

Delay-adaptive Control of Nonlinear Systems with Approximate Neural Operator Predictors

Luke Bhan, Miroslav Krstic, Yuanyuan Shi

Abstract—In this work, we propose a rigorous method for implementing predictor feedback controllers in nonlinear systems with unknown and arbitrarily long actuator delays. To address the analytically intractable nature of the predictor, we approximate it using a learned neural operator mapping. This mapping is trained once, offline, and then deployed online, leveraging the fast inference capabilities of neural networks. We provide a theoretical stability analysis based on the universal approximation theorem of neural operators and the transport partial differential equation (PDE) representation of the delay. We then prove, via a Lyapunov-Krasovskii functional, semi-global practical convergence of the dynamical system dependent on the approximation error of the predictor and delay bounds. Finally, we validate our theoretical results using a biological activator/repressor system, demonstrating speedups of 15 times compared to traditional numerical methods.

I. INTRODUCTION

In this work, we extend the concept of neural operator approximate predictors to delay systems with constant but *unknown* delays. The presence of unknown delays presents a significant challenge, as the approximation of the predictor introduces an additive error requiring analysis akin to robust adaptive control. Furthermore, unlike robust adaptive control, the adaptive parameter estimated is the actuator delay which directly impacts system stability. Despite these challenges, we achieve a result similar to [6], ensuring the semi-global practical stability of the feedback system, dependent on the approximation error and an additional error term depending on the delay bounds. This represents the first result for implementing any type of approximate predictor when the delay is constant but unknown.

A. Predictor feedback designs and implementations

Predictor feedback approaches for compensating actuator delays in dynamical systems have been studied for well-over 50 years [13], [15]–[17], [21], [31], [37]. Furthermore, predictor feedback compensation has benefited a variety of applications including traffic [1], [32], aerospace vehicles [12], [40], and robotics [3]–[5]. Despite this, in nonlinear systems, predictor feedback controllers still suffer from implementation limitations as the predictor itself is an implicit ordinary differential equation [2]. To alleviate this issue, the book [18] detailed a variety of numerical schemes combining finite differencing with successive approximations for approximating predictors for both linear and nonlinear systems with corresponding stability guarantees. However,

[18] does not explore the delay-adaptive case and thus no stability results exist for approximating the predictor when the delay is unknown. Furthermore, these schemes suffer from significant computational cost due to the requirement of a small discretization size. Therefore, [6] introduced operator learning to approximate the predictor mapping achieving 1000x speedups compared to numerical solvers. However, [6] was for the simplest case of a nonlinear system - a constant known delay. Thus, we extend [6] to unknown delays.

B. Operator learning in control

We briefly discuss the advances of operator learning as a tool for real-time implementation of controllers. Neural operators were first introduced in [10] and then more recently popularized for learning PDE solutions in [29] and [30]. The goal is to approximate any infinite dimensional operator between function spaces by finite dimensional neural networks. In control theory, operator learning was first introduced in [7], [23] for approximating the gain kernel in linear PDEs, and has now expanded to output-feedback PDEs [39], delayed PDEs systems [36], [38], gain-scheduling [26] and even adaptive control [8], [25]. Notably, [41] presented the first application of operator learning in real-world traffic scenarios, marking a significant milestone for practical deployment. We further review neural operators in Section II-B.

C. Notation

For functions, $u : [0, D] \times \mathbb{R} \rightarrow \mathbb{R}$, we use $u_x(x, t) = \frac{\partial u}{\partial x}(x, t)$ and $u_t(x, t) = \frac{\partial u}{\partial t}(x, t)$ to denote derivatives. We use $C^1([t - D, t]; \mathbb{R}^m)$ to denote the set of functions with continuous first derivatives mapping the interval $[t - D, t]$ to \mathbb{R}^m . For a n -vector, we use $|\cdot|$ for the Euclidean norm. For functions, we define the spatial L^p norms as $\|u(t)\|_{L^p[0, D]} = (\int_0^D |u(x, t)|^p dx)^{\frac{1}{p}}$ for $p \in [1, \infty)$.

