldots + \dot{q}_{ik} (p_{ik} \times r_{ik}), \tag{7}$$

where p_{ij} is the direction vector of angular velocity for jth rotation joint, i.e., $\dot{q}_{ij}$, and r_{ij} is the position vector from jth joint to $(j+1)$th joint. Given $r_{ij} = l_{ij} \epsilon_{ij}$, with ϵ_{ij} and l_{ij} denoting the direction vector and length of r_{ij}, $p_{ij} \times r_{ij} = l_{ij} f_{ij}$ and f_{ij} is a (3×1) vector function of measurable variables q_{ij} and the azimuth angles of the joint axis. Substituting $p_{ij} \times r_{ij} = l_{ij} f_{ij}$ into Eq. (7) leads to

$$\begin{aligned} \dot{x} &= \dot{q}_{i1} l_{i1} f_{i1} + \ldots + \dot{q}_{ij} l_{ij} f_{ij} + \ldots + \dot{q}_{ik} l_{ik} f_{ik} \\ &= \left(l_{i1} f_{i1}, \ldots, l_{ij} f_{ij}, \ldots, l_{ik} f_{ik}\right) \left(\dot{q}_{i1}, \ldots, \dot{q}_{ij}, \ldots, \dot{q}_{ik}\right)^T = J_i \dot{q}_i. \end{aligned} \tag{8}$$

Based on its definition in ref. [1], A_p can be rewritten as

$$A_p = \begin{bmatrix} l_{1p_1} f_{1p_{1x}} & \cdots & l_{1p_p} f_{1p_{px}} & l_{2p_1} f_{2p_{1x}} & \cdots & l_{2p_p} f_{2p_{px}} & & & & & & \\ l_{1p_1} f_{1p_{1y}} & \cdots & l_{1p_p} f_{1p_{py}} & l_{2p_1} f_{2p_{1y}} & \cdots & l_{2p_p} f_{2p_{py}} & & & & & & \\ l_{1p_1} f_{1p_{1z}} & \cdots & l_{1p_p} f_{1p_{pz}} & l_{2p_1} f_{2p_{1z}} & \cdots & l_{2p_p} f_{2p_{pz}} & & & & & & \\ l_{1p_1} f_{1p_{1x}} & \cdots & l_{1p_p} f_{1p_{px}} & & & & l_{3p_1} f_{3p_{1x}} & \cdots & l_{3p_p} f_{3p_{px}} & & & \\ l_{1p_1} f_{1p_{1y}} & \cdots & l_{1p_p} f_{1p_{py}} & & & & l_{3p_1} f_{3p_{1y}} & \cdots & l_{3p_p} f_{3p_{py}} & & & \\ l_{1p_1} f_{1p_{1z}} & \cdots & l_{1p_p} f_{1p_{pz}} & & & & l_{3p_1} f_{3p_{1z}} & \cdots & l_{3p_p} f_{3p_{pz}} & & & \\ \vdots & \vdots & \vdots & & & & & & & \ddots & & \\ l_{1p_1} f_{1p_{1x}} & \cdots & l_{1p_p} f_{1p_{px}} & & & & & & & & l_{np_1} f_{np_{1x}} & \cdots & l_{np_p} f_{np_{px}} \\ l_{1p_1} f_{1p_{1y}} & \cdots & l_{1p_p} f_{1p_{py}} & & & & & & & & l_{np_1} f_{np_{1y}} & \cdots & l_{np_p} f_{np_{py}} \\ l_{1p_1} f_{1p_{1z}} & \cdots & l_{1p_p} f_{1p_{pz}} & & & & & & & & l_{np_1} f_{np_{1z}} & \cdots & l_{np_p} f_{np_{pz}} \end{bmatrix}$$

where l_{ip_l} corresponds to the length of a link starting from a passive (driven) joint in the ith leg, $f_{ip_{lx}}$, $f_{ip_{ly}}$, $f_{ip_{lz}}$ are the first, second and third elements of vector function f_{ip_l}. p is the number of passive joints in one "leg" and assume all "legs" have the same number of passive joints. Then, $np = N - n_T$ denotes the number of all the passive joints.

To get the properties of $\left|A_p^T A_p\right|$, first $A_p^T A_p$ needs to be calculated. The calculation process and results of $A_p^T A_p$ are shown below.

The component of matrix $A_p^T A_p$ in row 1, column 1 is

$$A_p^T A_p (1,1) = n \left(l_{1p_1}^2 f_{1p_{1x}}^2 + l_{1p_1}^2 f_{1p_{1y}}^2 + l_{1p_1}^2 f_{1p_{1z}}^2\right) = l_{1p_1}^2 f_{1,1}.$$

The same operation is applied to other components of matrix $A_p^T A_p$. And to simplify the expression, unify the sequence number, namely, $l_{1p_1} = l_1, \ldots, l_{1p_p} = l_p, l_{2p_1} = l_{p+1}, \ldots, l_{2p_p} = l_{2p}, \ldots, l_{np_p} = l_{np}$, then $A_p^T A_p$ will be converted into

The analysis of the properties of $|A_p^T A_p|$ can be implemented through the knowledge of $A_p^T A_p$ above.

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

$$
A_p^T A_p = \begin{bmatrix}
l_1 l_1 f_{1,1} & \cdots & l_1 l_p f_{1,p} & l_1 l_{p+1} f_{1,p+1} & \cdots & l_1 l_{2p} f_{1,2p} & l_1 l_{2p+1} f_{1,2p+1} & \cdots & l_1 l_{3p} f_{1,3p} & \cdots & l_1 l_{(n-1)p+1} f_{1,(n-1)p+1} & \cdots & l_1 l_{np} f_{1,np} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\
l_p l_1 f_{p,1} & \cdots & l_p l_p f_{p,p} & l_p l_{p+1} f_{p,p+1} & \cdots & l_p l_{2p} f_{p,2p} & l_p l_{2p+1} f_{p,2p+1} & \cdots & l_p l_{3p} f_{p,3p} & \cdots & l_p l_{(n-1)p+1} f_{p,(n-1)p+1} & \cdots & l_p l_{np} f_{p,np} \\
l_{p+1} l_1 f_{p+1,1} & \cdots & l_{p+1} l_p f_{p+1,p} & l_{p+1} l_{p+1} f_{p+1,p+1} & \cdots & l_{p+1} l_{2p} f_{p+1,2p} & & & & & & & \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & & & & & & \\
l_{2p} l_1 f_{2p,1} & \cdots & l_{2p} l_p f_{2p,p} & l_{2p} l_{p+1} f_{2p,p+1} & \cdots & l_{2p} l_{2p} f_{2p,2p} & & & & & & & \\
l_{2p+1} l_1 f_{2p+1,1} & \cdots & l_{2p+1} l_p f_{2p+1,p} & & & & l_{2p+1} l_{2p+1} f_{2p+1,2p+1} & \cdots & l_{2p+1} l_{3p} f_{2p+1,3p} & & & & \\
\vdots & \ddots & \vdots & & & & \vdots & \ddots & \vdots & & & & \\
l_{3p} l_1 f_{3p,1} & \cdots & l_{3p} l_p f_{3p,p} & & & & l_{3p} l_{2p+1} f_{3p,2p+1} & \cdots & l_{3p} l_{3p} f_{3p,3p} & & & & \\
\vdots & \vdots & \vdots & & & & & & & \ddots & & & \\
l_{(n-1)p+1} l_1 f_{(n-1)p+1,1} & \cdots & l_{(n-1)p+1} l_p f_{(n-1)p+1,p} & & & & & & & & l_{(n-1)p+1} l_{(n-1)p+1} f_{(n-1)p+1,(n-1)p+1} & \cdots & l_{(n-1)p+1} l_{np} f_{(n-1)p+1,np} \\
\vdots & \ddots & \vdots & & & & & & & & \vdots & \ddots & \vdots \\
l_{np} l_1 f_{np,1} & \cdots & l_{np} l_p f_{np,p} & & & & & & & & l_{np} l_{(n-1)p+1} f_{np,(n-1)p+1} & \cdots & l_{np} l_{np} f_{np,np}
\end{bmatrix}
$$

If $n \geq 3$, which applies for parallel robots with no less than three "legs",

$$\left|A_p^T A_p\right| = l_1^2 \ldots l_{np}^2 F_1$$

F_1 is a scalar function of measurable variables q_{ij} and the azimuth angles of the joint axis.

For $n = 2$, which applies for parallel robots with two "legs",

$$\left|A_p^T A_p\right| = l_1^2 l_2^2 \ldots l_{2p}^2 F_2$$

F_2 is also a scalar function of measurable variables q_{ij} and the azimuth angles of the joint axis.

Therefore, $|A_p^T A_p|$ is linear in a combination of physical parameters $\theta_4 = l_1^2 l_2^2 \ldots l_{np}^2 (n \geq 2)$, and $\frac{1}{|A_p^T A_p|}$ is also linear in a combination of physical parameters $\theta_5 = \frac{1}{l_1^2 l_2^2 \ldots l_{np}^2}$. Thus

$$\left|A_p^T A_p\right| = Y_4(q, Q_a)\theta_4 \; and \; \frac{1}{\left|A_p^T A_p\right|} = Y_5(q)\theta_5 \tag{9}$$

Bring Eq. (9) back into Eq. (5),

$$\frac{J_j S_j S_p^T A_{pp} A_p^T S_{n_T} \dot{Q}_a}{\left|A_p^T A_p\right|} = Y_3\left(q, \dot{Q}_a\right)\theta_3 Y_5(q)\theta_5 = Y_6\left(q, \dot{Q}_a\right)\theta_6 \tag{10}$$

where $Y_6(q, \dot{Q}_a)$ is the regressor matrix and θ_6 is the collection of physical parameters in θ_3 and θ_5.

The second part of Eq. (3), i.e., $J_j S_j S_p^T A_p^+ A_d S_{n_T} \dot{Q}_a$ is linear in a set of physical parameters $\theta_6 = (\theta_{61}, \theta_{62}, \ldots, \theta_{6m_6})$ Substituting Eqs. (10) and (5) into Eq. (3) yields

$$\dot{x} = J_T \dot{Q}_a = Y_2\left(q, \dot{Q}_a\right)\theta_2 + Y_6\left(q, \dot{Q}_a\right)\theta_6 = Y_7\left(q, \dot{Q}_a\right)\theta_7, \tag{11}$$

where $Y_7(q, \dot{Q}_a)$ is the regressor matrix and θ_7 is the collection of physical parameters in θ_2 and θ_6.

Hence, the kinematics functions (or the kinematic model) of the proposed parallel robot is linear in a set of physical parameters $\theta_7 = (\theta_{71}, \theta_{72}, \ldots, \theta_{7m_7})$.

2.2. Dynamics analysis

The dynamic model of a parallel robot with uncertain parameters is:

$$\tau = M_1(q)\ddot{q}_a + C_1(q, \dot{q})\dot{q}_a + G_1(q) \tag{12}$$

where $\ddot{q}_a$ and $\dot{q}_a$ are the angular acceleration and angular velocity of the active joints, $M_1(q) \in R^{n \times n}$ is the inertia matrix, $C_1(q, \dot{q})\dot{q}_a \in R^n$ is a vector function containing Coriolis and centrifugal forces and $G_1(q) \in R^n$ is a vector function consisting of gravitational forces.

There are several properties for the dynamic equation:

Property 1: The inertia matrix $M_1(q)$ is symmetric and uniformly positive-definite for all $q \in R^n$.

Property 2: The matrix $(\frac{1}{2}\dot{M}_1(q) - C_1(q, \dot{q}))$ is skew-symmetric so that $v^T(\frac{1}{2}\dot{M}_1(q) - C_1(q, \dot{q}))v = 0$ for all $v \in R^n$.

Property 3: The dynamic model as described by (10) is linear in a set of physical parameters $\theta_1 = (\theta_{11}, \theta_{12}, \ldots, \theta_{1m_1})^T$ as

$$M_1(q)\ddot{q}_a + C_1(q, \dot{q})\dot{q}_a + G(q) = Y_1(q, \dot{q}, \dot{q}_a, \ddot{q}_a)\theta_1$$

where $Y_1(q, \dot{q}, \dot{q}_a, \ddot{q}_a) \in R^{n \times m}$ is called the dynamic regressor matrix.

