Robotica (2014) volume 32, pp. 1153–1169. © Cambridge University Press 2014
doi:10.1017/S0263574713001203

Adaptive control of rigid-link electrically driven robots with parametric uncertainties in kinematics and dynamics and without acceleration measurements

Mahboubeh Ahmadipour, Alireza Khayatian* and Maryam Dehghani

Department of Power and Control Engineering, Shiraz University, Shiraz, Iran

(Accepted November 25, 2013. First published online: January 20, 2014)

SUMMARY
In this paper, the backstepping strategy is used to design an adaptive tracking controller for rigid-link electrically driven robots in the presence of parametric uncertainties in kinematics, manipulator dynamics, and actuator dynamics. To avoid acceleration measurements, two techniques are exploited. One technique adds compensation control terms to the control law signal. The other uses a linear in variable property of the Jacobian matrix. Global asymptotic convergence of the end-effector motion tracking errors is shown via Lyapunov analysis. Simulation results are presented to show the effectiveness of the proposed control scheme.

KEYWORDS: Adaptive control; RLED robot; Uncertain kinematics; Uncertain actuator dynamics; Uncertain manipulator dynamics; Adaptive observer.

1. Introduction

Physical parameters of robot manipulators are not exactly known. In addition, robot manipulators pick up various tools to perform tasks, hence the kinematics and dynamics of robot manipulators are uncertain.

Many control schemes have been proposed for robot manipulators with uncertain dynamics.[1–8] The objective of most of these controllers is joint-space tracking. However, in most cases the desired path for the robot is designed in Cartesian space in terms of position and motion features of the end-effector. In the literature, two approaches were suggested to achieve task-space control. The first one needs to solve inverse kinematics problem to transform the desired task-space references into the corresponding joint-space references and uses joint-space controllers. This method has a disadvantage that solving inverse kinematics problem increases the computational load of joint-space controllers. In addition, due to kinematics uncertainties, the joint-space references cannot be exactly achieved. The second approach suggests designing control scheme directly in the task-space. The early proposed controllers based on the second approach assume that the exact Jacobian matrix is known. However, the Jacobian matrix is not known in the presence of kinematics uncertainties.

Several task-space controllers based on the second approach were proposed to overcome the problem of uncertain kinematics.[9–16] All these controllers neglect the actuator dynamics and they are designed at the torque input level. As demonstrated in ref. [17], the inclusion of actuator dynamics is important especially in high-speed movements or highly varying loads. Therefore, ignoring the actuator dynamics may cause these controllers result in degraded performance or instability problems.

* Corresponding author. E-mail: khayatia@shirazu.ac.ir

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

Considering actuator dynamics complicates both controller design and stability analysis. Liu and Cheah[18] and Liu and Poignet[19] proposed two controllers for rigid-link electrically driven (RLED) robots in the presence of kinematics and dynamics uncertainties. These controllers consider only the regulation problem. Liu *et al.*[20] proposed an adaptive Jacobian tracking controller for RLED robots in the presence of parametric uncertainties in kinematics and dynamics.[20] This controller belongs to the passivity-based controllers, which adopted approximate transpose Jacobian feedback to accomplish the end-point tracking of the manipulator. However, the performance of the transpose Jacobian controller is not satisfactory over the entire workspace of the robot and the dynamic response of such controllers will vary with the arm configuration.[14,21]

In this paper, an adaptive task-space tracking controller for RLED robots with parametric uncertainties in kinematics and dynamics is proposed. This controller is designed by backstepping algorithm in two steps. At first, the armature current vector is viewed as the input for the manipulator subsystem and a desired armature current vector is designed based on the inverse manipulator dynamics. At the second step, an input control voltage is designed so that the armature current vector tracks the desired armature current vector.

Ordinary inverse dynamics controllers for robot manipulators need acceleration measurements for parameters adaptation. To avoid acceleration measurements for parameters adaptation, compensation control terms are added to the desired armature current vector. Observed joint-space velocity and a new property of the Jacobian matrix are used to avoid the use of acceleration in the input control voltage.

The paper is organized as follows: in Section 2, the robot kinematics and dynamics and some important properties are given. In Section 3, an adaptive controller is proposed and the stability of the closed-loop system is shown. In Section 4, simulation results of the proposed controller applied on a two-link RLED robot are presented and compared with ref. [20]. The simulation results show that in similar accuracies the suggested controller can reduce the amplitude and deviation of the input control voltage. Section 5 concludes the paper.

2. Robot Kinematics and Dynamics

The dynamic model for the RLED robot manipulator with n degrees of freedom is given by:[22]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = K_N I \tag{1}$$

$$L\dot{I} + RI + K_E\dot{q} = u \tag{2}$$

where $q \in R^n$ is the joint-space position vector and $M(q) \in R^{n\times n}$ is the inertia matrix. $C(q,\dot{q})\dot{q} \in R^n$ is the vector of centripetal and coriolis forces. $g(q)$ is the vector of gravitational forces. $I \in R^n$ is the vector of armature currents. $u \in R^n$ denotes the vector of armature voltages. $K_N \in R^{n\times n}$ is a positive definite constant diagonal matrix that characterizes the electromechanical conversion between current and torque. $L \in R^{n\times n}$ and $R \in R^{n\times n}$ are the actuators inductance matrix and the actuators resistance matrix, respectively. $K_E \in R^{n\times n}$ is the matrix that characterizes the voltage constant of the actuators. L, R, and K_E are positive definite constant diagonal matrices. In this paper, Eqs. (1) and (2) are called the manipulator subsystem and the actuator subsystem, respectively.

It is straightforward to write Eq. (1) as follows:

$$\bar{M}(q)\ddot{q} + \bar{C}(q,\dot{q})\dot{q} + \bar{g}(q) = I \tag{3}$$

where

$$\bar{M}(q) = K_N^{-1}M(q) \tag{4}$$

$$\bar{C}(q,\dot{q}) = K_N^{-1}C(q,\dot{q}) \tag{5}$$

$$\bar{g}(q) = K_N^{-1}g(q) \tag{6}$$

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

The relation between joint-space position vector $q \in R^n$ and task-space position vector $x \in R^m$ for a general robot manipulator is as follows:[23]

$$x = h(q) \tag{7}$$

where $h(.) \in R^n \rightarrow R^m$ is generally a nonlinear transformation.

Differentiating Eq. (7) with respect to time, the relation between joint-space velocity $\dot{q}$ and task-space velocity $\dot{x}$ is obtained:

$$\dot{x} = J(q)\dot{q} \tag{8}$$

where $J(q) = \frac{\partial h}{\partial q} \in R^{m \times n}$ is the Jacobian matrix.

In the following, some important properties of the robot dynamics and kinematics that will be used in the controller design are given:

Property 1. The inertia matrix $M(q)$ is symmetric and positive definite and satisfies:[24]

$$\beta_1 \leq \|M(q)\| \leq \beta_2(q) \tag{9}$$

where β_1 is a positive constant.

Property 2. Matrix $\frac{1}{2}\dot{M}(q) - C(q, \dot{q})$ is skew-symmetric.[24]

Property 3. The Matrix $C(q, \dot{q})$ satisfies:[24]

$$C(q, x)y = C(q, y)x \quad \forall x, y \in R^n \tag{10}$$

Property 4. The matrix $C(q, \dot{q})$ satisfies:[24]

$$\|C(q, \dot{q})\| \leq c_0(q)\|\dot{q}\| \tag{11}$$

Property 5. The gravity vector $g(q)$ satisfies:[24]

$$\|g(q)\| \leq g_0(q) \tag{12}$$

$\beta_2(q)$, $c_0(q)$, and $g_0(q)$ in the above properties are known scalar functions. If the robot has only revolute joints, these functions are constant. If the robot has some prismatic joints, these functions may depend on q.

Property 6. It is assumed that the electromechanical conversion matrix K_N is upperbounded and lowerbounded, which means:

$$\gamma_1 \leq \|K_N\| \leq \gamma_2 \tag{13}$$

where γ_1 and γ_2 are positive constants.

Property 7. The right-hand side of Eq. (8) is linear in a set of kinematic parameters $a_k = (a_{k1}, a_{k2}, \ldots, a_{kj})^T$ such as link lengths. Hence, Eq. (8) can be expressed as:[12]

$$\dot{x} = J(q)\dot{q} = Y_k(q, \dot{q})a_k \tag{14}$$

where $Y_k(q, \dot{q}) \in R^{m \times j}$ is called the kinematic regressor matrix.

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

Property 8. The left-hand side of Eq. (1) is linear in a set of dynamic parameters $a_d = (a_{d1}, a_{d2}, \ldots, a_{dp})^T$:[23]

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = Y_d(q, \dot{q}, \dot{q}, \ddot{q})a_d \tag{15}$$

where $Y_d(q, \dot{q}, \dot{q}, \ddot{q}) = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \in R^{n \times p}$ is usually called the dynamic regressor matrix.

