

Alexander Bertino
Dynamic Systems and Control Laboratory,
Department of Mechanical Engineering,
San Diego State University,
San Diego, CA 92182
e-mail: abertino6245@sdsu.edu

Peiman Naseradinmousavi
Associate Professor
Dynamic Systems and Control Laboratory,
Department of Mechanical Engineering,
San Diego State University,
San Diego, CA 92182
e-mail: pnaseradinmousavi@sdsu.edu

Atul Kelkar
D. W. Reynolds Distinguished Professor
Department Chair of Mechanical Engineering,
Clemson University,
Clemson, SC 29632
e-mail: atul@clemson.edu

Analytical and Experimental Decentralized Adaptive Control of a High-Degrees-of-Freedom Robot Manipulator

In this paper, we study the analytical and experimental control of a seven degrees-of-freedom (7DOF) robot manipulator. A model-free decentralized adaptive control strategy is presented for the tracking control of the manipulator. The problem formulation and experimental results demonstrate the computational efficiency and simplicity of the proposed method. The results presented here are one of the first known experiments on a redundant 7DOF robot. The efficacy of the adaptive decentralized controller is demonstrated experimentally by using the Baxter robot to track a desired trajectory. Simulation and experimental results clearly demonstrate the versatility, tracking performance, and computational efficiency of this method. [DOI: 10.1115/1.4049795]

1 Introduction

As the global trend is toward increased automation, robot manipulators have seen widespread use in many industrial applications. While the research in adaptive and nonlinear control has seen significant advances, most robot manipulators utilized in industry are driven by simple decentralized proportional-integral-derivative (PID) controllers due to their simplicity in their design and implementation [1,2]. While these controllers are effective at driving robot manipulators to specific set points, they have difficulty in tracking an arbitrary desired trajectory. Furthermore, due to the strong interconnected nonlinearities inherently present in the dynamic model of such systems, a given set of PID gains will only work well for a specific joint configuration and end-effector mass. While many pick-and-place type operations in industry not needing navigation through obstacles can be performed effectively using PID type controllers, the tasks requiring sophisticated path planning and tracking need advanced controls. In order to maintain acceptable performance across a larger range of joint configurations, one might consider utilizing a gain scheduling PID controller, such as presented in Refs. [3] and [4]. While these controllers can theoretically achieve desirable performance under such circumstances, most implementations of these controllers will require determining acceptable PID gains for a multitude of linearized models at different operating conditions. For a seven degrees-of-freedom (7DOF) manipulator tracking an arbitrary trajectory, the number of such linearizations required will be too large and cumbersome. Additionally, such a method would not account for an unknown end-effector mass. As society looks toward the use of robot manipulators that can interact with humans in social settings, rescue operations, and potential medical applications, the requirement that such manipulators must adhere to an arbitrary desired trajectory during motion becomes an important task. The decentralized adaptive control approach presented here provides one effective control strategy for high-performance robot operations for which PID control might not give desirable performance. Such an approach retains much of the simplicity and computational efficiency of the decentralized PID approach, while offering a wide range of applicability with extended joint configuration space and variability of end-effector masses.

Due to the strength of the dynamic interconnection between joints, a model-based approach in which the system is split into a set of decoupled systems is not feasible for robot manipulators. Instead, there are several different methods designed to work around this constraint to achieve desirable performance. First, neural network-based methods [5,6], as well as the disturbance observer method by Yang et al. [7], and model-reference method such as by Sundareshan and Koenig [8], attempt to obtain a model of certain system behaviors during the operation of the robot manipulator. Such adaptive-model based methods do not suffer from unmodeled system dynamics and are well suited for tasks in which the joint dynamics change during the operation of a task. Another popular approach to the decentralized adaptive control of robot manipulators is the model-free approach [9–12], in which the adaptive control law is governed purely from the performance of the manipulator in the tracking task. Model-free approaches, such as that by Seraji [9], can bear strong similarity to the decentralized PID approach. In such approaches, the static gains associated with the PID approach are replaced with adaptive gains, which change during the execution of the task to better track the desired trajectory. Other research efforts for decentralized control of various systems can be found in Refs. [13–29].

The goal of this paper is to develop a control formulation and conduct an experimental verification of the model-free decentralized adaptive method using Baxter, a 7DOF redundant robot manipulator. This work is novel in that the experimental verification of a decentralized adaptive controller for a 7DOF manipulator is not currently addressed in literature. The decentralized adaptive control of such a manipulator is an important and challenging task. The increased degrees-of-freedom of the robot manipulator leads to an increased dynamic interconnection between joints, which is a challenge for decentralized approaches. Also, the Baxter arm configuration is a more likely choice for the complex tasks to be performed in an industrial setting. Through the analytical formulation and experimental verification of the decentralized adaptive approach, we seek to demonstrate the feasibility and computational effectiveness of said approach, in order to facilitate its adoption into industry practices. For this purpose, the model-free decentralized adaptive approach examined in this paper is an effective choice, as its structure is similar to the decentralized PID controllers currently utilized in industry.

It is important to mention the existence of a similar model-free decentralized approach, known as model-free control (MFC). MFC is a decentralized method developed in order to compensate for uncertainties in nonlinear systems and has been shown to be

Contributed by the Dynamic Systems Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received August 7, 2020; final manuscript received January 5, 2021; published online February 19, 2021. Assoc. Editor: Pranav Bhounsule.

effective in the control of many uncertain dynamical systems, including robot manipulators [30–32]. In order to compensate for these uncertainties, such as changes in the inertia matrix during the motion of the robot manipulator, the uncertainties are estimated directly utilizing the torque and state information from the previous time-step. In addition to this compensation, a decentralized PID controller is typically employed to drive the robot manipulator toward the desired trajectory. Thus, this method is capable of adjusting to a wide range of operating conditions without needing to tune adaptive gains during the procedure. However, despite the simplicity and effectiveness of this method, it has a few important drawbacks, especially when considering its possible implementation on Baxter. First, in order to estimate the nonlinear uncertainties of a robot manipulator with the MFC approach, it is necessary to numerically calculate the angular acceleration of each joint. This numerical approximation of the second derivative is highly susceptible to noise, as it amplifies the noise already encountered when calculating the angular velocity of each joint. Thus, the uncertainty compensation employed by MFC is susceptible to noise when implemented on a robot manipulator. Second, calculating the system uncertainties based on data from the previous time-step introduces bias into the uncertainty estimate. This bias can be large when the system dynamics change quickly, such as changes in frictional terms when the angular velocity of a joint changes sign. Additionally, the controller time-step must be sufficiently small in order to make the bias negligible. As Baxter is typically sampled at 100 Hz, this sampling rate may not be fast enough to ensure a low enough bias. This bias introduces a disturbance in the manipulator error dynamics that can lead to imperfect tracking of the desired trajectory. Due to these potential drawbacks of the MFC method when applied to robot manipulators, the authors believe the model-free decentralized adaptive approach studied here to be the more promising method for the decentralized control of Baxter.

The paper is organized as follows: In Sec. 2, we present a brief overview of the dynamics of Baxter's right manipulator. Additionally, we also present a decentralized model of Baxter's joint dynamics, as well as the structure of the model-free decentralized adaptive approach. In Sec. 3.1, we utilize Lyapunov method to derive the update law for the adaptive gains of the controller, demonstrating asymptotic stability in the process. In Sec. 4, we demonstrate and analyze the performance of the decentralized adaptive approach on a simulation of Baxter executing the desired trajectory, paying close attention to tracking performance, controller effort, and selection of adaptive gains. In Sec. 5, we repeat the same procedure on the Baxter robot in practice, and thoroughly compare the experimental performance to that derived from the simulation. Finally, in Sec. 6, we present the case that the decentralized adaptive method is computationally efficient, simple to implement, effective at tracking a desired trajectory, and is a better alternative to both decentralized PID controllers and centralized controllers for robot manipulators.

2 Mathematical Modeling

The redundant manipulator, which is being studied here, has 7DOF as shown in Figs. 1 and 2. The Baxter manipulator's Denavit–Hartenberg parameters are shown in Table 1 provided by the manufacturer.