II. TECHNICAL BACKGROUND

A. Delay-adaptive control

We study the nonlinear plant

$$\dot{X}(t) = f(X(t), U(t - D)), \quad (1)$$

where $X \in \mathbb{R}^n$, $U \in \mathbb{R}$, $f \in C^2(\mathbb{R}^n \times \mathbb{R}; \mathbb{R}^n)$ such that $f(0, 0) = 0$, and D is an unknown delay within the interval $[\underline{D}, \bar{D}]$ where $\underline{D} > 0$. As standard in the predictor feedback literature [9], we introduce the following assumptions:

Assumption 1: The plant $\dot{X} = f(X, \Omega)$ with Ω scalar is strongly forward complete.

All three authors are with the University of California, San Diego.
{lbhan, mkrstic, yyshi}@ucsd.edu.

Code and datasets available at <https://github.com/lukebhan/Neural-Operator-Delay-Adaptive-Predictor-Feedback>.

Assumption 2: There exists $\kappa \in C^2(\mathbb{R}^n; \mathbb{R})$ such that the feedback law $U(t) = \kappa(X(t))$ guarantees that the delay-free plant is globally exponentially stable.

Note that, under Assumption 2, [19, Theorem 4.14] implies that there exists $\lambda > 0$ and a class C^∞ radially unbounded positive definite function V such that for any $X \in \mathbb{R}^n$, we have

$$\frac{dV}{dX}(X)f(X, \kappa(X)) \leq -\lambda V(X), \quad (2)$$

$$|X|^2 \leq V(X) \leq C_1|X|^2, \quad (3)$$

$$\left| \frac{dV}{dX}(X) \right| \leq C_2|X|, \quad (4)$$

where $C_1, C_2 > 0$.

Assumption 3: Let $f(X, U)$ be the plant dynamics as in (1) and $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}$ be compact domains with bounds \bar{X}, \bar{U} respectively. Then, there exists a constant $C_f(\bar{X}, \bar{U}) > 0$ such that f satisfies the Lipschitz condition

$$|f(X_1, u_1) - f(X_2, u_2)| \leq C_f(|X_1 - X_2| + |u_1 - u_2|), \quad (5)$$

for all $X_1, X_2 \in \mathcal{X}$ and $u_1, u_2 \in \mathcal{U}$.

Assumption 4: The function f , the Jacobian matrix $\partial f / \partial X$, the control law κ and its derivative satisfy the following growth conditions

$$|f(X, U)| \leq M_1(|X| + |U|), \quad \left| \frac{\partial f}{\partial X}(X, U) \right| \leq M_2,$$

$$|\kappa(X)| \leq M_3|X|, \quad \left| \frac{d\kappa}{dX}(X) \right| \leq M_4,$$

for constants $M_1, M_2, M_3, M_4 > 0$.

Assumption 1 ensures the plant (1) does not escape before the input reaches the system at $t = D$. Assumption 2 of global exponential stability is needed for the semi-global results in Section III, though it can be relaxed for a local result [9]. Assumption 3 is the only new assumption compared to [9] and is required for approximating the predictor. Assumption 4 is needed to show global asymptotic stability without any predictor approximation. Assumption 4 makes Assumption 1 redundant and is implied by Lipschitzness, but is included for clarity.

As standard in predictor feedback designs, we reformulate the plant (1) into the following ODE-PDE cascade where the delay is absorbed into the transport PDE with velocity D :

$$\dot{X}(t) = f(X(t), u(0, t)), \quad (6a)$$

$$Du_t(x, t) = u_x(x, t), \quad (6b)$$

$$u(1, t) = U(t). \quad (6c)$$

The representation of the delay through a transport PDE was first introduced in [22] where the analytical solution of the transport PDE recovers the original plant. Namely, we have that

$$u(x, t) = U(t + D(x - 1)), \quad x \in [0, 1], \quad (7)$$

and thus when $x = 0$ in the PDE representation, we recover $U(t - D)$ as in (1). To compensate for the delay, we estimate

the state D timesteps in the future with the predictor given for all $x \in [0, 1]$

$$p(x, t) = X(t + Dx) = X(t) + D \int_0^x f(p(y, t), u(y, t)) dy, \quad (8)$$

which is itself, an implicit ODE whose analytical form is unknown for most nonlinear plant dynamics f .