It can be concluded from Property 8 that the left side of Eq. (3) is linear in a set of augmented parameters. Hence:

$$\bar{M}(q)\ddot{q} + \bar{C}(q, \dot{q})\dot{q} + \bar{g}(q) = \bar{Y}(q, \dot{q}, \dot{q}, \ddot{q})\bar{a} \tag{16}$$

where $\bar{Y}(q, \dot{q}, \dot{q}, \ddot{q})$ and $\bar{a}$ can be constructed from $Y_d(q, \dot{q}, \dot{q}, \ddot{q})$, a_d, and K_N. Let $K_N = diag\{K_N^1, K_N^2, \ldots, K_N^n\}$. Then, $\bar{Y}(q, \dot{q}, \dot{q}, \ddot{q})$ and $\bar{a}$ are as follows:

$$\bar{Y} = blockdiag\{y_1, y_2 \ldots, y_n\} = \begin{bmatrix} y_1 & 0_{1\times p} & . & 0_{1\times p} & 0_{1\times p} \\ 0_{1\times p} & y_2 & . & 0_{1\times p} & 0_{1\times p} \\ . & . & . & . & . \\ 0_{1\times p} & 0_{1\times p} & . & 0_{1\times p} & y_n \end{bmatrix} \tag{17}$$

$$\bar{a} = [\bar{a}_1^T, \bar{a}_2^T, \ldots, \bar{a}_n^T]^T \tag{18}$$

where $\bar{a}_i \in R^p$ is given by:

$$\bar{a}_i = \left(\frac{1}{K_N^i}\right) * a_d, \quad i = 1, 2, \ldots, n \tag{19}$$

In addition, it can be shown that:

$$K_N\bar{Y}(q, \dot{q}, \dot{q}, \ddot{q}) = \bar{Y}(q, \dot{q}, \dot{q}, \ddot{q})K_{Na} \tag{20}$$

where

$$K_{Na} = diag\{K_N^1 I_p, K_N^2 I_p, \ldots, K_N^n I_p\} \tag{21}$$

and I_p denotes the $p \times p$ identity matrix.

In the following, two lemmas are introduced that are used for establishing the convergence of the proposed controller in the next sections.

Lemma 1: Suppose $H(p)$ is a strictly proper and exponentially stable transfer function and $y(t) = H(p)\{f(t)\}$. If $f(t) \in L_\infty$, then $y(t)$, $\dot{y}(t) \in L_\infty$ and $y(t)$ is uniformly continuous.[25]

Lemma 2: Suppose $H(p)$ is a strictly proper and exponentially stable transfer function and $y(t) = H(p)\{f(t)\}$. If $f(t) \in L_2$, then $y(t) \in L_2 \cap L_\infty$, $\dot{y}(t) \in L_2$ and $y(t) \to 0$ as $t \to \infty$.[25]

3. Controller Design

Backstepping is a recursive procedure that breaks a controller design problem for the full system into a sequence of controller design problems for lower-order systems. At first, only a small subsystem is considered and a virtual control law is constructed. Then, the design is extended in several steps until a control law for the full system is constructed. In this section, the backstepping control strategy[26,27] will be used to design an adaptive task-space controller for RLED robots. The control objective is to design a control law for the actuator voltages such that the manipulator task-space positions track the desired task-space positions in the presence of parametric uncertainties in kinematics and dynamics. The design procedure is organized in two steps. First, the manipulator subsystem is considered. The

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

armature current vector I is viewed as the input and a desired armature current vector I_d is designed so that the tracking errors converge. Then, an input control voltage vector u is designed so that the armature current vector I tracks the desired armature current vector I_d.

3.1. First step of controller design

In this section, the first step of backstepping strategy is explained. The desired armature current vector is designed such that the need for acceleration measurements in all control procedure is removed. To overcome the problem of parametric uncertainties in manipulator dynamics and electromechanical conversion matrix, adaptive estimation of the augmented parameters will be used in the desired armature current vector.

Suppose the desired armature current vector I_d has been designed so that the task-space tracking errors converge to zero. Then, the manipulator subsystem can be rewritten as:

$$\bar{M}(q)\ddot{q} + \bar{C}(q,\dot{q})\dot{q} + \bar{g}(q) = I_d + \tilde{I} \tag{22}$$

where $\tilde{I} = I - I_d$ is an error signal, which will be made as small as possible in the second design step.

Let x_d be the desired task-space trajectory that is bounded and differentiable up to the third order. Considering Eq. (22), the desired armature current vector can be proposed based on inverse manipulator dynamics as follows:

$$I_d = \hat{\bar{M}}(q)\hat{J}^{\dagger}(q)\{\ddot{x}_d - K_v\Delta\hat{\dot{x}} - K_p\Delta x - \dot{\hat{J}}(q)\dot{q}\} + \hat{\bar{C}}(q,\dot{q})\dot{q} + \hat{\bar{g}}(q) \tag{23}$$

where K_v and K_p are positive definite constant diagonal matrices. $\hat{J}(q)$ is the approximated Jacobian matrix corresponding to estimated kinematic parameters $\hat{a}_k$, which will be adopted by a kinematic adaptation law. $\hat{J}^{\dagger}(q) = \hat{J}^T(q)(\hat{J}(q)\hat{J}^T(q))^{-1}$ is the generalized inverse of $\hat{J}(q)$. $\hat{\bar{M}}(q)$, $\hat{\bar{C}}(q,\dot{q})$, and $\hat{\bar{g}}(q)$ have the same functional form as $\bar{M}(q)$, $\bar{C}(q,\dot{q})$, and $\bar{g}(q)$ with estimated parameters $\hat{\bar{a}}$, which will be adopted by an adaptation law. $\Delta x = x - x_d$ is the end-effector position tracking error. $\Delta\hat{\dot{x}} = \hat{\dot{x}} - \dot{x}_d$ is the estimated task-space velocity deviation and $\hat{\dot{x}}$ is the estimated task-space velocity:

$$\hat{\dot{x}} = \hat{J}(q)\dot{q} \tag{24}$$

Ordinary adaptive inverse dynamics controllers for both joint-space and task-space trajectory tracking need acceleration measurements for parameters adaptation law. In addition, as it will be seen in the next design step, the input control voltage utilizes the time derivative of the desired armature current vector. Therefore, the desired armature current vector (23) implies the need for acceleration in the input control voltage. In order to avoid the need for acceleration in parameters adaptation law and the input control voltage, the desired armature current vector (23) should be modified. The techniques used to avoid acceleration measurements are summarized as follows:

(a) Adding compensation control terms to the desired armature current vector.
(b) Using linear in variable property of the Jacobian matrix.
(c) Designing joint-space velocity observer.

In the following, the above techniques are explained. To eliminate acceleration from dynamic parameters adaptation law, Kelly proposed to add compensation control terms to the ordinary adaptive inverse dynamics controller for joint-space trajectory tracking.[8] In this paper, similar to the idea of ref. [8], compensation control terms are added to the desired armature current vector (23) to avoid the need for acceleration measurements in the augmented parameters adaptation law. The modified desired armature current vector that can remove the need for acceleration measurements is proposed as follows:

$$I_d = \hat{\bar{M}}(q)\hat{J}^{\dagger}(q)\{\ddot{x}_d - K_v\Delta\hat{\dot{x}} - K_p\Delta x - \dot{\hat{J}}(q)\dot{q} + \dot{\hat{J}}(q)v\} + \hat{\bar{C}}(q,\dot{q})\dot{q} - \hat{\bar{C}}(q,\dot{q})v + \hat{\bar{g}}(q) \tag{25}$$

This control law has two additional terms $\hat{\bar{M}}(q)\hat{J}^{\dagger}(q)\dot{\hat{J}}(q)v$ and $-\hat{\bar{C}}(q,\dot{q})v$ compared to the ordinary inverse dynamics control law (23). The signal v, which plays an important role in elimination

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

of acceleration from adaptation law for augmented parameters, is defined as:

$$v = \hat{J}^{\dagger}(q)\frac{1}{p+\lambda}\{\Delta\dot{\hat{x}} + K_v\Delta\hat{x} + K_p\Delta x\} \tag{26}$$

where $\Delta\dot{\hat{x}} = \frac{d\Delta\hat{x}}{dt} = \dot{\hat{x}} - \ddot{x}_d$ and λ is a positive constant. $'p'$ is the Laplace variable.

In order to avoid the need for acceleration in the input control voltage, the desired armature current vector (25) should be modified such that no joint-space velocity $\dot{q}$ appears in the desired armature current vector. Therefore, the joint-space velocity $\dot{q}$ with the observed joint-space velocity $\hat{\dot{q}}_o$ should be replaced. However, this replacement of $\dot{q}$ in the desired armature vector (25) is not straightforward because $\dot{\hat{J}}(q)$ depends on the joint-space velocity implicitly. To solve this problem, the following property is mentioned.

Property 9. There exist known matrices $R_k(.)$ and $P(.)$ such that the approximate Jacobian matrix satisfies:

$$\dot{\hat{J}}(q)z = R_k(q, \dot{\hat{a}}_k)z + P(q, \hat{a}_k, z)\dot{q} \tag{27}$$

In addition, for any two vectors $x, y \in R^n$, the matrix $P(.)$ satisfies:

$$P(q, \hat{a}_k, x) - P(q, \hat{a}_k, y) = P(q, \hat{a}_k, z) \tag{28}$$

where $z = x - y$.