Therefore, for the parallel robots proposed in this paper, namely, a parallel robot connected by rotational joints and with the same number of linkages in each "leg", both their kinematic and dynamic

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

models are linear in sets of physical parameters or sets of combination of physical parameters. Since all uncertain parameters in both dynamics and kinematics are those physical parameters, they can be separated and arranged into uncertain parameters vectors. Uncertain parameters in dynamics are collected in vector θ_1 and uncertain parameters in kinematics are collected in vector θ_7. Adaptive control can then be applied to estimate those uncertainties and compensate for them. And the designed controller would have robust performance with regard to uncertain dynamics and kinematics.

3. Adaptive Backstepping Control for Parallel Robot with Uncertainties in Kinematics and Dynamics

This part of this paper is focused on designing a controller that gives an asymptotical tracking result in task-space for the proposed parallel robot while it is robust to kinematics and dynamics uncertainties.

3.1. Lyapunov based design of the controller

Let e, x, x_d denote the tracking error of the end-effector, the position of the end-effector and the destination position of the end-effector. And for simplicity, replace $Y_7(q, \dot{q}_a)$ with Y_7. Then

$$e = x - x_d. \tag{13}$$

Taking the time derivative of both sides of Eq. (13) and substituting Eq. (11) into it

$$\begin{aligned} \dot{e} &= \dot{x} - \dot{x}_d \\ &= J_T \dot{q}_a - \dot{x}_d \,. \\ &= Y_7 \theta_7 - \dot{x}_d \end{aligned} \tag{14}$$

Here, backstepping control is introduced through adding and subtracting $\hat{J}_T \dot{q}_a$ and $\hat{J}_T \dot{q}_d$ on the right side of Eq. (14). $\hat{J}_T$ is the estimate of the Jacobian matrix of the parallel robot, where all uncertain elements of θ_7 in the Jacobian matrix J_T are replaced by corresponding elements in $\hat{\theta}_7$, which are the estimators of those uncertain elements in θ_7. $\hat{J}_T \dot{q}_a = Y_7 \hat{\theta}_7$. $\dot{q}_d$ is a value which can be designed to achieve specified goals. And Eq. (14) is transformed into

$$\begin{aligned} \dot{e} &= Y_7 \theta_7 - \hat{J}_T \dot{q}_a + \hat{J}_T \dot{q}_a - \hat{J}_T \dot{q}_d + \hat{J}_T \dot{q}_d - \dot{x}_d \\ &= Y_7 \theta_7 - Y_7 \hat{\theta}_7 + \hat{J}_T \rho + \hat{J}_T \dot{q}_d - \dot{x}_d \\ &= Y_7 \tilde{\theta}_7 + \hat{J}_T \rho + \hat{J}_T \dot{q}_d - \dot{x}_d \end{aligned} \quad , \tag{15}$$

where $\tilde{\theta}_7 = \theta_7 - \hat{\theta}_7$ is the error between the set of physical parameters θ_7 and the estimator of the same set of physical parameters $\hat{\theta}_7$; $\rho = \dot{q}_a - \dot{q}_d$, and take the derivative of ρ resulting in $\dot{\rho} = \ddot{q}_a - \ddot{q}_d$.

Now $\dot{q}_d$ can be designed as $\dot{q}_d = \hat{J}_T^{-1}(\dot{x}_d - k_1 e)$

$$\dot{e} = Y_7 \tilde{\theta}_7 + \hat{J}_T \rho + \dot{x}_d - k_1 e - \dot{x}_d = Y_7 \tilde{\theta}_7 + \hat{J}_T \rho - k_1 e. \tag{16}$$

Substituting $\rho = \dot{q}_a - \dot{q}_d$ and $\dot{\rho} = \ddot{q}_a - \ddot{q}_d$ into the dynamics function of the parallel robot, i.e., Eq. (12)

$$\begin{aligned} \tau &= M_1(q)\ddot{q}_a + C_1(q, \dot{q})\dot{q}_a + G(q) = M_1(q)(\dot{\rho} + \ddot{q}_d) + C_1(q, \dot{q})(\rho + \dot{q}_d) + G(q) \\ &= M_1(q)\dot{\rho} + C_1(q, \dot{q})\rho + M_1(q)\ddot{q}_d + C_1(q, \dot{q})\dot{q}_d + G(q). \end{aligned} \tag{17}$$

Applying **Property 3** to Eq. (17)

$$\tau = M_1(q)\dot{\rho} + C_1(q, \dot{q})\rho + Y_1(q, \dot{q}, \dot{q}_d, \ddot{q}_d)\theta_1 \tag{18}$$

Eq. (18) can be reformulated as

$$M_1(q)\dot{\rho} = -C_1(q, \dot{q})\rho - Y_1(q, \dot{q}, \dot{q}_d, \ddot{q}_d)\theta_1 + \tau \tag{19}$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

$\tilde{\theta}_1 = \theta_1 - \hat{\theta}_1$ is defined as the error between the set of physical parameters θ_7 and the estimator of the same set of physical parameters $\hat{\theta}_1$. The Lyapunov candidate is selected as

$$V = \frac{1}{2}e^T e + \frac{1}{2}\rho^T M_1(q)\rho + \frac{1}{2}\tilde{\theta}_1^T \beta_1 \tilde{\theta}_1 + \frac{1}{2}\tilde{\theta}_7^T \beta_2 \tilde{\theta}_7. \tag{20}$$

The derivative of the Lyapunov candidate is

$$\dot{V} = e^T \dot{e} + \rho^T M_1(q)\dot{\rho} + \frac{1}{2}\rho^T \dot{M}_1(q)\rho + \tilde{\theta}_1^T \beta_1 \dot{\tilde{\theta}}_1 + \tilde{\theta}_7^T \beta_2 \dot{\tilde{\theta}}_7. \tag{21}$$

Substituting Eqs. (16) and (18) into Eq. (21) and apply **Property 2**

$$\begin{aligned}
\dot{V} &= e^T\left(Y_7\tilde{\theta}_7 + \hat{J}_T\rho - k_1 e\right) + \rho^T\left(-C_1(q,\dot{q})\rho - Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\theta_1 + \tau\right) + \frac{1}{2}\rho^T\dot{M}_1(q)\rho \\
&\quad + \tilde{\theta}_1^T\beta_1\dot{\tilde{\theta}}_1 + \tilde{\theta}_7^T\beta_2\dot{\tilde{\theta}}_7 \\
&= e^T Y_7\tilde{\theta}_7 + e^T\hat{J}_T\rho - e^T k_1 e + \rho^T\left(-Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\theta_1 + \tau\right) + \rho^T\left(\frac{1}{2}\dot{M}_1(q) - C_1(q,\dot{q})\rho\right)\rho \\
&\quad + \tilde{\theta}_1^T\beta_1\dot{\tilde{\theta}}_1 + \tilde{\theta}_7^T\beta_2\dot{\tilde{\theta}}_7 \\
&= e^T Y_7\tilde{\theta}_7 + e^T\hat{J}_T\rho - e^T k_1 e + \rho^T\left(-Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\theta_1 + \tau\right) + \tilde{\theta}_1^T\beta_1\dot{\tilde{\theta}}_1 + \tilde{\theta}_7^T\beta_2\dot{\tilde{\theta}}_7
\end{aligned} \tag{22}$$

The input controller τ is designed as

$$\tau = Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\hat{\theta}_1 - \hat{J}_T^T e - k_2\rho. \tag{23}$$

Substituting Eq. (23) into Eq. (22)

$$\begin{aligned}
\dot{V} &= e^T Y_7\tilde{\theta}_7 + e^T\hat{J}_T\rho - e^T k_1 e + \rho^T\left(-Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\theta_1 + Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\hat{\theta}_1 - \hat{J}_T^T e - k_2\rho\right) \\
&\quad + \tilde{\theta}_1^T\beta_1\dot{\tilde{\theta}}_1 + \tilde{\theta}_7^T\beta_2\dot{\tilde{\theta}}_7 \\
&= e^T Y_7\tilde{\theta}_7 - e^T k_1 e - \rho^T Y_1(q,\dot{q},\dot{q}_d,\ddot{q}_d)\tilde{\theta}_1 - \rho^T k_2\rho - \tilde{\theta}_1^T\beta_1\dot{\hat{\theta}}_1 - \tilde{\theta}_7^T\beta_2\dot{\hat{\theta}}_7.
\end{aligned} \tag{24}$$

For simplicity, replace $Y_7(q,\dot{q},\dot{q}_d,\ddot{q}_d)$ with Y_7. The adaptation laws for $\dot{\hat{\theta}}_1$ and $\dot{\hat{\theta}}_7$ are proposed as

$$\dot{\hat{\theta}}_1 = -\frac{1}{\beta_1}Y_1^T\rho \tag{25}$$

$$\dot{\hat{\theta}}_7 = \frac{1}{\beta_2}Y_7^T e \tag{26}$$

where β_1 and β_2 are designed positive numbers.

Substituting Eqs. (25), (26) into Eq. (24)

$$\begin{aligned}
\dot{V} &= e^T Y_7\tilde{\theta}_7 - e^T k_1 e - \rho^T Y_1\tilde{\theta}_1 - \rho^T k_2\rho + \tilde{\theta}_1^T Y_1^T\rho - \tilde{\theta}_7^T Y_7^T e \\
&= -e^T k_1 e - \rho^T k_2\rho
\end{aligned}, \tag{27}$$

where $\dot{V}$ is a negative semi-definite function.

Barbalate's Lemma Corollary (sufficient condition): If a scalar function $V = V(t,x)$, is such that

- $V = V(t,x)$ is lower bounded by zero
- $\dot{V}(t,x) \le -g^2(t)$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

- $g(t) \in L_2$ and $g(t)$ is uniformly continuous in time
  Then $g(t) \to 0$, as $t \to \infty$.

It has already been proven that $\dot{V}$ is a negative semi-definite function along the trajectories of $\dot{x} = f(t, x)$ and V is a positive definite function, which mean V is decreasing and the value of V is always bigger than 0. Therefore, V could be lower bounded by 0. Moreover

$$\begin{cases} V \text{ is positive definite} \\ \dot{V} \text{ is negative semi - definite} \end{cases} \Rightarrow V \epsilon L_\infty \tag{28}$$

Under the reasonable assumption that $V(0) \epsilon L_\infty$, it can be deduced from Eq. (28) that

$$V(t) - V(0) = \int_0^t \left(\frac{1}{2} e^t e + \frac{1}{2} \rho^T M_1(q) \rho + \frac{1}{2} \tilde{\theta}_1^T \beta_1 \tilde{\theta}_1 + \frac{1}{2} \tilde{\theta}_7^T \beta_2 \tilde{\theta}_7 \right) d\omega \in L_\infty$$

$$\Rightarrow \begin{cases} e \in L_2 \\ \rho \epsilon L_2 \\ \tilde{\theta}_1 \epsilon L_2 \\ \tilde{\theta}_7 \epsilon L_2 \end{cases} \quad \text{and} \quad \begin{cases} e \in L_\infty \\ \rho \epsilon L_\infty \\ \tilde{\theta}_1 \epsilon L_\infty \\ \tilde{\theta}_7 \epsilon L_\infty \end{cases} \tag{29}$$

Equation (27) could then be rewritten as

$$\dot{V} = -e^T k_1 e - \rho^T (d - Dsgn(\rho)) - \rho^T k_2 \rho \leq -e^T k_1 e = -g^2,$$

where $g = \sqrt{k_1} e$. Take the derivative of g and substitute Eq. (16) into the derivative

$$\dot{g} = \sqrt{k_1} \dot{e} = \sqrt{k_1} \left(Y_7 \tilde{\theta}_7 + \hat{J}_T \rho - k_1 e \right).$$

According to the results in Eq. (29)

$$\dot{g} = \sqrt{k_1} \left(Y_7 \tilde{\theta}_7 + \hat{J}_T \rho - k_1 e \right) \epsilon L_\infty \Rightarrow g \textit{ is uniformly continuous}$$

$$e \in L_2 \Rightarrow g = \sqrt{k_1} e \in L_2$$

Consequently, (1) V is lower bounded by 0; (2) $\dot{V} \leq -g^2 = -e^T k_1 e$; and (3) $g = \sqrt{k_1} e \in L_2$ and g is uniformly continuous. All the conditions in **Barbalate's Lemma Corollary** are satisfied. Applying **Barbalate's Lemma Corollary** to the Lyapunov candidate V in Eq. (21) leads to the conclusion $g = \sqrt{k_1} e \to 0$, as $t \to \infty$, i.e., $e \to 0$, as $t \to \infty$.

