

Delay-Adaptive Control of a 7-DOF Robot Manipulator: Design and Experiments

Alexander Bertino[✉], Peiman Naseradinmousavi[✉], and Miroslav Krstić[✉], *Fellow, IEEE*

Abstract—We present an analytical design and experimental verification of trajectory-tracking control of a 7-DOF robot manipulator with an unknown long actuator delay. To compensate for this unknown delay, we formulate a delay-adaptive prediction-based control strategy to simultaneously estimate the unknown delay while driving the robot manipulator toward the desired trajectory. To the best of the authors' knowledge, this article is the first to present a delay-adaptive approach for a nonlinear system with multiple inputs. Through Lyapunov analysis, we first establish local input-to-state stability with respect to temporal derivatives of the reference trajectory, along with regulation of the tracking errors when the reference trajectory approaches a stationary configuration. Then, through both simulation and experiment, we demonstrate that the proposed controller is capable of tracking the desired trajectory with desirable performance despite a large initial delay mismatch, which would cause nonadaptive prediction-based controllers to become unstable.

Index Terms—Delay-adaptive control, experimental validation, high-DOF robots, unknown delay.

I. INTRODUCTION

IN THIS article, we pursue an experimental verification of an analytically designed control of a 7-DOF robot manipulator subjected to an unknown constant input delay. Such delays are frequently observed in the control of remote manipulators [1], [6], [17], [27], [30], where a long, slowly time-varying (often assumed to be constant) communication delay is likely present. To account for a known delay, a variety of predictor-based and sliding mode approaches have been developed for linear systems [2], [3], [14], [15], [20]–[22], [24], [25], [35], nonlinear systems [4], [5], [18], [23], [31], and systems with a time-varying delay [7], [8], [26], [33], [34]. Although such techniques notably improve the transient performance of a controller in the presence of a known delay,

they are also well-known to be sensitive to delay mismatch. In the case where a long, slowly time-varying communication delay is difficult to accurately predict or measure, a delay-adaptive control approach has the potential to significantly increase the transient performance of the robot manipulator, through compensation of the delay mismatch.

In recent articles [10]–[13], [19], [36]–[39], adaptive control strategies were developed to estimate an unknown delay while simultaneously compensating for this delay with a predictor-based approach. To achieve this, most of these articles represent the constant delay at the input of an ordinary differential equation (ODE) as a transport partial differential equation (PDE) whose convective speed is inversely proportional to the unknown delay. This approach introduces the delay parameter into the model in a linear manner and is, therefore, suitable for adaptive design. In the article by Bresch-Pietri and Krstić [13], this strategy was extended to nonlinear dynamics subjected to a constant input delay. The authors examined the case of a measured distributed input, in which a global delay-adaptive stability result is achieved, as well as the more realistic case of an unmeasured distributed input, in which a local delay-adaptive stability result is achieved.

In this effort, we formulate the local technique developed by Bresch-Pietri and Krstić [13] for an unmeasured distributed input, to handle the case of trajectory tracking with multiple actuators. This formulation yields local input-to-state stability with respect to temporal derivatives of the reference trajectory, as well as regulation of tracking errors when the reference trajectory approaches a stationary configuration. Furthermore, through the experimental verification of this delay-adaptive control strategy on Baxter, a 7-DOF redundant robot manipulator, we demonstrate desirable controller performance even in the presence of a significant delay mismatch. Two cases are studied here, an underestimation of 0.9 s (0-s initial prediction, 0.9-s actual delay) and an overestimation of 0.5 s (0.9-s initial prediction, 0.4-s actual delay). Thus, the delay-adaptive control strategy is both theoretically sound and effective in practice, significantly improving the tracking performance of the predictor-based approach when the delay is unknown. This is the main contribution of this article.

The organization of this article is as follows. In Section II, we present a brief overview of the dynamics of Baxter's right manipulator. In Section III, we formulate the delay adaptation task in mathematical terms, as well as state several assumptions on the system dynamics, feedback law, and

Manuscript received 14 November 2021; revised 15 November 2021; accepted 14 February 2022. Date of publication 4 March 2022; date of current version 21 October 2022. This work was supported by the National Science Foundation under Award #1823951-1823983. Recommended by Associate Editor A. Macchelli. (Corresponding author: Peiman Naseradinmousavi.)

Alexander Bertino is with the Department of Mechanical Engineering, San Diego State University, San Diego, CA 92115 USA, and also with the Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093 USA (e-mail: abertino6245@sdsu.edu).

Peiman Naseradinmousavi is with the Department of Mechanical Engineering, San Diego State University, San Diego, CA 92115 USA (e-mail: pnaseradinmousavi@sdsu.edu).

Miroslav Krstić is with the Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093 USA (e-mail: krstic@ucsd.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TCST.2022.3152363>.

Digital Object Identifier 10.1109/TCST.2022.3152363

1063-6536 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See <https://www.ieee.org/publications/rights/index.html> for more information.

desired trajectories that are used in the Lyapunov analysis of the delay-adaptive method. In Sections IV–VI, we present the delay adaptation approach and demonstrate the local delay-adaptive stability of the method through Lyapunov analysis using the $\mathcal{L}_1$ norm. In Section VII, we present the simulation and experimental results of the proposed method implemented on Baxter's right manipulator, accounting for a large delay mismatch both in the case of delay underestimation and overestimation. Finally, in Section VIII, we present the case that the proposed delay-adaptive method has the potential to significantly increase the transient performance of a robot manipulator subjected to an unknown delay, through the compensation of an initial delay mismatch.

Notations: In the following, we use the common definitions of class $\mathcal{K}$ and $\mathcal{K}_\infty$ given in [16]. $|\cdot|$ and $|\cdot|_1$ refer to the Euclidean and $\mathcal{L}_1$ norms, respectively, and the matrix norm is defined accordingly; for $M \in \mathcal{M}_\ell(\mathbb{R})$ ($\ell \in \mathbb{N}^*$), as $|M| = \sup_{|x| \leq 1} |Mx|$ and the spatial $\mathcal{L}_1$ norm is defined as follows:

$$\|u(t)\|_1 = \int_0^1 |u(x, t)|_1 dx.$$

For $(a, b) \in \mathbb{R}^2$ such that $a < b$, we define the standard projection operator on the interval $[a, b]$ as a function of two scalar arguments f (denoting the parameter being updated) and g (denoting the nominal update law) in the following manner:

$$\text{Proj}_{[a,b]}(f, g) = g \begin{cases} 0, & \text{if } f = a \text{ and } g < 0 \\ 0, & \text{if } f = b \text{ and } g > 0 \\ 1, & \text{otherwise.} \end{cases}$$

For a distributed function of (x, t) or (y, t) , a lowercase subscript indicates differentiation by the corresponding parameter. For example

$$f_{xt}(x, t) = \frac{\partial^2 f(x, t)}{\partial x \partial t}.$$

II. MATHEMATICAL MODELING

The redundant manipulator, which is being studied here, has a 7-DOF as shown in Fig. 1. The Euler–Lagrange formulation leads to a set of seven coupled nonlinear second-order ODEs

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

where

$$N(q, \dot{q}) = C(q, \dot{q})\dot{q} + G(q) \quad (2)$$

in which $q, \dot{q}$, and $\ddot{q} \in \mathbb{R}^7$ 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, and $G(q) \in \mathbb{R}^7$ is a vector of gravitational loading.

Our verified coupled nonlinear dynamic model of the robot [4], [5], [9] is used as the basis of the delay-adaptive approach. Note that the inertia matrix $M(q)$ is symmetric, positive definite, and consequently invertible. This property is

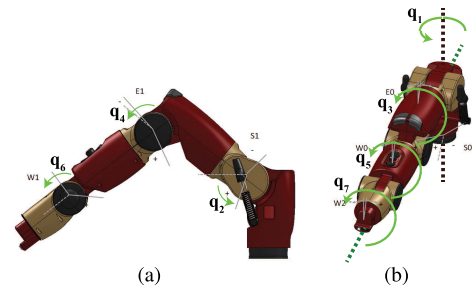


Fig. 1. Joints' configuration: (a) sagittal view and (b) top view.

used in the subsequent development. The multiinput nonlinear system (1) can be written as the 14th-order system of ODEs with the following general state-space form:

$$\begin{aligned} \dot{X} &= f_0(X, U) \\ &= \begin{bmatrix} -M(q)^{-1}N(q, \dot{q}) \end{bmatrix} + \begin{bmatrix} 0 \\ M(q)^{-1} \end{bmatrix} U \end{aligned} \quad (3)$$

$$X = [q_1, \dots, q_7, \dot{q}_1, \dots, \dot{q}_7]^T \quad (4)$$

where $X \in \mathbb{R}^{14}$ is the 14-D vector of states, and $U = \tau \in \mathbb{R}^7$ represents the input torques to the system (3).

To track a desired trajectory, we reformulate (3) in terms of error dynamics

$$\begin{aligned} \dot{Z} &= f(Z, U, X_R) \\ &= \begin{bmatrix} -M(z + q_R)^{-1}N(z + q_R, \dot{z} + \dot{q}_R) - \ddot{q}_R \end{bmatrix} \\ &\quad + \begin{bmatrix} 0 \\ M(z + q_R)^{-1} \end{bmatrix} U \end{aligned} \quad (5)$$

$$X_R = [q_R^T, \dot{q}_R^T, \ddot{q}_R^T]^T \quad (6)$$

$$Z = [z^T, \dot{z}^T]^T \quad (7)$$

$$z = q - q_R \quad (8)$$

where $X_R \in \mathbb{R}^{21}$ is the state reference trajectory, $Z \in \mathbb{R}^{14}$ is the state error vector, $z \in \mathbb{R}^7$ is the positional error of the robot manipulator, and $q_R \in \mathbb{R}^7$ are the reference joint trajectories to track.

Furthermore, we make the following assumption regarding the reference joint trajectories:

Assumption 1: The desired joint trajectories $q_R(t) \in \mathbb{R}^7$ are class $\mathcal{C}^5$ functions, and $X_R(t)$, $\dot{X}_R(t)$, and $\ddot{X}_R(t)$ are uniformly bounded for all $t \geq 0$.

III. PROBLEM STATEMENT

Consider the following nonlinear plant:

$$\dot{Z}(t) = f(Z(t), U(t - D), X_R(t)) \quad (9)$$

in which D is an unknown delay introduced to the error dynamic model of the Baxter manipulator (5), belonging to the interval $[\underline{D}, \overline{D}]$, with $\underline{D} > 0$. The objective of the delay-adaptive approach is to stabilize the error dynamics with input delay (9), despite the length of the delay being initially unknown. To assist the Lyapunov stability analysis in Section IV, the following assumptions are made regarding the

nonlinear plant (9), the corresponding feedback law, and the state reference trajectory $X_R(t)$.

Assumption 2: The plant (5) is forward complete.

Assumption 3: There exists a C^2 feedback law $U = \kappa(Z, X_R)$ such that the closed-loop delay-free plant (5) is globally exponentially stable, i.e., there exist $\lambda > 0$ and a class C^1 radially unbounded positive definite function V such that for all $Z \in \mathbb{R}^{14}$

$$\frac{\partial V}{\partial Z}(Z) f(Z, \kappa(Z, X_R), X_R) \leq -\lambda V(Z) \quad (10)$$

$$|Z|^2 \leq V(Z) \leq c_1 |Z|^2 \quad (11)$$

$$\left| \frac{\partial V}{\partial Z}(Z) \right| \leq c_2 |Z|. \quad (12)$$

Assumption 4: The values for the state reference trajectory $X_R(t)$ are known at least $\bar{D}$ seconds in advance.

Assumption 2 assures that (9) does not escape in finite time. This assumption is necessary to ensure that the system does not escape before the input $U(t - D)$ reaches the system, and it has been proven to hold for robot manipulators [4], [5]. Assumption 3 is a stronger than necessary condition used to prove the local stability of the delay-adaptive approach. Assumption 4 ensures that the state error vector $Z(t)$ can be predicted up to $\bar{D}$ seconds in advance. This is a necessary assumption for any predictor-based control strategy involving trajectory tracking since if $Z(t + D)$ cannot be predicted, $U(t)$ cannot be chosen to compensate for the delay present in the system. While such a control strategy is technically noncausal with respect to the state reference $X_R(t)$, this is not a concern in practice since $X_R(t)$ is a user-defined signal that is independent of the current joint state $X(t)$ and input $U(t)$, and thus can be determined an arbitrary time in advance.

To analyze the closed-loop stability despite delay uncertainties, we use the systematic Lyapunov tools introduced in [18] and first reformulate plant (9) in the form

$$\dot{Z}(t) = f(Z(t), u(0, t), X_R(t)) \quad (13)$$

$$Du_t(x, t) = u_x(x, t) \quad (14)$$

$$u(1, t) = U(t) \quad (15)$$

by introducing the following distributed input:

$$u(x, t) = U(t + D(x - 1)), \quad x \in [0, 1]. \quad (16)$$

Thus, the input delay is now represented as a coupling with a transport PDE driven by an input with unknown convection speed $1/D$.

