

Correspondence

Comments on “Redesign of Hybrid Adaptive/Robust Motion Control of Rigid-Link Electrically-Driven Robot Manipulators”

M. S. de Queiroz, F. Zhang, and W. Dixon

Abstract—The above paper¹ presents the design of an adaptive/robust controller for uncertain electrically-driven robots with no velocity measurements. This note shows that the claim that velocity measurements are not required for control implementation is incorrect.

Index Terms—Adaptive control, electrically-driven robots, robot control, velocity measurements.

This note addresses the adaptive/robust controller proposed in the above paper¹ by Su and Stepanenko for the position control of uncertain, electrically-driven robots without velocity measurements. In the paper, the authors claim that the implementation of the proposed control law does not require joint velocity measurements; however, this claim is invalid due to the reasons described in the following discussion.

In the above-mentioned paper, a linear parametrization of the robot dynamics is defined as shown below

$$(D(q) + J) \ddot{q}_d + B(q, \dot{q}_d) \dot{q}_d + G(q) = \Phi_a(q, \dot{q}_d, \ddot{q}_d) \alpha_a \quad (1)$$

where $\Phi_a(\cdot) \in \mathbb{R}^{n \times nm}$ is a regressor matrix, $\alpha_a \in \mathbb{R}^{nm}$ is an unknown, constant parameter vector, $q(t) \in \mathbb{R}^n$ is the joint position vector, and $\dot{q}_d(t), \ddot{q}_d(t) \in \mathbb{R}^n$ denote the desired velocity and acceleration trajectory vectors, respectively. A new unknown parameter vector is then defined as follows:

$$K_N^{-1} \Phi_a \alpha_a = \Phi_a (K_{N_a}^{-1} \alpha_a) = \Phi_a \alpha_{ak} \quad (2)$$

where $K_N = \text{diag}\{k_{Ni}\}$, $i = 1, \dots, n$, contains the torque constants of the individual DC motors and $K_{N_a} = \text{diag}\{k_{Ni} I_m\}$. Since α_{ak} is unknown, a parameter estimate vector $\hat{\alpha}_{ak}$ is generated using the projection update algorithm

$$\dot{\hat{\alpha}}_{aki} = \begin{cases} 0, & \text{if } \hat{\alpha}_{aki} = \theta_{i \max} \\ & \text{and } \sigma(\Phi_a^T z)_i < 0 \\ & [\theta_{i \min} < \hat{\alpha}_{aki} < \theta_{i \max}], \text{ or} \\ & [\hat{\alpha}_{aki} = \theta_{i \max} \text{ and} \\ & \sigma(\Phi_a^T z)_i \geq 0], \text{ or} \\ -\sigma(\Phi_a^T z)_i, & \text{if } \sigma(\Phi_a^T z)_i \geq 0, \text{ or} \\ & [\hat{\alpha}_{aki} = \theta_{i \min} \text{ and} \\ & \sigma(\Phi_a^T z)_i \leq 0] \\ 0, & \text{if } \hat{\alpha}_{aki} = \theta_{i \min} \\ & \text{and } \sigma(\Phi_a^T z)_i > 0. \end{cases} \quad (3)$$

Manuscript received August 17, 1998; revised June 1, 1999 and January 2, 2000. This paper was recommended for publication by Associate Editor J. Yuh and Editor S. Salcudean upon evaluation of the reviewers' comments.

M. S. de Queiroz was with the Department of Mechanical Engineering, Polytechnic University, Brooklyn, NY 11201 USA. He is now with the Department of Mechanical Engineering, Louisiana State University, Baton Rouge, LA 70803-6413 USA (e-mail: dequeiroz@me.lsu.edu).

F. Zhang and W. Dixon are with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634-0915 USA (e-mail: fzhang@eng.clemson.edu; wdixon@eng.clemson.edu).

Publisher Item Identifier S 1042-296X(00)06953-6.

¹C.Y. Su and Y. Stepanenko, *IEEE Trans. Robot. Automat.*, vol. 14, pp. 651–655, Aug. 1998.

In (3), $\theta_{i \min}, \theta_{i \max}$ represent known parameter bounds with $i = 1, \dots, nm$, σ is a positive constant, and $z(t) \in \mathbb{R}^n$ is defined as

$$z = \dot{\tilde{q}} - \frac{1}{\gamma} w + \frac{k}{\gamma} \tilde{q} \quad (4)$$

where

$$\tilde{q}(t) = q(t) - q_d(t) \quad \dot{\tilde{q}}(t) = \dot{q}(t) - \dot{q}_d(t) \quad (5)$$

represent the joint position and velocity tracking errors, respectively, γ, k are positive constants, and $w(t) \in \mathbb{R}^n$ is a filter output which does not depend on velocity measurements. In the following, we will show that the projection algorithm-based computation of $\hat{\alpha}_{ak}$ from (3) does require velocity measurements.

First, in Remark 3 of Section B of the paper, the authors state that the role of the projection algorithm is crucial for the stability analysis. Specifically, the boundedness of $\hat{\alpha}_{ak}$, which is guaranteed by the projection algorithm, enables the proof of the semiglobal stability result. Unfortunately, notice from (3) that each case of the projection algorithm requires the evaluation of the sign of $\sigma(\Phi_a^T z)_i$. Since the definition of z in (4) involves $\dot{\tilde{q}}$, this rule-based decision process in the projection algorithm will involve velocity measurements.