The designed controller could achieve asymptotical tracking for the proposed parallel robot.

3.2. Verification on the implementation of the controller

Assume $\dot{x}_d$ is a given desired value and $\dot{x}_d \epsilon L_\infty$. θ_1 and θ_7 are the sets of some constant uncertain physical parameters and would never expand to infinity, thus, $\theta_1, \theta_7 \epsilon L_\infty$. Singularities in kinematics and dynamics could be avoided by the selection of the working area, which guarantees $Y_1, Y_7 \in L_\infty$, $J_T \in L_\infty$. Apply the conclusions from Eq. (29) to Eqs. (25), (26), $\tilde{\theta}_1 = \theta_1 - \hat{\theta}_1, \tilde{\theta}_7 = \theta_7 - \hat{\theta}_7$, $\dot{q}_d = \hat{J}_T^{-1} (\dot{x}_d - k_1 e)$, $\rho = \dot{q}_a - \dot{q}_d$ and $\dot{x} = J_T \dot{q}_a = Y_7 \theta_7$ gives the following results

$$\begin{cases} e \in L_\infty \\ \rho \epsilon L_\infty \\ \tilde{\theta}_1 \epsilon L_\infty \\ \tilde{\theta}_7 \epsilon L_\infty \\ \dot{x}_d \epsilon L_\infty \end{cases} \Rightarrow \begin{cases} \hat{\theta}_1 \in L_\infty \\ \hat{\theta}_7 \in L_\infty \\ \dot{\hat{\theta}}_1 \in L_\infty \\ \dot{\hat{\theta}}_7 \in L_\infty \\ \dot{q}_a \epsilon L_\infty \\ \dot{q}_d \epsilon L_\infty \end{cases} \Rightarrow \hat{J}_T \in L_\infty.$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

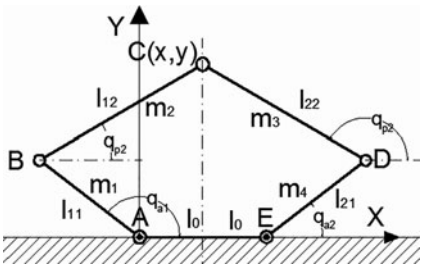

Fig. 1. 2-DOF parallel robot.

Introducing the above results into the equation $\tau = Y_1\hat{\theta}_1 - \hat{J}_T^T e - k_2\rho$

$$\tau \in l_\infty$$

.

Therefore, all those designed and measured values have been proven to be bounded.

Since, e could be measured and Y_1, Y_7 are made of measurable parameters, from Eq. (26), $\dot{\hat{\theta}}_7$ is achievable. Through integration of $\dot{\hat{\theta}}_7$, $\hat{\theta}_7$ is obtained. The value of $\hat{J}_T$ can be acquired from the equation $\hat{J}_T\dot{q}_a = Y_7\hat{\theta}_7$. $\dot{x}_d$ is a given desired value and known, $\dot{q}_d$ can be calculated through $\dot{q}_d = \hat{J}_T^{-1}(\dot{x}_d - k_1 e)$ and $\ddot{q}_d$ is calculated by taking the derivative of $\dot{q}_d$. $\dot{q}_a$ is measurable, then ρ is available from $\rho = \dot{q}_a - \dot{q}_d$. Using Eq. (25), the value of $\dot{\hat{\theta}}_1$ is accessible, and the integration of $\dot{\hat{\theta}}_1$ gives $\hat{\theta}_1$.

Consequently, all elements of the control input τ can be constructed. All designed and measured values are bounded and the control input τ is implementable. Therefore, the controller is implementable.

4. Simulations

4.1. Simulation model and its parameters

In this section, simulation results are presented to illustrate the performance of the adaptive backstepping controller. For simplicity, consider a 2-DOF parallel robot. The structure of the robot is shown in Fig. 1

A, B, C, D, E are the five revolute joints of the robot, with A and E the active joints and mounted on the ground and B and D the passive joints. C is the joint where the end-effector is located. AB, BC, CD, DE are four links of the robot with the length of l_{11}, l_{12}, l_{22}, l_{21} and weight of m_1, m_2, m_3, m_4. The distance between A and E is $2l_0$. And q_{a1}, q_{p1}, q_{p2}, q_{a2} shown in Fig. 1 are the four angles to locate the direction of the AB, BC, CD, DE links.

The structure parameters of the 2-DOF parallel robot in the simulation are designed as

$$\begin{array}{l} m_1 = 0.21kg, m_2 = 0.19kg, m_3 = 0.21kg, m_4 = 0.2kg \\ l_{11} = 0.18m, l_{12} = 0.19m, l_{22} = 0.2m, l_{21} = 0.21m, 2l_0 = 0.21m \end{array}. \quad \text{(a)}$$

For this simulation implementation, the parameter estimates are initialized as

$$\begin{array}{l} m_1 = m_2 = m_3 = m_4 = 0.2kg \\ l_{11} = l_{12} = l_{22} = l_{21} = 2l_0 = 0.2m \end{array}. \quad \text{(b)}$$

The Jacobian Matrix mapping from the angular velocities of the active joints A, E to the velocity of the end-effector is

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

$$J_T = \begin{bmatrix} -l_{11}\sin q_{a1} - \frac{l_{11}\sin q_{p1}\sin(q_{a1}-q_{p2}) + l_{11}\sin q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})} & -l_{21}\sin q_{a2} - \frac{l_{21}\sin q_{p1}\sin(q_{p2}-q_{a2}) + l_{21}\sin q_{p2}\sin(q_{p1}-q_{p2})}{\sin(q_{p2}-q_{p1})} \\ l_{11}\cos q_{a1} + \frac{l_{11}\cos q_{p1}\sin(q_{a1}-q_{p2}) + l_{11}\cos q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})} & l_{21}\sin q_{a2} + \frac{l_{21}\cos q_{p1}\sin(q_{p2}-q_{a2}) + l_{21}\cos q_{p2}\sin(q_{p1}-q_{a2})}{\sin(q_{p2}-q_{p1})} \end{bmatrix}$$

Separating the kinematic uncertain physical parameters and collecting them in the vector θ_7

$$J_T \dot{q}_a = Y_7(q, \dot{q}_a)\theta_7,$$

where

$$\theta_7 = \begin{bmatrix} l_{11} \\ l_{21} \end{bmatrix}$$

$$Y_7(q, \dot{q}_a) = \begin{bmatrix} -\dot{q}_{a1}\left(\sin q_{a1} + \frac{\sin q_{p1}\sin(q_{a1}-q_{p2}) + \sin q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})}\right) & -\dot{q}_{a2}\left(\sin q_{a2} + \frac{\sin q_{p1}\sin(q_{p2}-q_{a2}) + \sin q_{p2}\sin(q_{p1}-q_{a2})}{\sin(q_{p2}-q_{p1})}\right) \\ \dot{q}_{a1}\left(\cos q_{a1} + \frac{\cos q_{p1}\sin(q_{a1}-q_{p2}) + \cos q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})}\right) & \dot{q}_{a2}\left(\cos q_{a2} + \frac{\cos q_{p1}\sin(q_{p2}-q_{a2}) + \cos q_{p2}\sin(q_{p1}-q_{a2})}{\sin(q_{p2}-q_{p1})}\right) \end{bmatrix}.$$

The dynamic model of the proposed 2-DOF parallel robot manipulator without friction is given as

$$M_1(q)\ddot{q}_a + C_1(q, \dot{q})\dot{q}_a + G(q) = T.$$

According to **Property 3**

$$M_1(q)\ddot{q}_a + C_1(q, \dot{q})\dot{q}_a + G(q) = Y_1(q, \dot{q}, \dot{q}_d, \ddot{q}_d)\theta_1,$$

where

$$\theta_1 = \begin{bmatrix} m_1 l_{11}^2 & m_2 l_{11}^2 & m_3 l_{11}^2 & m_2 l_{11} l_{21} & m_3 l_{11} l_{21} & m_1 l_{11} & m_2 l_{11} & m_3 l_{11} & m_2 l_{21}^2 & m_3 l_{21}^2 \\ & m_4 l_{21}^2 & m_2 l_{21} & m_3 l_{21} & m_4 l_{21} & & & & & \end{bmatrix}^T$$

$$Y_1(q, \dot{q}, \dot{q}_a, \ddot{q}_a) = \begin{bmatrix} y_{11} & y_{12} & y_{13} & y_{14} & y_{15} & y_{16} & y_{17} & y_{18} & & \\ & & & y_{24} & y_{25} & & & & y_{29} & y_{210} & y_{211} & y_{212} & y_{213} & y_{214} \end{bmatrix}$$

with all the details of the elements in $Y_1(q, \dot{q}, \dot{q}_a, \ddot{q}_a)$ shown in the Appendix.

The coefficients are tuned with the value $\beta_1 = \beta_2 = k_1 = k_2 = 1$ which gives an acceptable control result. The adaptive backstepping controller is then designed as

$$\dot{\hat{\theta}}_7 = \begin{bmatrix} \dot{\hat{l}}_{11} \\ \dot{\hat{l}}_{21} \end{bmatrix} = Y_7^T e, \quad \text{with} \quad \hat{\theta}_7(0) = \begin{bmatrix} 0.2 \\ 0.2 \end{bmatrix} m$$

$$\hat{J}_T = \begin{bmatrix} -\hat{l}_{11}\sin q_{a1} - \frac{\hat{l}_{11}\sin q_{p1}\sin(q_{a1}-q_{p2}) + \hat{l}_{11}\sin q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})} & -\hat{l}_{21}\sin q_{a2} - \frac{\hat{l}_{21}\sin q_{p1}\sin(q_{p2}-q_{a2}) + \hat{l}_{21}\sin q_{p2}\sin(q_{p1}-q_{a2})}{\sin(q_{p2}-q_{p1})} \\ \hat{l}_{11}\cos q_{a1} + \frac{\hat{l}_{11}\cos q_{p1}\sin(q_{a1}-q_{p2}) + \hat{l}_{11}\cos q_{p2}\sin(q_{a1}-q_{p1})}{\sin(q_{p2}-q_{p1})} & \hat{l}_{21}\cos q_{a2} + \frac{\hat{l}_{21}\cos q_{p1}\sin(q_{p2}-q_{a2}) + \hat{l}_{21}\cos q_{p2}\sin(q_{p1}-q_{a2})}{\sin(q_{p2}-q_{p1})} \end{bmatrix}$$

$$\dot{q}_d = \hat{J}_T^{-1}(\dot{x}_d - e)$$

$$\dot{\hat{\theta}}_1 = -Y_1^T \rho, \quad \text{with} \quad \hat{\theta}_1(0)$$

$$= \begin{bmatrix} 0.008 & 0.008 & 0.008 & 0.008 & 0.008 & 0.04 & 0.04 & 0.04 & 0.008 & 0.008 & 0.008 & 0.04 & 0.04 & 0.04 \end{bmatrix}^T$$

$$\times \left(unit\ in\ kg^* m^2\ or\ kg^* m\right)$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