Then, under the assumption that the delay is known, the stabilizing control law is given by $U(t) = \kappa(p(1, t))$. When the delay is unknown, as in the problem setting of this work, the control law is chosen by

$$U(t) = \kappa(P(1, t)), \quad (9)$$

where the predictor is given by

$$P(x, t) = X(t) + \check{D}(t) \int_0^x f(P(y, t), u(y, t)) dy. \quad (10)$$

with the delay estimate $\check{D}(t)$. Following [9], $\check{D}(t)$ is updated via

$$\dot{\check{D}}(t) = \gamma \text{Proj}_{[\underline{D}, \bar{D}]} \left\{ \check{D}(t), \phi(t) \right\}, \quad (11)$$

$$\phi(t) = - \frac{\int_0^1 (1+x) q_1(x, t) w(x, t) dx}{1 + V(x) + b \int_0^1 (1+x) w(x, t)^2 dx}, \quad (12)$$

where b is a user-specified parameter, V is the positive definite Lyapunov function in (2), (3), (4), w is given by the backstepping transformation

$$w(x, t) = u(x, t) - \kappa(P(x, t)), \quad (13)$$

and the scalar function q_1 is defined as

$$q_1(x, t) = \frac{d\kappa}{dP}(P(x, t)) \Phi(x, 0, t) f(P(0, t), u(0, t)), \quad (14)$$

where Φ is the transition matrix associated with the space-varying time-parameterized equation $(dr/dx)(x) = \check{D}(t)(\partial f / \partial P)(P(x, t), u(x, t))r(x)$.

Then, under the control feedback law (9), [9, Theorem 1] states that the plant is globally asymptotically stable. However, as mentioned, (10) is not analytically known and therefore needs to be approximated for any real-world implementation. Thus, we introduce an approximation for the predictor P and show that, under the universal approximation of neural operators, semi-global practical asymptotic stability of the closed-loop system is achieved.

B. Neural operators

For the readers reference, we briefly review the related background and theoretical results used in the analysis in Section III. Neural operators are finite dimensional approximations of nonlinear operators (e.g. the solution operator to any nonlinear ODE) which map across function spaces. Thus, a neural operator takes a representation of the input function $c(x)$ and its evaluation point x as inputs. Additionally, as input, it takes a value y , which specifies the point to evaluate the target function after applying the operator to c . The neural operator then provides an approximation of the value of the

operator applied to $c(x)$ at the point y in the target function space via finite neural networks.

We now describe neural operators from a rigorous perspective. Namely, let $\Omega_u \subset \mathbb{R}^{d_{u_1}}$, $\Omega_v \subset \mathbb{R}^{d_{v_1}}$ be bounded domains and let $\mathcal{F}_c \subset C^0(\Omega_u; \mathbb{R}^c)$, $\mathcal{F}_v \subset C^0(\Omega_v; \mathbb{R}^v)$ be continuous function spaces. Then, we define a neural operator approximation for the nonlinear operator Ψ as any function satisfying the following form:

Definition 1: [27, Section 1.2] Given the parameter for channel dimension d_c , we call any $\hat{\Psi}$ a neural operator given it satisfies the compositional form $\hat{\Psi} = \mathcal{Q} \circ \mathcal{L}_L \circ \dots \circ \mathcal{L}_1 \circ \mathcal{R}$ where \mathcal{R} is a lifting layer, $\mathcal{L}_l, l = 1, \dots, L$ are the hidden layers, and \mathcal{Q} is a projection layer. That is, \mathcal{R} is given by

$$\mathcal{R} : \mathcal{F}_c(\Omega_u; \mathbb{R}^c) \rightarrow \mathcal{F}_s(\Omega_s; \mathbb{R}^{d_c}), \quad c(x) \mapsto R(c(x), x), \quad (15)$$

where $\Omega_s \subset \mathbb{R}^{d_{s_1}}$, $\mathcal{F}_s(\Omega_s; \mathbb{R}^{d_c})$ is a Banach space for the hidden layers and $R : \mathbb{R}^c \times \Omega_u \rightarrow \mathbb{R}^{d_c}$ is a learnable neural network acting between finite-dimensional Euclidean spaces. For $l = 1, \dots, L$, each hidden layer is given by

$$(\mathcal{L}_l v)(x) := s(W_l v(x) + b_l + (\mathcal{K}_l v)(x)), \quad (16)$$

where weights $W_l \in \mathbb{R}^{d_c \times d_c}$ and biases $b_l \in \mathbb{R}^{d_c}$ are learnable parameters, $s : \mathbb{R} \rightarrow \mathbb{R}$ is a smooth, infinitely differentiable activation function that acts component wise on inputs and \mathcal{K}_l is the nonlocal operator given by