Proof: The proof is given in the Appendix.

Using Property 9, the desired armature current vector (25) can be rewritten as follows:

$$\begin{aligned} I_d = {} & \widehat{\bar{M}}(q)\hat{J}^{\dagger}(q)\{\ddot{x}_d - K_v\Delta\dot{\hat{x}} - K_p\Delta x - R_k(q, \dot{\hat{a}}_k)\dot{q} - P(q, \hat{a}_k, \dot{q})\dot{q} + R_k(q, \dot{\hat{a}}_k)v + P(q, \hat{a}_k, v)\dot{q}\} \\ & + \widehat{\bar{C}}(q, \dot{q})\dot{q} - \widehat{\bar{C}}(q, \dot{q})v + \hat{\bar{g}}(q) \end{aligned} \tag{29}$$

Now, replacing the joint-space velocity $\dot{q}$ with the observed joint-space velocity $\hat{\dot{q}}_o$ in Eq. (29), the following is obtained:

$$\begin{aligned} I_d = {} & \widehat{\bar{M}}(q)\hat{J}^{\dagger}(q)\{\ddot{x}_d - K_v\Delta\dot{\hat{x}}_o - K_p\Delta x - R_k(q, \dot{\hat{a}}_k)\hat{\dot{q}}_o - P(q, \hat{a}_k, \hat{\dot{q}}_o)\hat{\dot{q}}_o + R_k(q, \dot{\hat{a}}_k)v \\ & + P(q, \hat{a}_k, v)\hat{\dot{q}}_o\} + \widehat{\bar{C}}(q, \hat{\dot{q}}_o)(\hat{\dot{q}}_o - v) + \hat{\bar{g}}(q) \end{aligned} \tag{30}$$

where $\Delta\dot{\hat{x}}_o = \dot{\hat{x}}_o - \dot{x}_d$ and $\dot{\hat{x}}_o$ obtains as follows:

$$\dot{\hat{x}}_o = \hat{J}(q)\hat{\dot{q}}_o \tag{31}$$

Moreover, modifying the signal v is needed as follows:

$$v = \hat{J}^{\dagger}(q)\frac{1}{p+\lambda}\{\Delta\dot{\hat{x}}_o + K_v\Delta\dot{\hat{x}}_o + K_p\Delta x\} \tag{32}$$

Using Eq. (16), the desired armature current vector (30) can be described as:

$$I_d = \bar{Y}(q, \hat{\dot{q}}_o, \dot{q}_z, \ddot{q}_r)\hat{\bar{a}} \tag{33}$$

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

where $\dot{q}_z$ and $\ddot{q}_r$ are given as follows:

$$\dot{q}_z = \hat{\dot{q}}_o - v \tag{34}$$

$$\ddot{q}_r = \hat{J}^{\dagger}(q)\{\ddot{x}_d - K_v\Delta\hat{\dot{x}}_o - K_p\Delta x - R_k(q,\hat{\dot{a}}_k)\hat{\dot{q}}_o - P(q,\hat{a}_k,\hat{\dot{q}}_o)\hat{\dot{q}}_o + R_k(q,\hat{\dot{a}}_k)v + P(q,\hat{a}_k,v)\hat{\dot{q}}_o\} \tag{35}$$

According to the above discussion, the desired armature current vector is given by Eq. (30). In the following, the stability analysis of the closed-loop manipulator subsystem with this desired armature current vector is given.

3.1.1. Parameters estimation and stability analysis

In this subsection, adaptation laws for the estimated kinematic parameters and augmented parameters will be proposed and the joint-space velocity observer will be designed. In addition, the stability analysis of the closed-loop manipulator subsystem with the desired armature current vector (30) will be given.

At first, the closed-loop dynamics of the manipulator subsystem is obtained. Let $\Delta\bar{a} = \hat{\bar{a}} - \bar{a}$ denotes the augmented parameter error vector. Using $I = \tilde{I} + I_d$ and Eqs. (1) and (33), the following is procured:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = K_N\bar{Y}(q,\hat{\dot{q}}_o,\dot{q}_z,\ddot{q}_r)(\Delta\bar{a} + \bar{a}) + K_N\tilde{I} \tag{36}$$

Using Eqs. (16) and (20), Eq. (36) can be rewritten as:

$$M(q)(\ddot{q} - \ddot{q}_r) + C(q,\dot{q})\dot{q} - C(q,\hat{\dot{q}}_o)\dot{q}_z = \bar{Y}(q,\hat{\dot{q}}_o,\dot{q}_z,\ddot{q}_r)K_{Na}\Delta\bar{a} + K_N\tilde{I} \tag{37}$$

On the other hand, differentiating Eq. (24) with respect to time and using Eq. (27), the following is acquired:

$$\ddot{q} = \hat{J}^{\dagger}(q)\{\hat{\ddot{x}} - R_k(q,\hat{\dot{a}}_k)\dot{q} - P(q,\hat{a}_k,\dot{q})\dot{q}\} \tag{38}$$

Substituting Eq. (38) into Eq. (37) and using Eq. (28) concludes:

$$M(q)\hat{J}^{\dagger}(q)\{\Delta\hat{\ddot{x}} + K_v\Delta\hat{\dot{x}}_o + K_p\Delta x - R_k(q,\hat{\dot{a}}_k)(v - \tilde{\dot{q}}) - P(q,\hat{a}_k,\dot{q})\dot{q} - P(q,\hat{a}_k,v - \hat{\dot{q}}_o)\hat{\dot{q}}_o\} + C(q,\dot{q})\dot{q} - C(q,\hat{\dot{q}}_o)(\hat{\dot{q}}_o - v) = \bar{Y}(q,\hat{\dot{q}}_o,\dot{q}_z,\ddot{q}_r)K_{Na}\Delta\bar{a} + K_N\tilde{I} \tag{39}$$

Now, the following Lyapunov function is proposed to analyze the stability of the closed-loop manipulator subsystem with the desired armature current vector (30):

$$V = \frac{1}{2}(v - \tilde{\dot{q}})^T M(q)(v - \tilde{\dot{q}}) + \frac{1}{2}\tilde{q}^T\tilde{q} + \frac{1}{2}\Delta\bar{a}^T K_{Na}\bar{L}^{-1}\Delta\bar{a} + \frac{1}{2}\Delta a_k^T L_k^{-1}\Delta a_k \tag{40}$$

where $\Delta a_k = \hat{a}_k - a_k$ is the kinematic parameter estimation error and $\tilde{\dot{q}} = \hat{\dot{q}}_o - \dot{q}$. $\bar{L}$ and L_k are positive definite constant diagonal matrices.

Differentiating V with respect to time follows:

$$\dot{V} = \frac{1}{2}(v - \tilde{\dot{q}})^T\dot{M}(q)(v - \tilde{\dot{q}}) + (v - \tilde{\dot{q}})^T M(q)(\dot{v} - \dot{\tilde{\dot{q}}}) + \tilde{\dot{q}}^T\dot{\tilde{q}} + \Delta\bar{a}^T K_{Na}\bar{L}^{-1}\dot{\hat{\bar{a}}} + \Delta a_k^T L_k^{-1}\dot{\hat{a}}_k \tag{41}$$

On the other hand, from Eq. (32), we have:

$$\dot{\hat{J}}(q)v + \hat{J}(q)\dot{v} + \lambda\hat{J}(q)v = \Delta\hat{\ddot{x}}_o + K_v\Delta\hat{\dot{x}}_o + K_p\Delta x \tag{42}$$

$\Delta\hat{\ddot{x}}_o$ can be calculated by the derivative of the following equation:

$$\Delta\hat{\dot{x}}_o = \hat{\dot{x}}_o - \hat{\dot{x}} + \hat{\dot{x}} - \dot{x}_d = \hat{J}(q)\tilde{\dot{q}} + \Delta\hat{\dot{x}} \tag{43}$$

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

Equation (42) can be rewritten as:

$$\hat{J}(q)(\dot{v}-\dot{\tilde{q}}) = \Delta\dot{\hat{x}} + K_v\Delta\hat{x}_o + K_p\Delta x - \dot{\hat{J}}(q)(v-\tilde{q}) - \lambda\hat{J}(q)v \tag{44}$$

Using Eq. (27), Eq. (44) can be reformulated as the following:

$$\hat{J}(q)(\dot{v}-\dot{\tilde{q}}) = \Delta\dot{\hat{x}} + K_v\Delta\hat{x}_o + K_p\Delta x - R_k(q,\dot{\hat{a}}_k)(v-\tilde{q}) - P(q,\hat{a}_k,v-\tilde{q})\dot{q} - \lambda\hat{J}(q)v \tag{45}$$