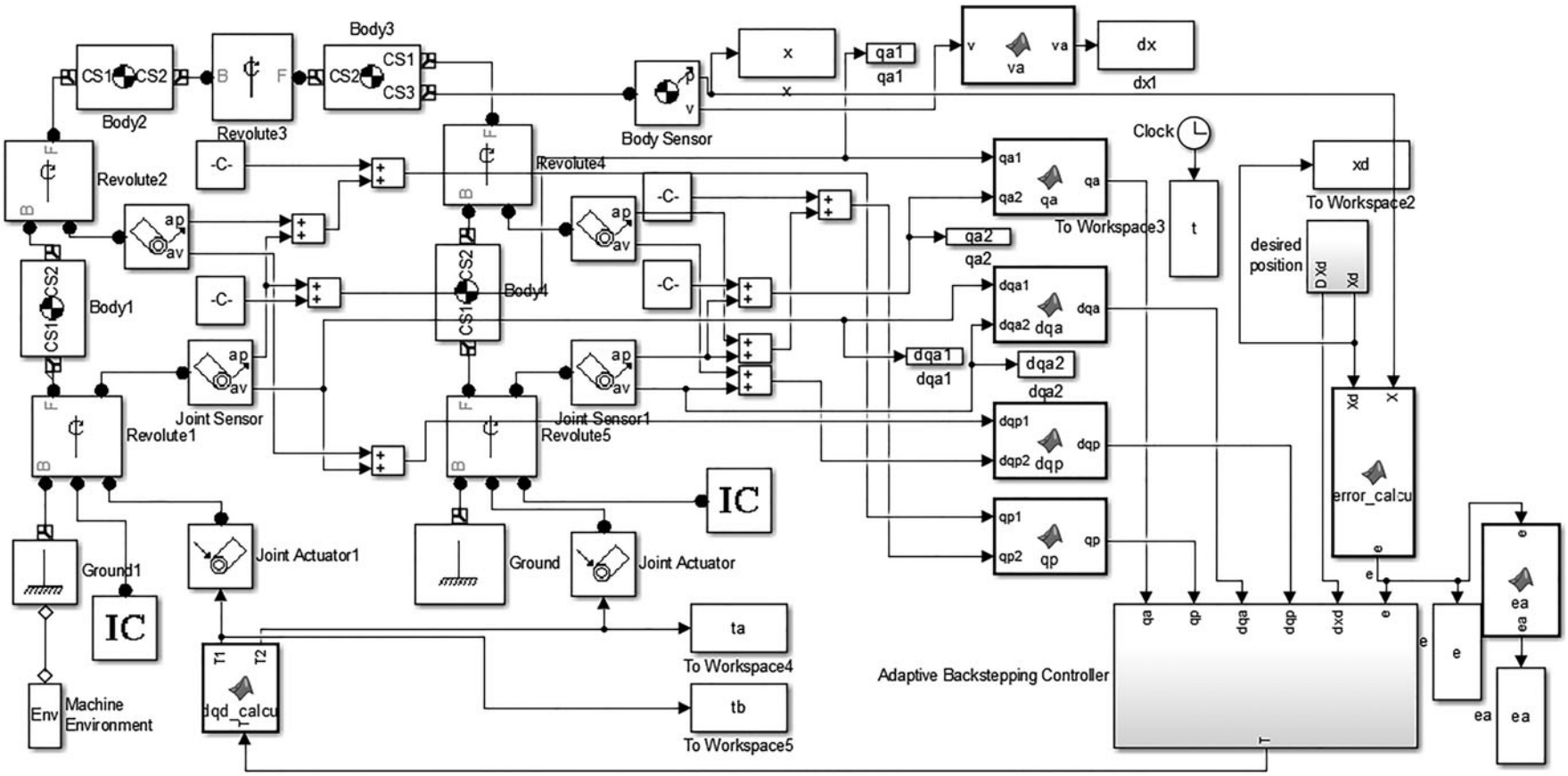

Fig. 2. Simulation block diagram for 2-DOF parallel robot controlled by adaptive backstepping controller with kinematics and dynamics uncertainties in MATLAB Simulink.

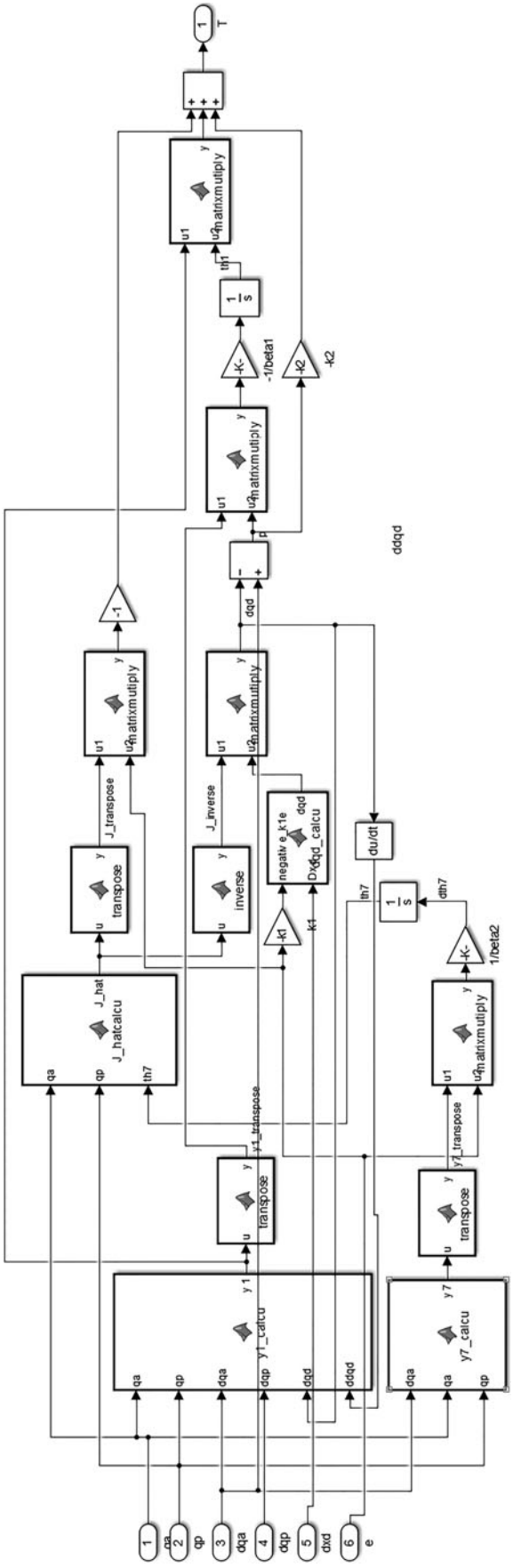

Fig. 3. Simulation block diagram for the adaptive backstepping controller.

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

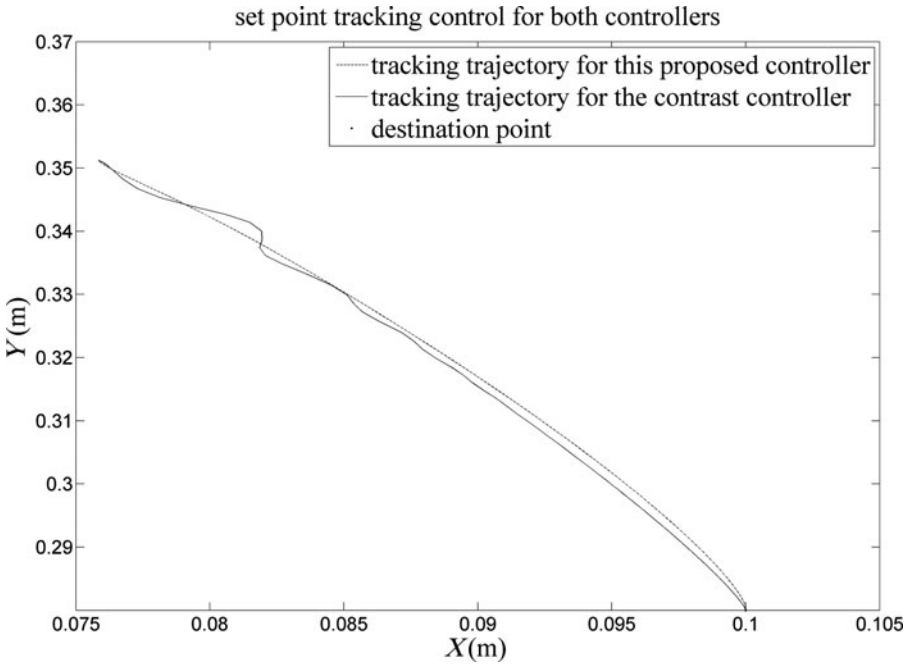

Fig. 4. Destination point and tracking trajectories for both controllers (set point tracking).

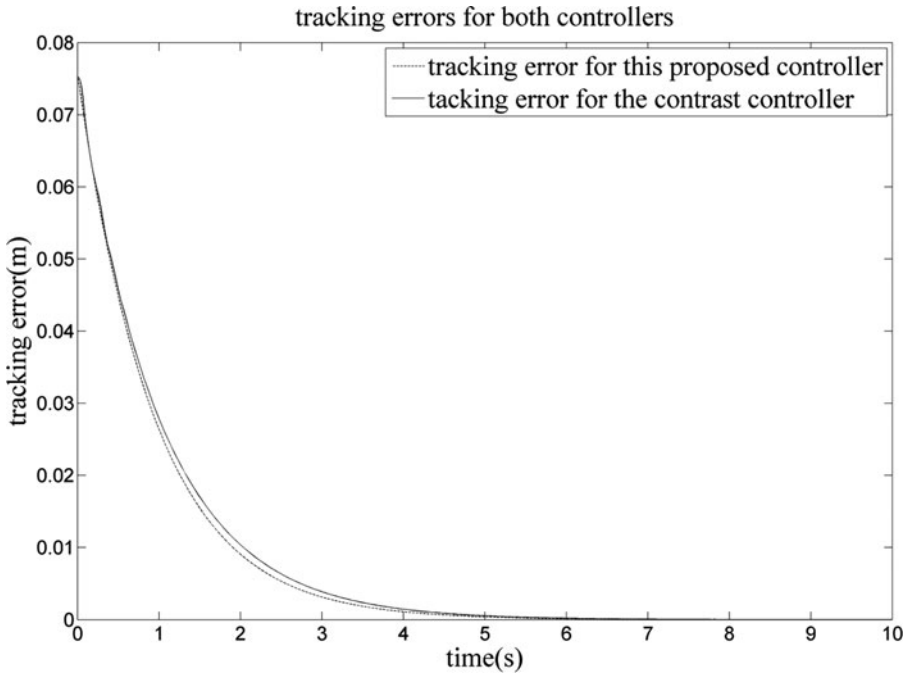

Fig. 5. Tracking errors for both controllers (set point tracking).

$$\tau = Y_1\hat{\theta}_1 - \hat{J}_T e - \rho.$$

These steps were followed and a controller was built in MATLAB SimMechanics. The simulation block of the 2-DOF parallel robot controlled by the adaptive backstepping controller is shown in Fig. 2 and the controller block is shown in Fig. 3.

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

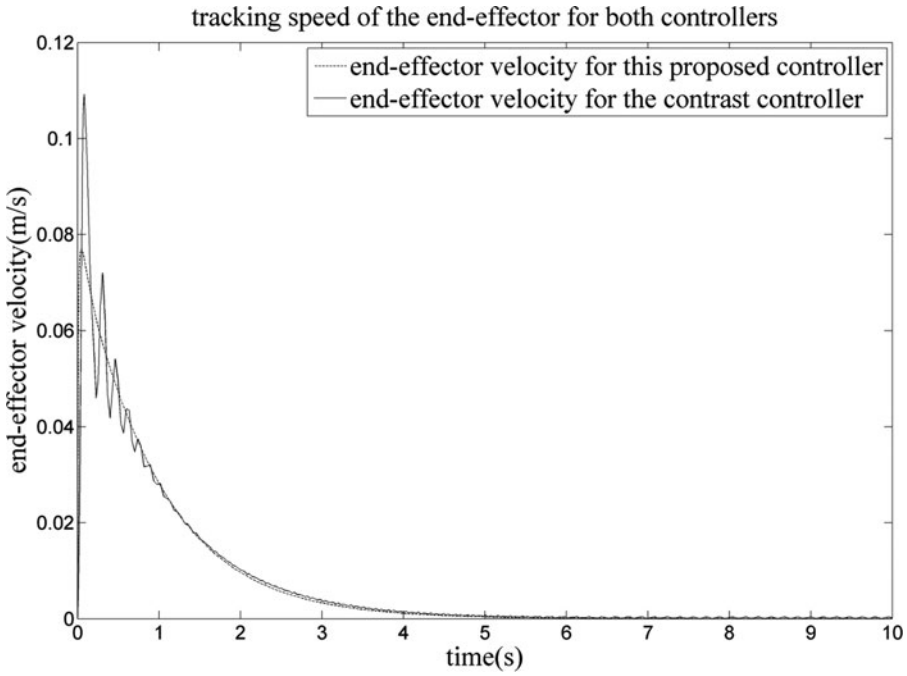

Fig. 6. Tracking speeds of the end-effector for both controllers (set point tracking).