The Euler–Lagrange formulation leads to a set of seven coupled nonlinear second-order ordinary differential equations

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau + F(\dot{q}) \quad (1)$$

where $q, \dot{q}, \ddot{q} \in \mathbb{R}^7$ are angles, angular velocities, and angular accelerations of joints, respectively, and $\tau \in \mathbb{R}^7$ indicates the vector of joints' driving torques. Also, $M(q) \in \mathbb{R}^{7 \times 7}$ is a symmetric mass-inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{7 \times 7}$ is a matrix of Coriolis coefficients, $G(q) \in \mathbb{R}^7$ is a vector of gravitational loading, and $F(\dot{q}) \in \mathbb{R}^7$ represents a vector of frictional torques. For the purpose of

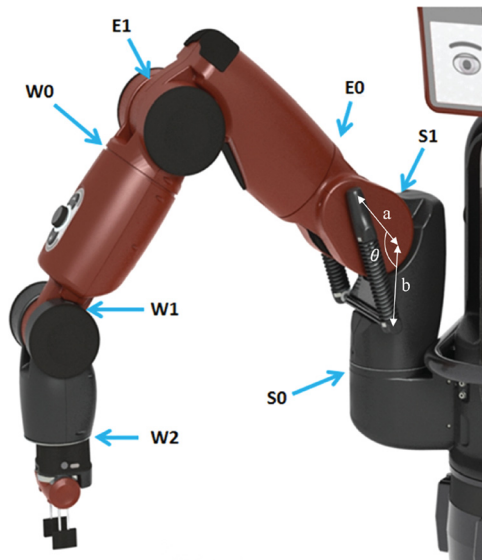


Fig. 1 7DOF Baxter's arm. Each of Baxter's seven joints is appropriately labeled. Note that S, E, and W refer to joints located on Baxter's shoulder, elbow, and wrist, respectively.

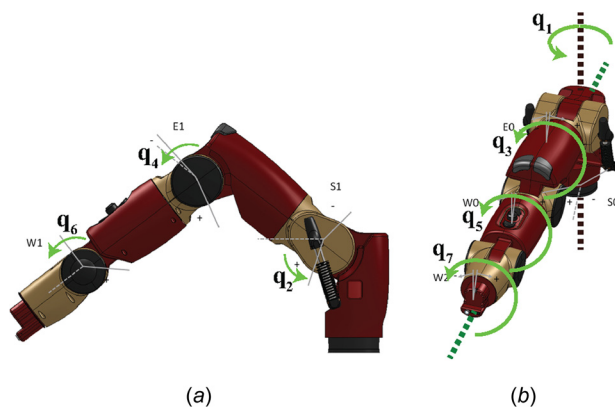


Fig. 2 Joints' configuration: (a) sagittal view and (b) top view

Table 1 Baxter's Denavit–Hartenberg [33] parameters

Link/joint	a_i	d_i	α_i	q_i
1/S ₀	0.069	0.27035	$-\pi/2$	q_1
2/S ₁	0	0	$\pi/2$	$q_2 + \pi/2$
3/E ₀	0.069	0.36435	$-\pi/2$	q_3
4/E ₁	0	0	$\pi/2$	q_4
5/W ₀	0.010	0.37429	$-\pi/2$	q_5
6/W ₁	0	0	$\pi/2$	q_6
7/W ₂	0	0.3945	0	q_7

simulation, the frictional torque F , which represents friction in both the joints and their corresponding actuators, is modeled utilizing the hyperbolic tangent function in order to approximate the behavior of Coulomb friction

$$F = -c_1^T \tanh(c_2 \dot{q}) \quad (2)$$

where $c_1 \in \mathbb{R}^7$ represents the magnitude of the Coulomb friction at each joint, and $c_2 \in \mathbb{R}$ is a parameter chosen so that the model of Coulomb friction is sufficiently smooth to enable accurate numerical simulation of (1). Our verified coupled nonlinear

dynamic model of the robot [34–42] is used as the basis of the decentralized adaptive approach. Also, the following assumption is made for the desired joint trajectories.

ASSUMPTION 1. *The desired joint trajectories are designed such that $q_r(t)$, $\dot{q}_r(t)$, and $\ddot{q}_r(t) \in \mathbb{R}^7$ exist and are bounded for all $t \geq 0$.*

2.1 Decentralized Model Formulation. In order to derive the decentralized adaptive controller, it is necessary to model the dynamics of a single joint, rather than the system as a whole. Rewriting (1) as series of seven differential equations yields

$$m_{ii}(q)\ddot{q}_i + \left[\sum_{j=1, j \neq i}^n m_{ij}(q)\ddot{q}_j \right] + c_i(q, \dot{q})\dot{q} + g_i(q) = T_i(t) + F_i(\dot{q}) \quad (3)$$

where m_{ij} is the element in the mass matrix located at (i, j) , $c_i(q, \dot{q})$ is the i th row of the Coriolis matrix, $g_i(q)$ is the i th element of the gravity vector, $T_i(t)$ is the input torque at joint i , and $F_i(\dot{q})$ is the frictional torque at joint i . Note that this equation represents the angular acceleration at joint i as a function of the input torque only at joint i , and the dynamics of each link $q, \dot{q}, \ddot{q}$. Thus, (1) can be reduced to a series of seven dynamically interconnected SISO systems. In order to further express this concept, we rewrite (3) as

$$m_{ii}(q)\ddot{q}_i + d_i(q, \dot{q}, \ddot{q}) = T_i(t) \quad (4)$$

where $d_i(q, \dot{q}, \ddot{q}) = [\sum_{j=1, j \neq i}^n m_{ij}(q)\ddot{q}_j] + c_i(q, \dot{q})\dot{q} + g_i(q) - F_i(\dot{q})$ represents the dynamic interconnection between joints.

3 Decentralized Adaptive Controller

In order to track an arbitrary desired trajectory, we employ the following decentralized adaptive control structure

$$T_i(t) = f_i(t) + k_{i1}(t)e_i(t) + k_{i2}(t)\dot{e}_i(t) + z_{i1}(t)\dot{q}_{ri}(t) + z_{i2}(t)\ddot{q}_{ri}(t) \quad (5)$$

where $q_{ri}(t)$ is the desired reference trajectory, $e_i(t) = q_{ri}(t) - q_i(t)$ is the tracking error, and $f_i(t)$, $k_{i1}(t)$, $k_{i2}(t)$, $z_{i1}(t)$, $z_{i2}(t)$ are adaptive control signals to be determined through the application of Lyapunov methods. In this formulation, $f_i(t)$ is termed the auxiliary signal and is the primary driver of the system state $q_i, \dot{q}_i$ toward the desired trajectory. $k_{i1}(t)$, $k_{i2}(t)$ are adaptive proportional-derivative gains intended to account for current error in the tracking performance, adjusting to the dynamics of the current joint configuration. Similarly, $z_{i1}(t)$, $z_{i2}(t)$ are adaptive feed-forward velocity and acceleration gains, intended to ensure that the joint stays on the desired trajectory.

3.1 Derivation of Update Law. In order to derive the equations of the adaptive control signals, we first make the following assumption:

ASSUMPTION 2. *The mass element m_{ii} , and the dynamic interconnection between the joints $d_i(q, \dot{q}, \ddot{q})$, are slowly time varying with respect to the desired trajectory $q_{ri}(t)$. That is, $\dot{m}_{ii} \approx 0$ and $\dot{d}_i \approx 0$.*

Utilizing this assumption, the decentralized model (4), and the controller law (5), we can express the model plus controller dynamics as

$$m\ddot{q} + d = f + k_1 e + k_2 \dot{e} + z_1 \dot{q}_r + z_2 \ddot{q}_r \quad (6)$$

Note that the i th subscript, as well as notations indicating functions of time and joint configuration $(t, q, \dot{q}, \ddot{q})$, have been removed for the sake of notational simplicity. This equation can be rearranged to obtain:

$$m\ddot{e} + k_2 \dot{e} + k_1 e = d - f - z_1 \dot{q}_r + (m - z_2) \ddot{q}_r \quad (7)$$

Furthermore, defining the error state vector as $X = [e, \dot{e}]^T$, (7) can be rewritten in state-space form to obtain:

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ \frac{-k_1}{m} & \frac{-k_2}{m} \end{bmatrix} X + \begin{bmatrix} 0 \\ \frac{d-f}{m} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{-z_1}{m} \end{bmatrix} \dot{q}_r + \begin{bmatrix} 0 \\ \frac{m-z_2}{m} \end{bmatrix} \ddot{q}_r \quad (8)$$

In order to ensure that the robot manipulator follows the desired trajectory, we define the desired performance of the tracking error $e_s(t)$, which we define with the following second-order homogeneous differential equation:

$$\ddot{e}_s + 2\xi\omega_n \dot{e}_s + \omega_n^2 e_s = 0 \quad (9)$$

where ω_n is the natural frequency of the desired performance and ξ is the damping ratio. Similar to (7), we define the reference state vector $X_s = [e_s, \dot{e}_s]^T$ and rewrite (9) in state-space form to obtain:

$$\dot{X}_s = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix} X_s = A X_s \quad (10)$$

Next, we use the following theorem to prove a crucial property of the reference model (10).