Now, using Eqs. (41) and (45), we have:

$$\begin{aligned}\dot{V} = {} & \frac{1}{2}(v-\tilde{q})^T\dot{M}(q)(v-\tilde{q}) - \lambda(v-\tilde{q})^T M(q)v + (v-\tilde{q})^T M(q)\hat{J}^{\dagger}\{\Delta\dot{\hat{x}} + K_v\Delta\hat{x}_o \\ & + K_p\Delta x - R_k(q,\dot{\hat{a}}_k)(v-\tilde{q}) - P(q,\hat{a}_k,v-\tilde{q})\dot{q}\} + \tilde{q}^T\dot{\tilde{q}} + \Delta\bar{a}^T K_{Na}\bar{L}^{-1}\dot{\hat{\bar{a}}} + \Delta {a_k}^T L_k^{-1}\dot{\hat{a}}_k\end{aligned} \tag{46}$$

Using Eqs. (39) and (28), we have:

$$\begin{aligned}\dot{V} = {} & \frac{1}{2}(v-\tilde{q})^T\dot{M}(q)(v-\tilde{q}) + (v-\tilde{q})^T\{\bar{Y}(q,\hat{q}_o,\dot{q}_z,\ddot{q}_r)K_{Na}\Delta\bar{a} + K_N\tilde{I} + C(q,\hat{q}_o)(\hat{q}_o - v) \\ & - C(q,\dot{q})\dot{q} - \lambda M(q)v + M(q)\hat{J}^{\dagger}(q)P(q,\hat{a}_k,v-\hat{q}_o)\tilde{q}\} + \tilde{q}^T\dot{\tilde{q}} + \Delta\bar{a}^T K_{Na}\bar{L}^{-1}\dot{\hat{\bar{a}}} + \Delta {a_k}^T L_k^{-1}\dot{\hat{a}}_k\end{aligned} \tag{47}$$

On the other hand, by using Property 3, we have:

$$\begin{aligned}C(q,\hat{q}_o)(\hat{q}_o - v) - C(q,\dot{q})\dot{q} &= C(q,\hat{q}_o - v)\hat{q}_o - C(q,\dot{q})\dot{q} \\ &= C(q,\hat{q}_o - v)\tilde{q} + C(q,\hat{q}_o - v)\dot{q} - C(q,\dot{q})\dot{q} \\ &= C(q,\hat{q}_o - v)\tilde{q} + C(q,\dot{q})(\hat{q}_o - v) - C(q,\dot{q})\dot{q} \\ &= C(q,\hat{q}_o - v)\tilde{q} - C(q,\dot{q})(v-\tilde{q})\end{aligned} \tag{48}$$

Substituting Eq. (48) into Eq. (47) implies:

$$\begin{aligned}\dot{V} = {} & \frac{1}{2}(v-\tilde{q})^T\dot{M}(q)(v-\tilde{q}) + (v-\tilde{q})^T\{\bar{Y}(q,\hat{q}_o,\dot{q}_z,\ddot{q}_r)K_{Na}\Delta\bar{a} + K_N\tilde{I} + C(q,\hat{q}_o - v)\tilde{q} \\ & - C(q,\dot{q})(v-\tilde{q}) - \lambda M(q)v + M(q)\hat{J}^{\dagger}(q)P(q,\hat{a}_k,v-\hat{q}_o)\tilde{q}\} + \tilde{q}^T\dot{\tilde{q}} \\ & + \Delta\bar{a}^T K_{Na}\bar{L}^{-1}\dot{\hat{\bar{a}}} + \Delta {a_k}^T L_k^{-1}\dot{\hat{a}}_k\end{aligned} \tag{49}$$

By using Property 2, Eq. (49) can be simplified as:

$$\begin{aligned}\dot{V} = {} & +(v-\tilde{q})^T\{\bar{Y}(q,\hat{q}_o,\dot{q}_z,\ddot{q}_r)K_{Na}\Delta\bar{a} + K_N\tilde{I} + C(q,\hat{q}_o - v)\tilde{q} - \lambda M(q)v \\ & + M(q)\hat{J}^{\dagger}(q)P(q,\hat{a}_k,v-\hat{q}_o)\tilde{q}\}\tilde{q}^T\dot{\tilde{q}} + \Delta\bar{a}^T K_{Na}\bar{L}^{-1}\dot{\hat{\bar{a}}} + \Delta {a_k}^T L_k^{-1}\dot{\hat{a}}_k\end{aligned} \tag{50}$$

Now, propose the adaptation laws for $\hat{\bar{a}}$ and $\hat{a}_k$ as follows:

$$\dot{\hat{\bar{a}}} = -\bar{L}\bar{Y}^T(q,\hat{q}_o,\dot{q}_z,\ddot{q}_r)(v-\tilde{q}) \tag{51}$$

$$\dot{\hat{a}}_k = -L_k {W_k}^T(t)R_{kk}(W_k(t)\hat{a}_k - y) \tag{52}$$

where R_{kk} is a positive definite matrix and y is the output of a filter with the task-space velocity as its input and with zero initial value, i.e., $y(0) = 0$. The low-pass filter for y signal is as follows:[14]

$$\dot{y} + \lambda_k y = \lambda_k\dot{x} \tag{53}$$

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

Equation (53) can be represented in the following form:

$$y = \lambda_k/(\lambda_k + p)\dot{x} = W_k(t)a_k \tag{54}$$

where $W_k(t) = \lambda_k/(\lambda_k + p)Y_k(q, \dot{q})$ and the initial value of W_k is zero.

Substituting the adaptation laws (51) and (52) into Eq. (50) concludes:

$$\begin{aligned}\dot{V} = (v - \tilde{\dot{q}})^T\{K_N\tilde{I} + C(q, \hat{\dot{q}}_o - v)\tilde{q} - \lambda M(q)v + M(q)\hat{J}^{\dagger}(q)P(q, \hat{a}_k, v - \hat{\dot{q}}_o)\tilde{\dot{q}}\} + \tilde{q}^T\dot{\tilde{q}} \\ + (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k)\end{aligned} \tag{55}$$

Using the following equation:

$$\dot{\tilde{q}} = \dot{\hat{q}}_o - \ddot{q} = \dot{\hat{q}}_o - M^{-1}(q)\{K_N I - C(q, \dot{q})\dot{q} - g(q)\} \tag{56}$$

Eq. (55) can be rewritten as follows:

$$\dot{V} = (v - \tilde{\dot{q}})^T K_N\tilde{I} - \lambda v^T M(q)v + \tilde{q}^T(h + \dot{\hat{q}}_o) - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) \tag{57}$$

where h is given as follows:

$$\begin{aligned}h = +C^T(q, \hat{\dot{q}}_o - v)(v - \tilde{\dot{q}}) + \lambda M(q)v + M^{-1}(q)\{-K_N I + C(q, \dot{q})\dot{q} + g(q)\} \\ + P^T(q, \hat{a}_k, v - \hat{\dot{q}}_o)\hat{J}^{\dagger T}(q)M(q)(v - \tilde{\dot{q}})\end{aligned} \tag{58}$$

According to the Properties 1, 4, 5, and 6, we have:

$$\|h\|_1 \leq \alpha(v, \hat{\dot{q}}_o, \dot{q}, \hat{a}_k) \tag{59}$$

Applying the above norm inequality to Eq. (57), we have:

$$\dot{V} \leq (v - \tilde{\dot{q}})^T K_N\tilde{I} - \lambda v^T M(q)v + \|\tilde{\dot{q}}\|_1\alpha(v, \hat{\dot{q}}_o, \dot{q}, \hat{a}_k) + \tilde{\dot{q}}^T\dot{\hat{q}}_o - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) \tag{60}$$

Now, the adaptation law for the observed joint-space velocity $\hat{\dot{q}}_o$ is defined as follows:

$$\dot{\hat{q}}_o = -\alpha(v, \hat{\dot{q}}_o, \dot{q}, \hat{a}_k)sgn(\tilde{\dot{q}}) - \Gamma\tilde{\dot{q}} \tag{61}$$

Substituting Eq. (61) into Eq. (60) implies:

$$\dot{V} \leq (v - \tilde{\dot{q}})^T K_N\tilde{I} - \lambda v^T M(q)v - \tilde{\dot{q}}^T\Gamma\tilde{\dot{q}} - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) \tag{62}$$

Therefore, $\dot{V}$ is negative semi-definite in the absence of armature current error.

In the following, a theorem is stated to guarantee the convergence of the end-effector motion tracking errors.

Theorem 1: Consider the desired current vector (30) together with the parameter adaptation laws (51), (52) and the observer (61). Then, in the absence of armature current error, the end-effector motion tracking errors converge to zero. That is $\Delta x \to 0$, $\Delta\dot{x} \to 0$ as $t \to \infty$. Moreover $\tilde{\dot{q}}$, $W_k(t)\Delta a_k$, $\Delta\hat{\dot{x}} \to 0$ as $t \to \infty$.

Proof: It is obvious that V in Eq. (40) is positive definite and $\dot{V}$ in Eq. (62) is negative semi-definite in the absence of armature current error. Therefore, it is concluded that $v, \tilde{\dot{q}} \in L_2 \cap L_\infty$, $W_k(t)\Delta a_k \in L_2$ and $\Delta\bar{a}, \Delta a_k \in L_\infty$. Since $\bar{a}, a_k \in L_\infty$, we conclude that $\hat{\bar{a}}, \hat{a}_k \in L_\infty$. Matrix $\hat{J}(q)$ is a trigonometric function of q and $\hat{a}_k \in L_\infty$; it is concluded that $\hat{J}(q) \in L_\infty$. Consequently, $\hat{J}(q)v \in L_2 \cap L_\infty$.