IV. DELAY-ADAPTIVE CONTROL DESIGN

Due to the fact that the distributed input is unmeasured, we introduce an estimate of the distributed input

$$\hat{u}(x, t) = U(t + \hat{D}(t)(x - 1)), \quad x \in [0, 1] \quad (17)$$

where $\hat{D}(t)$ is the current estimate of the input delay. To stabilize (13)–(15), we must first predict the state of the system (13)–(15) once the delayed input reaches the system. To achieve this, we introduce a distributed predictor estimate

$$\hat{p}(x, t) = Z(t + \hat{D}(t)x)$$

$$= Z(t) + \hat{D}(t) \int_0^x f(\hat{p}(y, t), \hat{u}(y, t), \hat{r}(y, t)) dy \quad (18)$$

in which

$$\hat{r}(x, t) = X_R(t + \hat{D}(t)x) \quad (19)$$

is the distributed trajectory estimate. If the input delay was known, the control law $U(t) = \kappa(Z(t + D), X_R(t + D))$ could be used to stabilize the system, exactly compensating for the delay present in the system. Therefore, by the certainty equivalence principle, we choose the control law as

$$\begin{aligned} U(t) &= \kappa(Z(t + \hat{D}(t)), X_R(t + \hat{D}(t))) \\ &= \kappa(\hat{p}(1, t), \hat{r}(1, t)). \end{aligned} \quad (20)$$

To derive an adaptation update law for the estimated delay, we define and use at time t a prediction of the current system state $X(t)$, starting from a recent previous state $X(t - \beta)$ with $\beta > 0$, and assuming the correct value of the input delay is $\hat{D}(t)$

$$\begin{aligned} X_P(x, t, \hat{D}) &= [q_P(x, t, \hat{D})^T, \dot{q}_P(x, t, \hat{D})^T]^T \\ &= X(t - \beta) + \beta \int_0^x f_0(X_P, U_P) dy \end{aligned} \quad (21)$$

where

$$U_P(x, t, \hat{D}) = U(t - \hat{D} + \beta(x - 1)), \quad x \in [0, 1]. \quad (22)$$

Note that in this section, notations indicating nested functions of $(y, t, \hat{D})$ have been removed for the sake of brevity. An important property of this prediction is the following:

$$X_P(x, t, D) = X(t + \beta(x - 1)) \quad (23)$$

$$X_P(1, t, D) = X(t) \quad (24)$$

and thus if the estimated value of the delay $\hat{D}(t)$ equals the true value of the delay D , then our prediction of the current system state $X_P(1, t, \hat{D})$ is equivalent to the current system state $X(t)$. Leveraging this useful property, we can use the mean square error between the predicted system state $X_P(1, t, \hat{D})$ and the current system state $X(t)$ through a gradient descent algorithm, updating the estimate of the delay $\hat{D}(t)$ to minimize this prediction error. For this purpose, we use the following instantaneous cost function, initially proposed in [10] for a linear plant:

$$J : (t, \hat{D}) \in \left[t_0, \infty \left[\rightarrow \frac{1}{2} |X_P(1, t, \hat{D}) - X(t)|^2 \right] \right]. \quad (25)$$

To obtain the gradient of this cost function with respect to the estimated delay $\hat{D}$, it is first necessary to determine the partial derivative of X_P with respect to $\hat{D}$

$$\begin{aligned} \frac{\partial X_P}{\partial \hat{D}}(x, t, \hat{D}) &= \beta \int_0^x \left(\frac{\partial f_0}{\partial X_P}(X_P, U_P) \frac{\partial X_P}{\partial \hat{D}}(y, t, \hat{D}) \right. \\ &\quad \left. + \frac{\partial f_0}{\partial U_P}(X_P, U_P) \frac{\partial U_P}{\partial \hat{D}}(y, t, \hat{D}) \right) dy \\ &= \beta \int_0^x \left(\frac{\partial f_0}{\partial X_P}(X_P, U_P) \frac{\partial X_P}{\partial \hat{D}}(y, t, \hat{D}) \right. \\ &\quad \left. - \begin{bmatrix} 0 \\ M(q_P)^{-1} \end{bmatrix} \dot{U}_P(y, t, \hat{D}) \right) dy. \end{aligned} \quad (26)$$

By taking the derivative of this equation with respect to x , it can be seen that $(\partial^2 X_P / \partial \hat{D} \partial x)$ satisfies the following ODE:

$$\frac{\partial^2 X_P}{\partial \hat{D} \partial x}(x, t, \hat{D}) = \beta \left(\frac{\partial f_0}{\partial X_P}(X_P, U_P) \frac{\partial X_P}{\partial \hat{D}}(x, t, \hat{D}) - \left[M(q_P)^{-1} \right] \dot{U}_P(x, t, \hat{D}) \right) \quad (27)$$

$$\frac{\partial X_P}{\partial \hat{D}}(0, t, \hat{D}) = 0. \quad (28)$$

By defining the transition matrix Φ_0 associated with the corresponding homogeneous equation, one solves (27) and (28) to obtain

$$\frac{\partial X_P}{\partial \hat{D}}(x, t, \hat{D}) = -\beta \int_0^x \Phi_0(x, y, t, \hat{D}) \times \left[M(q_P)^{-1} \right] \dot{U}_P(y, t, \hat{D}) dy \quad (29)$$

where $\Phi_0(x, y, t, \hat{D})$ is the solution to the following homogeneous ODE:

$$\frac{\partial \Phi_0}{\partial x}(x, y, t, \hat{D}) = \beta \frac{\partial f_0}{\partial X_P}(X_P, U_P) \Phi_0(x, y, t, \hat{D}) \quad (30)$$

$$\Phi_0(y, y, t, \hat{D}) = I, \quad y \in [0, 1], \quad x \in [y, 1]. \quad (31)$$

Taking the gradient of the instantaneous cost function (25), the following equation is obtained:

$$\frac{\partial J}{\partial \hat{D}}(t, \hat{D}) = \left(X_P(1, t, \hat{D}) - X(t) \right)^T \frac{\partial X_P}{\partial \hat{D}}(1, t, \hat{D}). \quad (32)$$

Using this computed gradient, the rate of change of the delay estimate is designed as

$$\dot{\hat{D}}(t) = \gamma \text{Proj}_{[\underline{D}, \overline{D}]} \{ \hat{D}(t), \rho_D(t) \} \quad (33)$$

where

$$\rho_D(t) = \frac{-\frac{\partial J}{\partial \hat{D}}(t, \hat{D}(t))}{1 + \left| \frac{\partial X_P}{\partial \hat{D}}(1, t, \hat{D}(t)) \right|^2} \quad (34)$$

and $\gamma > 0$ is the adaptation rate of the delay estimate. The projection operator is used to ensure that the delay estimate remains in the interval $[\underline{D}, \overline{D}]$. Note that normalization by the regressor is used in the adaptation of the delay estimate (34) to reduce the speed of adaptation when there are large changes in the input. Using a steepest descent argument [29] along with an appropriate bounding of terms, one obtains the following properties of (34), provided that the initial delay estimate is close enough to the true value of the delay:

Lemma 1: There exist positive parameters $H > 0$ and $\tilde{D}_{\max} > 0$ such that if $|\tilde{D}(0)| < \tilde{D}_{\max}$, and $X(t)$, $U(t)$, and $\dot{U}(t)$ are uniformly bounded

$$\tilde{D}(t) \rho_D(t) \geq 0 \quad (35)$$

$$|\rho_D(t)| \leq H \quad (36)$$

where $\tilde{D}(t) = D - \hat{D}(t)$ is the current estimation error of the delay.

Proof: Using a steepest descent argument [29], one obtains the property $\tilde{D}(t) \rho_D(t) \geq 0$. To obtain the property $|\rho_D(t)| \leq H$, it is necessary to establish the uniform boundedness of X_P and $(\partial X_P / \partial \hat{D})$ in (34). Due to the forward

completeness property of f_0 , the uniform boundedness of X and U , and the fixed integration distance in (21), one obtains the uniform boundedness of X_P . To establish a bound for $(\partial X_P / \partial \hat{D})$, Lemma 3 is used to prove the uniform boundedness of Φ_0 . Due to the uniform boundedness of Φ_0 , X_P , U , and $\dot{U}$, along with the fact that f_0 is a class C^2 function, one obtains the uniform boundedness of $(\partial X_P / \partial \hat{D})$ from (29). Applying the uniform boundedness of X_P and $(\partial X_P / \partial \hat{D})$ to (34), one obtains the property $|\rho_D(t)| \leq H$. ■

To predict the input delay of the robot manipulator, we use a gradient-based method minimizing the difference between the current system state and a prediction of what the system state should be if the currently estimated delay is equal to the true value of the delay. It is important to note that contrary to the implementation of the current state predictor X_P in previous works by Bresch-Pietri *et al.* [10], [13], in which the current state is estimated by simulating from the initial condition $X(t_0)$, the predictor method present in this article estimates the current state by simulating from the more recent state $X(t - \beta)$. This modification to the predictor-based update law ensures that the computational cost of performing this method remains consistent due to the fixed bounds of the integral in (21), as well as serving to bound the maximum growth of the state transition matrix $\Phi_0(x, y, t, \hat{D})$ and prediction error $X_P(1, t, \hat{D}) - X(t)$. In essence, this cost function is a simulated replay (21) of the last β seconds of the robot manipulator motion, and the update of the delay estimate aims to match this simulated replay to the observed value of the system state during this period. If the estimate of the delay is correct, the simulated replay should match perfectly with the motion of the robot manipulator during the most recent β seconds of motion.

Due to the fact that the properties stated in Lemma 1 hold for any $\beta > 0$, the selection of the replay length β is motivated primarily by practical considerations. If β is chosen to be too small, measurement noise and external disturbances will make up a large portion of the difference between the predicted and actual current system state, meaning that the update of the estimated delay will be susceptible to high-frequency noise. If β is too large, the accuracy of the simulated replay will suffer due to accumulated inaccuracies, such as small inconsistencies in the robot manipulator model, as well as accumulated error from the numerical methods used in the simulation. Additionally, a larger β increases the computational cost of the method, as a longer simulation will need to be performed. Thus, the selection of β is a balance between susceptibility to high-frequency noise with a small β and susceptibility to low-frequency noise and increased computational burden with a large β . Through simulations and experiments, we have found that selecting β to be roughly an order of magnitude smaller than the settling time of the control law $\kappa(Z, X_R)$ is a suitable choice.

Using the delay estimate update law (33), as well as the control law (20), we are now ready to present the stability theorem for the delay-adaptive controller operating on a robot manipulator with unknown input delay.

Theorem 1: Consider the closed-loop system consisting of the error dynamics of the robot manipulator (5), control

law (20), delay estimate update law (33), and desired joint trajectories $q_R(t)$ satisfying Assumptions 1–4. Define the following functionals:

$$\begin{aligned}\Gamma(t) &= |Z(t)| + \|\tilde{u}(t)\|_1 \\ &\quad + \int_0^1 |\hat{u}(x, t) - \hat{u}_R(x, t)|_1 dx \\ &\quad + \int_0^1 |\hat{u}_x(x, t) - \hat{u}_{R,x}(x, t)|_1 dx \quad (37)\end{aligned}$$

$$\Gamma_R(t) = \|\hat{x}(1, t)\|_1 + \|\hat{x}(t)\|_1 + \|\hat{x}_x(t)\|_1 \quad (38)$$

in which $\hat{x}$ is defined in (19),

$$\hat{u}_R(x, t) = \kappa(\hat{p}(x, t), \hat{r}(x, t)) \quad (39)$$

is the predicted distributed input reference, and

$$\tilde{u}(x, t) = u(x, t) - \hat{u}(x, t) \quad (40)$$

is the distributed input estimation error. Then, there exist R^* , Γ^* , δ^* , γ^* , c_3 , $c_4 > 0$, and a class $\mathcal{K}_\infty$ function α^* such that if $X_R(t)$, $\dot{X}_R(t)$, and $\ddot{X}_R(t)$ are uniformly bounded by R^* , $\Gamma(0) \leq \Gamma^*$, $|\dot{D}(0)| < \delta^*$, and $\gamma < \gamma^*$, then

$$\Gamma(t) \leq c_3 \Gamma(0) e^{-c_4 t} + \alpha^* \left(\sup_{s \in [0, t]} \{\Gamma_R(s)\} \right) \quad \forall t \geq 0 \quad (41)$$

Furthermore, if $\dot{X}_R(t) \rightarrow 0$ as $t \rightarrow \infty$, then

$$Z(t) \xrightarrow[t \rightarrow \infty]{} 0. \quad (42)$$

V. LYAPUNOV ANALYSIS

In this section, the proof of Theorem 1 resembles the proof of Theorem 3 in [13], since the delay-adaptive control approach in our article is an extension of their approach. However, significant changes to this proof were necessary to adapt it to a trajectory tracking task using a robot manipulator. In the original backstepping transformation used by Bresch-Pietri and Krstic in [13], the nonlinear plant f and the corresponding control law κ are assumed to be autonomous. As both f and κ are nonautonomous when tracking a time-varying reference trajectory $X_R(t)$, it was necessary to reformulate the backstepping transformation to handle the nonautonomous case. This reformulation of the backstepping transformation introduces new terms in Lyapunov analysis, which need to be carefully bounded. To bound these terms, it is necessary to do so in terms of the current tracking errors, the distributed input torques, and the temporal derivatives of the reference trajectory. As a result of terms being bound by the temporal derivatives of the reference trajectory, local input-to-state stability is established with respect to these derivatives, with regulation of the tracking errors when the reference trajectory approaches a stationary configuration. Thus, the analysis presented here extends upon the stability results of [13] when there is a time-varying reference to track, while preserving the original local asymptotic stability result of [13] when the reference is stationary. Additionally, due to the control affine nature of robot manipulators, the application of the mean value theorem is not necessary to bound terms such as $f_{\tilde{u}}$ appearing in this section. This key property of robot manipulators allows

for a large reduction in the number of terms that are necessary for the Lyapunov analysis presented in this section when compared with that of [13].