THEOREM 1. *Consider the linear state-space model $\dot{x} = Ax$. The equilibrium $x=0$ is globally asymptotically stable if and only if $\exists P = P^T > 0$, $\exists Q = Q^T > 0$ such that the following Lyapunov equation holds [43]:*

$$PA + A^T P = -Q \quad (11)$$

Since we are free to define ξ and ω_n in such a manner as to ensure (10) is globally asymptotically stable, then by Theorem 1 there exists a unique symmetric positive definite matrix P that solves (11) for the linear system (10). We denote the elements in P

$$P = \begin{bmatrix} P_1 & P_2 \\ P_2 & P_3 \end{bmatrix} \quad (12)$$

Next, we define $E = X_s - X$, and combine (8) and (10) to obtain the tracking error state-space model:

$$\begin{aligned} \dot{E} = & \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\xi\omega_n \end{bmatrix} E + \begin{bmatrix} 0 & 1 \\ \frac{k_1}{m} - \omega_n^2 & \frac{k_2}{m} - 2\xi\omega_n \end{bmatrix} X \\ & + \begin{bmatrix} 0 \\ \frac{f-d}{m} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{z_1}{m} \end{bmatrix} \dot{q}_r + \begin{bmatrix} 0 \\ \frac{z_2-m}{m} \end{bmatrix} \ddot{q}_r \end{aligned} \quad (13)$$

In order to determine the stability properties of (13), it is first necessary to define a Lyapunov function for the system. For this system, we define the following Lyapunov function:

$$\begin{aligned} V = & E^T P E + Q_0 \left(\frac{f-d}{m} - f^* \right)^2 + Q_1 \left(\frac{k_1}{m} - \omega_n^2 - k_1^* \right)^2 \\ & + Q_2 \left(\frac{k_2}{m} - 2\omega_n \xi - k_2^* \right)^2 + Q_3 \left(\frac{z_1}{m} - z_1^* \right)^2 \\ & + Q_4 \left(\frac{z_2-m}{m} - z_2^* \right)^2 \end{aligned} \quad (14)$$

where $Q_0, \dots, Q_4$ are positive scalars, and $f^*, k_1^*, k_2^*, z_1^*, z_2^*$ are functions of time to be determined later. Differentiating (14) with respect to time and applying Assumption 2 yields:

$$\begin{aligned}\dot{V} = & -E^T Q E + 2\left(\frac{f-d}{m}\right)\left[Q_0\left(\frac{\dot{f}}{m}-\dot{f}^*\right)-r\right] \\ & -2Q_0^*\left(\frac{\dot{f}}{m}-\dot{f}^*\right)+2\left(\frac{k_1-\omega_n^2}{m}\right)\left[Q_1\left(\frac{\dot{k}_1}{m}-\dot{k}_1^*\right)-re\right] \\ & -2Q_1^*\left(\frac{\dot{k}_1}{m}-\dot{k}_1^*\right)+2\left(\frac{k_2-2\zeta\omega_n}{m}\right)\left[Q_2\left(\frac{\dot{k}_2}{m}-\dot{k}_2^*\right)-r\dot{e}\right] \\ & -2Q_2^*\left(\frac{\dot{k}_2}{m}-\dot{k}_2^*\right)+2\left(\frac{z_1}{m}\right)\left[Q_3\left(\frac{\dot{z}_1}{m}-\dot{z}_1^*\right)-r\dot{q}_r\right] \\ & -2Q_3^*\left(\frac{\dot{z}_1}{m}-\dot{z}_1^*\right)+2\left(\frac{z_2-m}{m}\right)\left[Q_4\left(\frac{\dot{z}_2}{m}-\dot{z}_2^*\right)-r\ddot{q}_r\right] \\ & -2Q_4^*\left(\frac{\dot{z}_2}{m}-\dot{z}_2^*\right)\end{aligned}\quad (15)$$

where $r = P_2 e + P_3 \dot{e}$ is the weighted error. Before continuing the derivation, we make note of the following theorem:

THEOREM 2. Let $X \in \mathbb{R}^n = 0$ be an equilibrium point of the system $\dot{X} = f(X)$, and let $V : \mathbb{R}^n \rightarrow \mathbb{R}$ [43]:

- (1) If $V(0) = 0$, $V(X) > 0 \forall X \neq 0$, $\dot{V} \leq 0 \forall X \neq 0$, then $X = 0$ is globally stable
- (2) If $V(X) \rightarrow \infty$ as $\|X\| \rightarrow \infty$, then $V(X)$ is radially unbounded
- (3) If $X = 0$ is stable, $V(X)$ is radially unbounded, and $\dot{V} < 0 \forall X \neq 0$, then $X = 0$ is globally asymptotically stable

We first note that per our definition of V in (14), V is both positive when $E \neq 0$ and radially unbounded. Thus, we seek to derive adaptation parameters $f^*, k_1^*, k_2^*, z_1^*, z_2^*$ such that V is negative definite, and thus $E = 0$ is globally asymptotically stable. First, we set the following terms in (15) to 0:

$$\begin{aligned}Q_0\left(\frac{\dot{f}}{m}-\dot{f}^*\right)-r &= 0, \quad Q_1\left(\frac{\dot{k}_1}{m}-\dot{k}_1^*\right)-re = 0 \\ Q_2\left(\frac{\dot{k}_2}{m}-\dot{k}_2^*\right)-r\dot{e} &= 0, \quad Q_3\left(\frac{\dot{z}_1}{m}-\dot{z}_1^*\right)-r\dot{q}_r = 0 \\ Q_4\left(\frac{\dot{z}_2}{m}-\dot{z}_2^*\right)-r\ddot{q}_r &= 0\end{aligned}\quad (16)$$

Substituting Eq. (16) into Eq. (15) yields the following equation:

$$\dot{V} = -E^T Q E - 2f^* r - 2k_1^* re - 2k_2^* r\dot{e} - 2z_1^* r\dot{q}_r - 2z_2^* r\ddot{q}_r \quad (17)$$

We then define the following terms

$$\begin{aligned}f^* &= Q_0^* r k_1^* = Q_1^* re k_2^* = Q_2^* r\dot{e} \\ z_1^* &= Q_3^* r\dot{q}_r z_2^* = Q_4^* r\ddot{q}_r\end{aligned}\quad (18)$$

where $Q_0^*, \dots, Q_4^*$ are positive scalars. Substituting (18) into (17) yields

$$\begin{aligned}\dot{V} = & -E^T Q E - 2Q_0^* r^2 - 2Q_1^* r^2 e^2 - 2Q_2^* r^2 \dot{e}^2 - 2Q_3^* r^2 \dot{q}_r^2 \\ & - 2Q_4^* r^2 \ddot{q}_r^2\end{aligned}\quad (19)$$

which is negative for all $E \neq 0$, thus Theorem 2 is satisfied and $E = 0$ is globally asymptotically stable. However, we must now determine the values of the parameters f, k_1, k_2, z_1, z_2 to satisfy (16), which are as follows:

$$\begin{aligned}\dot{f} &= mQ_0^* \dot{r} + \frac{m}{Q_0} r \\ \dot{k}_1 &= mQ_1^* \frac{d}{dt}(re) + \frac{m}{Q_1} re \\ \dot{k}_2 &= mQ_2^* \frac{d}{dt}(r\dot{e}) + \frac{m}{Q_2} r\dot{e} \\ \dot{z}_1 &= mQ_3^* \frac{d}{dt}(r\dot{q}_r) + \frac{m}{Q_3} r\dot{q}_r \\ \dot{z}_2 &= mQ_4^* \frac{d}{dt}(r\ddot{q}_r) + \frac{m}{Q_4} r\ddot{q}_r\end{aligned}\quad (20)$$

We then define the following terms so that (20) is independent of m :

$$\begin{aligned}Q_0^* &= \frac{\rho}{m} Q_0 = \frac{m}{\delta} \\ Q_1^* &= \frac{\beta_1}{m} \quad Q_1 = \frac{m}{\alpha_1} \\ Q_2^* &= \frac{\beta_2}{m} \quad Q_2 = \frac{m}{\alpha_2} \\ Q_3^* &= \frac{\lambda_1}{m} \quad Q_3 = \frac{m}{\gamma_1} \\ Q_4^* &= \frac{\lambda_2}{m} \quad Q_4 = \frac{m}{\gamma_2}\end{aligned}\quad (21)$$

Substituting Eq. (21) into Eq. (20) and integrating with respect to time yields the following equations for the decentralized adaptive parameters:

$$\begin{aligned}f(t) &= f(0) + \delta \int_0^t r(t) dt + \rho r(t) \\ k_1(t) &= k_1(0) + \alpha_1 \int_0^t r(t) e(t) dt + \beta_1 r(t) e(t) \\ k_2(t) &= k_2(0) + \alpha_2 \int_0^t r(t) \dot{e}(t) dt + \beta_2 r(t) \dot{e}(t) \quad (t) \\ z_1(t) &= z_1(0) + \gamma_1 \int_0^t r(t) \dot{q}_r(t) dt + \lambda_1 r(t) \dot{q}_r(t) \\ z_2(t) &= z_2(0) + \gamma_2 \int_0^t r(t) \ddot{q}_r(t) dt + \lambda_2 r(t) \ddot{q}_r(t)\end{aligned}\quad (22)$$

Now that we have successfully derived the decentralized adaptive gains, we make the following notes of its structure. First, the auxiliary signal can be interpreted as a decentralized PID signal, acting to guide the system toward the desired trajectory in a generalized approach. Second, each adaptive gain is updated based on the performance of the signal it multiplies in (5), as well as the weighted error. This update law is purely performance based and does not rely on a model of the system. Finally, the update of each parameter is a simple computation, where a trapezoidal approximation can be used to estimate the value of the integral at each time-step.