It is known that x is bounded, since $x = h(q)$ is a trigonometric function of q. Assuming $x_d \in L_\infty$, $\Delta x \in L_\infty$.

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

From Eq. (32):

$$\{\Delta\hat{\dot{x}}_o\} = (pI_E + K_v)^{-1}(p+\lambda)\{\hat{J}(q)v\} - (pI_E + K_v)^{-1}\{K_p\Delta x\} \tag{63}$$

and substituting Eq. (43) into Eq. (63) implies that:

$$\{\Delta\hat{\dot{x}}\} = (pI_E + K_v)^{-1}(p+\lambda)\{\hat{J}(q)v\} - \hat{J}(q)\tilde{\dot{q}} - (pI_E + K_v)^{-1}\{K_p\Delta x\} \tag{64}$$

According to Eq. (64), $\Delta\hat{\dot{x}}$ is the summation of the bounded signal $\hat{J}(q)\tilde{\dot{q}}$ with the outputs of one strictly proper stable filter and one proper stable filter with bounded inputs. Lemma 1 concludes that $\Delta\hat{\dot{x}}$ is bounded. Therefore, $\hat{\dot{x}}$ is also bounded if $\dot{x}_d$ is bounded. According to Eq. (24), we conclude that $\dot{q}$ is bounded if $\hat{J}(q)$ is nonsingular.

To establish the results of the theorem, we show that $\tilde{\dot{q}} \to 0$ as $t \to \infty$ firstly.

It can be seen from Eq. (30) that I_d is bounded. In the absence of armature current error, Eq. (22) implies that $\ddot{q}$ is bounded. Equation (61) implies that $\dot{\hat{q}}_o$ is bounded. Therefore, $\dot{\tilde{q}}$ is bounded and since $\tilde{\dot{q}} \in L_2 \cap L_\infty$, Barbalat's Lemma implies that $\tilde{\dot{q}} \to 0$ as $t \to \infty$.

Secondly, we show that $W_k(t)\Delta a_k$, $Y_k(q,\dot{q})\Delta a_k \to 0$ as $t \to \infty$.

Matrix $Y_k(q,\dot{q})$ is bounded, since $Y_k(q,\dot{q})$ is a trigonometric function of q and $\dot{q}$ is bounded. Since $W_k(t)$ is the output of a strictly proper stable filter with $Y_k(q,\dot{q})$ as input, it is bounded according to Lemma 1. Therefore, $W_k(t)a_k$, $W_k(t)\Delta a_k \in L_\infty$. From Eq. (54), we deduce that y is bounded. Therefore, $\dot{\hat{a}}_k$ is bounded according to Eq. (52). Taking the time derivative of $W_k(t)\Delta a_k$, we obtain:

$$\frac{d}{dt}(W_k(t)\Delta a_k) = \dot{W}_k(t)\Delta a_k + W_k(t)\dot{\hat{a}}_k = \lambda_k(Y_k(q,\dot{q}) - W_k(t))\Delta a_k + W_k(t)\dot{\hat{a}}_k \tag{65}$$

Therefore, $\frac{d}{dt}(W_k(t)\Delta a_k)$ is bounded and since $W_k(t)\Delta a_k \in L_2 \cap L_\infty$, Barbalat's Lemma implies that $W_k(t)\Delta a_k \to 0$ as $t \to \infty$. The convergence of $W_k(t)\Delta a_k$ to zero can be the result of either $\Delta a_k \to 0$ or singularity of $W_k(t)$. Since $Y_k(q,\dot{q}) = ((p+\lambda_k)/\lambda_k)W_k(t)$, singularity of $W_k(t)$ implies the same singularity of $Y_k(q,\dot{q})$. So, in either case $Y_k(q,\dot{q})\Delta a_k \to 0$ as $t \to \infty$.

Finally, it is shown that Δx, $\Delta\dot{x} \to 0$ as $t \to \infty$.

From Eq. (32):

$$(pI_E + K_v)\{\Delta\hat{\dot{x}}_o\} = (p+\lambda)\{\hat{J}(q)v\} - \{K_p\Delta x\} \tag{66}$$

By using Eq. (43) and the identity:

$$\Delta\hat{\dot{x}} = \hat{\dot{x}} - \dot{x}_d = \hat{\dot{x}} - \dot{x} + \dot{x} - \dot{x}_d = Y_k(q,\dot{q})\Delta a_k + \Delta\dot{x} \tag{67}$$

Eq. (66) can be rewritten as:

$$\Delta\ddot{x} + K_v\Delta\dot{x} + K_p\Delta x = (p+\lambda)\{\hat{J}(q)v\} - (pI_E + K_v)\{\hat{J}(q)\tilde{\dot{q}}\} - (pI_E + K_v)\{Y_k(q,\dot{q})\Delta a_k\} \tag{68}$$

From Eq. (68), it can be seen that task-space tracking error is the sum of outputs of three strictly proper and exponentially stable filters with $\hat{J}(q)v$, $\hat{J}(q)\tilde{\dot{q}}$, and $Y_k(q,\dot{q})\Delta a_k$ as their inputs. Since $Y_k(q,\dot{q})\Delta a_k$, $\hat{J}(q)\tilde{\dot{q}} \to 0$ as $t \to \infty$, $\hat{J}(q)v \in L_2$, Lemma 2 implies that $\Delta x \to 0$ as $t \to \infty$.

It should be mentioned that $\ddot{q}$ is bounded in the absence of current error, so $\ddot{x} = J(q)\ddot{q} + \dot{J}(q)\dot{q}$ is also bounded. Hence, $\Delta\ddot{x}$ is bounded if $\ddot{x}_d$ is bounded. This implies that $\Delta\dot{x}$ is uniformly continuous; also we know that Δx is bounded, then according to Barbalat's Lemma, $\Delta\dot{x} \to 0$ as $t \to \infty$. This result completes the proof.

Remark 1. Since $\hat{J}^\dagger(q)$ has been used in the control variables, a projection algorithm should be used to ensure that the estimated kinematic parameters $\hat{a}_k$ remain in a prespecified range such that $\hat{J}(q)$ is nonsingular.

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

3.2. Second step of controller design

In this section, the input control voltage will be designed such that the armature current error converges to zero. Using $\tilde{I} = I - I_d$, the actuator subsystem (2) can be transformed into the form:

$$L\dot{\tilde{I}} + R\tilde{I} = u - L\dot{I}_d - RI_d - K_E\dot{q} \tag{69}$$

The actuators coefficient matrices L, R, K_E and electromechanical conversion matrix cannot be used in the input control voltage since these matrices are not known. To overcome this problem, we use adaptive approximate coefficient matrices $\hat{L}$, $\hat{R}$, $\hat{K}_E$, and $\hat{K}_N$, which are diagonal and their adaptation laws are given later. We propose the input control voltage as follows:

$$u = \hat{L}\dot{I}_d + \hat{R}I_d + \hat{K}_E\dot{q} - \hat{K}_N(v - \tilde{\dot{q}}) - K_I\tilde{I} \tag{70}$$

where K_I is a positive definite diagonal matrix. Substituting Eq. (70) into Eq. (69) concludes:

$$L\dot{\tilde{I}} + R\tilde{I} = \hat{L}\dot{I}_d + \hat{R}I_d + \hat{K}_E\dot{q} - \hat{K}_N(v - \tilde{\dot{q}}) - K_I\tilde{I} - L\dot{I}_d - RI_d - K_E\dot{q} \tag{71}$$

and using the idea of ref. [20], Eq. (71) can be rewritten as:

$$L\dot{\tilde{I}} + R\tilde{I} = N_L(\dot{I}_d)(\hat{\theta}_L - \theta_L) + N_R(I_d)(\hat{\theta}_R - \theta_R) + N_E(\dot{q})(\hat{\theta}_{K_E} - \theta_{K_E}) - N_N(v - \tilde{\dot{q}})\hat{\theta}_{K_N} - K_I\tilde{I} \tag{72}$$

where $N_L(\dot{I}_d) = diag(\dot{I}_{d1}, \dot{I}_{d2}, \ldots, \dot{I}_{dn})$, $N_R(I_d) = diag(I_{d1}, I_{d2}, \ldots, I_{dn})$, $N_E(\dot{q}) = diag(\dot{q}_1, \dot{q}_2, \ldots, \dot{q}_n)$, $N_N(v - \tilde{\dot{q}}) = diag((v - \tilde{\dot{q}})_1, (v - \tilde{\dot{q}})_2, \ldots, (v - \tilde{\dot{q}})_n)$, and $\hat{\theta}_L, \theta_L, \hat{\theta}_R, \theta_R, \hat{\theta}_{K_E}, \theta_{K_E}, \hat{\theta}_{K_N}$ are vectors of diagonal elements in $\hat{L}, L, \hat{R}, R, \hat{K}_E, K_E$, and $\hat{K}_N$, respectively.