It is important to note that both the original method presented in [13] and the method we present here use the $\mathcal{L}_1$ norm rather than the $\mathcal{L}_2$ norm as is typical when performing Lyapunov analysis. For an additional reference in the use of $\mathcal{L}_1$ norms in Lyapunov analysis, the interested reader can refer to [32].

To perform Lyapunov stability analysis, a backstepping transformation is first used to reformulate (13)–(15).

Lemma 2: The backstepping transformation of the distributed input estimates (17)

$$\hat{w}(x, t) = \hat{u}(x, t) - \kappa(\hat{p}(x, t), \hat{r}(x, t)) \quad (43)$$

in which the distributed predictor estimate is defined in (18), and together with the control law (20), it transforms plants (13)–(15) into

$$\begin{aligned}\dot{Z}(t) &= f(Z(t), \kappa(Z(t), \hat{r}(0, t))) \\ &\quad + \hat{w}(0, t) + \tilde{u}(0, t), \hat{r}(0, t)) \quad (44)\end{aligned}$$

$$D\tilde{u}_t(x, t) = \tilde{u}_x(x, t) - \dot{D}(t)g_1(x, t) - \dot{D}(t)g_2(x, t) \quad (45)$$

$$\tilde{u}(1, t) = 0 \quad (46)$$

$$\dot{D}(t)\hat{w}_t(x, t) = \hat{w}_x(x, t) + \dot{D}(t)h_1(x, t) - h_2(x, t)f_{\tilde{u}}(t) \quad (47)$$

$$\hat{w}(1, t) = 0. \quad (48)$$

Additionally, the spatial derivative of the backstepping transformation (43) satisfies the following PDE:

$$\begin{aligned}\dot{D}(t)\hat{w}_{xt}(x, t) &= \hat{w}_{xx}(x, t) + \dot{D}(t)h_3(x, t) - h_4(x, t)f_{\tilde{u}}(t) \\ &\quad (49)\end{aligned}$$

$$\hat{w}_x(1, t) = -\dot{D}(t)h_1(1, t) + h_2(1, t)f_{\tilde{u}}(t). \quad (50)$$

The expressions for the terms g_1 , g_2 , h_1 , h_2 , h_3 , h_4 , and $f_{\tilde{u}}$ can be found in Appendix A.

Proof: Note that in this proof, notations indicating distributed functions of (x, t) and (y, t) have been removed for the sake of brevity.

First, (44) is obtained by applying (43) and (40) to that of the nonlinear plants (13)–(15). Next, by examining the spatial and temporal derivatives of the distributed input estimate (17) and distributed trajectory estimate (19), the following relationships are obtained:

$$\dot{D}(t)\hat{u}_t = \hat{u}_x + \dot{D}(t)(x-1)\hat{u}_x \quad (51)$$

$$\hat{u}(1, t) = U(t) \quad (52)$$

$$\dot{D}(t)\hat{r}_t = \hat{r}_x + \dot{D}(t)x\hat{r}_x \quad (53)$$

$$\hat{r}(0, t) = X_R(t). \quad (54)$$

By combining (51) with that of the relationship between the spatial and temporal derivatives of the inputs (14) and (15), and applying backstepping transformation, (43), (45), and (46) are obtained.

To obtain the governing equation for $\hat{w}$, we first substitute (43) and (53) into (51) to obtain the following expression:

$$\begin{aligned}\dot{D}(t)\hat{w}_t &= \hat{w}_x - \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r})(\dot{D}(t)\hat{p}_t - \hat{p}_x) \\ &\quad + \dot{D}(t)(x-1)\hat{u}_x - \dot{D}(t)x \frac{\partial \kappa}{\partial \hat{r}}(\hat{p}, \hat{r})\hat{r}_x. \quad (55)\end{aligned}$$

To obtain (47) and (48) from (55), it is necessary to study the behavior of the temporal and spatial derivatives of the distributed predictor. By taking the temporal and spatial derivatives of $\hat{p}$ and substituting in (51) and (53), the following relationships are obtained:

$$\begin{aligned}\hat{p}_t &= f(\hat{p}(0, t), u(0, t), \hat{r}(0, t)) \\ &+ \int_0^x \left(\frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \hat{D}(t) \hat{p}_t \right. \\ &\quad + \frac{\partial f}{\partial \hat{u}}(\hat{p}, \hat{u}, \hat{r}) [\hat{u}_y + \dot{\hat{D}}(t)(y-1)\hat{u}_y] \\ &\quad + \frac{\partial f}{\partial \hat{r}}[\hat{r}_y + \dot{\hat{D}}(t)y\hat{r}_y] \Big) dy + \dot{\hat{D}}(t) \\ &\quad \times \int_0^x f(\hat{p}, \hat{u}, \hat{r}) dy\end{aligned}\quad (56)$$

$$\begin{aligned}\hat{p}_x &= \hat{D}(t) f(\hat{p}, \hat{u}, \hat{r}) \\ &= \hat{D}(t) f(\hat{p}(0, t), \hat{u}(0, t), \hat{r}(0, t)) \\ &\quad + \hat{D}(t) \int_0^x \left(\frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \hat{p}_y + \frac{\partial f}{\partial \hat{u}}(\hat{p}, \hat{u}, \hat{r}) \hat{u}_y \right. \\ &\quad \left. + \frac{\partial f}{\partial \hat{r}}(\hat{p}, \hat{u}, \hat{r}) \hat{r}_y \right) dy.\end{aligned}\quad (57)$$

Then, we define $\phi = \hat{D}(t)\hat{p}_t - \hat{p}_x$. It is observed from the substitution of (56) and (57) into this definition that ϕ satisfies the following equation in x , parametrized in t :

$$\frac{\partial \phi}{\partial x} = \hat{D}(t) \frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \phi + \dot{\hat{D}}(t) \hat{D}(t) \psi \quad (58)$$

$$\phi(0, t) = \hat{D}(t) f_{\hat{u}}(t) \quad (59)$$

where $f_{\hat{u}}(t)$ and ψ are defined in (A.7) and (A.8), respectively.

By defining the transition matrix Φ associated with the corresponding homogeneous equation, one solves this equation to obtain

$$\begin{aligned}\hat{D}(t)\hat{p}_t - \hat{p}_x &= \dot{\hat{D}}(t)\hat{D}(t) \int_0^x \Phi(x, y, t) \psi dy \\ &\quad + \Phi(x, 0, t) \hat{D}(t) f_{\hat{u}}(t).\end{aligned}\quad (60)$$

Substituting this equation along with the backstepping transformation (43) into (55) yields (47) and (48).

By taking the spatial derivative of (47), one obtains the governing equation (49). Additionally, the boundary condition (50) is directly obtained by rearranging the governing equation (47), along with the knowledge that $\hat{w}_t(1, t) = 0$ due to taking the time derivative of the boundary condition (48). ■

For the purpose of Lyapunov analysis, we consider the following Lyapunov–Krasovskii functional candidate:

$$\begin{aligned}W(t) &= V_0(Z(t)) + b_0 D \int_0^1 (1+x) |\tilde{u}(x, t)|_1 dx \\ &\quad + b_1 \hat{D}(t) \int_0^1 (1+x) |\hat{w}(x, t)|_1 dx \\ &\quad + b_2 \hat{D}(t) \int_0^1 (1+x) |\hat{w}_x(x, t)|_1 dx\end{aligned}\quad (61)$$

in which $V_0 = \sqrt{V}$, which was previously defined in Assumption 3. This functional measures the current tracking

errors of the robot manipulator, as well as the difference between the distributed input and the desired distributed input. Using the properties of Assumption 3, along with the state space model of the robot manipulator (5), the following inequality is obtained:

$$\begin{aligned}\dot{V}_0(t) &\leq -\frac{\lambda}{2} |Z(t)| + \frac{c_2}{2} |M(q)^{-1}| |\tilde{u}(0, t) + \hat{w}(0, t)|_1 \\ &\leq -\frac{\lambda}{2} |Z(t)| + M_1 |\tilde{u}(0, t) + \hat{w}(0, t)|_1\end{aligned}\quad (62)$$

where

$$M_1 = \max_{q \in [0, 2\pi]} \frac{c_2}{2} |M(q)^{-1}|. \quad (63)$$

Using (49), the temporal derivative of the fourth term of $W(t)$ can be bounded as follows:

$$\begin{aligned}&\frac{d}{dt} \left[b_2 \hat{D}(t) \int_0^1 (1+x) |\hat{w}_x(x, t)|_1 dx \right] \\ &= b_2 \hat{D}(t) \int_0^1 (1+x) \hat{w}_{xt}(x, t) \cdot \text{sign}(\hat{w}_x(x, t)) dx \\ &\quad + b_2 \dot{\hat{D}}(t) \int_0^1 (1+x) |\hat{w}_x(x, t)|_1 dx \\ &\leq b_2 \int_0^1 (1+x) \hat{w}_{xx}(x, t) \cdot \text{sign}(\hat{w}_x(x, t)) dx \\ &\quad + b_2 |\dot{\hat{D}}(t)| \int_0^1 (1+x) |h_3(x, t)|_1 dx \\ &\quad + b_2 \int_0^1 (1+x) |h_4(x, t) f_{\hat{u}}(t)|_1 dx \\ &\quad + b_2 |\dot{\hat{D}}(t)| \int_0^1 (x+1) |\hat{w}_x(x, t)|_1 dx.\end{aligned}\quad (64)$$

Using integration by parts, the first term in this inequality can be simplified as follows:

$$\begin{aligned}&b_2 \int_0^1 (1+x) \hat{w}_{xx}(x, t) \cdot \text{sign}(\hat{w}_x(x, t)) dx \\ &= (1+x) |\hat{w}_x(x, t)|_1 \Big|_0^1 - \int_0^1 |\hat{w}_x(x, t)|_1 dx \\ &= 2b_2 |\hat{w}_x(1, t)|_1 - b_2 |\hat{w}_x(0, t)|_1 - b_2 \|\hat{w}_x(t)\|_1.\end{aligned}\quad (65)$$

Using the same method for the terms containing $\hat{w}$ and $\tilde{u}$ within (61), the inequality (62), and applying the boundary conditions (46) and (48), the following inequality is obtained:

$$\begin{aligned}\dot{W}(t) &\leq -\frac{\lambda}{2} |Z(t)| + M_1 |\tilde{u}(0, t) + \hat{w}(0, t)|_1 - b_0 \|\tilde{u}(t)\|_1 \\ &\quad - b_0 |\tilde{u}(0, t)|_1 - b_1 \|\hat{w}(t)\|_1 - b_1 |\hat{w}(0, t)|_1 \\ &\quad - b_2 \|\hat{w}_x(t)\|_1 + 2b_2 |\hat{w}_x(1, t)|_1 - b_2 |\hat{w}_x(0, t)|_1 \\ &\quad + b_0 |\tilde{D}(t)| \int_0^1 (x+1) |g_1(x, t)|_1 dx \\ &\quad + b_0 |\dot{\hat{D}}(t)| \int_0^1 (x+1) |g_2(x, t)|_1 dx \\ &\quad + b_1 |\dot{\hat{D}}(t)| \int_0^1 (x+1) |h_1(x, t)|_1 dx \\ &\quad + b_1 \int_0^1 (x+1) |h_2(x, t) f_{\hat{u}}(t)|_1 dx\end{aligned}$$

$$\begin{aligned}
& +b_2|\dot{D}(t)| \int_0^1 (1+x)|h_3(x,t)|_1 dx \\
& +b_2 \int_0^1 (x+1)|h_4(x,t)f_{\bar{u}}(t)|_1 dx \\
& +b_1|\dot{D}(t)| \int_0^1 (x+1)|\hat{w}(x,t)|_1 dx \\
& +b_2|\dot{D}(t)| \int_0^1 (x+1)|\hat{w}_x(x,t)|_1 dx. \quad (66)
\end{aligned}$$

To bound the positive terms in the previous expression, we define the following combined functional:

$$S(t) = \frac{1}{2}(W(t) + \Gamma_R(t)) \quad (67)$$

where $\Gamma_R(t)$ is defined in (38). This functional measures the current tracking errors of the robot manipulator, the difference between the distributed input and the desired distributed input, and the magnitude of temporal derivatives of the desired trajectory.