4 Simulation Results

In order to assess the performance of this decentralized adaptive controller, we first apply the control law described in Sec. 3 to Baxter's dynamic model (1). We apply our control methodology to a tracking problem where the desired tracking trajectories for the joints were created for a specific end-effector maneuver in Ref. [41]. While the maneuver was a pick-and-place task in Ref.

Table 2 Controller parameters for simulation and experiment

Joint	1	2	3	4	5	6	7
p_{i2}	1	1	1	1	1	1	1
p_{i3}	0.3	0.3	0.3	0.3	0.3	0.3	0.3
δ_i	30	60	40	30	7	40	2
ρ_i	30	60	40	30	7	6	2
α_{i1}	6000	6000	6000	6000	10,200	102,000	1200
β_{i1}	600	600	600	600	1020	10,200	120
α_{i2}	6	6	6	6	6	6	6
β_{i2}	0.6	0.6	0.6	0.6	0.6	0.6	0.6
γ_{i1}	60	60	60	60	60	60	60
λ_{i1}	6	6	6	6	6	6	6
γ_{i2}	60	60	60	60	60	60	60
λ_{i2}	6	6	6	6	6	6	6

[41] in which the desired joint trajectories were generated online, for our problem our interest is in using these previously generated trajectories as a reference for tracking. In this simulation, we introduce a sampling rate of 100 Hz in order to effectively model the effect of discrete sampling on the continuous-time controller. Furthermore, the controller parameters we used during this simulation can be observed in Table 2.

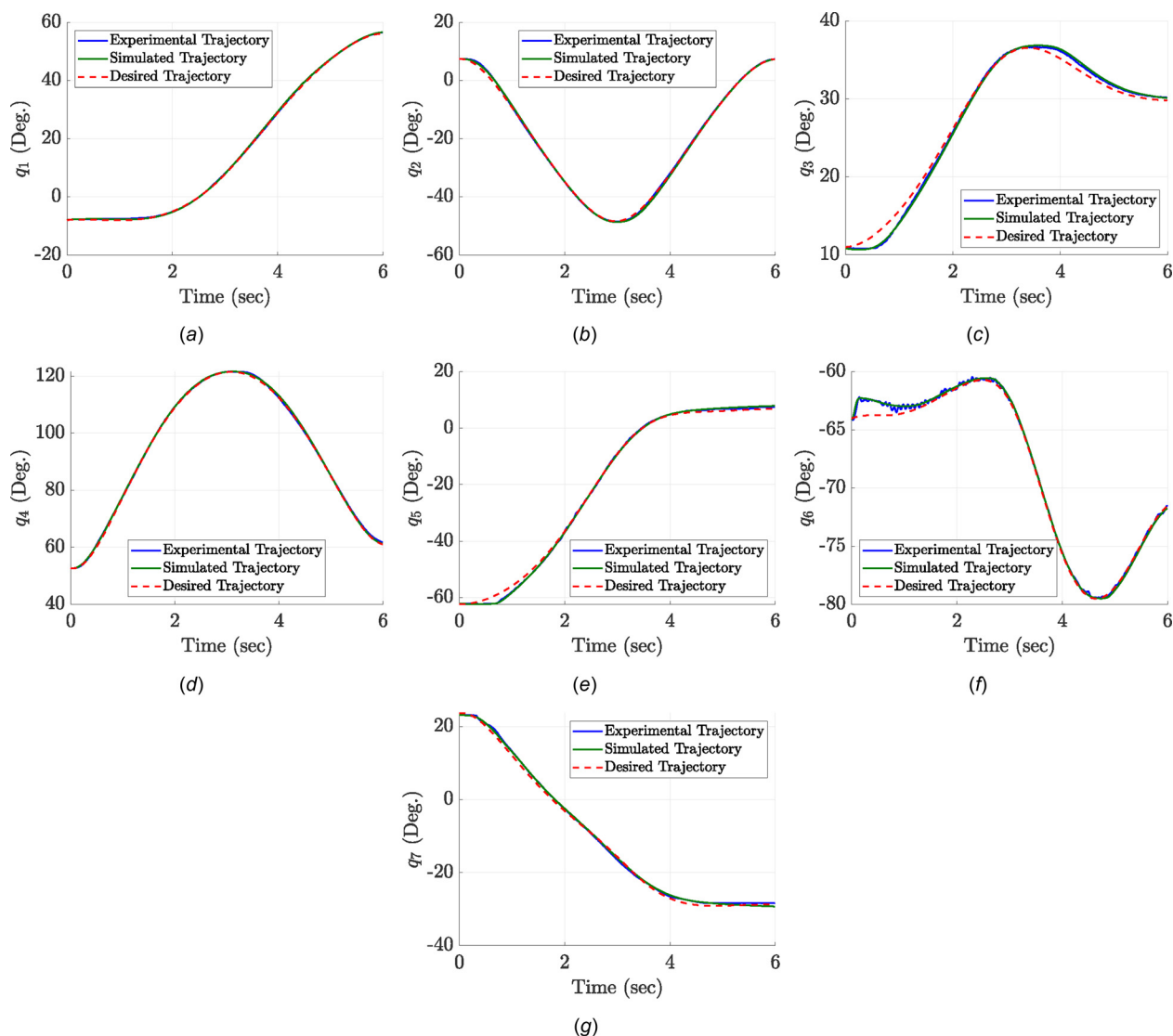
In order to determine the controller parameters to implement, the following general procedure can be performed:

- (1) Initialize all controller parameters to 0.
- (2) Choose parameters p_2, p_3, δ, ρ of the auxiliary signal $f(t)$ such that the controller adequately tracks the desired trajectory across multiple different joint configurations, ensuring satisfactory general performance. It is important to note that the auxiliary signal is equivalent to a PID controller with the following gains:

$$\begin{aligned} \mathbf{K}_P &= \delta p_3 + \rho p_2 \\ \mathbf{K}_I &= \delta p_2 \\ \mathbf{K}_D &= \rho p_3 \end{aligned} \quad (23)$$

Thus, traditional techniques used to tune the decentralized PID controllers can be used in order to determine the parameters of the auxiliary signal (Fig. 2).

- (3) Choose the minimum values of the parameters α_1, β_1 of the adaptive proportional gain $k_1(t)$ that reduce tracking error in the angular position signal during motion of the robot manipulator to a desired amount. At this stage, choose the same parameters for each joint, and set $\beta_1 = \alpha_1/10$.

**Fig. 3 Experimental (blue line), simulated (green line), and desired (red dashed line) joint trajectories of Baxter**

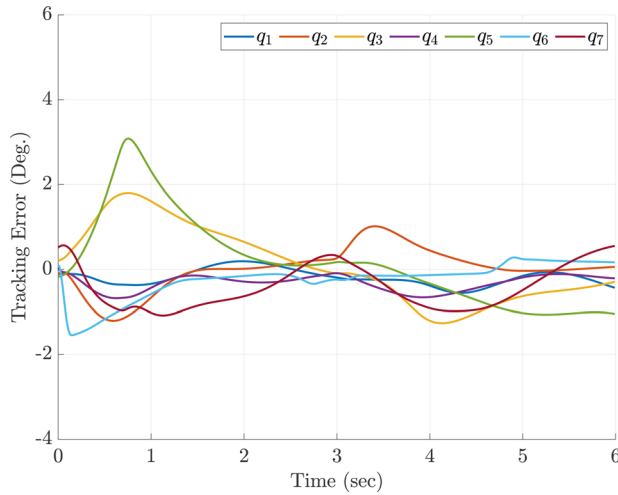


Fig. 4 Simulated tracking error

- (4) Adjust α_{i1}, β_{i1} of each joint individually if a specific joint updates too slowly or too quickly.
- (5) Repeat steps 3 and 4 on the parameters α_2, β_2 of the adaptive derivative gain $k_2(t)$ in order to reduce the tracking error in the angular velocity signal during motion to a desirable amount.
- (6) Repeat steps 3 and 4 on the parameters γ_1, λ_1 of the adaptive feedforward velocity gain $z_1(t)$ in order to adequately counteract the effect of friction in the beginning of motion.
- (7) Repeat steps 3 and 4 on the parameters γ_2, λ_2 of the adaptive feedforward acceleration gain $z_2(t)$ in order to adequately overcome the robot manipulator's inertia in the regions that the desired acceleration is large.