Consider the augmented Lyapunov function:

$$\begin{aligned} V_2 = V &+ \frac{1}{2}\tilde{I}^T L\tilde{I} + \frac{1}{2}(\hat{\theta}_L - \theta_L)^T L_L^{-1}(\hat{\theta}_L - \theta_L) + \frac{1}{2}(\hat{\theta}_R - \theta_R)^T L_R^{-1}(\hat{\theta}_R - \theta_R) \\ &+ \frac{1}{2}(\hat{\theta}_{K_E} - \theta_{K_E})^T L_E^{-1}(\hat{\theta}_{K_E} - \theta_{K_E}) + \frac{1}{2}(\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}(\hat{\theta}_{K_N} - \theta_{K_N}) \end{aligned} \tag{73}$$

where V is defined as in Eq. (40) and L_L, L_R, L_E, L_N are positive definite diagonal matrices. θ_{K_N} is the diagonal elements vector of the electromechanical conversion matrix.

Differentiating V_2 with respect to time, we have:

$$\begin{aligned} \dot{V}_2 = \dot{V} &+ \tilde{I}^T L\dot{\tilde{I}} + (\hat{\theta}_L - \theta_L)^T L_L^{-1}\dot{\hat{\theta}}_L + (\hat{\theta}_R - \theta_R)^T L_R^{-1}\dot{\hat{\theta}}_R + (\hat{\theta}_{K_E} - \theta_{K_E})^T L_E^{-1}\dot{\hat{\theta}}_{K_E} \\ &+ (\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}\dot{\hat{\theta}}_{K_N} \end{aligned} \tag{74}$$

Using Eq. (72), Eq. (74) can be reformulated as:

$$\begin{aligned} \dot{V}_2 = \dot{V} &- \tilde{I}^T(R + K_I)\tilde{I} + \tilde{I}^T N_L(\dot{I}_d)(\hat{\theta}_L - \theta_L) + \tilde{I}^T N_R(I_d)(\hat{\theta}_R - \theta_R) + \tilde{I}^T N_E(\dot{q})(\hat{\theta}_{K_E} - \theta_{K_E}) \\ &- \tilde{I}^T N_N(v - \tilde{\dot{q}})\hat{\theta}_{K_N} + (\hat{\theta}_L - \theta_L)^T L_L^{-1}\dot{\hat{\theta}}_L + (\hat{\theta}_R - \theta_R)^T L_R^{-1}\dot{\hat{\theta}}_R + (\hat{\theta}_{K_E} - \theta_{K_E})^T L_E^{-1}\dot{\hat{\theta}}_{K_E} \\ &+ (\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}\dot{\hat{\theta}}_{K_N} \end{aligned} \tag{75}$$

Now, propose the adaptation laws for the actuator coefficients as follows:

$$\dot{\hat{\theta}}_L = -L_L N_L(\dot{I}_d)\tilde{I} \tag{76}$$

$$\dot{\hat{\theta}}_R = -L_R N_R(I_d)\tilde{I} \tag{77}$$

$$\dot{\hat{\theta}}_{K_E} = -L_E N_E(\dot{q})\tilde{I} \tag{78}$$

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

Substituting the adaptation laws (76)–(78) into Eq. (75) implies:

$$\dot{V}_2 = \dot{V} - \tilde{I}^T(R + K_I)\tilde{I} - \tilde{I}^T N_N(v - \tilde{\dot{q}})\hat{\theta}_{K_N} + (\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}\dot{\hat{\theta}}_{K_N} \tag{79}$$

and by using Eq. (62), we have:

$$\begin{aligned}\dot{V}_2 \leq\ & (v - \tilde{\dot{q}})^T K_N \tilde{I} - \lambda v^T M(q) v - \tilde{\dot{q}}^T \Gamma \tilde{\dot{q}} - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) \\ & - \tilde{I}^T(R + K_I)\tilde{I} - \tilde{I}^T \hat{K}_N(v - \tilde{\dot{q}}) + (\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}\dot{\hat{\theta}}_{K_N}\end{aligned} \tag{80}$$

The above inequality can be simplified as follows:

$$\begin{aligned}\dot{V}_2 \leq\ & -\tilde{I}^T N_N(v - \tilde{\dot{q}})(\hat{\theta}_{K_N} - \theta_{K_N}) - \lambda v^T M(q) v - \tilde{\dot{q}}^T \Gamma \tilde{\dot{q}} - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) \\ & - \tilde{I}^T(R + K_I)\tilde{I} + (\hat{\theta}_{K_N} - \theta_{K_N})^T L_N^{-1}\dot{\hat{\theta}}_{K_N}\end{aligned} \tag{81}$$

Now, propose the adaptation law for estimated elements of the electromechanical conversion matrix as follows:

$$\dot{\hat{\theta}}_{K_N} = L_N N_N(v - \tilde{\dot{q}})\tilde{I} \tag{82}$$

Substituting the adaptation law (82) into Eq. (81) concludes:

$$\dot{V}_2 \leq -\lambda v^T M(q) v - \tilde{\dot{q}}^T \Gamma \tilde{\dot{q}} - (W_k(t)\Delta a_k)^T R_{kk}(W_k(t)\Delta a_k) - \tilde{I}^T(R + K_I)\tilde{I} \tag{83}$$

Theorem 2: If the input control voltage (70) with the parameter adaptation laws (51), (52), (82), and (76)–(78) and the observer (61) is applied on the robot manipulator, then $\Delta x, \Delta\dot{x}, \tilde{\dot{q}}, W_k(t)\Delta a_k, \tilde{I} \to 0$ as $t \to \infty$.

Proof: Clearly, V_2 is positive definite and $\dot{V}_2$ is negative semi-definite. Therefore, $\tilde{I}, v, \tilde{\dot{q}} \in L_2 \cap L_\infty$, $W_k(t)\Delta a_k \in L_2$ and $(\hat{\theta}_L - \theta_L)$, $(\hat{\theta}_R - \theta_R)$, $(\hat{\theta}_{K_E} - \theta_{K_E})$, $(\hat{\theta}_{K_N} - \theta_{K_N})$, $\Delta\bar{a}$, $\Delta a_k \in L_\infty$. $\hat{\theta}_{K_N}$ is also bounded, since $\theta_{K_N}$is bounded. Similarly, as in the proof for Theorem 1, It is verified that I_d, Δx, $J(q)$, $\dot{q}$ and $\dot{J}(q)$ are bounded and Δx, $W_k(t)\Delta a_k$, $\tilde{\dot{q}} \to 0$ as $t \to \infty$. From Eq. (22), it can be concluded that $\ddot{q}$ is bounded. Therefore, $\ddot{x} = J(q)\ddot{q} + \dot{J}(q)\dot{q}$ is also bounded. Hence, $\Delta\ddot{x}$ is bounded if $\ddot{x}_d$ is bounded. This implies that $\Delta\dot{x}$ is uniformly continuous; also we know that Δx is bounded, then according to Barbalat's Lemma, $\Delta\dot{x} \to 0$ as $t \to \infty$.

4. Simulation Results

In this section, we present simulation results of the proposed controller applied on a two-link RLED robot as shown in Fig. 1. The proposed controller is compared with the controller presented in ref. [20]. The dynamic model for this robot can be described in the form (1), (2) with

$$M(q) = \begin{bmatrix} a + bcos(q_2) & c + \frac{b}{2}\cos(q_2) \\ c + \frac{b}{2}\cos(q_2) & c \end{bmatrix} \tag{84}$$

$$C(q, \dot{q}) = \begin{bmatrix} -\frac{b}{2}\sin(q_2)\dot{q}_2 & -\frac{b}{2}\sin(q_2)(\dot{q}_1 + \dot{q}_2) \\ \frac{b}{2}\sin(q_2)\dot{q}_1 & 0 \end{bmatrix} \tag{85}$$

where $a = m_2 l_2^2 + (m_1 + m_2) l_1^2$, $b = 2 l_1 l_2 m_2$, $c = m_2 l_2^2$. l_1 and l_2 denote the lengths of the first and second links, respectively. q_1 and q_2 denote joint angle of the first and second links, respectively. m_1 and m_2 are the masses of the two links.

For simplicity, the gravitational forces are assumed to be zero. The exact robot parameters are $l_1 = 1m, l_2 = 1m, m_1 = 1kg$, and $m_2 = 1kg$. Coefficient matrices of the actuator dynamics are $L = diag\{.1, .1\}$,$R = diag\{1, 1\}$, $K_E = diag\{1, 1\}$ and $K_N = diag\{2, 2\}$.

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

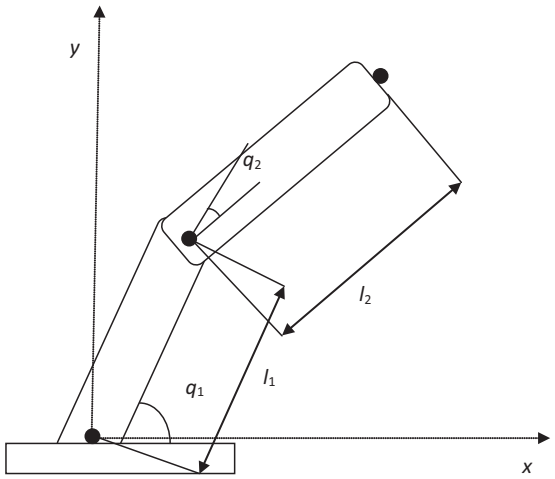


Fig. 1. A two-link robot.