$$(\mathcal{K}_l v)(x) = \int_{\mathcal{X}} K_l(x, y) v(y) dy, \quad (17)$$

where $K_l(x, y)$ is a kernel function containing learnable parameters. Lastly, the projection layer \mathcal{Q} is given by

$$\mathcal{Q} : \mathcal{F}_s(\Omega_s; \mathbb{R}^{d_c}) \rightarrow \mathcal{F}_v(\Omega_v; \mathbb{R}^v), \quad s(x) \mapsto Q(s(x), y), \quad (18)$$

where Q is a finite dimensional neural network from $\mathbb{R}^{d_c} \times \Omega_v \rightarrow \mathbb{R}^v$.

This abstract formulation of neural operators covers a wide range of architectures including the Fourier Neural Operator (FNO) [29] as well as DeepONet [30] where the differentiator between these approaches lies in the implementation of the kernel function K_l . However, in [27], the authors showed that a single hidden layer neural operator with a kernel given by the averaging kernel $K_l(x, y) = 1/|\mathcal{X}|$ where $|\mathcal{X}|$ is the diameter of the domain is sufficient for universal approximation in the following sense

Theorem 1: [27, Theorem 2.1] Let $\Omega_u \subset \mathbb{R}^{d_{u_1}}$ and $\Omega_v \subset \mathbb{R}^{d_{v_1}}$ be two bounded domains with Lipschitz boundary. Let $\Psi : C^0(\overline{\Omega_u}; \mathbb{R}^{d_{u_1}}) \rightarrow C^0(\overline{\Omega_v}; \mathbb{R}^{d_{v_1}})$ be a continuous operator and fix a compact set $K \subset C^0(\overline{\Omega_u}; \mathbb{R}^{d_{u_1}})$. Then for any $\epsilon > 0$, there exists a channel dimension $d_c > 0$ such that a single hidden layer neural operator $\hat{\Psi} : K \rightarrow C^0(\overline{\Omega_v}; \mathbb{R}^{d_{v_1}})$ with kernel function $K_l(x, y) = 1/|\mathcal{X}|$ satisfies

$$\sup_{u \in K} |\Psi(u)(y) - \hat{\Psi}(u)(y)| \leq \epsilon, \quad (19)$$

for all values $y \in \Omega_v$.

The key corollary to Theorem 1 is that, many of the architectures aforementioned (e.g. FNO, Laplace neural operator, DeepONet) contain kernel functions K_l that recover

the averaging kernel and thus are universal under the setting of Theorem 1. Henceforth, for the remainder of this study, we assume that any reference to a neural operator is an architecture of the form given in Definition 1 with a viable kernel function such that Theorem 1 holds. We are now ready to introduce the predictor operator to be approximated.

III. NEURAL OPERATOR APPROXIMATE PREDICTORS

Definition 2: Let $X \in \mathbb{R}^n$, $U \in C^2([0, 1]; \mathbb{R})$, $\varphi \in \mathbb{R}^+$. Then, we define the **predictor operator** as the mapping $\mathcal{P} : (X, U, \varphi) \rightarrow P$ where $P(s) = \mathcal{P}(X, U, \varphi)(s)$ satisfies for all $s \in [0, 1]$,

$$P(s) - X - \varphi \int_0^s f(P(s), U(s)) ds = 0. \quad (20)$$

Notice that, by definition the predictor operator yields the solution to (10) where φ , as an input to the operator, is the estimated delay. Furthermore, in order to approximate this mapping with a neural operator such that Theorem 1 holds, we require that the predictor operator is continuous. Thus, we prove the following Lemma:

Lemma 1: Let Assumption 3 hold. Then, for any $X_1, X_2 \in \mathcal{X}$, $U_1, U_2 \in C^2([0, 1]; \mathcal{U})$, and $\varphi_1, \varphi_2 \in (\underline{D}, \overline{D})$ the predictor operator \mathcal{P} given in Definition 2 satisfies

$$\begin{aligned} & \|\mathcal{P}(X_1, U_1, \varphi_1) - \mathcal{P}(X_2, U_2, \varphi_2)\|_{L^\infty[0,1]} \\ & \leq C_{\mathcal{P}} (|X_1 - X_2| + \|U_1 - U_2\|_{L^\infty[0,1]} + |\varphi_1 - \varphi_2|), \end{aligned} \quad (21)$$

with Lipschitz constant

$$C_{\mathcal{P}} = e^{\overline{D}C_f} \max\{1, \Xi, \overline{D}C_f\}, \quad (22)$$

$$\Xi = C_f \left[\overline{U} + e^{\overline{D}C_f} (\overline{X} + C_f \overline{D} \overline{U}) \right]. \quad (23)$$