The simulated joint trajectories along with the desired joint trajectories can be observed in Fig. 3. From these graphs, it can be seen that the decentralized adaptive controller achieves close tracking of the desired trajectories. Although the effects of the simulated frictional torque and gravity negligibly impact the tracking performance during the beginning of motion, as can be seen in the performance of joints 3, 5, and 6, these effects are quickly accounted for by the adaptive controller. Furthermore,

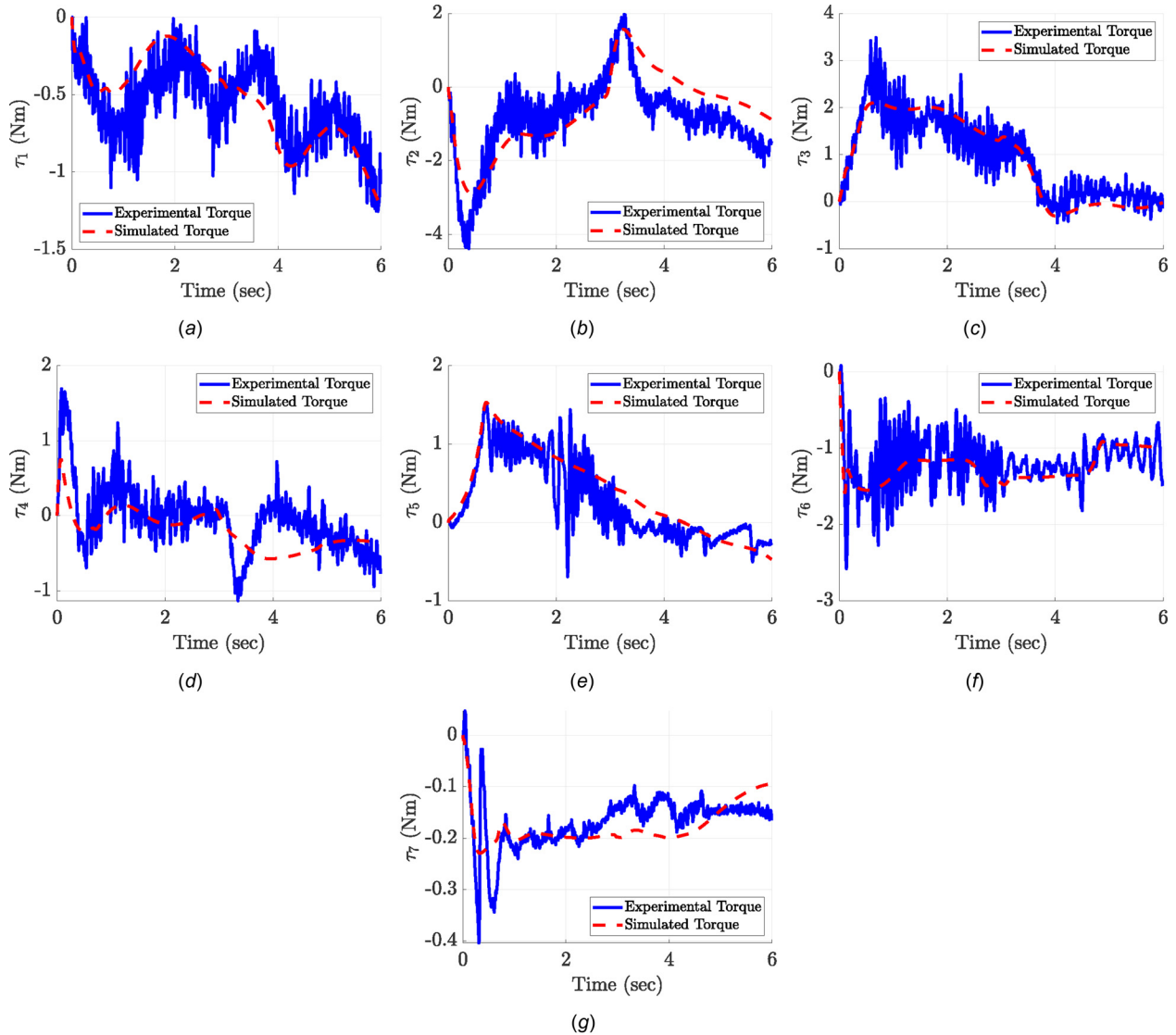


Fig. 5 Experimental (blue line) and simulated (red dashed line) joint torques of Baxter

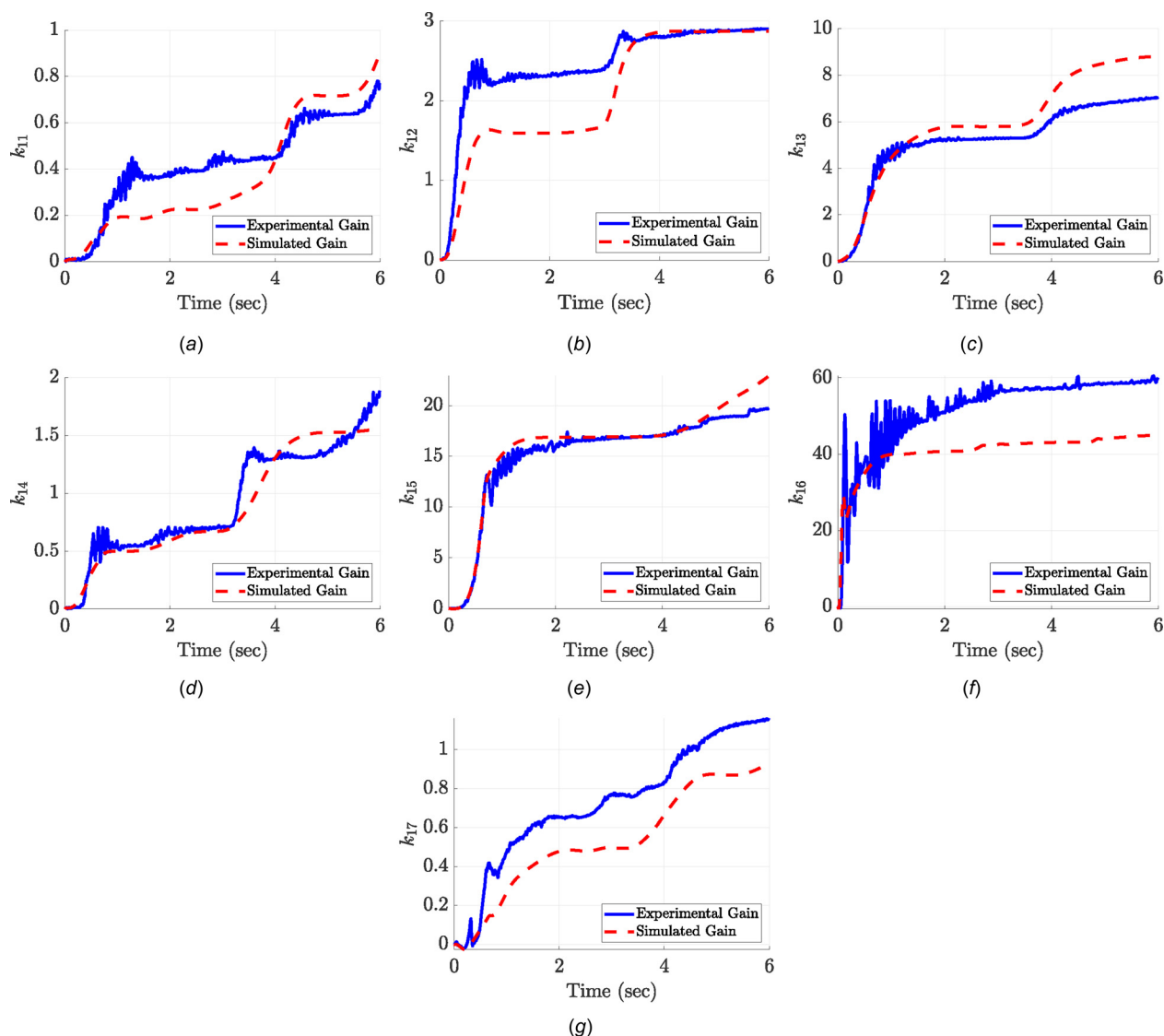


Fig. 6 Tuning of adaptive gain k_1 during experimentation (blue line) and simulation (red dashed line) of Baxter

despite large changes in the joint configuration throughout the course of the operation, the performance based control scheme remains effective at consistently driving each joint toward the desired trajectory. These behaviors can also be observed in Fig. 4, as the tracking error remains less than 1.5 deg for all joints after 1.5 s of operation.

The torques generated by the decentralized adaptive controller can be observed in Fig. 5. It is important to note that these torques are significantly lower than the maximum torque output of Baxter's joints, which are 50 Nm for joints 1–4, and 15 N m for joints 5–7, meaning that saturation of torque is not an issue for this decentralized adaptive scheme. Furthermore, this demonstrates energy efficiency of this control scheme, as the torques generated are consistently small in magnitude. Additionally, it can be observed that the torques generated are smooth throughout the operation, which is potentially beneficial to the motors that are used to generate these torques in practice.