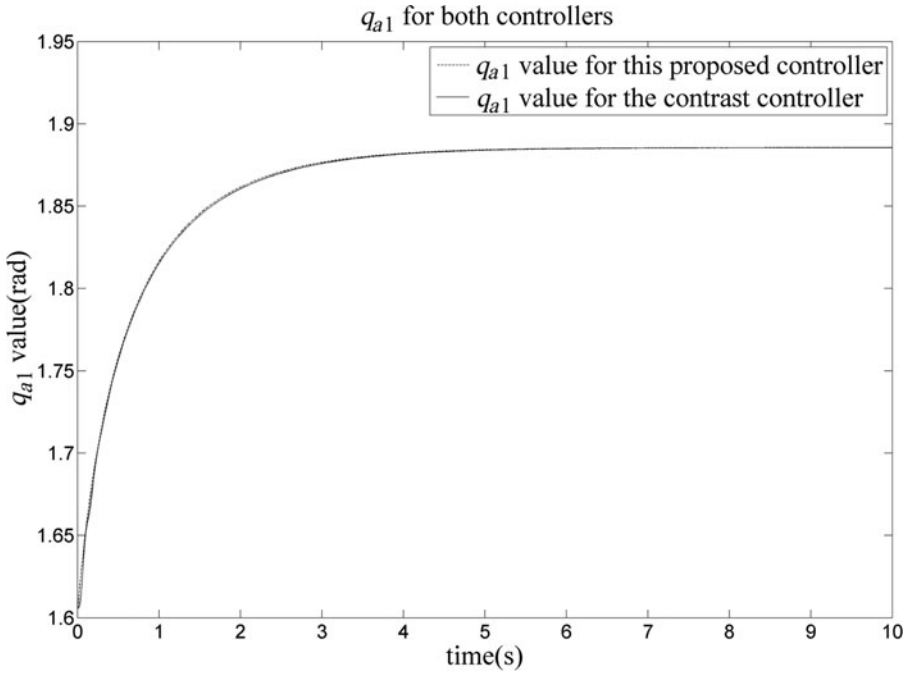

Fig. 7. The value of q_{a1} for both controllers (set point tracking).

4.2. Simulation process and result

In this paper, set point tracking control is implemented. The assignment for the controller is to adjust the input torques on the active joints so that the end-effector could eventually reach the destination point. As a contrast, set point control using the controller designed in ref. [7] is carried out on the same 2-DOF parallel robot.

Set point tracking control and trajectory tracking control are performed on the 2-DOF parallel robot with the proposed adaptive backstepping control. The desired point for set point tracking is given as $(0.1, 0.28)\,m$. The desired trajectory function for trajectory tracking control is given

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

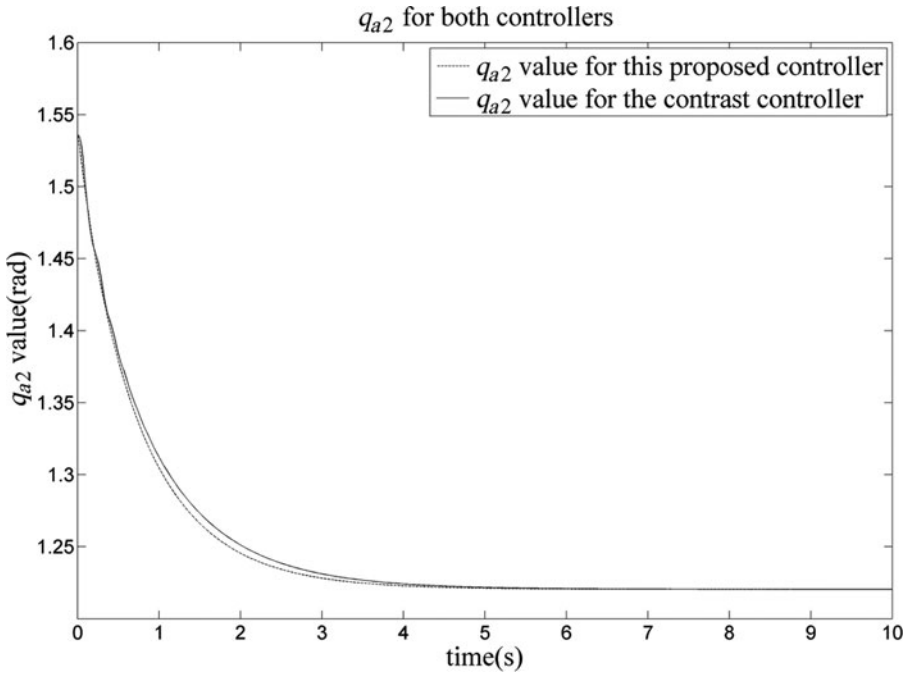

Fig. 8. The value of q_{a2} for both controllers (set point tracking).

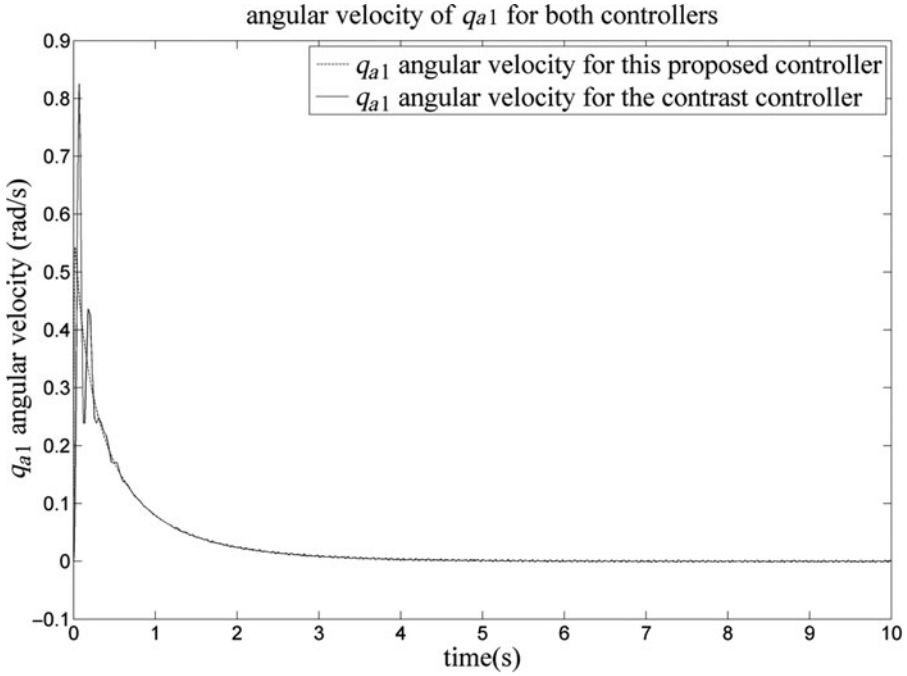

Fig. 9. The angular velocities of q_{a1} for both controllers (set point tracking).

as $x = 0.1 + 0.05\cos(t)$, $y = 0.05\sin(t)$. For set point tracking control, the destination point and tracking trajectories for the controller proposed in this paper and the contrast controller are displayed in Fig. 4. Corresponding tracking errors between the end-effector and the destination point during this process are shown in Fig. 5. The speeds of the end-effector for both controllers are shown in Fig. 6. For both controllers, the value of the angles q_{a1}, q_{a2}, the angular velocities $\dot{q}_{a1}$, $\dot{q}_{a2}$ and the input torques at the revolute joints A and E are shown in Figs. 7–12. For trajectory tracking control, the destination trajectory and tracking trajectories for both controllers are displayed in Fig. 13. Corresponding tracking errors between the end-effector and the destination trajectory are shown in

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

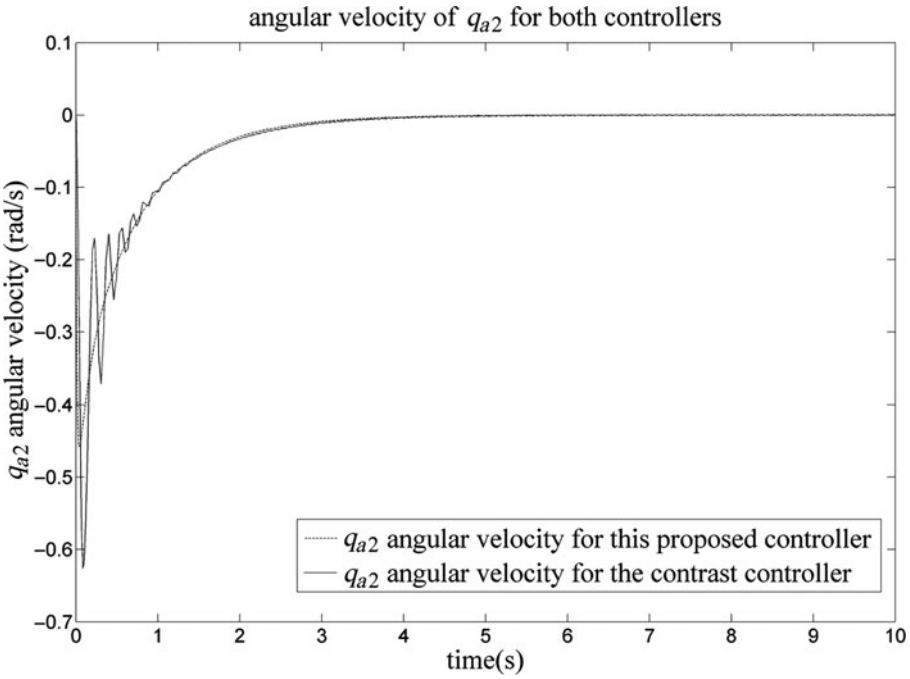

Fig. 10. The angular velocities of q_{a2} for both controllers (set point tracking).

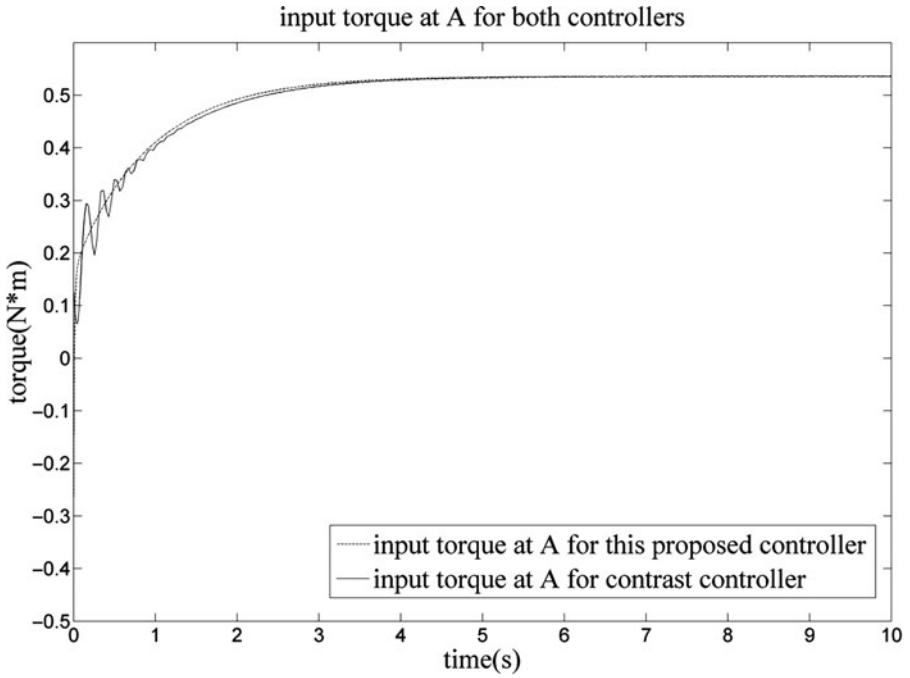

Fig. 11. Input torques at revolute joint A for both controllers (set point tracking).

Fig. 14. The speeds of the end-effector for both controllers are shown in Fig. 15. For both controllers, the value of tq_{a1}, q_{a2}, the angular velocities $\dot{q}_{a1}$, $\dot{q}_{a2}$ and input torques at the revolute joints A and E are shown in Figs. 16–21.