Through application of Assumption 3, it is found that $|Z(t)|$ satisfies the following inequality:

$$\frac{\lambda}{2}|Z(t)| \geq \frac{\lambda}{2\sqrt{c_1}}V_0(Z(t)). \quad (68)$$

Furthermore, the term $b_2\|\hat{w}_x(t)\|_1$ satisfies the following inequality:

$$\begin{aligned}
b_2\|w_x(t)\|_1 & \geq \frac{1}{2}b_2 \int_0^1 (1+x)|\hat{w}_x(x,t)|_1 dx \\
& \geq \frac{1}{2\bar{D}}b_2\hat{D}(t) \int_0^1 (1+x)|\hat{w}_x(x,t)|_1 dx. \quad (69)
\end{aligned}$$

Similar inequalities can be formulated for the terms $b_0\|\bar{u}(t)\|_1$ and $b_1\|\hat{w}(t)\|_1$ appearing in (66). Thus, by applying Lemma 6 located in the Appendix, and introducing $\eta = (1/2)\min\{\lambda/\sqrt{c_1}, 1/\bar{D}\}$, one bounds (66) in terms of class $\mathcal{K}_\infty$ functions $\alpha_i(S(t))$ with $i = 5, \dots, 13$ of (67) and constants $M_2, M_3, M_4 > 0$

$$\begin{aligned}
\dot{W}(t) \leq & -\left(\eta W(t) - |\dot{D}(t)| \left(b_0\alpha_7(S(t)) + b_1\alpha_8(S(t)) \right. \right. \\
& \quad \left. \left. + b_2\alpha_9(S(t)) + 2b_2\alpha_{12}(S(t)) \right. \right. \\
& \quad \left. \left. + \frac{2}{\bar{D}}(b_1 + b_2)S(t) \right) \right. \\
& \quad \left. - |\tilde{D}(t)|b_0\alpha_6(S(t)) \right) - |\tilde{u}(0,t)|_1 \\
& \times \left(b_0 - b_1\alpha_{10}(S(t)) \right. \\
& \quad \left. - b_2\alpha_{11}(S(t)) - 2b_2\alpha_{13}(S(t)) - M_1 - b_1M_2 \right. \\
& \quad \left. - b_2(M_3 + 2M_4) \right) - |\hat{w}(0,t)|_1 \left(b_1 - M_1 \right) \\
& \quad - |\hat{w}_x(0,t)|_1 b_2 \left(1 - |\dot{D}(t)| \right). \quad (70)
\end{aligned}$$

To ensure that the terms corresponding to $|\tilde{u}(0,t)|_1$ and $|\hat{w}(0,t)|_1$ in (70) are negative $\forall t \geq 0$, we define the following constant parameters:

$$W^* > 0, \quad \Gamma_R^* > 0, \quad S^* = \frac{W^* + \Gamma_R^*}{2}. \quad (71)$$

Furthermore, we choose the constant parameters of the functional $W(t)$ such that

$$b_0 > b_1\alpha_{10}(S^*) + b_2\alpha_{11}(S^*) + 2b_2\alpha_{13}(S^*) + M_1 + b_1M_2 + b_2(M_3 + 2M_4) \quad (72)$$

$$b_1 > M_1. \quad (73)$$

To further reduce (70), we apply Lemma 1 and introduce the following functions:

$$\begin{aligned}
\alpha_1^*(S(t)) & = b_0\alpha_7(S(t)) + b_1\alpha_8(S(t)) + b_2\alpha_9(S(t)) \\
& \quad + 2b_2\alpha_{12}(S(t)) + \frac{2}{\bar{D}}(b_1 + b_2)S(t) \quad (74)
\end{aligned}$$

$$\alpha_2^*(S(t)) = b_0\alpha_6(S(t)). \quad (75)$$

For $W(t) \leq W^*$ and $\Gamma_R(t) \leq \Gamma_R^*$, (70) reduces to

$$\begin{aligned}
\dot{W}(t) \leq & -\left(\eta W(t) - \gamma H\alpha_1^*(W(t)) - \gamma H\alpha_1^*(\Gamma_R(t)) \right. \\
& \quad \left. - |\tilde{D}(t)|\alpha_2^*(W(t)) - |\tilde{D}(t)|\alpha_2^*(\Gamma_R(t)) \right) \\
& \quad - b_2(1 - \gamma H)|\hat{w}_x(0,t)|_1. \quad (76)
\end{aligned}$$

Noting that $\Gamma_R(t)$ is uniformly bounded due to Assumption 1, by choosing for a given $\nu \in (0, 1)$

$$\gamma < \frac{1}{H} \min \left\{ 1, \frac{(1-\nu)\eta}{2 \max_{x \in [0, W^*]} \alpha_1^*(x)}, \frac{(1-\nu)\nu\eta W^*}{2\alpha_1^*(\Gamma_R^*)} \right\} \quad (77)$$

$$|\tilde{D}(0)| < \min \left\{ \tilde{D}_{\max}, \frac{(1-\nu)\eta}{2 \max_{x \in [0, W^*]} \alpha_2^*(x)}, \frac{(1-\nu)\nu\eta W^*}{2\alpha_2^*(\Gamma_R^*)} \right\} \quad (78)$$

$$W(0) \leq W^* \quad (79)$$

$$\Gamma_R(t) \leq \Gamma_R^* \quad \forall t \geq 0 \quad (80)$$

one ensures that

$$\dot{W}(t) \leq -\nu^2\eta W(t), \quad \text{for } W(t) \geq \alpha_3^*(\Gamma_R(t)) \quad (81)$$

$$W(t) \leq W^* \quad \forall t \geq 0 \quad (82)$$

where

$$\alpha_3^*(\Gamma_R(t)) = \frac{\gamma H\alpha_1^*(\Gamma_R(t)) + |\tilde{D}(t)|\alpha_2^*(\Gamma_R(t))}{(1-\nu)\nu\eta}. \quad (83)$$

Through careful examination of (81), the following inequality is obtained:

$$W(t) \leq W(0)e^{-\nu^2\eta t} + \alpha_3^* \left(\sup_{s \in [0, t]} \{\Gamma_R(s)\} \right). \quad (84)$$

To provide a stability result in terms of Γ and Γ_R , Assumption 3 is used to prove the existence of two constants $c_1^*, c_2^* > 0$ such that $c_1^*\Gamma(t) \leq W(t) \leq c_2^*\Gamma(t)$. By combining this inequality with (84), the property (41) stated in Theorem 1 is obtained, with $c_3 = c_2^*/c_1^*$, $c_4 = \nu^2\eta$, and $\alpha^*(\Gamma_R(s)) = \alpha_3^*(\Gamma_R(s))/c_1^*$.

To prove convergence when $\dot{X}_R(t) \rightarrow 0$ as $t \rightarrow \infty$, we first analyze the convergence properties of $\alpha_3^*(\Gamma_R(t))$. As $q_R \in \mathcal{C}^5$ due to Assumption 1, it can be observed that $\Gamma_R(t)$ and

consequently $\alpha_3^*(\Gamma_R(t))$ converge to 0 as $t \rightarrow \infty$. Thus, $\forall \epsilon > 0, \exists T_1 > 0$ such that

$$\alpha_3^*(\Gamma_R(t)) \leq \epsilon \quad \forall t \geq T_1 \quad (85)$$

and thus consequently

$$\dot{W}(t) \leq -v^2 \eta W(t), \quad \text{for } W(t) \geq \epsilon \quad \forall t \geq T_1. \quad (86)$$

Integrating this inequality from T_1 to t , it can be observed that $\forall \epsilon > 0, \exists T_2 > 0$ such that

$$W(t) \leq \epsilon \quad \forall t \geq T_2 \quad (87)$$

and thus we conclude

$$\lim_{t \rightarrow \infty} W(t) = \lim_{t \rightarrow \infty} Z(t) = 0. \quad (88)$$

Finally, we verify the assumed uniform boundedness of $X(t)$, $U(t)$, and their derivatives in Lemma 1. From the relationship (82), it can be observed that $W(t)$, and thus $S(t)$ and $Z(t)$ are uniformly bounded. As both $Z(t)$ and $X_R(t)$ are uniformly bounded, $X(t)$ is consequently uniformly bounded. Additionally, through the application of Lemma 4 and Assumption 1 to (20), $U(t)$ can be bounded in terms of $W(t)$ and is consequently uniformly bounded.

To obtain a bound for $\dot{U}(t)$, (17) is substituted into (43). Taking the partial derivative of this result with respect to x yields

$$\begin{aligned} & \hat{D}(t)\dot{U}(t + \hat{D}(t)(x - 1)) \\ &= \hat{w}_x(x, t) + \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}(x, t), \hat{r}(x, t))\hat{p}_x(x, t) \\ & \quad + \frac{\partial \kappa}{\partial \hat{r}}(\hat{p}(x, t), \hat{r}(x, t))\hat{r}_x(x, t). \end{aligned} \quad (89)$$

Evaluating this equation at $x = 1$ yields

$$\begin{aligned} \hat{D}(t)\dot{U}(t) &= \hat{w}_x(1, t) + \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}(1, t), \hat{r}(1, t))\hat{p}_x(1, t) \\ & \quad + \frac{\partial \kappa}{\partial \hat{r}}(\hat{p}(1, t), \hat{r}(1, t))\hat{r}_x(1, t). \end{aligned} \quad (90)$$

Through the application of Lemma 4 to bound $\hat{p}(1, t)$, Lemma 5 to bound $\hat{p}_x(1, t)$, and Lemma 6 and Lemma 1 to bound $\hat{w}_x(1, t)$, noting that $|\tilde{u}(0, t)|_1$ is uniformly bounded due to the uniform boundedness of $U(t)$, and the fact that $\hat{D}(t) \in [\underline{D}, \overline{D}]$, $\dot{U}(t)$ can be bounded in terms of $S(t)$ and $U(t)$ and is consequently uniformly bounded.

VI. REMARKS ON DELAY-ADAPTIVE CONTROL LAW

In this research effort, our proposed delay-adaptive control law for a high-DOF robot manipulator consists of a predictor (18) of the system state after the delayed input reaches the system, a globally exponentially stable controller (20) for the delay-free system, and a gradient-based estimator of the input delay (33). It should be noted that this structure of the delay-adaptive control law allows for a wide selection of possible controllers for the delay-free system. One such permissible controller with (global) exponential stability is the feedback linearization-based controller, a popular control strategy for nonlinear systems that we have previously

used in a prediction-based control law for a known input delay [4], [5]. Compared with our previous work, this research effort possesses the following key differences.

- 1) In our previous work, the input delay was a known constant parameter. In this research effort, it is an unknown constant parameter.
- 2) In our previous work, a predictor was used without delay adaptation to compensate for a known delay. In this research effort, we perform both prediction (18) and delay adaptation (33) to compensate for an unknown delay.
- 3) In our previous work, we compared the performance of a control law with prediction with a control law without prediction in simulations and experiments. In this research effort, we compare the performance of a control law with prediction and delay adaptation with a control law with prediction, but without delay adaptation, in simulations and experiments.

It is important to note that the properties of the delay estimator (33) stated in Lemma 1 are contingent on the initial delay estimation error $|\tilde{D}(0)|$ being less than a critical value $\tilde{D}_{\max}$. Due to the strongly nonlinear relationship between the estimated delay $\hat{D}$ and the gradient of X_P given by (29), obtaining even a conservative mathematical expression for $\tilde{D}_{\max}$ is a difficult task [28]. As a mathematical expression for $\tilde{D}_{\max}$ is not currently known, and $|\tilde{D}(0)|$ is initially unknown, the necessary conditions for Lemma 1 may seem restrictive. However, it is still possible to determine an estimate for $\tilde{D}_{\max}$ through repeated simulations and experiments. As we demonstrate in Section VII through both simulations and experiments, the delay estimator (33) is capable of correcting a significant initial delay mismatch, both in cases of overestimation and underestimation. Thus, when a reasonable initial estimate of the delay is available, and thus we can upper bound $|\tilde{D}(0)|$, the properties stated in Lemma 1 can safely be assumed to hold.

To implement the proposed delay-adaptive control strategy we have proposed, several numerical approximations are necessary. The governing equation (18) for $\hat{p}$ and the governing equation (21) for X_P are both ODEs, and thus can be numerically solved by a variety of different methods, such as Euler's method, Heun's method, and RK4. In simulations and experiments, we have observed that using Euler's method provides sufficient accuracy in the estimation of $\hat{p}$ and X_P at the least computational burden out of the tested methods. Once X_P is determined, $(\partial X_P / \partial \hat{D})$ can be determined through a trapezoidal Reimann sum, as (29) is a definite integral. However, solving for $(\partial X_P / \partial \hat{D})$ is complicated by the presence of Φ_0 , which is governed by the ODE given in (30) with initial condition (31). While this ODE can technically be evaluated in the same manner as $\hat{p}$ and X_P , it is infeasible to compute in practice due to the function $(\partial f_0 / \partial X_P)$ present in the ODE containing computationally expensive terms to calculate such as $(\partial C / \partial X_P)$, which is a $7 \times 7 \times 14$ tensor. However, for a sufficiently small value of β , $(\partial \Phi_0 / \partial X_P) \approx 0$ and thus Φ_0 can be approximated as the identity matrix. Alternatively, it is possible to avoid calculation of the integral (29) through the use of finite

difference methods

$$\frac{\partial X_P}{\partial \hat{D}}(1, t, \hat{D}) \approx \frac{X_P(1, t, \hat{D} + h) - X_P(1, t, \hat{D})}{h} \quad (91)$$

where h is the timestep of the controller. In this method, $X_P(1, t, \hat{D} + h)$ is determined in the same manner as $X_P(1, t, \hat{D})$, and thus involves the solution of an additional ODE. In practice, approximating Φ_0 as the identity matrix and using the presented finite difference method have yielded near-identical results. In simulations and experiments, we have opted to approximate Φ_0 as the identity matrix.