Proof: For all $s \in [0, 1]$, let $\bar{P}_1(s) := \mathcal{P}(X_1, U_1, \varphi_1)$ and likewise $\bar{P}_2(s) := \mathcal{P}(X_2, U_2, \varphi_2)$. Note that, for $s \in [0, 1]$ the predictor is uniformly bounded. First, by definition and Assumption 3, we have

$$\begin{aligned} \bar{P}_1(s) &= X_1 + \varphi_1 \int_0^s f(\bar{P}_1(\theta), U_1(\theta)) d\theta \\ &\stackrel{(5)}{\leq} X_1 + \varphi_1 \int_0^s C_f (|\bar{P}_1(\theta)| + |U_1(\theta)|) d\theta \\ &\leq X_1 + \varphi_1 C_f \|U_1\|_{L^\infty[0,1]} + \varphi_1 C_f \int_0^1 |\bar{P}_1(\theta)| d\theta \end{aligned} \quad (24)$$

Applying Gronwall's inequality yields

$$|\bar{P}_1(s)| \leq e^{\overline{D}C_f} (\overline{X} + C_f \overline{D} \overline{U}). \quad (25)$$

Then, applying (25) yields the following calculation

$$\begin{aligned}
\bar{P}_1(s) - \bar{P}_2(s) &= X_1 - X_2 + \varphi_1 \int_{-1}^s f(\bar{P}_1(\theta), U_1(\theta)) d\theta \\
&\quad - \varphi_2 \int_{-1}^s f(\bar{P}_2(\theta), U_2(\theta)) d\theta \\
&\leq |X_1 - X_2| \\
&\quad + (\varphi_1 - \varphi_2) \int_{-1}^s f(\bar{P}_1(\theta), U_1(\theta)) d\theta \\
&\quad + \varphi_2 \int_{-1}^s \left(f(\bar{P}_1(\theta), U_1(\theta)) \right. \\
&\quad \left. - f(\bar{P}_2(\theta), U_2(\theta)) \right) d\theta \\
&\stackrel{(5)}{\leq} |X_1 - X_2| + (\varphi_1 - \varphi_2) C_f \|U_1\|_{L^\infty[0,s]} \\
&\quad + (\varphi_1 - \varphi_2) C_f \int_{-1}^s |P(\theta)| d\theta \\
&\quad + \varphi_2 C_f \int_{-1}^s |\bar{P}_1(\theta) - P_2(\theta)| \\
&\quad + |U_1(\theta) - U_2(\theta)| d\theta \\
&\stackrel{(25)}{\leq} |X_1 - X_2| \\
&\quad + |\varphi_1 - \varphi_2| C_f \left[\bar{U} + e^{C_f \bar{D}} (\bar{X} + C_f \bar{D} \bar{U}) \right] \\
&\quad + \bar{D} C_f \|U_1 - U_2\|_{L^\infty[-1,s]} \\
&\quad + \bar{D} C_f \int_{-1}^s |\bar{P}_1(\theta) - \bar{P}_2(\theta)| d\theta. \quad (26)
\end{aligned}$$

Applying Gronwall's again yields the desired result. \blacksquare

Given the continuity in Lemma 1, we can then apply Theorem 1 to the predictor operator yielding the existence of an arbitrarily close neural operator approximation:

Theorem 2: Fix a compact set $K \subset \mathcal{X} \times C^2([0, 1]; \mathcal{U}) \times \mathcal{D}$. Then, for all $\bar{X}, \bar{U}, \bar{D}, \epsilon > 0$, there exists a neural operator approximation $\hat{\mathcal{P}} : K \rightarrow C^1([0, 1]; \mathbb{R}^n)$ such that

$$\sup_{(X,U,\varphi) \in K} |\mathcal{P}(X, U, \varphi)(s) - \hat{\mathcal{P}}(X, U, \varphi)(s)| < \epsilon, \quad (27)$$

for all $s \in [0, 1]$.