We now turn our attention to the calculation of $\hat{\alpha}_{ak}$ itself. With this respect, in Remark 2 of Section B of the paper, the authors state that although $\hat{\alpha}_{ak}$ depends on the velocity \dot{q} , the signal $\hat{\alpha}_{ak}$ does not. To examine the validity of this statement, we consider the computation of $\hat{\alpha}_{ak}$ for the second case of (3), i.e., when

$$\dot{\hat{\alpha}}_{aki} = -\sigma(\Phi_a^T z)_i. \quad (6)$$

For the sake of simplifying the following derivations, we assume that $n = m = 1$ and $\sigma = 1$. After integrating (6) over time, we obtain

$$\begin{aligned} \hat{\alpha}_{ak}(t) - \hat{\alpha}_{ak}(t_0) &= -\int_{t_0}^t \Phi_a(q(\tau), \dot{q}_d(\tau), \ddot{q}_d(\tau)) z(\tau) d\tau \\ &= -\int_{t_0}^t \Phi_a(\cdot) \dot{\tilde{q}}(\tau) d\tau + \int_{t_0}^t \Phi_a(\cdot) \left(\frac{1}{\gamma} w(\tau) - \frac{k}{\gamma} \tilde{q}(\tau) \right) d\tau \\ &= -\int_{t_0}^t \Phi_a(\cdot) \dot{\tilde{q}}(\tau) d\tau + \int_{t_0}^t \Phi_a(\cdot) \\ &\quad \cdot \left(\dot{q}_d(\tau) + \frac{1}{\gamma} w(\tau) - \frac{k}{\gamma} \tilde{q}(\tau) \right) d\tau \end{aligned} \quad (7)$$

upon the use of (4) and (5). The computation of the second time integral of (7) does not constitute a problem since none of the terms depend on velocity. However, the same cannot be said about the first time integral of (7) due to the presence of $\dot{\tilde{q}}$. One attempt to eliminate the need for $\dot{\tilde{q}}$ is to transform the time integral into the following line integral along q :

$$\int_{t_0}^t \Phi_a(q(\tau), \dot{q}_d(\tau), \ddot{q}_d(\tau)) \dot{\tilde{q}}(\tau) d\tau = \int_{q(t_0)}^{q(t)} \Phi_a(u, \dot{q}_d(u), \ddot{q}_d(u)) du. \quad (8)$$

Unfortunately, (8) only holds for the time domain where the function $q(t)$ is invertible. For example, consider the simple case where

$$\Phi_a(q, \dot{q}_d, \ddot{q}_d) = \sin(q) \dot{q}_d \quad \dot{q}_d(t) = \sin(t) \quad q(t) = \cos(t). \quad (9)$$

According to (8), we have

$$\int_{t_0}^t -\sin(\cos(\tau)) \sin^2(\tau) d\tau = \int_{q(t_0)}^{q(t)} \sin(u) \sin(\cos^{-1}(u)) du \quad (10)$$

where the inverse of $\cos(t)$ is only defined for $0 \leq t \leq \pi$. More importantly, since the position is being measured on-line, one does not know in practice the function that describes $q(t)$ for all time; thus, one cannot determine *a priori* that the domain for which (8) is valid.

The problem outlined above dictates that the aforementioned integral of (7) must be performed over time. A common, alternative way of calculating this time integral is via the following integration by parts:

$$\int_{t_0}^t \Phi_a(q, \dot{q}_d, \ddot{q}_d) \dot{q}(\tau) d\tau = \Phi_a(\cdot) q(\tau) \Big|_{t_0}^t - \int_{t_0}^t \frac{d\Phi_a(\cdot)}{d\tau} q(\tau) d\tau. \quad (11)$$

Unfortunately, due to the dependence of $\Phi_a(\cdot)$ on q , the calculation of

$(d\Phi_a(\cdot))/dt$ in (11) will require velocity measurements, i.e.,

$$\frac{d\Phi_a(q, \dot{q}_d, \ddot{q}_d)}{dt} = \frac{\partial \Phi_a(\cdot)}{\partial q} \dot{q} + \frac{\partial \Phi_a(\cdot)}{\partial \dot{q}_d} \ddot{q}_d + \frac{\partial \Phi_a(\cdot)}{\partial \ddot{q}_d} \dots \ddot{q}_d. \quad (12)$$

We note that the dependence of (12) on velocity can be overcome by defining (as in [18] of the paper¹) a new linear parametrization of the robot dynamics as follows:

$$(D(q_d) + J) \ddot{q}_d + B(q_d, \dot{q}_d) \dot{q}_d + G(q_d) = \Phi_{da}(q_d, \dot{q}_d, \ddot{q}_d) \alpha_a \quad (13)$$

where $\Phi_{da}(q_d, \dot{q}_d, \ddot{q}_d)$ denotes the *desired* regressor matrix. Since $\Phi_{da}(\cdot)$ is only a function of the desired motion trajectory, the computation of $(d\Phi_{da}(\cdot))/dt$ will not require the actual velocity. Note that in this case, an additional term will have to be included in the design of the embedded control input I_d to account for the “mismatch” between the desired regressor $\Phi_{da}(\cdot)$ and the actual regressor $\Phi_a(\cdot)$ (see [18] in the paper¹ for details).