VII. SIMULATION AND EXPERIMENTAL RESULTS

To assess the performance of the delay-adaptive approach, we perform both a series of simulations using ODE methods on Baxter's dynamic (1) and several corresponding experiments. In each simulation and experiment, Baxter must track a trajectory designed for a pick-and-place task in [9], while initially suffering from a large delay mismatch. Two cases are studied here, an underestimation of 0.9 s (0-s initial prediction, 0.9-s actual delay) and an overestimation of 0.5 s (0.9-s initial prediction, 0.4-s actual delay). These large delay mismatches are intentionally chosen to demonstrate the ability of the delay-adaptive approach to achieve stability under conditions that would cause a purely predictor-based approach to fail. In each simulation and experiment, the robot manipulator is commanded to remain stationary for a length of time equal to the initial estimated delay and then follow the 6-s pick-and-place trajectory described previously. This initial stationary period was chosen so that if the initial estimated delay were in fact equal to the actual input delay, the robot manipulator would be able to perfectly track the desired trajectory.

In both the simulation and experiment, we used an adaptation rate of $\gamma = 40$ and a replay length of $\beta = 0.1$ s. For the control law for the delay-free system described in Assumption 3, we used the following feedback linearization-based controller proposed in [4] and [5]:

$$\kappa(Z, X_R) = M(z + q_R)[\ddot{q}_R - k_1 k_2 \dot{z} - (k_1 + k_2)\dot{z}] + N(z + q_R, \dot{z} + \dot{q}_R) \quad (92)$$

with $k_1 = k_2 = 5$.

A. Trajectory Tracking Without Delay Adaptation When the Delay Is Underestimated

To compare the performance of the delay-adaptive approach studied in this research effort to a predictive approach without delay adaptation, we first performed several simulations in which the input delay is underestimated. Simulations are performed at an estimated delay of 0.9, 0.85, 0.8, and 0.78 s to examine the destabilizing effect of several magnitudes of delay mismatch.

The simulated and desired joint trajectories of Baxter can be seen in Fig. 2. When the estimated delay is 0.85 s, the predictive approach without delay adaptation still manages to closely track the desired trajectory. This indicates that even without delay adaptation, the predictive approach has a small degree of robustness to a mismatch in delay. However, when

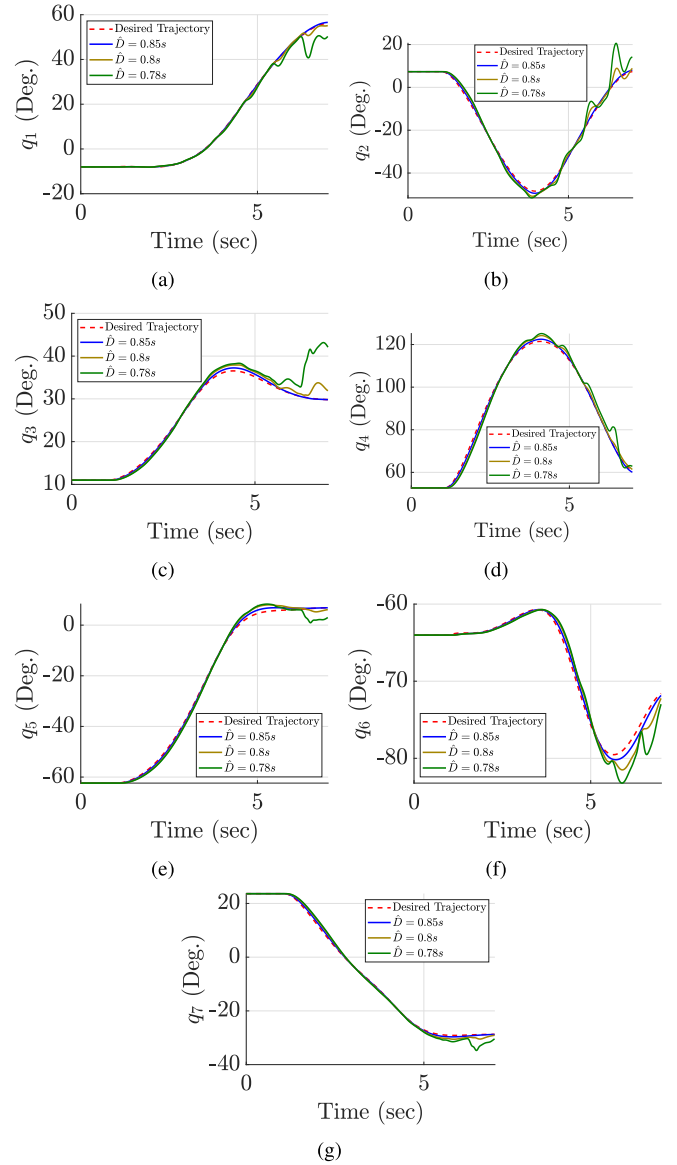


Fig. 2. Simulated and desired (red dashed line) joint trajectories of Baxter performing a pick-and-place task without delay adaptation, with an input delay of 0.9 s. Simulations are performed at an estimated delay of 0.85 s (blue line), 0.8 s (yellow line), and 0.78 s (green line).

the estimated delay is 0.8 s, the trajectory tracking appears to become worse throughout the procedure, with several large oscillations observed in the last second of the simulation. At a slightly larger delay mismatch when the estimated delay is 0.78 s, oscillations are observed throughout the procedure, with a large divergence from the desired trajectory at the end of the simulation. Thus, without delay adaptation, the tracking performance of the predictive approach is significantly reduced in the presence of a delay mismatch.

The simulated joint torque input signals can be seen in Fig. 3. Significant issues can be observed in the behavior of these torque signals. When the estimated delay is 0.85 s, several large oscillations can be observed in the torque input signal, which are not present when the estimated delay is equivalent to the true input delay. This indicates that even a relatively small delay mismatch can noticeably impact

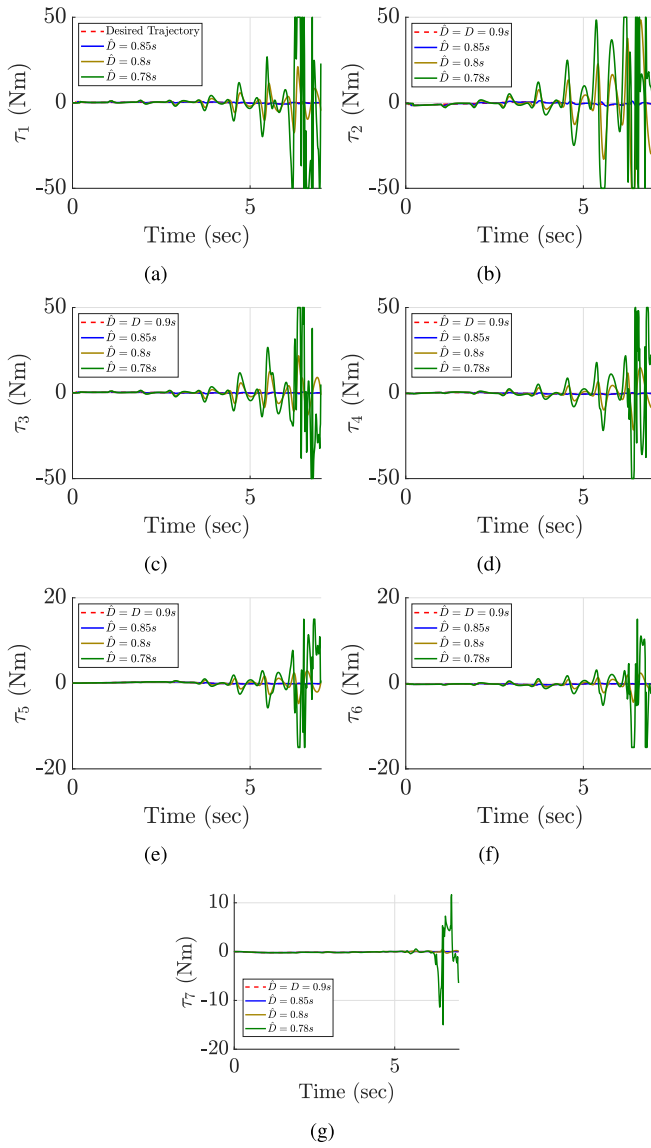


Fig. 3. Simulated joint torque input signals of Baxter performing a pick-and-place task without delay adaptation, with an input delay of 0.9 s. Simulations are performed at an estimated delay of 0.9 s (red dashed line), 0.85 s (blue line), 0.8 s (yellow line), and 0.78 s (green line).

the predictive approach without delay adaptation. Observing the behavior of the 0.8-s estimated delay, as well as that of the 0.78-s estimated delay, it is clear that both torque signals are unstable. In particular, at an estimated delay of 0.78 s, the input torque signal rapidly oscillates between the minimum and maximum torque outputs of Baxter during the last second of the procedure. Such a torque signal would likely damage the actuators of the robot manipulator and demonstrate dangerous behavior by a control scheme.

B. Trajectory Tracking When the Delay Is Underestimated

The experimental, simulated, and desired joint trajectories of Baxter can be seen in Fig. 4. Despite the large initial delay mismatch, the delay-adaptive approach is effective at driving the robot manipulator toward the desired trajectory. After the initial 0.9 s of operation, in which the robot manipulator was

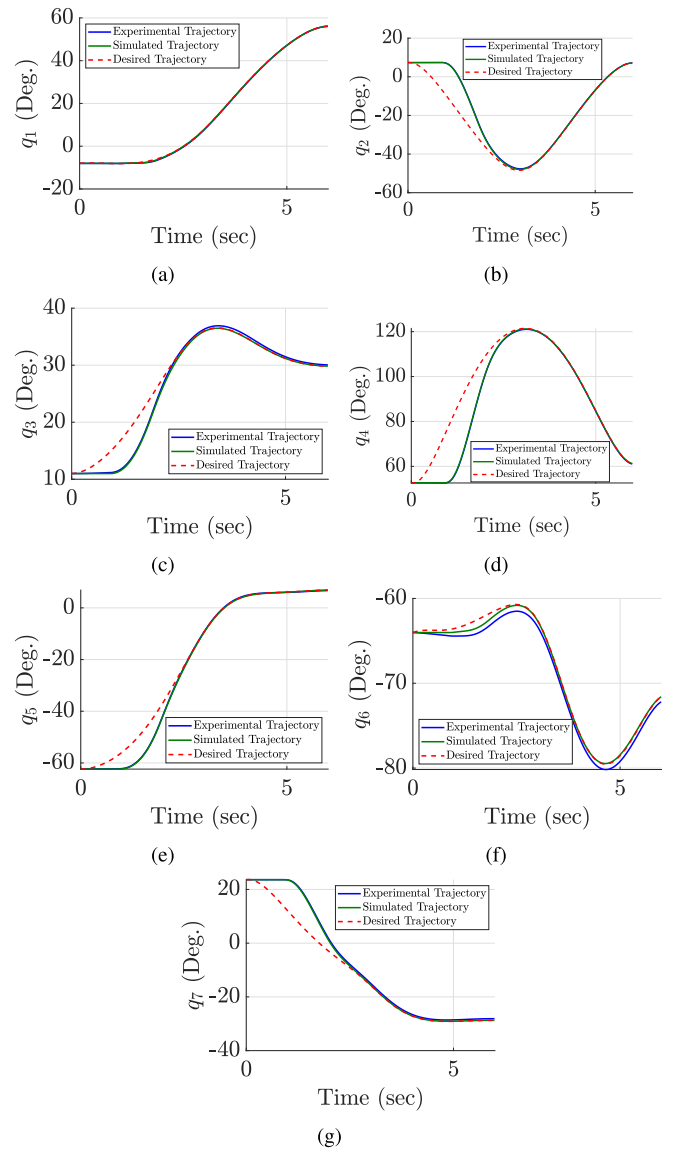


Fig. 4. Experimental (blue line), simulated (green line), and desired (red dashed line) joint trajectories of Baxter. The input delay of the system is initially underestimated (0-s initial prediction, 0.9-s actual delay).