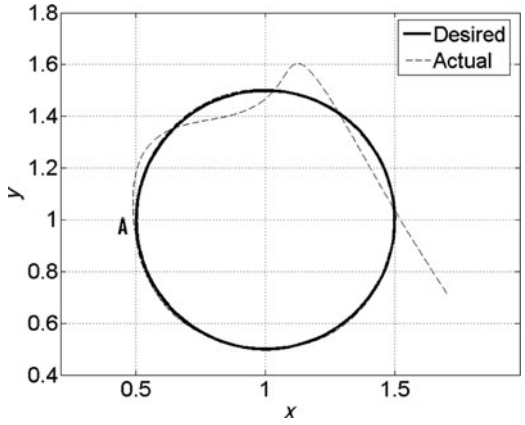


Fig. 2. Desired and actual trajectory.

The manipulator Jacobian matrix $J(q)$ mapping from joint-space to task-space for this robot can be derived as:

$$J(q) = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1 + q_2) & -l_2 \sin(q_1 + q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) & l_2 \cos(q_1 + q_2) \end{bmatrix} \tag{86}$$

The desired trajectory is defined in task-space as:

$$x_d = \begin{bmatrix} 1 + 0.5\cos(t) \\ 1 + 0.5\sin(t) \end{bmatrix} \tag{87}$$

The initial joint position of the robot is set as $q(0) = [\pi/4 \ -\pi/4]^T$. Initial estimation of the parameters is set as $\hat{\tilde{a}}(0) = [1.33\ 0.89\ 0.44\ 1.33\ 0.89\ 0.44\ ]^T$, $\hat{\theta}_R(0) = \hat{\theta}_{K_E}(0) = [0.8\ 0.8]^T$, $\hat{a}_k(0) = [1.2\ 1.2]^T$, $\hat{\theta}_{K_N}(0) = [1.8\ 1.8]^T$, $\hat{\theta}_L(0) = [0.05\ 0.05]^T$, $\hat{\dot{q}}_o(0) = [0\ 0]^T$. The control gains of the proposed controller are set as $K_v = 2I_E$, $K_p = 15I_E$, $K_I = 10I_E$, $L_k = 0.08I_E$, $R_{kk} = 100I_E$, $\Gamma = I_E$, $\lambda = \lambda_k = 10$, $L_L = L_R = L_E = L_N = 0.05I_E$, $\bar{L} = diag\{0.2, 0.6, 0.4, 0.5, 1.5, 0.8\}$.

Figure 2 shows the actual and desired task-space trajectory. The solid line shows the desired task-space trajectory and the dashed line shows the actual task-space trajectory. The initial errors along x and y axes are 0.2 and 0.3 meter, respectively. The robot will continue tracking the desired circle for several iterations. It can be seen that the robot does not have a good performance in half of the first circle tracking, but the performance is satisfactory after that (point A in Fig. (2)).

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

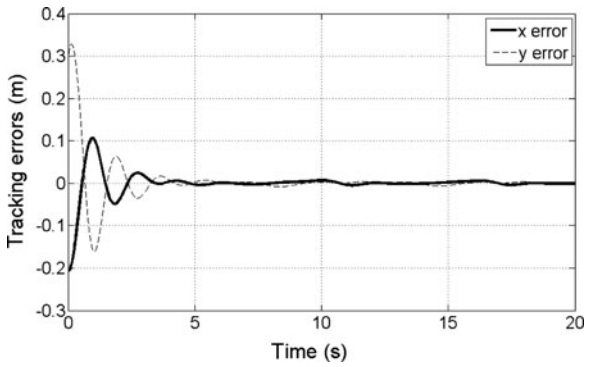


Fig. 3. End-effector tracking errors (proposed controller).

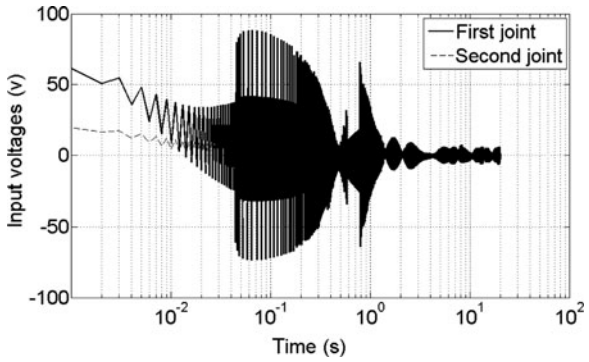


Fig. 4. Input control voltages (proposed controller).

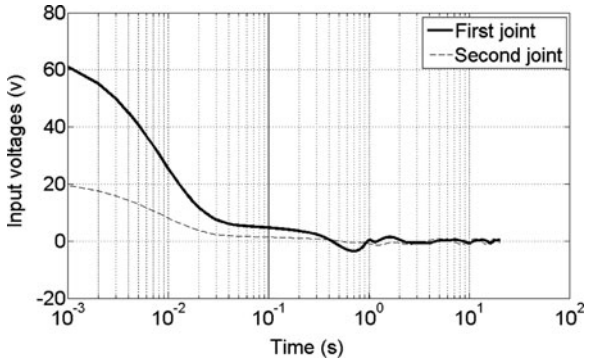


Fig. 5. Input control voltages (proposed controller with approximation signum function).

Figure 3 shows the end-effector tracking errors for the proposed controller. The solid line shows the end-effector tracking error along x axis and the dashed line shows the end-effector tracking error along y axis. Looking deeply into Fig. 3, we see that the robot cannot track the desired trajectory in the first 4 seconds, but it can estimate the parameters and variables after that and the performance is satisfactory after 4 seconds. Input control voltages for the first and second joints are shown in Fig. 4. The solid line shows the input control voltage for the first joint and the dashed line shows the input control voltage for the second joint. From Fig. 4, we see that the proposed controller introduces chattering into the robot system. To eliminate the chattering effects, the signum function $sgn(\tilde{\dot{q}})$ can be replaced by $\tanh(\varepsilon\tilde{\dot{q}})$. Input control voltages of applying the proposed controller with $\tanh(\varepsilon\tilde{\dot{q}})$, $\varepsilon = 10$ are shown in Fig. 5. It can be seen that the chattering is completely eliminated.

In order to compare the results of the proposed controller with the previous works, we apply the passivity-based controller in ref. [20] on the robot. Initial conditions are the same as the proposed

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

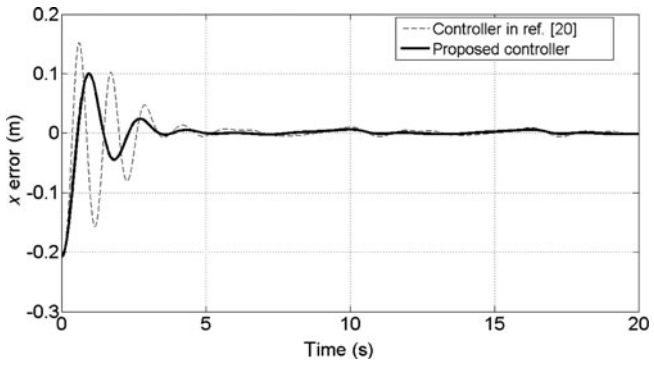


Fig. 6. End-effector tracking errors along x-axis.

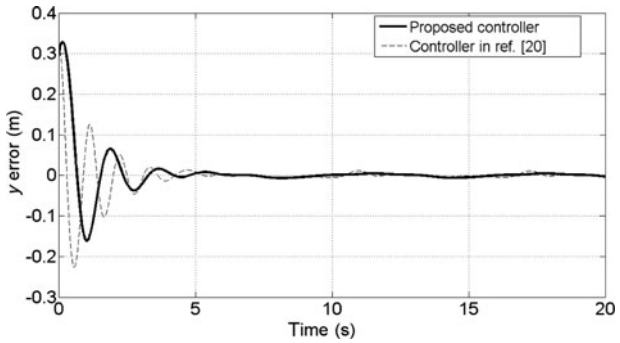


Fig. 7. End-effector tracking errors along y-axis.

approach of this paper. Other conditions are set as:

$$\hat{a}_d(0) = [\,2.4\ 1.6\ 0.8\,]^T, \hat{\theta}_a(0) = \hat{\tilde{\theta}}_a(0) = [\,0\ 0\,]^T, \hat{\theta}_{\tilde{K}_N}(0) = [\,1.8\ 1.8\,]^T, \hat{K}_N = 1.8 I_E.$$

The control gains of this controller are set as:

$$K_p = 30 I_E, K_d = 2 I_E, K_x = K_q = K_I = \Lambda = 10 I_E, L_k = diag\{0.2, 0.08\},$$
$$L_d = diag\{0.1, 1, 0.5\}, \alpha = 0.5, L_L = L_R = L_E = L_N = L_a = 0.05 I_E,$$
$$\hat{x}_o(0) = \hat{\tilde{q}}_o(0) = [\,0 \quad 0\,]^T$$

Figures 6 and 7 show the end-effector tracking errors of the controller in ref. [20] and the proposed controller in which the signum function $sgn(\tilde{q})$ is replaced by $\tanh(\varepsilon\tilde{q})$, $\varepsilon = 10$. As it is clear from Figs. (6) and (7), the end-effector tracking errors of both controllers converge to zero and the performance of the proposed controller is slightly better than the controller in ref. [20]. Figure 8 shows the input control voltages for the controller in ref. [20]. From Figs. (5) and (8), it can be seen that the amplitude of input voltage of the proposed controller is less than the controller in ref. [20]. In addition, the deviation of the proposed controller is less than the controller in ref. [20].