Now, we have established the existence of an arbitrarily close approximate predictor, but we did not discuss the detail on the number of parameters or data required to obtain this bound. Such analysis is beyond the goal of this paper, but we refer the reader to [33], [20], [28] for further details. Further, given the approximation is arbitrarily close in ϵ , this perturbation will affect the stability of the system compared to the global asymptotic result conducted with the exact predictor (which can never be implemented in practice). Therefore, in the next section, we identify the exact effect of the approximate predictor on the overall feedback loop.

IV. STABILITY ANALYSIS UNDER APPROXIMATE PREDICTORS

We are now ready to analyze the plant (6a), (6b), (6c) under the controller with the neural operator approximated

predictor

$$U(t) = \kappa \left(\hat{\mathcal{P}} \left(X(t), T_D(t)U, \check{D}(t) \right) \right), \quad (28)$$

where $T_D(t)$ is the historical control operator which applied to a function U yields

$$(T_{\check{D}}(t)U)(x) := U(t - D(t)(x - 1)), \quad x \in [0, 1]. \quad (29)$$

The operator $T_D(t)$ shifts the function U back by $t - D(x - 1)$ units such that, we have $u(x, t) = (T_D(t)U)(x)$ and therefore the resulting output of the exact predictor operator satisfies (10). Notice that, in this case, we consider the full actuator measurement such that $u(x, t)$ is always known which is not necessarily guaranteed for every application. However, as shown in Section V, this is a reasonable assumption for biological systems of which we study in this work.

We are now ready to present the main result of the paper.

Theorem 3: Let the system (6a), (6b), (6c) satisfy Assumptions 1, 2, 3, 4. Define the functional

$$\Gamma(t) = |X(t)|^2 + \int_{t-D}^t U(\theta)^2 d\theta + |\check{D}(t)|^2, \quad (30)$$

where $\check{D}(t) = D - \check{D}(t)$. Then, there exists constants γ^* , $b^*(\bar{D}, C_f)$, $\bar{\Gamma}(\bar{X}, \bar{U}, \bar{D}) > 0$, $\beta_1^* \in \mathcal{KL}$ and class \mathcal{K}_∞ functions $\alpha_1^*, \alpha_2^*, \alpha_3^*, \alpha_4^*, \alpha_5^*$ such that if $\gamma < \gamma^*$, $b > b^*$, $\epsilon < \epsilon^*$ where

$$\epsilon^* = (\alpha_1^*)^{-1} (\bar{\Gamma} - \alpha_2^*(\bar{\Delta D})), \quad (31)$$

where $\bar{\Delta D} = \bar{D} - \underline{D}$ and the initial state is constrained to

$$\Gamma(0) \leq \alpha_3^* (\bar{\Gamma} - \alpha_1^*(\epsilon) - \alpha_2^*(\bar{\Delta D})), \quad (32)$$

then,

$$\Gamma(t) \leq \alpha_3^* (\Gamma(0)) + \alpha_1^*(\epsilon) + \alpha_2^*(\bar{\Delta D}), \quad (33)$$

and

$$\begin{aligned}
|X(t)|^2 &\leq \beta_1^* (|X(0)|^2, t) + \alpha_4^*(\bar{\Delta D}) + \alpha_5^*(\epsilon), \quad (34) \\
\|u(t)\|_{L^2[0,D]}^2 &\leq \beta_1^* (\|u(0)\|_{L^2[-D,0]}^2, t) + \alpha_4^*(\bar{\Delta D}) + \alpha_5^*(\epsilon). \quad (35)
\end{aligned}$$

Notice that Theorem 3 is weaker than the result in [Theorem 7] because it depends on the delay projection bounds \bar{D}, \underline{D} . This is expected since the error introduced by the predictor is additive, leading to complications similar to those in robust adaptive control, where regulation depends on projector bounds (See [14]).

Additionally, it may seem counterintuitive that ϵ^* increases with $\bar{\Gamma}$, but this is expected: a larger radius of states allows for a larger transient, thus loosening the ϵ^* bound.