Finally, we observe the tuning of adaptive gains k_1 , z_1 , and z_2 throughout the simulation, as seen in Figs. 6–8, respectively. Each of these gains appear to adjust in 2 stages ($0 < t < 3$ s and $3 < t < 6$ s). These phases correspond to the picking up and placing down motion of the end manipulator, signifying that a different set of gains is necessary for each task. Thus, the tuning of

these parameters coincides with our expectations of their performance. It is also important to note that these gains are of a significant magnitude when compared to the auxiliary parameters δ_i and ρ_i , meaning that tunings were necessary in order to achieve the desired tracking performance. Furthermore, the joints 3, 5, and 6 with significantly tuned gains experienced the largest frictional torques and gravitational load. These results demonstrate the ability of the decentralized adaptive controller to adjust to different operating conditions. This beneficial quality of this scheme is of key importance when the robot manipulator is expected to reliably perform in a changing environment. From these results, it is evident that the decentralized adaptive controller is effective in simulation.

5 Experimental Results

Due to promising results during simulation, we now implement the control law described in Sec. 3 to Baxter in an experimental study. We utilize the same desired trajectories as in Sec. 4 with the same 100 Hz sampling rate. Note that several differences remain between the simulated and experimental study, which include measurement noise in the joint positions and velocities, differences between the idealistic Coulomb friction model and the

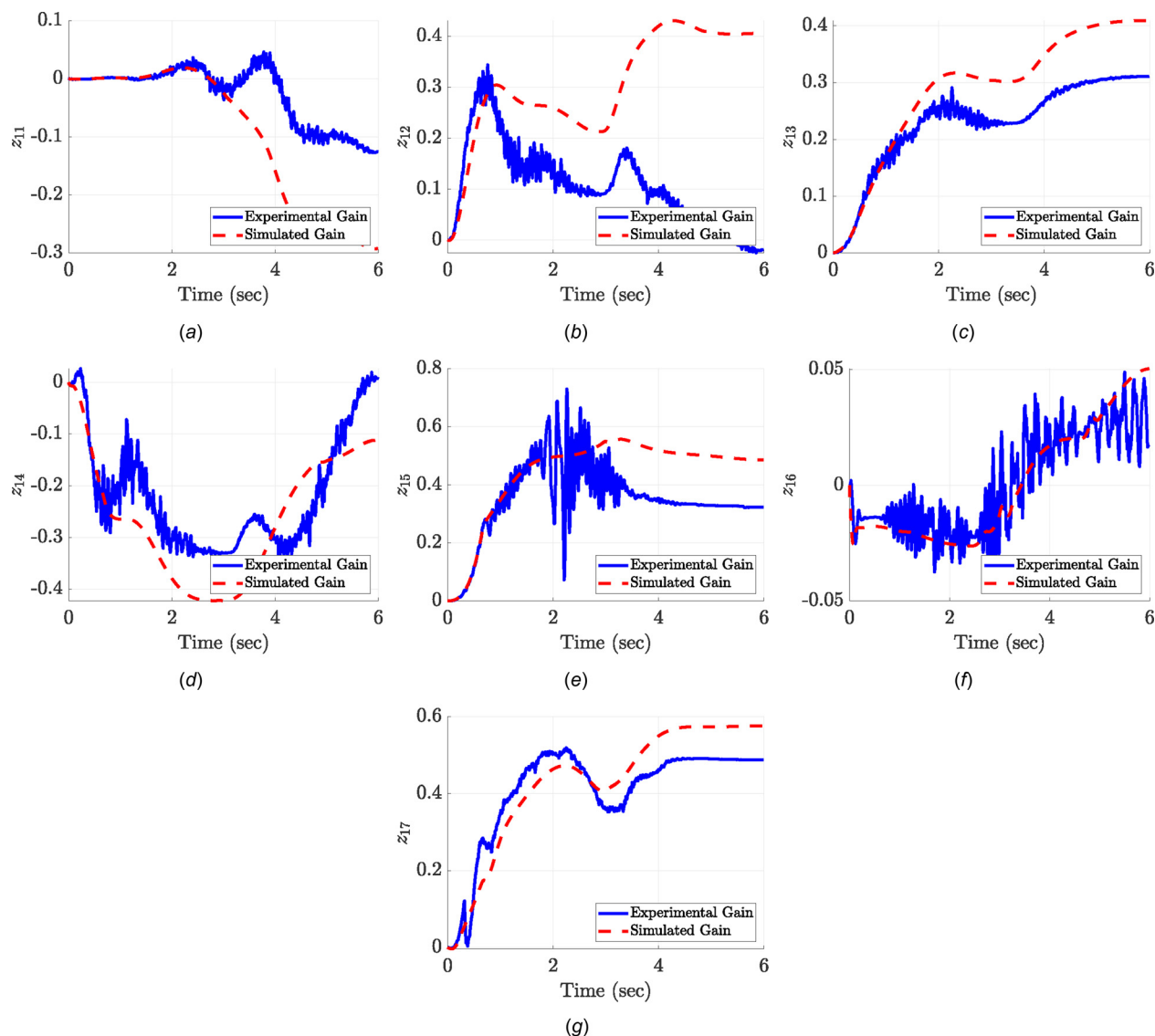


Fig. 7 Tuning of adaptive gain z_1 during experimentation (blue line) and simulation (red dashed line) of Baxter

actual friction dynamics, small potential inaccuracies in model parameters, and the actuator dynamics of each joint. These factors can lead to results slightly different than those experienced in simulation. For the experimental pick-and-place task, the controller parameters we used are the same as that of the simulation and can be observed in Table 2.

From Fig. 9(a), it can be observed that the decentralized adaptive controller is successful at executing the pick-and-place task in practice. The experimental joint trajectories along with the desired joint trajectories can be observed in Fig. 3. From these graphs, it can be seen that the decentralized adaptive controller exhibits close tracking of the desired trajectory, which is almost identical to that experienced during simulation. Similar to the experiment, it can be observed from the graphs that errors experienced in the beginning of the operation are quickly accounted for, and the controller returns to near perfect tracking. This behavior can also be observed in Fig. 10, as the tracking error remains less than 1.5 deg after 1.5 s of operation.

The torques generated by the decentralized adaptive controller can be observed in Fig. 5. While the presence of noise in measurements has caused similar variations in the joint torques, the torques still exhibit moderate continuity, as well as a magnitude much lower than the saturation torque of each joint. It can be seen from these graphs that the overall shape and magnitude of the

experimental torque of each joint match closely to that of the corresponding simulated torques. Thus, the differences in system dynamics between the simulation and experiment do not significantly affect the performance of the decentralized control algorithm.

Finally, we observe the tuning of the adaptive gains k_{1i} , z_{1i} , and z_{2i} throughout the experiment, as seen in Figs. 6–8, respectively. The behavior of these graphs is similar to that of the simulation in regard to both the stages of tuning, as well as the magnitude of the gains. Slight differences can be observed between the evolution of the gains in the simulation and experiment, which can reasonably be attributed to the small differences in dynamics between the simulated and actual system, such as the difference between the idealistic Coulomb friction model from the friction experienced in the real system. While these differences lead to the selection of different gains from simulation, the overall performance of the decentralized adaptive controller is not significantly affected by this difference in dynamics, as can be seen in Figs. 3 and 5. Thus, these adaptive gains are effective at maintaining desirable performance outside of the conditions in which the decentralized adaptive controller was designed. From these results, it is evident that the decentralized adaptive scheme performs well in experiments as well as in simulation.