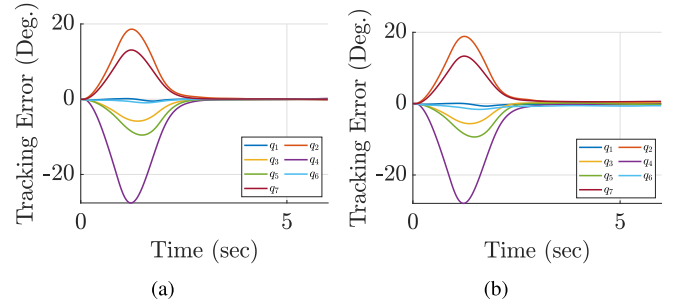


Fig. 5. (a) Simulated and (b) experimental joint tracking errors of Baxter. The input delay of the system is initially underestimated (0-s initial prediction, 0.9-s actual delay).

expectedly stationary due to the input delay, the robot manipulator quickly corrects itself toward the desired trajectory. This behavior can also be observed in Fig. 5, as both the simulated

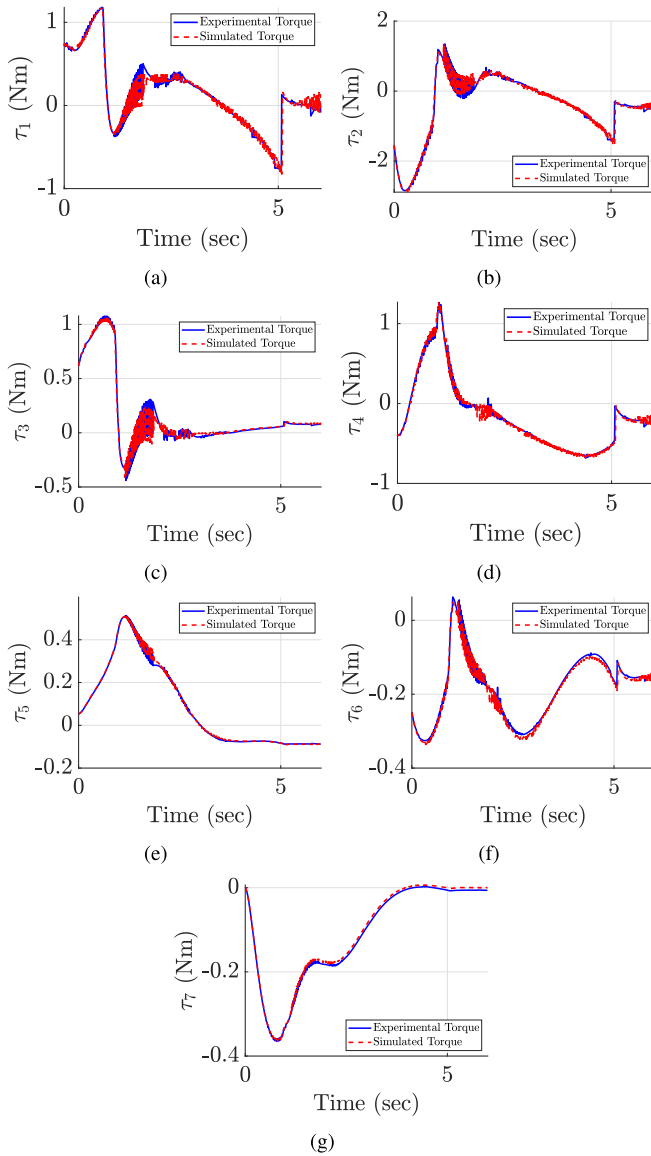


Fig. 6. Experimental (blue line) and simulated (red dashed line) joint torque input signals of Baxter. The input delay of the system is initially underestimated (0-s initial prediction, 0.9-s actual delay).

and experimental joint tracking errors decrease rapidly after around 1 s of operation. Furthermore, both the experimental and simulated trajectories appear to be smooth, indicating that the changes to the estimated delay during adaptation did not cause disturbances in the tracking performance of the manipulator. Thus, as the delay adaptive approach studied here is capable of handling an initial delay mismatch of 0.9 s in the case of underestimation, it is capable of handling a much larger delay mismatch than without delay adaptation, which could only safely handle a mismatch of 0.05 s.

The experimental and simulated joint torque input signals can be seen in Fig. 6. 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 Nm for joints 5–7. Thus, the delay-adaptive approach is able to compensate for a large delay underestimation without producing excessive joint torques. In the simulation and experiment,

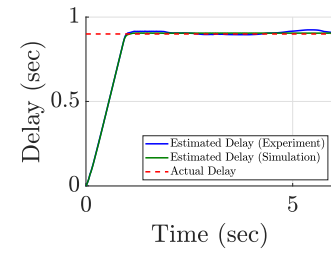


Fig. 7. Adaptation of the estimated delay in experiment (blue line) and simulation (green line), compared with the actual input delay (red dashed line). The input delay of the system is initially underestimated (0-s initial prediction, 0.9-s actual delay).

slight chattering can be observed in the input joint torque signal. This chattering, which is most prominent during the first 2 s of the simulation, is caused by large changes in the estimated delay in the beginning of the simulation. This behavior is not of concern however, as the chattering is of a small amplitude and is mostly eliminated after 2 s.

The adaptation of the estimated delay in the experiment and simulation can be seen in Fig. 7. In both the experiment and simulation, the estimated delay quickly converges to the actual delay. Furthermore, the following important observations can be made regarding the behavior of the adaptation. First, the rate of change of adaptation appears to be constant during the first 0.9 s of the simulation and experiment, until the estimated delay coincides with the actual delay. Due to the initial state of the robot manipulator being at rest, delays longer than the elapsed time t are indistinguishable from a delay of t seconds. Thus, as a consequence of the properties stated in Lemma 1, the estimated delay is upper bounded by the elapsed time in the simulation and experiment. Second, we observe that the estimation of the delay in simulation and experiment follows a nearly identical curve. This indicates that the delay-adaptive procedure does not suffer significantly from factors such as measurement noise of joint states which are present in the experiment but not in the simulation. Finally, it can be seen that there is slight overshoot in the estimated delay during the experiment. Although this behavior technically violates Lemma 1, it can reasonably be attributed to discretization of the control law.

C. Trajectory Tracking When the Delay Is Overestimated

The experimental, simulated, and desired joint trajectories of Baxter can be seen in Fig. 8. As was the case for an underestimated delay, the delay-adaptive approach is effective at driving the robot manipulator toward the desired trajectory. After the initial 0.4 s of operation, the manipulator starts following the curve of the desired trajectory. However, the manipulator was intended to remain stationary for 0.9 s, and thus has begun to accumulate error. After 1.5 s of operation, a shift in the manipulator behavior is observed, as the robot manipulator quickly moves to align to the desired trajectory. After 2 s have elapsed, the manipulator achieves near-perfect tracking of the desired trajectory for the remainder of the task. This behavior can also be observed in Fig. 9, as the errors increase after 0.4 s, reach a maximum at 1.5 s, and

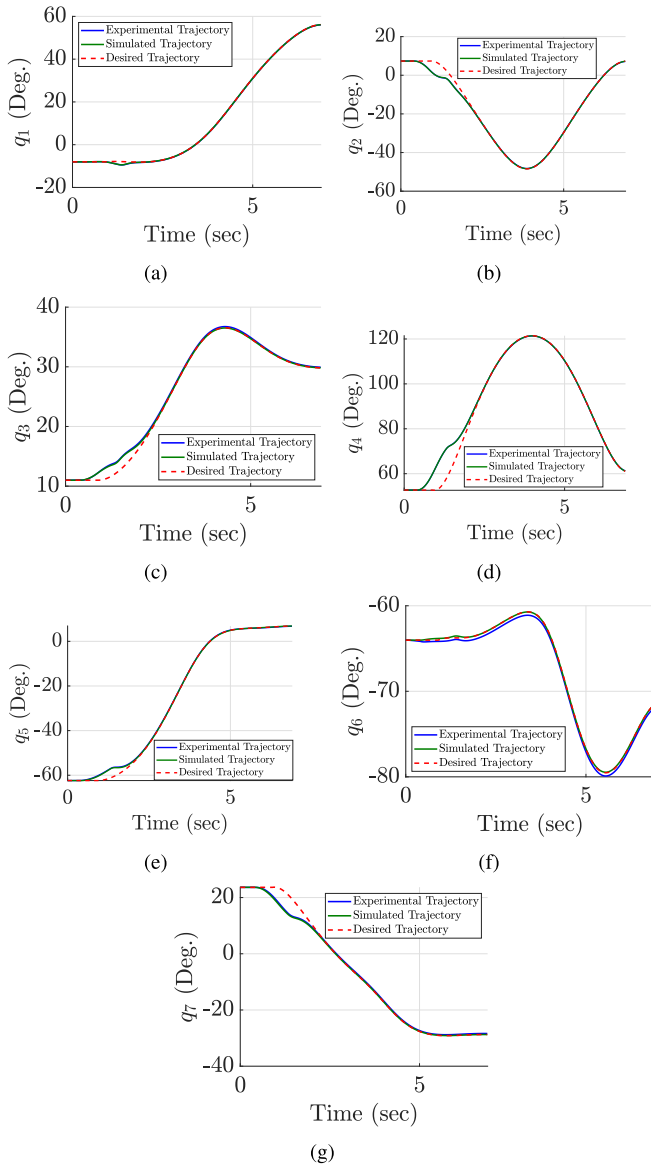


Fig. 8. Experimental (blue line), simulated (green line), and desired (red dashed line) joint trajectories of Baxter. The input delay of the system is initially overestimated (0.9-s initial prediction, 0.4-s actual delay).

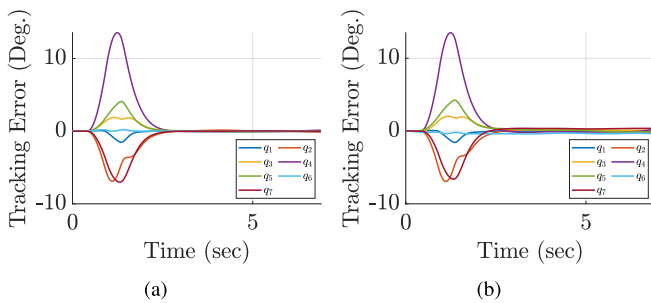


Fig. 9. (a) Simulated and (b) experimental joint tracking errors of Baxter. The input delay of the system is initially overestimated (0.9-s initial prediction, 0.4-s actual delay).

taper off after 2 s. As was the case for the underestimated delay, both the simulated and experimental trajectories appear to be smooth, indicating that the changes to the estimated delay

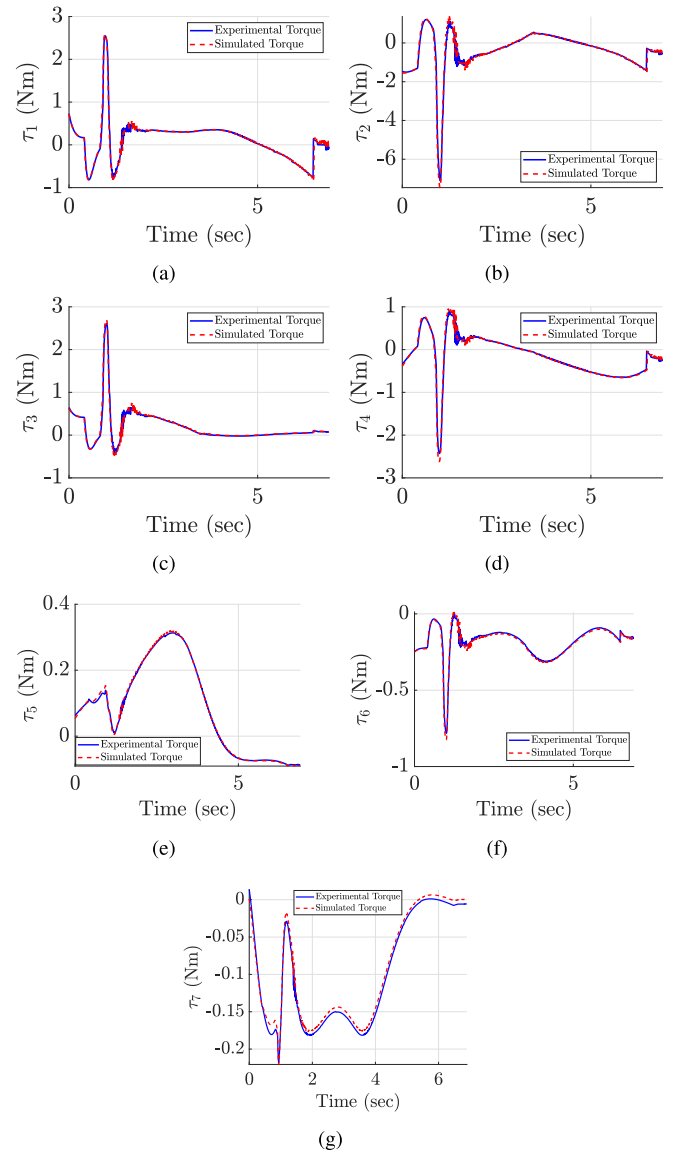


Fig. 10. Experimental (blue line) and simulated (red dashed line) joint torque input signals of Baxter. The input delay of the system is initially overestimated (0.9-s initial prediction, 0.4-s actual delay).

during adaptation did not cause disturbances in the tracking performance of the manipulator.

The experimental and simulated joint torque input signals can be seen in Fig. 10. As was the case with underestimation of the delay, the generated input torques are significantly lower than the maximum torque output of Baxter's joints. Thus, the delay-adaptive approach is able to compensate for a large delay overestimation without producing excessive joint torques. Compared with the case of underestimated delay, there does not appear to be chattering in the torque input signal. However, several joints exhibit a large spike in the torque input signal at around 1 s of operation, with smaller spikes at 0.4 and 1.5 s. These torque spikes are likely due to the initial unexpected robot manipulator motion, the start of delay adaptation, and the convergence of delay adaptation, respectively. Thus, qualitatively distinct behavior is observed between an underestimation and an overestimation of the

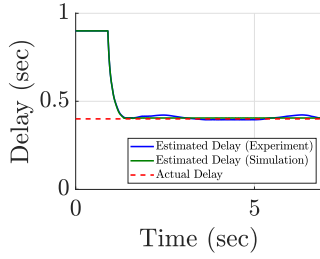


Fig. 11. Adaptation of the estimated delay in experiment (blue line) and simulation (green line), compared with the actual input delay (red dashed line). The input delay of the system is initially overestimated (0.9-s initial prediction, 0.4-s actual delay).

delay, although the robot manipulator displays near-perfect tracking throughout the majority of the procedure in both the cases.

The adaptation of the estimated delay in the experiment and simulation can be seen in Fig. 11. Several important observations can be made regarding the behavior of adaptation. During the first 0.9 s of operation, no changes are visible in the estimated delay. This behavior is due to the local nature of the gradient-based delay estimate approach. As previously described while examining the behavior of the delay adaptation when the delay was underestimated, delays longer than the elapsed time t are indistinguishable from a delay of t seconds. Thus, the gradient is 0 around the estimated delay of 0.9 s, until at least 0.9 s have elapsed in the procedure. After the operation time reaches 0.9 s, the estimated delay quickly converges to the actual delay. It is important to note that the behavior of the adaptation matches the observed behavior in the joint positions and torques. The estimated delay starts adapting at 0.9 s and finishes adapting around 1.5 s, which aligns with the spikes observed in the torque profiles, as well as the changes in the joint error signals.