From the simulation results for both the adaptive backstepping controller and the controller in ref [7], it can be seen that the performance of the adaptive backstepping controller is comparable with the controller in ref. [7]. Both the adaptive backstepping controller and the contrast controller can give asymptotic tracking results. As shown in the plots of the tracking errors in Fig. 5, the adaptive backstepping controller in this paper works better for set point tracking as the tracking error converges

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

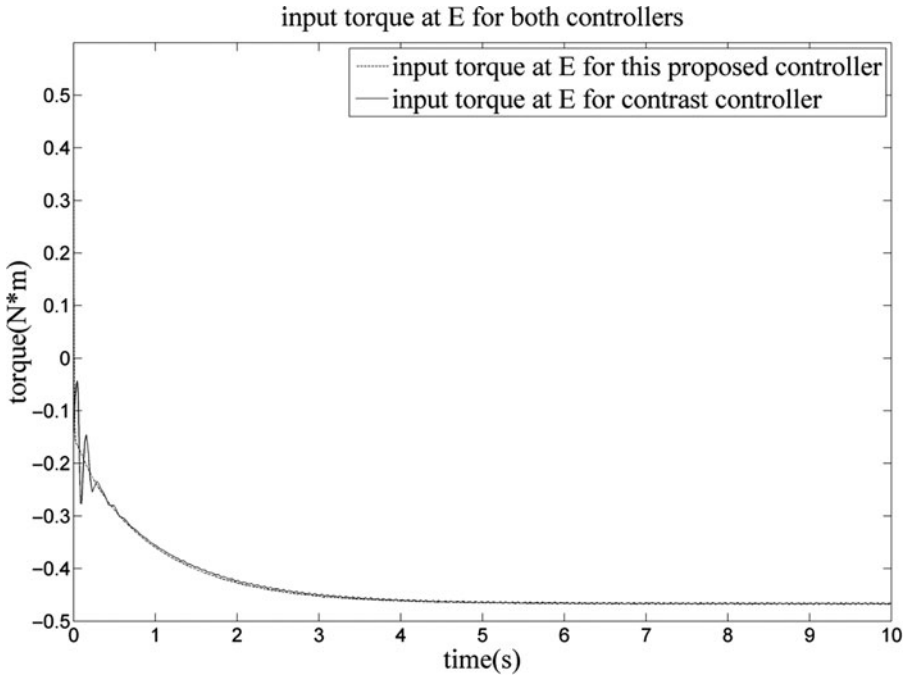

Fig. 12. Input torques at revolute joint E for both controllers (set point tracking).

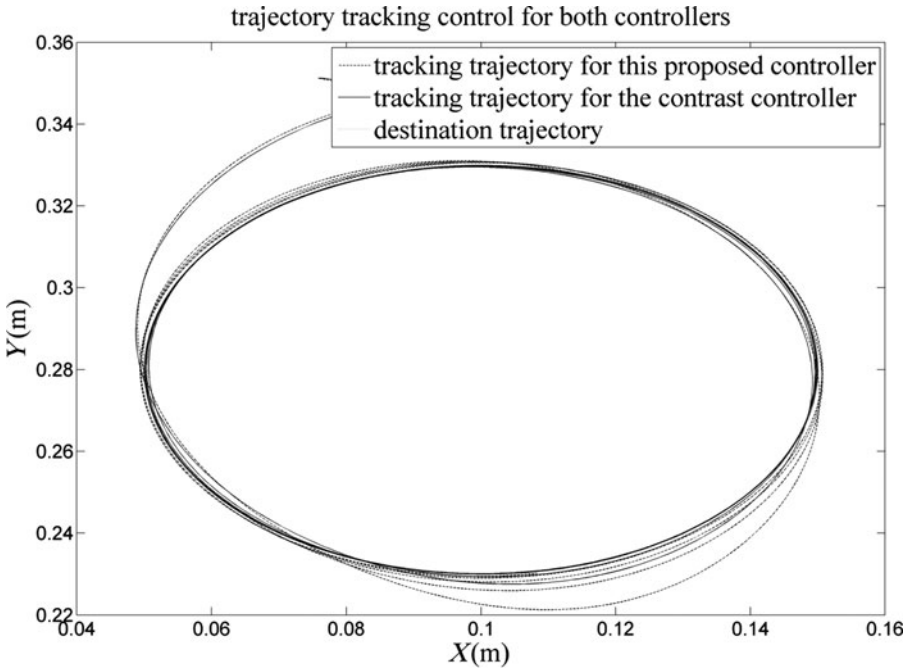

Fig. 13. Destination trajectory and tracking trajectories for both controllers (trajectory tracking).

to 0 faster. As shown in Figs. 6, 9 and 10, the working state of the system was more stable for the proposed controller, because there is less oscillation in the value of the speed of the end-effector and the angular velocities of q_{a1} and q_{a2} for the proposed controller. Meanwhile, the control output for the proposed controller in this paper is better than the contrast controller since the input torques at revolute joints A and E for the proposed controller in this paper are much smoother according to the plots in Figs. 11 and 12. For trajectory tracking, the tracking error for the contrast controller converges to 0 faster than the proposed controller based on Fig. 14. However, from Figs. 15, 18 and 19, the working state of the system for the proposed controller was more stable as there is less oscillation in

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

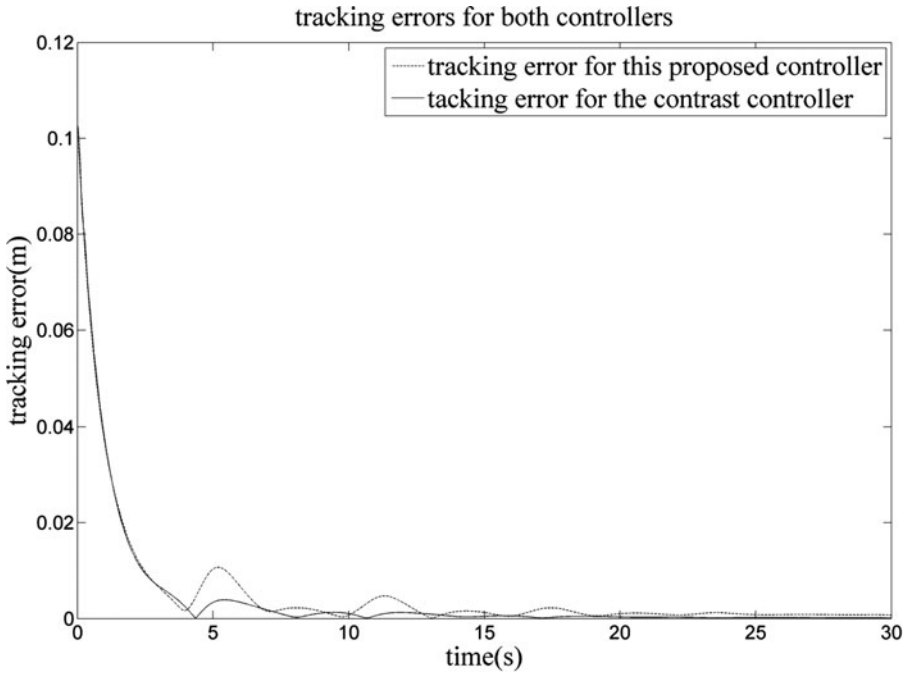

Fig. 14. Tracking errors for both controllers (trajectory tracking).

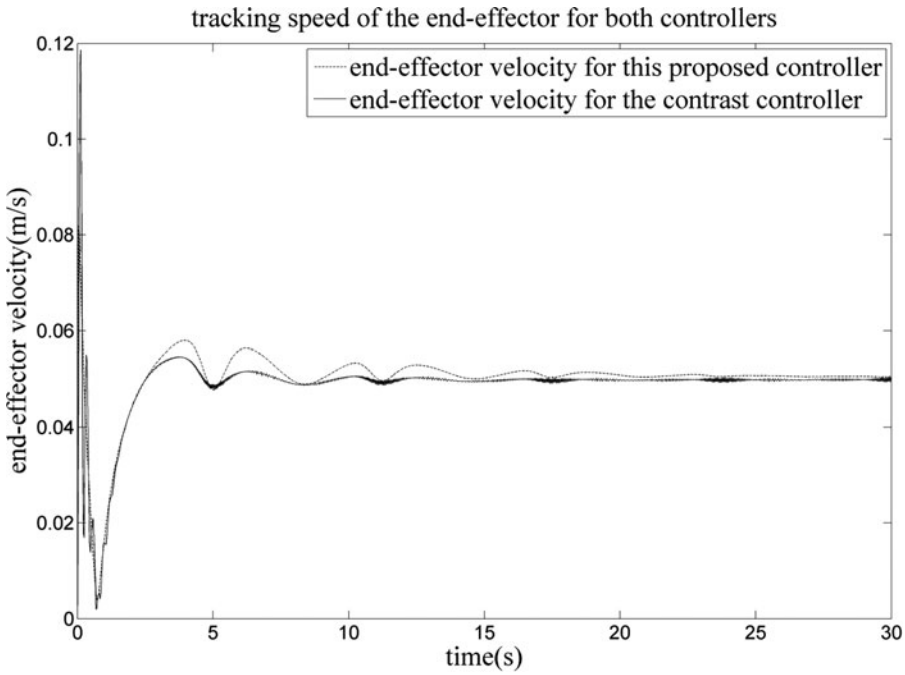

Fig. 15. Tracking speeds of the end-effector for both controllers (trajectory tracking).

the value of the speed of the end-effector and the angular velocities of q_{a1} and q_{a2} for the proposed controller. And the control output for the proposed controller is better than the contrast controller according to Figs. 20 and 21. Therefore, for trajectory tracking control, the proposed controller will be more desirable when relative stable working process is required and the demands on the performance of the executive system need to be reduced.

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

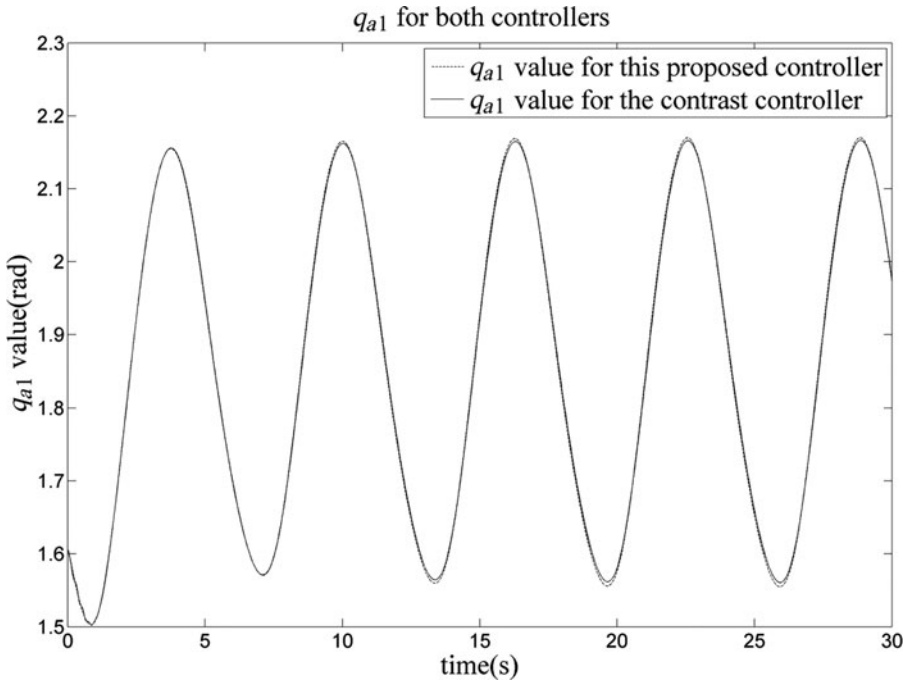

Fig. 16. The value of q_{a1} for both controllers (trajectory tracking).

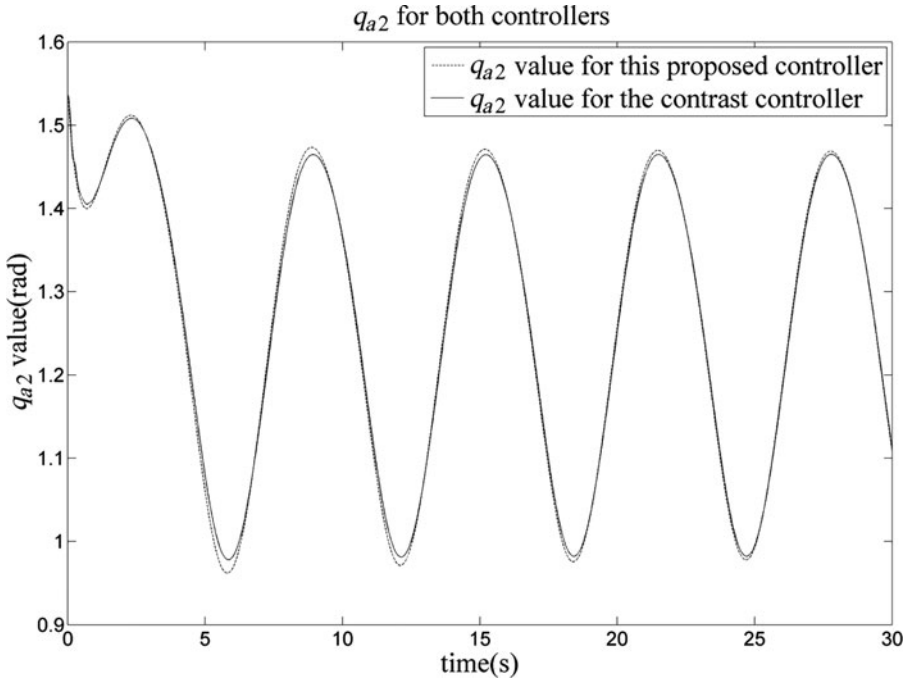

Fig. 17. The value of q_{a2} for both controllers (trajectory tracking).