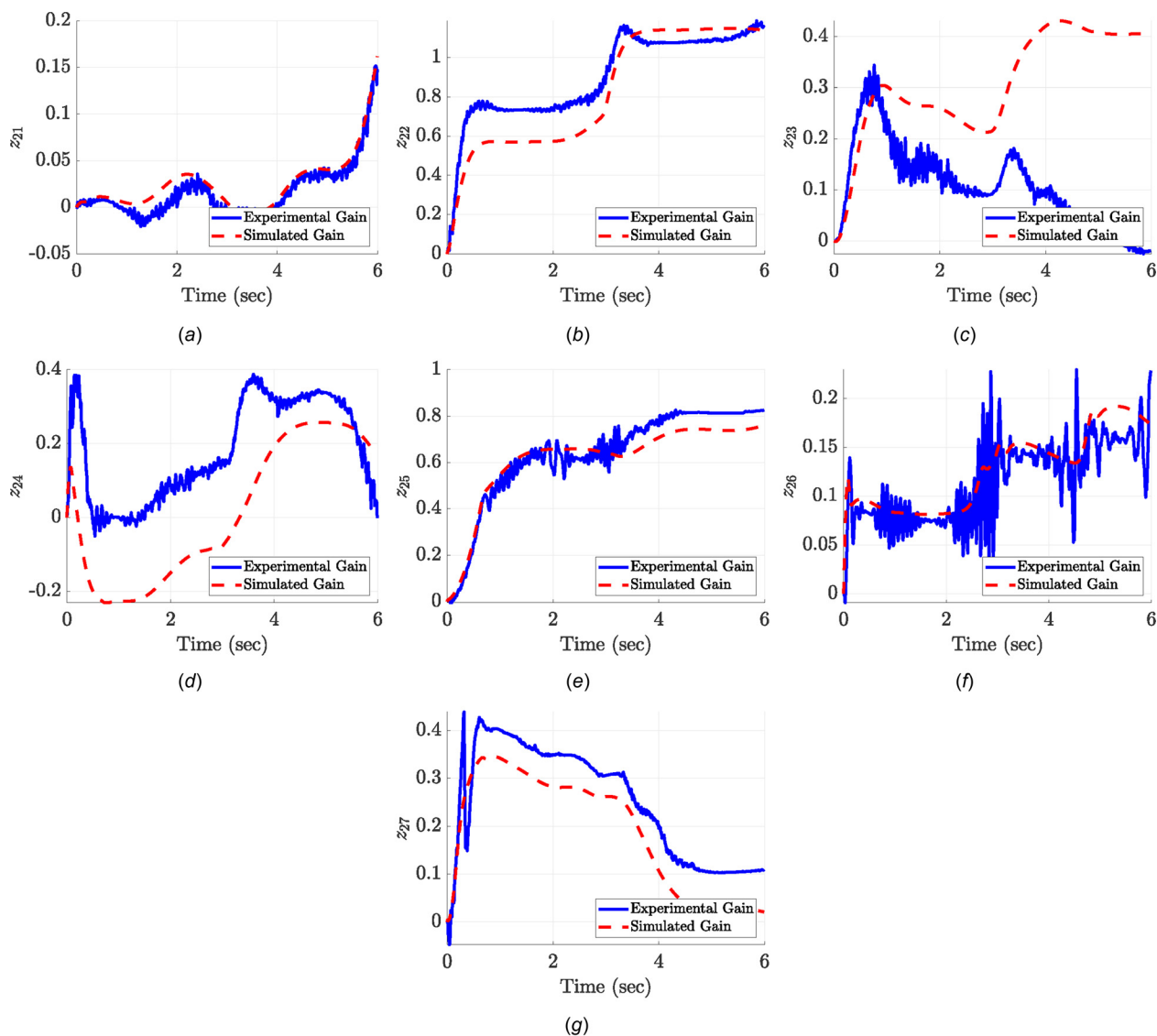


Fig. 8 Tuning of adaptive gain z_2 during experimentation (blue line) and simulation (red dashed line) of Baxter

Another crucial point to consider is the computational efficiency of the decentralized adaptive scheme compared to centralized ones. We previously carried out experimental work for a simple centralized model-based adaptive scheme to carry an unknown mass avoiding an obstacle, shown in Fig. 9(b). The mass of the end-effector was the only unknown parameter to be estimated and we again employed the damped least squares method to calculate desirable joint-space trajectories. The immediate challenge was the computation time of the control scheme in each loop, even when dealing with only one uncertainty, which was incompatible with the minimum time-step ($\Delta t_b = 0.001s$ or $f_b = 1\text{ kHz}$) of Baxter. The computation time of the centralized model-based adaptive scheme was in the range of $0.005s \leq t_c \leq 0.007s$ leading to the time delay in each control loop. Therefore, we had to address a critical tradeoff between the accuracy required and computational cost. To resolve this problem, we increased the Baxter's time-step to $\Delta t_b = 0.01s$ or $f_b = 0.1\text{ kHz}$, along with the sleep command of Python, in order to avoid such a time delay by sacrificing the accuracy needed. Shown in Fig. 9(b) is the experimental implementation of the centralized adaptive control of Baxter carried out at the DSC laboratory. We noticed that the estimation of even one uncertainty, without any external disturbance, caused at least three small operational interruptions.

Note that the decentralized adaptive scheme examined here reveals a significantly lower computation time of $\Delta t_b = 1.02\text{ ms}$ compared to the centralized one. Therefore, we did not observe the operational interruptions discussed for the centralized method whereas the decentralized scheme is at least five times faster than the centralized one. This would be highly beneficial for when we intend to control large-scale (high-DOF) systems.

6 Conclusion

In this paper, we investigated the performance of a model-free decentralized adaptive controller on a 7DOF redundant manipulator. We first formulated the theory behind the controller, demonstrating the global asymptotic stability of each local controller, as well as revealing the computationally efficient method of adapting each control parameter. Then, through the results of both our simulation and experiment of the decentralized adaptive controller implemented on Baxter, we demonstrated the following beneficial properties of the control scheme:

- (1) The algorithm is highly computationally efficient and at least five times faster than the centralized adaptive method examined here.

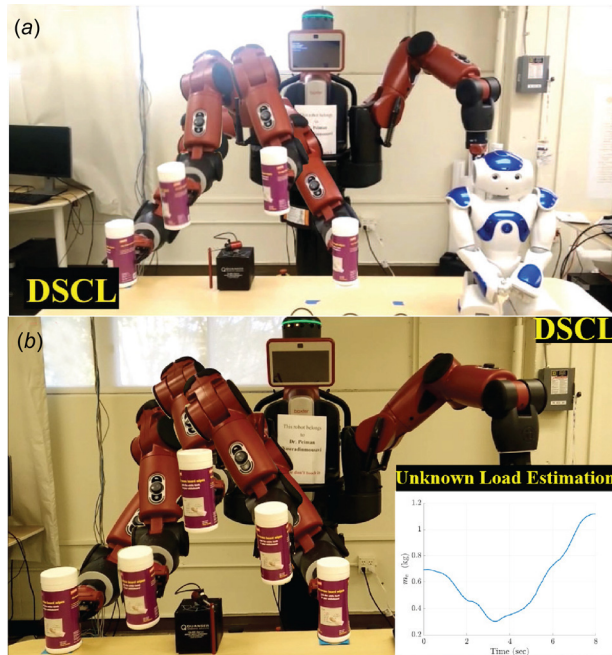


Fig. 9 Baxter tracking a desired trajectory under (a) the decentralized adaptive and (b) model-based centralized adaptive control schemes at Dynamic Systems and Control Laboratory (DSCL); see peimannm.sdsu.edu

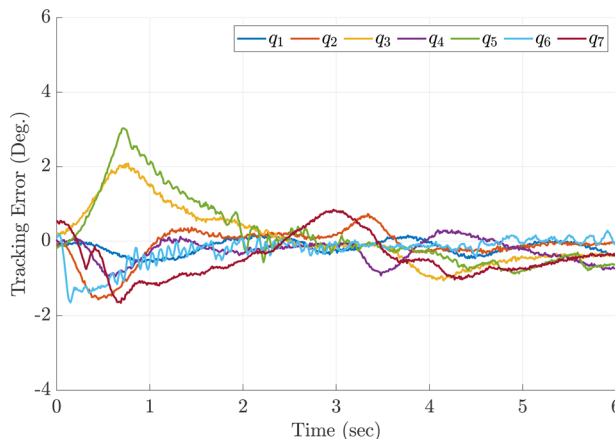


Fig. 10 Experimental tracking error

- (2) Close tracking of the desired trajectory is achieved throughout operation.
- (3) Large changes in the joint configuration throughout the procedure do not significantly affect the operation.
- (4) The generated torques are energy efficient and do not pose the risk of torque saturation.
- (5) The control scheme can adapt to and is effective outside of the conditions in which it was designed for.

Thus, we verified the effectiveness of the model-free decentralized adaptive control scheme and noted its promising potential for a wide variety of applications.

Acknowledgment

This article is based upon work supported by the National Science Foundation under Award No. 1823951. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