5. Conclusions

In this paper, a new adaptive backstepping control scheme is proposed for RLED robots with both kinematics and dynamics uncertainties. To design the desired armature current vector in the first step of backstepping algorithm, compensation control terms are added to the classical inverse dynamics control to avoid acceleration measurements. Moreover, using the estimate of observed joint-space velocity and linear in variable property of the Jacobian matrix, the need for acceleration measurements is completely removed in the backstepping design. Using Lyapunov analysis, the stability of the

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

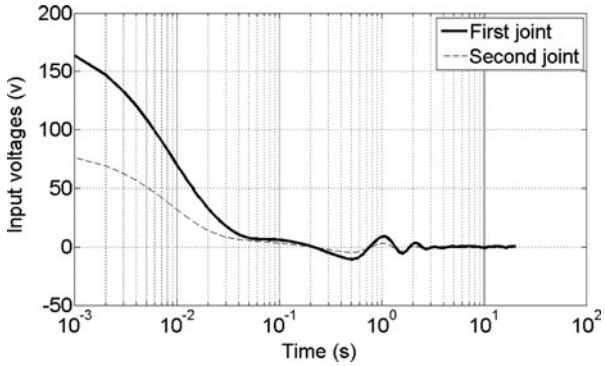


Fig. 8. Input control voltages (controller in ref. [20].)

closed-loop system is proved. Simulation results show that the performance of the proposed controller is comparable to the proposed passivity-based controllers for RLED robots.

Appendix

Proof of Property 9

Since $x = h(q)$ contains the terms such as $a_{k_j}\cos(q_i)$, $a_{k_j}q_i$, $a_{k_j}q_i\cos(q_k)\ldots\cos(q_l)$, and $a_{k_j}\cos(q_k)\ldots\cos(q_l)$, the estimated Jacobian matrix contains the terms such as $-\hat{a}_{k_j}\sin(q_i)$, $\hat{a}_{k_j}$, $\hat{a}_{k_j}\cos(q_k)\ldots\cos(q_l)$, $-\hat{a}_{k_j}q_i\sin(q_k)\ldots\cos(q_l)$, and $-\hat{a}_{k_j}\sin(q_k)\ldots\cos(q_l)$. Therefore, $\dot{\hat{J}}(q)z$ contains the terms such as $-\dot{\hat{a}}_{k_j}\sin(q_i)z_l$, $-\hat{a}_{k_j}\cos(q_i)\dot{q}_i z_l$, $\dot{\hat{a}}_{k_j}z_l$, $\dot{\hat{a}}_{k_j}\cos(q_k)\ldots\cos(q_l)z_l$, $-\hat{a}_{k_j}\sin(q_k)\ldots\cos(q_l)\dot{q}_k z_l$, $-\dot{\hat{a}}_{k_j}q_i\sin(q_k)\ldots\cos(q_l)z_l$, $-\hat{a}_{k_j}\dot{q}_i\sin(q_k)\ldots\cos(q_l)z_l - \hat{a}_{k_j}q_i\cos(q_k)\ldots\cos(q_l)\dot{q}_k z_l$, $-\dot{\hat{a}}_{k_j}\sin(q_k)\ldots\cos(q_l)z_l$, $-\hat{a}_{k_j}\cos(q_k)\ldots\cos(q_l)\dot{q}_k z_l$.

Consequently, the accuracy of Eq. (27) is obvious. Elements of the matrix $P(q, \hat{a}_k, z)$ are linear with respect to z_l, so Eq. (28) is verified.

References

1. J. J. E. Slotine and W. Li, "Adaptive manipulator control: A case study," *IEEE Trans. Autom. Control* **33**(11), 995–1003 (1988).
2. W. Li and J. J. E. Slotine, "An indirect adaptive robot controller," *Syst. Control Lett.* **12**, 259–266 (1989).
3. M. W. Spong and R. Ortega, "On adaptive inverse dynamics control of rigid robots," *IEEE Trans. Autom. Control* **35**(1), 92–95 (1990).
4. M. W. Spong, "On the robust control of robot manipulators," *IEEE Trans. Autom. Control* **37**(11), 1782–1786 (1992).
5. C. Canudas De Wit and N. Fixot, "Adaptive control of robot manipulators via velocity estimated feedback," *IEEE Trans. Autom. Control* **37**(8), 1234–1237 (1992).
6. D. M. Dawson, Z. Qu and J. J. Carroll, "Tracking control of rigid-link electrically-driven robot manipulators," *Int. J. Control* **56**(5), 991–1006 (1992).
7. C. Kwan, F. L. Lewis and D. M. Dawson, "Robust neural-network control of rigid-link electrically driven robots," *IEEE Trans. Neural Netw.* **9**(4), 581–588 (1998).
8. R. Kelly, "Adaptive computed torque plus compensation control for robot manipulators," *Mech. Mach. Theory* **25**(2), 161–165 (1990).
9. C. C. Cheah, S. Kawamura and S. Arimoto, "Feedback Control for Robotic Manipulator with Uncertain Kinematics and Dynamics," *Proceedings of the IEEE International Conference on Robotics and Automation*, Leuven, Belgium (1998) pp. 3607–3612.
10. H. Yazarel and C. C. Cheah, "Task space adaptive control of robotic manipulators with uncertainties in gravity regressor matrix and kinematics," *IEEE Trans. Autom. Control* **47**(9), 1580–1585 (2002).
11. C. C. Cheah, "Approximate Jacobian Robot Control with Adaptive Jacobian Matrix," *Proceedings of the IEEE International Conference on Decision and Control*, Maui, HI, USA (2003) pp. 5859–5864.
12. C. C. Cheah, C. Liu and J. J. E. Slotine, "Approximate Jacobian Adaptive Control for Robot Manipulators," *Proceedings of the IEEE International Conference on Robotics and Automation*, New Orleans, LA, USA (2004) pp. 3075–3080.

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press

13. C. C. Cheah, C. Liu and J. J. E. Slotine, "Adaptive Jacobian Tracking Control of Robots Based on Visual Task-Space Information," *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005) pp. 3498–3503.
14. H. Wang and Y. Xie, "Adaptive inverse dynamic control of robots with uncertain kinematics and dynamics," *Automatica* **45**, 2114–2119 (2009).
15. H. Wang and Y. Xie, "Prediction error based adaptive Jacobian tracking of robots with uncertain kinematics and dynamics," *IEEE Trans. Autom. Control* **54**(12), 2889–2894 (2009).
16. W. E. Dixon, "Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics," *IEEE Trans. Autom. Control* **52**(3), 488–493 (2007).
17. M. C. Good, L. M. Sweet and K. L. Strobel, "Dynamic models for control system design of integrated robot and drive systems," *ASME J. Dyn. Syst. Measurement Control* **107**, 53–59 (1985).
18. C. Liu and C. C. Cheah, "Adaptive Regulation of Rigid-Link Electrically Driven Robots with Uncertain Kinematics," *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (2005) pp. 3262–3267.
19. C. Liu and P. Poignet, "SP-ID Regulation of Rigid-Link Electrically-Driven Robots with Uncertain Kinematics," *Proceedings of the IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA (2010) pp. 4663–4668.
20. C. Liu, C. C. Cheah and J. J. E. Slotine, "Adaptive Jacobian tracking control of rigid-link electrically driven robots based on visual task-space information," *Automatica* **42**, 1491–1501 (2006).
21. J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 3rd ed. (Prentice Hall, Upper Saddle River, NJ, 2005).
22. T. J. Tarn, A. K. Bejczy, X. Yun and Z. Li, "Effect of motor dynamics on nonlinear feedback robot arm control," *IEEE Trans. Robot. Autom.* **7**(1), 114–122 (1991).
23. L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators* (McGraw-Hill, New York, 1996).
24. C. Canudas De Wit, B. Siciliano and G. Bastin, *Theory of Robot Control* (Springer, Berlin, 1996).
25. C. A. Desoer and M. Vidyasagar, *Feedback Systems: Input-Output Properties* (Academic Press, New York, 1975).
26. H. Khalil, *Nonlinear Systems*, 3rd ed. (Prentice Hall, Upper Saddle River, NJ, 2002).
27. M. Krstic, I. Kanellakopoulos and P. Kokotovic, *Nonlinear and Adaptive Control Design* (John Wiley, New York, 1995).

https://doi.org/10.1017/S0263574713001203 Published online by Cambridge University Press