5. Conclusion

In this paper, a new adaptive backstepping controller is proposed for parallel robots with uncertainties in kinematics and dynamics. Kinematic analysis on a parallel robot is carried out based on ref. [13], and leads to linearity in physical parameters for a parallel robot connected by revolute joints. Estimations are then made for both kinematic and dynamic uncertain parameters instead of just dynamic uncertainties. The backstepping variable structure and uncertain estimator are designed through Lyapunov-based analysis. Therefore, the stability of the close-loop system is guaranteed

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

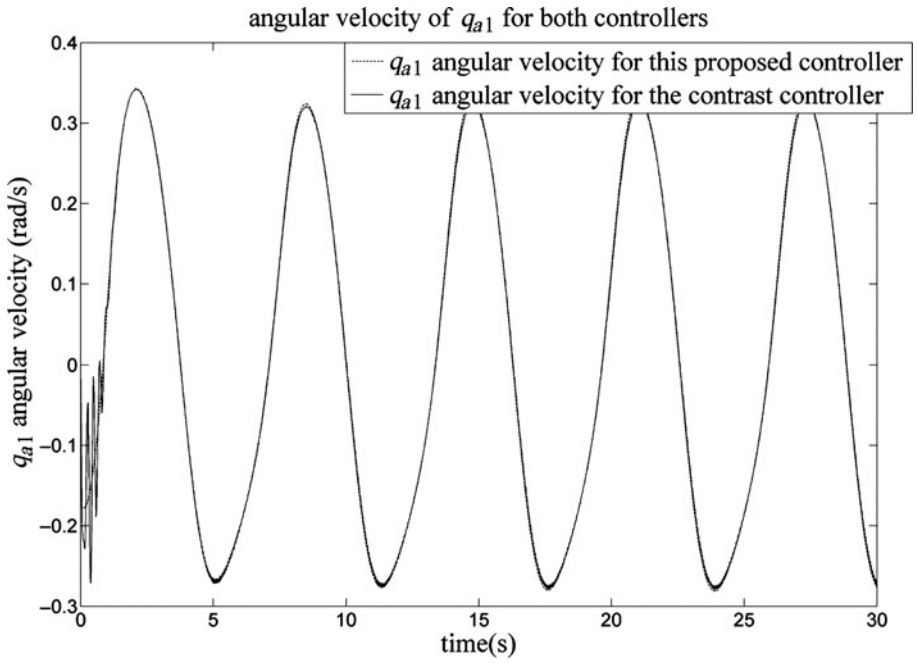

Fig. 18. The angular velocities of q_{a1} for both controllers (trajectory tracking).

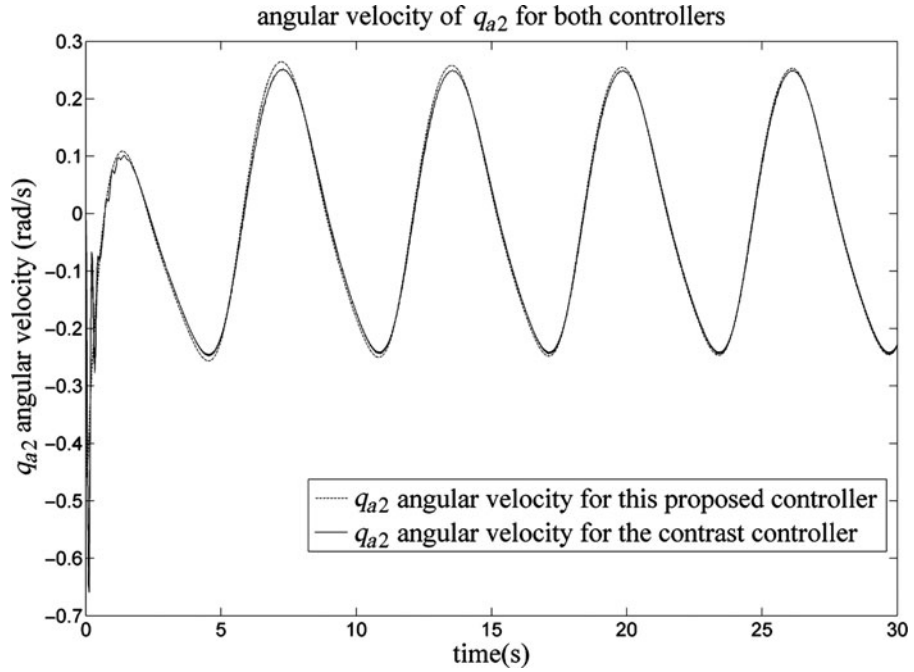

Fig. 19. The angular velocities of q_{a2} for both controllers (trajectory tracking).

and proven. The asymptotic result can be drawn through Barbalate's Lemma. From the simulation result for our proposed controller and the contrast controller in ref. [7], the adaptive backstepping controller proposed in this paper indeed gives asymptotic tracking results. The designed controller in this paper gives better results for set point tracking control as it takes less time for the tracking errors to converge to 0. And the control output for the proposed controller in this paper is better than the contrast controller in ref [7]. The contrast controller in ref. [7] provides faster convergence speed for trajectory tracking control, but the state of the working process for the proposed controller is more stable. The control output for the proposed controller for trajectory tracking control is also better

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

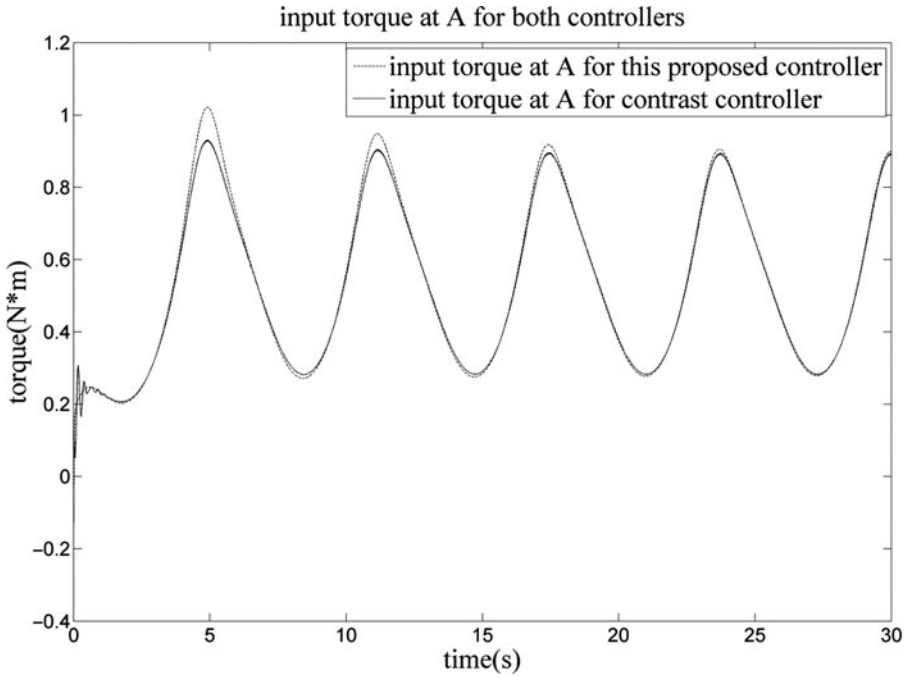

Fig. 20. Input torques at revolute joint A for both controllers (trajectory tracking).

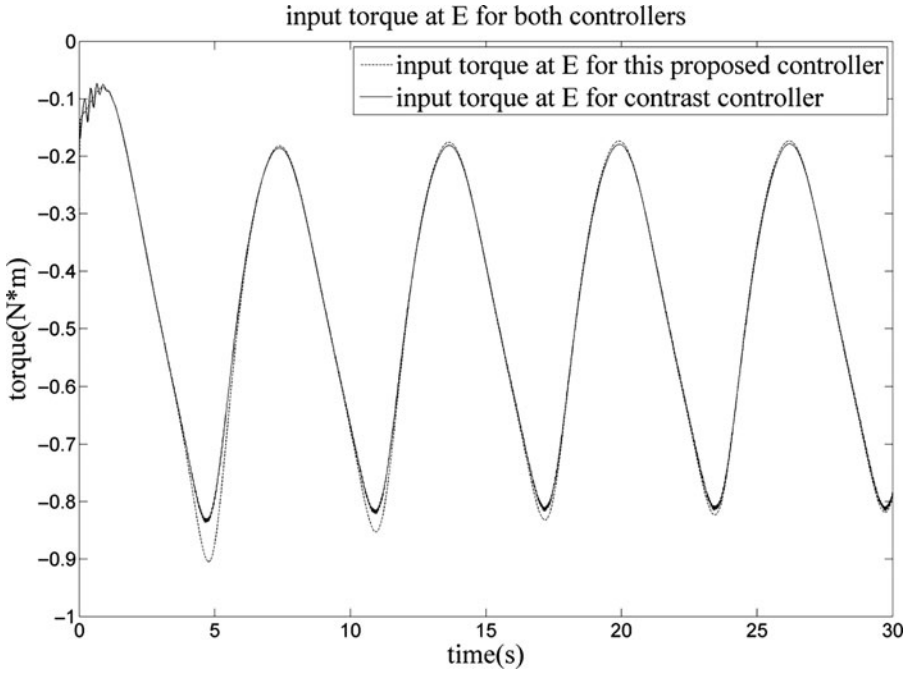

Fig. 21. Input torques at revolute joint E for both controllers (trajectory tracking).

than the contrast controller in ref. [7], which could reduce the demands on the executive system. The controller designed in this paper could have good performance in position control of the end-effector for parallel and serial robots even if their structure parameters are not known. It has great potential for fast and accurate position control.

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

Acknowledgements
The instruction in many of the techniques utilized in this paper by Profs. Warren Dixon and Carl D. Crane III is gratefully acknowledged.

References

1. W. W. Shang, S. Cong, Z. X. Li and S. L. Jiang, "Augmented nonlinear PD controller for a redundantly actuated parallel manipulator," *Adv. Robot.* **23**, 1725–1742 (2009).
2. H. Cheng, Y.-K. Yiu and Z. Li, "Dynamics and control of redundantly actuated parallel manipulators," *Trans. Mechatronics* **8**, 483–491 (2003).
3. J. Qinglong and C. Wenjie, "Adaptive Control of 6-DOF Parallel Manipulator," *30th Chinese Control Conference*, (2011) pp. 2440–2445.
4. M. Zeinali and L. Notash, "Adaptive sliding mode control with uncertainty estimator for robot manipulators," *Mech. Mach. Theory* **45**, 80–90 (2010).
5. J. Liu and X. Wang, *Advanced Sliding Mode Control for Mechanical Systems* (Tsinghua University Press, Beijing, China, 2011).
6. C. C. Cheah, C. Liu, and J. J. E. Slotine, "Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models," *IEEE Trans. Autom. Control* **51**, 1024–1029 (2006).
7. C. C. Cheah, C. Liu, and J. J. E. Slotine, "Approximate Jacobian Adaptive Control for Robot Manipulators," *IEEE International Conference on Robotics and Automation* (2004), vol. 3, pp. 3075–3080.
8. M. R. Soltanpour, J. Khalilpour and M. Soltani, "Robust nonlinear control of robot manipulator with uncertainties in kinematics, dynamics and actuator models," *Int. J. Innovative Comput. Inf. Control* **8**, 5487–5498 (2012).
9. S. Dutré, H. Bruyninckx and J. De Schutter, "The Analytical Jacobian and its Derivative for a Parallel Manipulator," *IEEE International Conference on Robotics and Automation*, Albuquerque (1997) vol. 4, pp. 2961–2966.
10. W. Deng, J.-W. Lee and H.-J. Lee, "Kinematics Simulation and Control of a New 2 DOF Parallel Mechanism Based on Matlab/SimMechanics," *ISECS International Colloquium on Computing, Communication, Control, and Management* (2009) vol. 3, pp. 233–236.
11. T. D. Le, H.-J. Kang and Y.-S. Ro, "Kinematic and Singularity Analysis of Symmetrical 2 DOF Parallel Manipulators," *7th International Forum on Strategic Technology* (2012) pp. 1–4.
12. W. E. Dixon, "Adaptive Regulation of Amplitude Limited Robot Manipulators with Uncertain Kinematics and Dynamics," *Proceedings of the American Control Conference* (2004) pp. 3844–3939.
13. M. Ahmadipour, A. Khayatian, and M. Dehghani, "Adaptive Backstepping Control of Rigid Link Electrically Driven Robots with Uncertain Kinematics and Dynamics," *2nd International Conference on Control, Instrumentation and Automation* (2011) pp. 911–916.