- [1] Alvarez-Ramirez, J., Santibanez, V., and Campa, R., 2008, "Stability of Robot Manipulators Under Saturated PID Compensation," *IEEE Trans. Control Syst. Technol.*, **16**(6), pp. 1333–1341.
- [2] Choi, Y., Chung, W. K., and Suh, I. H., 2001, "Performance and H/sub ∞ /spl infin// Optimality of PID Trajectory Tracking Controller for Lagrangian Systems," *IEEE Trans. Rob. Autom.*, **17**(6), pp. 857–869.
- [3] Jarrah, M. A., and Al-Jarrah, O. M., 1999, "Position Control of a Robot Manipulator Using Continuous Gain Scheduling," *IEEE International Conference on Robotics and Automation*, Detroit, MI, May 10–15, Paper No. 99CH36288C.
- [4] Sun, Y. L., and Er, M. J., 2004, "Hybrid Fuzzy Control of Robotics Systems," *IEEE Trans. Fuzzy Syst.*, **12**(6), pp. 755–765.
- [5] Karakasoglu, A., Sudharsanan, S. I., and Sundareshan, M. K., 1993, "Identification and Decentralized Adaptive Control Using Dynamical Neural Networks With Application to Robotic Manipulators," *IEEE Trans. Neural Networks*, **4**(6), pp. 919–930.
- [6] Tan, K. K., Huang, S., and Lee, T. H., 2009, "Decentralized Adaptive Controller Design of Large-Scale Uncertain Robotic Systems," *Automatica*, **45**(1), pp. 161–166.
- [7] Yang, Z., Fukushima, Y., and Qin, P., 2012, "Decentralized Adaptive Robust Control of Robot Manipulators Using Disturbance Observers," *IEEE Trans. Control Syst. Technol.*, **20**(5), pp. 1357–1365.
- [8] Sundareshan, M. K., and Koenig, M. A., 1985, "Decentralized Model Reference Adaptive Control of Robotic Manipulators," *American Control Conference*, Boston, MA, June 19–21, pp. 44–49.
- [9] Seraji, H., 1989, "Decentralized Adaptive Control of Manipulators: Theory, Simulation, and Experimentation," *IEEE Trans. Rob. Autom.*, **5**(2), pp. 183–201.
- [10] Colbaugh, R., Seraji, H., and Glass, K., 1994, "Decentralized Adaptive Control of Manipulators," *J. Rob. Syst.*, **11**(5), pp. 425–440.
- [11] Tarokh, M., 1996, "Decentralized Adaptive Tracking Control of Robot Manipulators," *J. Rob. Syst.*, **13**(12), pp. 803–816.
- [12] Liu, M., 1999, "Decentralized Control of Robot Manipulators: Nonlinear and Adaptive Approaches," *IEEE Trans. Autom. Control*, **44**(2), pp. 357–363.
- [13] Chen, Y. H., 1991, "Decentralized Adaptive Robust Control Design: The Uncertainty is Time Varying," *ASME J. Dyn. Syst., Meas., Control*, **113**(3), pp. 515–518.
- [14] Hua, C., Guan, X., and Shi, P., 2005, "Robust Decentralized Adaptive Control for Interconnected Systems With Time Delays," *ASME J. Dyn. Syst., Meas., Control*, **127**(4), pp. 656–662.
- [15] Lyou, J., and Bien, Z., 1985, "Decentralized Adaptive Stabilization of a Class of Large-Scale Interconnected Discrete Systems," *ASME J. Dyn. Syst., Meas., Control*, **107**(1), pp. 106–109.
- [16] Li, Y., Lu, Z., Zhou, F., Dong, B., Liu, K., and Li, Y., 2019, "Decentralized Trajectory Tracking Control for Modular and Reconfigurable Robots With Torque Sensor: Adaptive Terminal Sliding Control-Based Approach," *ASME J. Dyn. Syst., Meas., Control*, **141**(6), p. 061003.
- [17] Hernández-Alemán, R., Salas-Peña, O., and León-Morales, J. D., 2017, "Decentralized Formation Control Based on Adaptive Super Twisting," *ASME J. Dyn. Syst., Meas., Control*, **139**(4), p. 044502.
- [18] Watanabe, K., 1989, "A Decentralized Multiple Model Adaptive Filtering for Discrete-Time Stochastic Systems," *ASME J. Dyn. Syst., Meas., Control*, **111**(3), pp. 371–377.
- [19] Yan, J.-J., 2003, "Memoryless Adaptive Decentralized Sliding Mode Control for Uncertain Large-Scale Systems With Time-Varying Delays," *ASME J. Dyn. Syst., Meas., Control*, **125**(2), pp. 172–176.
- [20] Hashemipour, S. H., Vasegh, N., and Sedigh, A. K., 2017, "Decentralized MRAC for Large-Scale Interconnected Systems With State and Input Delays by Integrators Inclusion," *ASME J. Dyn. Syst., Meas., Control*, **139**(9), p. 091009.
- [21] Elmahdi, A., Taha, A. F., Sun, D., and Panchal, J. H., 2015, "Decentralized Control Framework and Stability Analysis for Networked Control Systems," *ASME J. Dyn. Syst., Meas., Control*, **137**(5), p. 051006.
- [22] Ghasemi, A. H., Hoagg, J. B., and Seigler, T. M., 2016, "Decentralized Vibration and Shape Control of Structures With Collocated Sensors and Actuators," *ASME J. Dyn. Syst., Meas., Control*, **138**(3), p. 031011.
- [23] Yeatman, M., Lv, G., and Gregg, R. D., 2019, "Decentralized Passivity-Based Control With a Generalized Energy Storage Function for Robust Biped Locomotion," *ASME J. Dyn. Syst., Meas., Control*, **141**(10), p. 101007.
- [24] Wang, C., and Li, D., 2011, "Decentralized PID Controllers Based on Probabilistic Robustness," *ASME J. Dyn. Syst., Meas., Control*, **133**(6), p. 061015.
- [25] Kalat, S. T., Faal, S. G., and Onal, C. D., 2018, "A Decentralized, Communication-Free Force Distribution Method With Application to Collective Object Manipulation," *ASME J. Dyn. Syst., Meas., Control*, **140**(9), p. 091012.
- [26] Kan, Z., Klotz, J. R., Shea, J. M., Doucette, E. A., and Dixon, W. E., 2017, "Decentralized Rendezvous of Nonholonomic Robots With Sensing and Connectivity Constraints," *ASME J. Dyn. Syst., Meas., Control*, **139**(2), p. 024501.
- [27] Pagilla, P. R., and Zhu, Y., 2005, "A Decentralized Output Feedback Controller for a Class of Large-Scale Interconnected Nonlinear Systems," *ASME J. Dyn. Syst., Meas., Control*, **127**(1), pp. 167–172.
- [28] Pavone, M., and Frazzoli, E., 2007, "Decentralized Policies for Geometric Pattern Formation and Path Coverage," *ASME J. Dyn. Syst., Meas., Control*, **129**(5), pp. 633–643.
- [29] Brahmi, B., Brahmi, A., Saad, M., Gauthier, G., and Rahman, M. H., 2019, "Robust Adaptive Tracking Control of Uncertain Rehabilitation Exoskeleton Robot," *ASME J. Dyn. Syst., Meas., Control*, **141**(12), p. 121007.

- [30] Al Younes, Y., Drak, A., Noura, H., Rabhi, A., and El Hajjaji, A., 2016, "Robust Model-Free Control Applied to a Quadrotor Uav," *J. Intell. Rob. Syst.*, **84**(1–4), pp. 37–52.
- [31] Villagra, J., Join, C., Haber, R., and Fliess, M., 2020, "Model-Free Control for Machine Tools," preprint [arXiv:2005.08546](https://arxiv.org/abs/2005.08546).
- [32] Han, D. K., and Chang, P.-H., 2010, "Robust Tracking of Robot Manipulator With Nonlinear Friction Using Time Delay Control With Gradient Estimator," *J. Mech. Sci. Technol.*, **24**(8), pp. 1743–1752.
- [33] Denavit, J., and Hartenberg, R. S., 1955, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," *ASME J. Appl. Mech.*, **77**, pp. 215–221.
- [34] Bagheri, M., Karafyllis, I., Naseradinmousavi, P., and Krstic, M., 2021, "Adaptive Control of a Two-Link Robot Using Batch Least-Square Identifier," *IEEE/CAA J. Automatica Sin.*, **8**(1), pp. 86–93.
- [35] Bagheri, M., and Naseradinmousavi, P., 2017, "Novel Analytical and Experimental Trajectory Optimization of a 7DOF Baxter Robot: Global Design Sensitivity and Step Size Analyses," *Int. J. Adv. Manuf. Technol.*, **93**(9–12), pp. 4153–4167.
- [36] Bagheri, M., Naseradinmousavi, P., and Morsi, R., 2017, "Experimental and Novel Analytical Trajectory Optimization of a 7DOF Baxter Robot: Global Design Sensitivity and Step Size Analyses," *ASME* Paper No. DSCC2017-5004.
- [37] Bagheri, M., Krstić, M., and Naseradinmousavi, P., 2018, "Joint-Space Trajectory Optimization of a 7-Dof Baxter Using Multivariable Extremum Seeking," *Annual American Control Conference (ACC)*, Milwaukee, WI, June 27–29, pp. 2176–2181.
- [38] Bagheri, M., Krstić, M., and Naseradinmousavi, P., 2018, "Multivariable Extremum Seeking for Joint-Space Trajectory Optimization of a High-Degrees-of-Freedom Robot," *ASME J. Dyn. Syst., Meas., Control*, **140**(11), p. 111017.
- [39] Bagheri, M., Krstić, M., and Naseradinmousavi, P., 2018, "Analytical and Experimental Predictor-Based Time Delay Control of Baxter Robot," *ASME* Paper No. DSCC2018-9101.
- [40] Bagheri, M., Naseradinmousavi, P., and Krstić, M., 2019, "Feedback Linearization Based Predictor for Time Delay Control of a High-Dof Robot Manipulator," *Automatica*, **108**, p. 108485.
- [41] Bertino, A., Bagheri, M., Krstić, M., and Naseradinmousavi, P., 2019, "Experimental Autonomous Deep Learning-Based 3D Path Planning for a 7-DOF Robot Manipulator," *ASME* Paper No. DSCC2019-8951.
- [42] Bagheri, M., Naseradinmousavi, P., and Krstić, M., 2019, "Time Delay Control of a High-DOF Robot Manipulator Through Feedback Linearization Based Predictor," *ASME* Paper No. DSCC2019-8915.
- [43] Khalil, H. K., 2015, *Nonlinear Control*, 3rd ed., Prentice Hall, Michigan State University, East Lansing, MI.