VIII. CONCLUSION

In this effort, we investigated the analytical and experimental trajectory-tracking control of a 7-DOF robot manipulator with an unknown long actuator delay. To compensate for this unknown delay, we formulated a delay-adaptive prediction-based control strategy to simultaneously estimate the unknown delay while driving the robot manipulator toward the desired trajectory. To the best of the authors' knowledge, this article is the first to present a delay-adaptive approach for a nonlinear system with multiple inputs. Through Lyapunov analysis using the $\mathcal{L}_1$ norm, we obtained a local asymptotic stability result of the proposed controller. Then, we demonstrated through both simulation and experiment that the proposed controller is capable of achieving desirable trajectory tracking performance, even in the case of a large initial delay mismatch. As shown in Fig. 12, the delay-adaptive approach significantly improves the tracking performance of the robot manipulator when there is a delay mismatch, without sacrificing the performance when the delay is properly identified. This research represents a large improvement upon the predictor-based approach in the case of an unknown delay, and thus has promising potential for use cases in which the delay is difficult to accurately predict or measure directly.



Fig. 12. Baxter performing a pick-and-place task while subjected to an input delay of 0.9 s (0-s initial delay estimate).

APPENDIX A MATHEMATICAL EXPRESSIONS FOR TERMS PRESENTED IN SECTION IV

Note that in this section, notations indicating distributed functions of (x, t) and (y, t) have been removed for the sake of brevity

$$g_1 = \frac{1}{\hat{D}(t)} \hat{u}_x \quad (\text{A.1})$$

$$g_2 = \frac{D}{\hat{D}(t)} (x-1) \hat{u}_x \quad (\text{A.2})$$

$$h_1 = (x-1) \hat{u}_x - \hat{D}(t) \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r}) \int_0^x \Phi(x, y, t) \psi dy - x \frac{\partial \kappa}{\partial \hat{r}}(\hat{p}, \hat{r}) \hat{r}_x \quad (\text{A.3})$$

$$h_2 = \hat{D}(t) \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r}) \Phi(x, 0, t) \quad (\text{A.4})$$

$$h_3 = \hat{u}_x + (x-1) \hat{u}_{xx} - \frac{d}{dx} \left[\hat{D}(t) \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r}) \int_0^x \Phi(x, y, t) \psi dy - x \frac{\partial \kappa}{\partial \hat{r}}(\hat{p}, \hat{r}) \hat{r}_x \right] \quad (\text{A.5})$$

$$h_4 = \hat{D}(t) \frac{d}{dx} \left[\frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r}) \right] \Phi(x, 0, t) + \hat{D}(t)^2 \frac{\partial \kappa}{\partial \hat{p}}(\hat{p}, \hat{r}) \frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \Phi(x, 0, t) \quad (\text{A.6})$$

$$\begin{aligned} f_{\tilde{u}}(t) &= f(\hat{p}(0, t), u(0, t), \hat{r}(0, t)) \\ &\quad - f(\hat{p}(0, t), \hat{u}(0, t), \hat{r}(0, t)) \\ &= \begin{bmatrix} 0 \\ M(q(t))^{-1} \end{bmatrix} \tilde{u}(0, t) \end{aligned} \quad (\text{A.7})$$

where

$$\psi = f(\hat{p}, \hat{u}, \hat{r}) + (x-1) \frac{\partial f}{\partial \hat{u}}(\hat{p}, \hat{u}, \hat{r}) \hat{u}_x + x \frac{\partial f}{\partial \hat{r}}(\hat{p}, \hat{u}, \hat{r}) \hat{r}_x \quad (\text{A.8})$$

$$\hat{u} = \hat{w} + \kappa(\hat{p}, \hat{r}) \quad (\text{A.9})$$

$$\hat{u}_x = \hat{w}_x + \frac{d}{dx} [\kappa(\hat{p}, \hat{r})] \quad (\text{A.10})$$

$$\hat{u}_{xx} = \hat{w}_{xx} + \frac{d^2}{dx^2} [\kappa(\hat{p}, \hat{r})] \quad (\text{A.11})$$

and $\Phi(x, y, t)$ is the solution to the following homogeneous ODE:

$$\frac{\partial \Phi}{\partial x}(x, y, t) = \hat{D}(t) \frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \Phi(x, y, t) \quad (\text{A.12})$$

$$\Phi(y, y, t) = I. \quad (\text{A.13})$$

APPENDIX B

TECHNICAL LEMMAS USED TO BOUND TERMS

Lemma 3: The transition matrix $\Phi(x, y, \bullet)$ associated with the space-varying homogeneous equation $(\partial \phi / \partial x)(x, \bullet) = A(x, \bullet) \phi(x, \bullet)$ with $y \in [0, 1]$ and $x \in [y, 1]$ satisfies the following property:

$$|\Phi(x, y, \bullet)| \leq \exp \max_{x \in [0, 1]} |A(x, \bullet)|. \quad (\text{B.1})$$

Furthermore, there exists a class $\mathcal{K}_\infty$ function α_1 such that for all $y \in [0, 1]$, $x \in [y, 1]$

$$|\Phi(x, y, \bullet)| \leq 1 + \alpha_1(\|A(\bullet)\|_1). \quad (\text{B.2})$$

Proof: By definition, for a given $y \in [0, 1]$ and $t \geq 0$, Φ satisfies the following differential equation in x :

$$\frac{\partial \Phi}{\partial x}(x, y, \bullet) = A(x, \bullet) \Phi(x, y, \bullet), \quad x \in [y, 1] \quad (\text{B.3})$$

$$\Phi(y, y, \bullet) = I. \quad (\text{B.4})$$

Therefore, its norm satisfies for any $x \in [y, 1]$

$$\begin{aligned} \left| \frac{\partial \Phi}{\partial x}(x, y, \bullet) \right| &\leq \left| \frac{\partial \Phi}{\partial x}(x, y, \bullet) \right| \\ &\leq |A(x, \bullet)| |\Phi(x, y, \bullet)| \end{aligned} \quad (\text{B.5})$$

with $|\Phi(y, y, \bullet)| = 1$. Therefore, there exists a $\mathcal{K}_\infty$ function α_1 such as introduced in this lemma. Furthermore, through taking the maximum of $|A(x, \bullet)|$ with respect to x and applying separation of variables, one obtains the upper bound (B.1) stated in this lemma. ■

Lemma 4: There exists a class $\mathcal{K}_\infty$ function α_2 such that for all $x \in [0, 1]$

$$|\hat{p}(x, t)| \leq \alpha_2(|Z| + \|\hat{w}(t)\|_1). \quad (\text{B.6})$$

Proof: The distributed predictor satisfies the following spatial ODE:

$$\begin{aligned} \hat{p}_x(x, t) &= \hat{D}(t) f(\hat{p}(x, t), \hat{w}(x, t) \\ &\quad + \kappa(\hat{p}(x, t), \hat{r}(x, t)), \quad \hat{r}(x, t)) \end{aligned} \quad (\text{B.7})$$

$$\hat{p}(0, t) = Z(t). \quad (\text{B.8})$$

Therefore, as $\hat{D}(t) \in [\underline{D}, \overline{D}]$, f is continuous, the plant (13)–(15) is forward complete, $f(0, \kappa(0, \hat{r}), \hat{r}) = 0$, and $\hat{r}$ is uniformly bounded, there exists a $\mathcal{K}_\infty$ function α_2 such as introduced in this lemma. ■

Lemma 5: There exist class $\mathcal{K}_\infty$ functions α_3 and α_4 such that for all $x \in [0, 1]$

$$|\hat{p}_x(x, t)| \leq \alpha_3(|\hat{p}(x, t)|_1 + |\hat{w}(x, t)|_1) \quad (\text{B.9})$$

$$\begin{aligned} |\hat{p}_{xx}(x, t)| &\leq \alpha_4(|\hat{p}(x, t)|_1 + |\hat{w}(x, t)|_1 + |\hat{w}_x(x, t)|_1 \\ &\quad + |\hat{r}_x(x, t)|_1). \end{aligned} \quad (\text{B.10})$$

Proof: Note that in this proof, notations indicating distributed functions of (x, t) have been removed for the sake of brevity.

The quantity $\hat{p}_x$ is equivalent to the following expression:

$$\begin{aligned} \hat{p}_x &= \hat{D}(t) f(\hat{p}, \hat{u}, \hat{r}) \\ &= \hat{D}(t) f(\hat{p}, \hat{w} + \kappa(\hat{p}, \hat{r}), \hat{r}). \end{aligned} \quad (\text{B.11})$$

Therefore, as $\hat{D}(t) \in [\underline{D}, \overline{D}]$, $f(0, \kappa(0, \hat{r}), \hat{r}) = 0$, and $\hat{r}$ is uniformly bounded, there exists a $\mathcal{K}_\infty$ function α_3 such as introduced in this lemma.

By taking the spatial derivative of $\hat{p}_x$, the following expression for $\hat{p}_{xx}$ is obtained:

$$\begin{aligned} \hat{p}_{xx} &= \hat{D}(t) \left[\frac{\partial f}{\partial \hat{p}}(\hat{p}, \hat{u}, \hat{r}) \hat{p}_x + \frac{\partial f}{\partial \hat{u}}(\hat{p}, \hat{u}, \hat{r}) \hat{u}_x + \frac{\partial f}{\partial \hat{r}}(\hat{p}, \hat{u}, \hat{r}) \hat{r}_x \right]. \end{aligned} \quad (\text{B.12})$$

For a given $\hat{D}$ and $\hat{r}$, this expression can be bounded by the terms $\hat{p}$, $\hat{u}$, $\hat{p}_x$, $\hat{u}_x$, and $\hat{r}_x$. $\hat{u}$ can be bounded by $\hat{p}$ and $\hat{w}$, and $\hat{p}_x$ can be bounded by α_2 . Through the investigation of (A.10), it can be seen that $\hat{u}_x$ can be bounded by $\hat{w}_x$, $\hat{p}$, $\hat{p}_x$, and $\hat{r}_x$, with $\hat{u}_x = 0$ if $\hat{w}_x = \hat{p}_x = \hat{r}_x = 0$. Furthermore, $p_{xx} = 0$ if $\hat{u}_x = \hat{p}_x = \hat{r}_x = 0$. Therefore, as $\hat{D}(t) \in [\underline{D}, \overline{D}]$, and $\hat{r}$ is uniformly bounded, there exists a $\mathcal{K}_\infty$ function α_4 such as introduced in this lemma. ■

Lemma 6: There exist class $\mathcal{K}_\infty$ functions α_i , with $i = 5, \dots, 13$ and constants $M_2, M_3, M_4 > 0$ such that

$$\begin{aligned} \int_0^1 (x+1) |g_1(x, t)|_1 dx &\leq \alpha_5(|Z| + \|\hat{w}(t)\|_1 + \|\hat{w}_x(t)\|_1 + \|\hat{r}_x(t)\|_1) \leq \alpha_6(S(t)) \end{aligned} \quad (\text{B.13})$$

$$\int_0^1 (x+1) |g_2(x, t)|_1 dx \leq \alpha_7(S(t)) \quad (\text{B.14})$$

$$\int_0^1 (x+1) |h_1(x, t)|_1 dx \leq \alpha_8(S(t)) \quad (\text{B.15})$$

$$\int_0^1 (1+x) |h_3(x, t)|_1 dx \leq |\hat{w}_x(0, t)|_1 + \alpha_9(S(t)) \quad (\text{B.16})$$

$$\int_0^1 (x+1) |h_2(x, t) f_{\hat{u}}(t)|_1 dx \leq (M_2 + \alpha_{10}(S(t))) |\hat{u}(0, t)|_1 \quad (\text{B.17})$$

$$\int_0^1 (x+1) |h_4(x, t) f_{\hat{u}}(t)|_1 dx \leq (M_3 + \alpha_{11}(S(t))) |\hat{u}(0, t)|_1 \quad (\text{B.18})$$

$$|\hat{w}_x(1, t)|_1 \leq |\dot{\hat{D}}(t)| \alpha_{12}(S(t)) + (M_4 + \alpha_{13}(S(t))) |\hat{u}(0, t)|_1. \quad (\text{B.19})$$

Proof: By the application of Lemma 4, Lemma 5, and Assumption 1, along with the fact that f and κ are class C^2 functions, one obtains the existence of α_5 as stated in this lemma. Then, through the application of the following

inequality:

$$\begin{aligned} & \left(|Z(t)| + \|\tilde{u}(t)\|_1 + \|\hat{w}(t)\|_1 + \|\hat{w}_x(t)\|_1 \right. \\ & \quad \left. + |\hat{f}_x(1, t)|_1 + \|\hat{f}_x(t)\|_1 + \|\hat{f}_{xx}(t)\|_1 \right) \\ & \leq 2 \max \left\{ 1, \frac{1}{b_0 D}, \frac{1}{b_1 D}, \frac{1}{b_2 D} \right\} S(t) \end{aligned} \quad (\text{B.20})$$

the existence of α_6 as stated in this lemma is obtained. Applying these same considerations, the existence of α_7 as stated in this lemma is also obtained.