Appendix

The details of the elements in $Y_1(q, \dot{q}, \dot{q}_a, \ddot{q}_a)$

$$y_{11} = \frac{1}{3}\ddot{q}_{d1}, \quad y_{12} = \ddot{q}_{d1} + y_{12}(1) + \frac{\dot{q}_{d1}}{\sin(q_{p2} - q_{p1})}(y_{12}(2) + y_{12}(3) - y_{12}(4)),$$

$$y_{13} = \frac{\ddot{q}_{d1} sin^2(q_{a1} - q_{p1})}{3\sin^2(q_{p2} - q_{p1})} + \frac{\dot{q}_{d1}(y_{13}(1) + y_{13}(2))}{3\sin^3(q_{p2} - q_{p1})}$$

$$y_{14} = \frac{\ddot{q}_{d2}}{\sin(q_{p2} - q_{p1})} y_{14}(1) + \frac{\dot{q}_{d2}}{\sin(q_{p2} - q_{p1})}(y_{14}(2) + y_{14}(3) - y_{14}(4)),$$

$$y_{15} = y_{15}(1) + \frac{\dot{q}_{d2}(y_{15}(2) + y_{15}(3))}{3\sin^3(q_{p2} - q_{p1})}, \quad y_{16} = \frac{1}{2} g cos q_{a1}$$

$$y_{17} = \frac{g cos q_{p1} sin(q_{a1} - q_{p2})}{2\sin(q_{p2} - q_{p1})} + g cos q_{a1}, \quad y_{18} = \frac{g cos q_{p2} \sin(q_{a1} - q_{p1})}{2\sin(q_{p2} - q_{p1})},$$

$$y_{29} = y_{29}(1) + \frac{\dot{q}_{d2}(y_{29}(2) + y_{29}(3))}{3\sin^3(q_{p2} - q_{p1})}$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

$$y_{210} = \ddot{q}_{d2} + \frac{\ddot{q}_{d2}}{\sin(q_{p2} - q_{p1})} y_{210}(1) + \frac{\dot{q}_{d2}}{\sin(q_{p2} - q_{p1})}(y_{210}(2) + y_{210}(3) - y_{210}(4))$$

$$y_{211} = \frac{1}{3}\ddot{q}_{d2}, \quad y_{24} = y_{24}(1) + \frac{\dot{q}_{d1}(y_{24}(2) + y_{24}(3))}{3\sin^3(q_{p2} - q_{p1})},$$

$$y_{25} = \frac{\ddot{q}_{d1}}{\sin(q_{p2} - q_{p1})} y_{25}(1) + \frac{\dot{q}_{d1}}{\sin(q_{p2} - q_{p1})}(y_{25}(2) + y_{25}(3) - y_{25}(4))$$

$$y_{212} = \frac{gcosq_{p1}\sin(q_{p2} - q_{a2})}{2\sin(q_{p2} - q_{p1})}, \quad y_{213} = \frac{gcosq_{p2}\sin(q_{p1} - q_{a2})}{2\sin(q_{p2} - q_{p1})} + gcosq_{a2}, \quad y_{214} = \frac{1}{2}gcosq_{a2}$$

$$y_{12}(1) = \frac{\ddot{q}_{d1}}{\sin(q_{p2} - q_{p1})}\left(\frac{1}{2}\cos(q_{p1} - q_{a1})sin(q_{a1} - q_{p1}) + \frac{\sin^2(q_{a1} - q_{p2})}{3\sin(q_{p2} - q_{p1})}\right)$$

$$y_{12}(2) = \dot{q}_{a1}\left(\frac{1}{2}\cos(q_{p1} - q_{a1})\cos(q_{p2} - q_{a1}) + \frac{\sin(q_{a1} - q_{p2})\cos(q_{p2} - q_{a1})}{3\sin(q_{p2} - q_{p1})}\right)$$

$$y_{12}(3) = \dot{q}_{p1}\left(\frac{\sin(q_{a1} - q_{p1})\sin(q_{a1} - q_{p2})}{3\sin^2(q_{p2} - q_{p1})} + \frac{\cos(q_{a1} - q_{p2})\sin(q_{a1} - q_{p2})}{6\sin(q_{p2} - q_{p1})}\right)$$

$$y_{12}(4) = \dot{q}_{p2}\left(\frac{\cos(q_{p1} - q_{a1})\sin(q_{a1} - q_{p1})}{2\sin(q_{p2} - q_{p1})} + \frac{\sin(q_{a1} - q_{p2})\sin(q_{a1} - q_{p1})}{3\sin^2(q_{p2} - q_{p1})}\right)$$

$$y_{13}(1) = \dot{q}_{a1}\sin(q_{a1} - q_{p1})\cos(q_{p1} - q_{a1})\sin(q_{p2} - q_{p1})$$

$$y_{13}(2) = \dot{q}_{p1}\sin(q_{a1} - q_{p1})\sin(q_{a1} - q_{p2}) - \dot{q}_{p2}sin^2(q_{a1} - q_{p1})\cos(q_{p2} - q_{p1})$$

$$y_{14}(1) = \frac{1}{2}\cos(q_{p1} - q_{a1})\sin(q_{p2} - q_{a2}) + \frac{\sin(q_{a1} - q_{p2})\sin(q_{p2} - q_{a2})}{3\sin(q_{p2} - q_{p1})}$$

$$y_{14}(2) = -\dot{q}_{a2}\left(\frac{1}{2}\cos(q_{p1} - q_{a1})\cos(q_{p2} - q_{a2}) + \frac{\sin(q_{a1} - q_{p2})\cos(q_{p2} - q_{a2})}{3\sin(q_{p2} - q_{p1})}\right)$$

$$y_{14}(3) = \dot{q}_{p1}\left(\frac{\sin(q_{a1} - q_{p1})\sin(q_{p2} - q_{a2})}{3\sin^2(q_{p2} - q_{p1})} + \frac{\cos(q_{a1} - q_{p2})\sin(q_{p2} - q_{a2})}{6\sin(q_{p2} - q_{p1})}\right)$$

$$y_{14}(4) = \dot{q}_{p2}\left(\frac{\cos(q_{p1} - q_{a1})\sin(q_{p1} - q_{a2})}{2\sin(q_{p2} - q_{p1})} + \frac{\sin(q_{a1} - q_{p2})\sin(q_{p1} - q_{a2})}{3\sin^2(q_{p2} - q_{p1})}\right)$$

$$y_{15}(1) = \frac{\ddot{q}_{d2}\sin(q_{a1} - q_{p1})\sin(q_{p1} - q_{a2})}{3\sin^2(q_{p2} - q_{p1})},$$

$$y_{15}(2) = -\dot{q}_{a2}\sin(q_{a1} - q_{p1})\cos(q_{p1} - q_{a2})\sin(q_{p2} - q_{p1})$$

$$y_{15}(3) = \dot{q}_{p1}\sin(q_{a1} - q_{p1})\sin(q_{p2} - q_{a2}) - \dot{q}_{p2}sin(q_{a1} - q_{p1})\cos(q_{p2} - q_{p1})\sin(q_{p1} - q_{a2})$$

$$y_{29}(1) = \frac{\ddot{q}_{d2}sin^2(q_{p2} - q_{a2})}{3\sin^2(q_{p2} - q_{p1})}, \quad y_{29}(2) = -\dot{q}_{a2}\sin(q_{p2} - q_{a2})\cos(q_{p2} - q_{a2})\sin(q_{p2} - q_{p1})$$

$$y_{29}(3) = \dot{q}_{p1}\sin^2(q_{p2} - q_{a2})\cos(q_{p2} - q_{p1}) - \dot{q}_{p2}\sin(q_{p2} - q_{a2})\sin(q_{p1} - q_{a2})$$

$$y_{210}(1) = \frac{1}{2}\cos(q_{a2} - q_{p2})sin(q_{p1} - q_{a2}) + \frac{\sin^2(q_{p1} - q_{a2})}{3\sin(q_{p2} - q_{p1})}$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press

$$y_{210}(2) = -\dot{q}_{a2}\left(\frac{1}{2}\cos(q_{a2}-q_{p2})\cos(q_{p1}-q_{a2}) + \frac{\sin(q_{p1}-q_{a2})\cos(q_{p1}-q_{a2})}{3\sin(q_{p2}-q_{p1})}\right)$$

$$y_{210}(3) = \dot{q}_{p1}\left(\frac{\cos(q_{a2}-q_{p2})\sin(q_{p2}-q_{a2})}{2\sin(q_{p2}-q_{p1})} + \frac{\sin(q_{p1}-q_{a2})\sin(q_{p2}-q_{a2})}{3\sin^2(q_{p2}-q_{p1})}\right)$$

$$y_{210}(4) = \dot{q}_{p2}\left(\frac{\cos(q_{p1}-q_{a2})\sin(q_{p1}-q_{a2})}{6\sin(q_{p2}-q_{p1})} + \frac{\sin(q_{p2}-q_{a2})\sin(q_{p1}-q_{a2})}{3\sin^2(q_{p2}-q_{p1})}\right)$$

$$y_{24}(1) = \frac{\ddot{q}_{d1}\sin(q_{p2}-q_{a2})\sin(q_{a1}-q_{p1})}{3\sin^2(q_{p2}-q_{p1})},$$

$$y_{24}(2) = \dot{q}_{a1}\sin(q_{p2}-q_{a2})\cos(q_{p2}-q_{a1})\sin(q_{p2}-q_{p1})$$

$$y_{24}(3) = \dot{q}_{p1}\sin(q_{p2}-q_{a2})\cos(q_{p2}-q_{p1})\sin(q_{a1}-q_{p2}) - \dot{q}_{p2}sin(q_{p2}-q_{a2})\sin(q_{a1}-q_{p1})$$

$$y_{25}(1) = \frac{1}{2}\cos(q_{a2}-q_{p2})\sin(q_{a1}-q_{p1}) + \frac{\sin(q_{p1}-q_{a2})\sin(q_{a1}-q_{p1})}{3\sin(q_{p2}-q_{p1})}$$

$$y_{25}(2) = \dot{q}_{a1}\left(\frac{1}{2}\cos(q_{a2}-q_{p2})\cos(q_{p1}-q_{a1}) + \frac{\sin(q_{p1}-q_{a2})\cos(q_{p1}-q_{a1})}{3\sin(q_{p2}-q_{p1})}\right)$$

$$y_{25}(3) = \dot{q}_{p1}\left(\frac{\cos(q_{a2}-q_{p2})\sin(q_{a1}-q_{p2})}{2\sin(q_{p2}-q_{p1})} + \frac{\sin(q_{p1}-q_{a2})\sin(q_{a1}-q_{p2})}{3\sin^2(q_{p2}-q_{p1})}\right)$$

$$y_{25}(4) = \dot{q}_{p2}\left(\frac{\cos(q_{p1}-q_{a2})\sin(q_{a1}-q_{p1})}{6\sin(q_{p2}-q_{p1})} + \frac{\sin(q_{p2}-q_{a2})\sin(q_{a1}-q_{p1})}{3\sin^2(q_{p2}-q_{p1})}\right).$$

https://doi.org/10.1017/S0263574714002410 Published online by Cambridge University Press