To bound the higher order spatial derivative term in h_3 , integration by parts may be used to bound this term as a function of w_x

$$\begin{aligned} \int_0^1 (x^2 - 1) |\hat{w}_{xx}(x, t)|_1 dx & \leq |\hat{w}_x(0, t)|_1 \\ & \quad + \int_0^1 2x |\hat{w}_x(x, t)|_1 dx. \end{aligned} \quad (\text{B.21})$$

Applying this bound, along with the previously stated considerations and Lemma 3, the existence of α_8 and α_9 as stated in this lemma is obtained.

To bound further terms present in this lemma, additional considerations must be made. First, due to the control affine property of robot manipulators, the term $f_{\tilde{u}}$ can be bounded as follows:

$$f_{\tilde{u}}(t) \leq \max_{q \in [0, 2\pi)} |M(q)^{-1}| \|\tilde{u}(0, t)\|_1. \quad (\text{B.22})$$

Second, using previous considerations, the terms h_2 and h_4 can be bounded in terms of S as follows:

$$|h_2(x, t)|_1 \leq C_{h_2} + \alpha_{h_2}(S(t)) \quad (\text{B.23})$$

$$|h_4(x, t)|_1 \leq C_{h_4} + \alpha_{h_4}(S(t)) \quad (\text{B.24})$$

where $C_{h_2}, C_{h_4} \geq 0$ and $\alpha_{h_2}, \alpha_{h_4}$ are class $\mathcal{K}_\infty$ functions. Note that constants C_{h_2} and C_{h_4} are necessary since we do not have $h_2 = h_4 = 0$ when $S = 0$. Applying all previously stated considerations, the existence of class $\mathcal{K}_\infty$ functions $\alpha_{10}, \alpha_{11}, \alpha_{12}$, and α_{13} and constants M_2, M_3 , and M_4 as stated in this lemma are obtained. ■

ACKNOWLEDGMENT

The authors in this research effort would like to extend their gratitude to Prof. Delphine Bresch-Pietri, both for her work in developing the original delay-adaptive method our research extends upon and providing valuable insight behind several formulations present in her research. 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] R. J. Anderson and M. W. Spong, "Bilateral control of teleoperators with time delay," in *Proc. 27th IEEE Conf. Decis. Control*, Aug. 1988, pp. 131–138.
- [2] M. Ariola and A. Pironti, "H ∞ optimal terminal control for linear systems with delayed states and controls," *Automatica*, vol. 44, no. 10, pp. 2676–2679, 2008.
- [3] Z. Artstein, "Linear systems with delayed controls: A reduction," *IEEE Trans. Autom. Control*, vol. AC-27, no. 4, pp. 869–879, Aug. 1982.
- [4] M. Bagheri, P. Naseradinmousavi, and M. Krstić, "Feedback linearization based predictor for time delay control of a high-DOF robot manipulator," *Automatica*, vol. 108, Oct. 2019, Art. no. 108485.
- [5] M. Bagheri, P. Naseradinmousavi, and M. Krstić, "Time delay control of a high-DOF robot manipulator through feedback linearization based predictor," in *Proc. Dyn. Syst. Control Conf.*, vol. 59162. New York, NY, USA: American Society of Mechanical Engineers, 2019, Art. no. V003T16A001.
- [6] A. K. Bejczy, W. S. Kim, and S. C. Venema, "The phantom robot: Predictive displays for teleoperation with time delay," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1990, pp. 546–551.
- [7] N. Bekiaris-Liberis and M. Krstic, "Compensation of time-varying input and state delays for nonlinear systems," *J. Dyn. Syst., Meas., Control*, vol. 134, no. 1, Jan. 2012, Art. no. 011009.
- [8] N. Bekiaris-Liberis and M. Krstic, *Nonlinear Control Under Nonconstant Delays*. Philadelphia, PA, USA: SIAM, 2013.
- [9] A. Bertino, M. Bagheri, M. Krstić, and P. Naseradinmousavi, "Experimental autonomous deep learning-based 3d path planning for a 7-DOF robot manipulator," in *Proc. ASME Dyn. Syst. Control Conf.*, vol. 2, Oct. 2019, Art. no. V002T14A002.
- [10] D. Bresch-Pietri, J. Chauvin, and N. Petit, "Adaptive control scheme for uncertain time-delay systems," *Automatica*, vol. 48, no. 8, pp. 1536–1552, Aug. 2012.
- [11] D. Bresch-Pietri and M. Krstic, "Adaptive trajectory tracking despite unknown input delay and plant parameters," *Automatica*, vol. 45, no. 9, pp. 2074–2081, 2009.
- [12] D. Bresch-Pietri and M. Krstic, "Delay-adaptive predictor feedback for systems with unknown long actuator delay," *IEEE Trans. Autom. Control*, vol. 55, no. 9, pp. 2106–2112, Sep. 2010.
- [13] D. Bresch-Pietri and M. Krstic, "Delay-adaptive control for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 59, no. 5, pp. 1203–1218, May 2014.
- [14] S. Gumussoy and H. Özbay, "Stable H ∞ controller design for time-delay systems," *Int. J. Control*, vol. 81, no. 4, pp. 546–556, 2008.
- [15] I. Karafyllis, "Stabilization by means of approximate predictors for systems with delayed input," *SIAM J. Control Optim.*, vol. 49, no. 3, pp. 1100–1123, Jan. 2011.
- [16] H. K. Khalil, *Nonlinear Systems*, vol. 3. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.
- [17] W. S. Kim, B. Hannaford, and A. K. Fejczy, "Force-reflection and shared compliant control in operating telemanipulators with time delay," *IEEE Trans. Robot. Autom.*, vol. 8, no. 2, pp. 176–185, Apr. 1992.
- [18] M. Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*. Cham, Switzerland: Springer, 2009.
- [19] M. Krstic and A. Banaszuk, "Multivariable adaptive control of instabilities arising in jet engines," *Control Eng. Pract.*, vol. 14, no. 7, pp. 833–842, Jul. 2006.
- [20] W. Kwon and A. Pearson, "Feedback stabilization of linear systems with delayed control," *IEEE Trans. Autom. Control*, vol. AC-25, no. 2, pp. 266–269, Apr. 1980.
- [21] A. Manitius and A. W. Olbrot, "Finite spectrum assignment problem for systems with delays," *IEEE Trans. Autom. Control*, vol. AC-24, no. 4, pp. 541–552, Aug. 1979.
- [22] F. Mazenc and S.-I. Niculescu, "Generating positive and stable solutions through delayed state feedback," *Automatica*, vol. 47, pp. 525–533, Mar. 2011.
- [23] F. Mazenc, S.-I. Niculescu, and M. Krstic, "Some remarks on Lyapunov–Krasovskii functionals and reduction approach for input delay compensation," in *Proc. Amer. Control Conf. (ACC)*, Jun. 2012, pp. 6229–6234.
- [24] T. G. Molnár, W. B. Qin, T. Insperger, and G. Orosz, "Application of predictor feedback to compensate time delays in connected cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 2, pp. 545–559, Feb. 2018.
- [25] S. Mondie and W. Michiels, "A safe implementation for finite spectrum assignment: Robustness analysis," in *Proc. 42nd IEEE Int. Conf. Decis. Control*, Dec. 2003, pp. 4569–4574.
- [26] M. T. Nihtila, "Finite pole assignment for systems with time-varying input delays," in *Proc. 30th IEEE Conf. Decis. Control*, Dec. 1991, pp. 927–928.
- [27] E. Nuño, L. Basañez, R. Ortega, and M. Spong, "Position tracking for non-linear teleoperators with variable time delay," *Int. J. Robot. Res.*, vol. 28, no. 7, pp. 895–910, 2009.
- [28] A. O'Dwyer, "A survey of techniques for the estimation and compensation," Tech. Rep., 2000.
- [29] E. Polak, *Optimization: Algorithms Consistent Approximations*, vol. 124. Cham, Switzerland: Springer, 2012.

- [30] T. B. Sheridan, "Space teleoperation through time delay: Review and prognosis," *IEEE Trans. Robot. Autom.*, vol. 9, no. 5, pp. 592–606, Oct. 1993.
- [31] Z.-L. Tang, S. S. Ge, and W. He, "Predictive control for a class of nonholonomic constrained mechanical systems with input delay," in *Proc. 11th World Congr. Intell. Control Autom.*, Jun. 2014, pp. 879–884.
- [32] G. Tao and P. V. Kokotovic, *Adapt. Control Systems With Actuator Sensor Nonlinearities*. Hoboken, NJ, USA: Wiley, 1996.
- [33] X.-G. Yan, S. K. Spurgeon, and C. Edwards, "Sliding mode control for time-varying delayed systems based on a reduced-order observer," *Automatica*, vol. 46, no. 8, pp. 1354–1362, Aug. 2010.
- [34] X.-G. Yan, S. K. Spurgeon, and Y. Orlov, "Sliding mode observer based-controller design for nonlinear systems with time varying delay," in *Recent Results Nonlinear Delay Control Systems*. Springer, 2016, pp. 347–365.
- [35] Y. Yildiz, A. Annaswamy, I. V. Kolmanovsky, and D. Yanakiev, "Adaptive posicast controller for time-delay systems with relative degree $n^* \leq 2$," *Automatica*, vol. 46, no. 2, pp. 279–289, 2010.
- [36] Y. Zhu and E. Fridman, "Sub-predictors for network-based control under uncertain large delays," *Automatica*, vol. 123, Jan. 2021, Art. no. 109350.
- [37] Y. Zhu and M. Krstic, *Delay-Adaptive Linear Control*, vol. 70, Princeton, NJ, USA: Princeton Univ. Press, 2020.
- [38] Y. Zhu, M. Krstic, and H. Su, "Adaptive output feedback control for uncertain linear time-delay systems," *IEEE Trans. Autom. Control*, vol. 62, no. 2, pp. 545–560, Feb. 2017.
- [39] Y. Zhu, M. Krstic, and H. Su, "PDE boundary control of multi-input LTI systems with distinct and uncertain input delays," *IEEE Trans. Autom. Control*, vol. 63, no. 12, pp. 4270–4277, Dec. 2018.



Alexander Bertino received the B.Sc. degree in mechanical engineering from the University of California, San Diego, La Jolla, CA, USA, in 2017, and the M.Sc. degree in mechanical engineering from San Diego State University, San Diego, CA, USA, in 2019. He is currently pursuing the Ph.D. degree in mechanical engineering with San Diego State University and the University of California, San Diego.

He is currently working as a Research Assistant with the Dynamic Systems and Controls Laboratory, San Diego State University. His research interests include robotics, control theory, adaptive control, delay systems, and machine learning.

Mr. Bertino was a recipient of the 2020 DSCD Robotics TC Best Student Paper for the research paper entitled "Experimental and Analytical Decentralized Adaptive Control of a 7-DOF Robot Manipulator."



Peiman Naseradinmousavi received the B.Sc. degree in mechanical engineering (dynamics and control) from the University of Tabriz, Tabriz, Iran, in 2002, and the Ph.D. degree in mechanical engineering (dynamics and control) from Villanova University, Villanova, PA, USA, in 2012.

He is currently an Associate Professor with the Dynamic Systems and Control Laboratory (DSC), Department of Mechanical Engineering, San Diego State University, San Diego, CA, USA. His research interests include robotics, smart flow distribution network, nonlinear dynamics, control theory, optimization, magnetic bearings, and mathematical modeling.

Dr. Naseradinmousavi was a recipient of the John J. Gallen Memorial Alumni Award, 2021. He serves as an Associate Editor of *ASME Letters in Dynamic Systems and Control* and the *Journal of Vibration and Control (JVC)*.



Miroslav Krstić (Fellow, IEEE) is a Distinguished Professor of mechanical and aerospace engineering, holds the Alspach Endowed Chair, and is the Founding Director of the Cymer Center for Control Systems and Dynamics, University California San Diego (UCSD), La Jolla, CA, USA. He also serves as a Senior Associate Vice Chancellor for Research with UCSD. He has coauthored 13 books on adaptive, nonlinear, and stochastic control, extremum seeking, control of PDE systems including turbulent flows, and control of delay systems.

Mr. Krstić has been an elected fellow of seven scientific societies-IFAC, ASME, SIAM, AAAS, IET (U.K.), and AIAA (Associate Fellow)-and as a foreign member of the Serbian Academy of Sciences and Arts and of the Academy of Engineering of Serbia. He won the UC Santa Barbara Best Dissertation Award and the Student Best Paper Awards at CDC and ACC, as a Graduate Student. He has received the SIAM Reid Prize, the ASME Oldenburger Medal, the Nyquist Lecture Prize, the Paynter Outstanding Investigator Award, the Ragazzini Education Award, the Chestnut textbook prize, Control Systems Society Distinguished Member Award, the PECASE, the NSF Career, the ONR Young Investigator Awards, the Axelby and Schuck paper prizes, and the first UCSD Research Award given to an Engineer. He has been awarded the Springer Visiting Professorship at UC Berkeley, the Distinguished Visiting Fellowship of the Royal Academy of Engineering, and the Invitation Fellowship of the Japan Society for the Promotion of Science. He serves as an Editor-in-Chief of *Systems and Control Letters* and has been serving as a Senior Editor in *Automatica* and *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, as an Editor of two Springer book series, and has served as the Vice President for Technical Activities of the IEEE Control Systems Society and as the Chair of the IEEE CSS Fellow Committee.