Finally, we briefly compare Theorem 3 with other adaptive control implementations using neural operators in PDEs. In [8, Theorem 4] and [25, Theorem 4], the authors approximate the gain kernel (not the controller), which introduces a multiplicative error and leads to global stability. However, this requires approximating the time derivative of the update parameter. In contrast, in Theorem 3, our approximation of P is similar to approximating the direct control law, as in

[7, Theorem 3] resulting in an additive error and yielding a semiglobal practical result, without the need to approximate the time derivative of P_t .

Proof: We first introduce a backstepping transform, that under the feedback law (28), transforms the system into a target system that we then use for our stability analysis.

Lemma 2: Under the backstepping transformation with the exact predictor given by

$$w(x, t) = u(x, t) - \kappa(P(x, t)), \quad (36)$$

the system (6a), (6b), (6c) with control law given by (28) is transformed into

$$\dot{X}(t) = f(X(t), \kappa(X(t)) + w(0, t)), \quad (37a)$$

$$Dw_t = w_x - \tilde{D}(t)q_1(x, t) - D\dot{\tilde{D}}(t)q_2(x, t), \quad (37b)$$

$$w(1, t) = \kappa(P(x, t)) - \kappa(\hat{p}(x, t)), \quad (37c)$$

where $\tilde{D} = D - \dot{\tilde{D}}(t)$, $\hat{p}(x, t) = P_1(x)$, and q_2 is given by

$$q_2 = \frac{\partial \kappa}{\partial P}(P(x, t)) \int_0^x \Phi(x, y, t) f(P(y, t), \kappa(P(y, t)) + w(y, t)) dy. \quad (38)$$

Since we used the exact backstepping transform in (36), the proof of Lemma 2 is exactly the same as the proof of [9, Lemma 1] except the boundary term at $x = 1$ which is obtained via direct substitution of $U(t) = \kappa(\hat{P}(\cdot)) - \kappa(P(\cdot))$ in (36).

To analyze (37a), (37b), (37c), we introduce the following Lyapunov-Krasovskii functional

$$W(t) = D \log N(t) + \frac{b}{\gamma} \tilde{D}(t)^2 \quad (39)$$

$$N(t) = 1 + V(x) + b \int_0^1 (1+x)w(x, t)^2 dx. \quad (40)$$

Taking the time derivative and substituting w_t yields

$$\begin{aligned} \dot{W}(t) = & \frac{1}{N(t)} \left(D \frac{\partial V(X)}{\partial X} \left[f(X(t), \kappa(X(t)) + w(0, t)) \right] \right) \\ & + \frac{2b}{N(t)} \int_0^1 (1+x)w(x, t) \left(w_x(x, t) - \tilde{D}(t)q_1(x, t) \right. \\ & \left. - D\dot{\tilde{D}}(t)q_2(x, t) \right) dx - \frac{2b}{\gamma} \tilde{D}\dot{\tilde{D}}(t). \end{aligned} \quad (41)$$

Using the Lipschitzness of f in Assumption 3 and the Lyapunov condition in Assumption 2, we have

$$\begin{aligned} \dot{W}(t) \leq & \frac{1}{N(t)} \left(-D\lambda|X(t)|^2 + DC_2C_f|X(t)||w(0, t)| \right) \\ & + \frac{2b}{N(t)} \int_0^1 (1+x)w(x, t) \left(w_x(x, t) - \tilde{D}(t)q_1(x, t) \right. \\ & \left. - D\dot{\tilde{D}}(t)q_2(x, t) \right) dx - \frac{2b}{\gamma} \tilde{D}\dot{\tilde{D}}(t). \end{aligned} \quad (42)$$

Distributing the second term, applying integration by parts,

and substituting (12) yields

$$\begin{aligned} \dot{W}(t) \leq & \frac{1}{N(t)} \left(-D\lambda|X(t)|^2 + DC_2C_f|X(t)||w(0, t)| \right. \\ & + 2bw^2(1, t) - 2bw^2(0, t) - 2b\|w(t)\|_2^2 \\ & + 2bD\dot{\tilde{D}}(t) \int_0^1 (1+x)w(x, t)q_2(x, t)dx \\ & \left. + \frac{2b}{\gamma} \tilde{D}(t)(\gamma\phi(t) - \dot{\tilde{D}}(t)) \right). \end{aligned} \quad (43)$$

Using [9, Lemma 2 and 3] which states that there exists $M_5, M_6, M_7 > 0$ such that

$$|P(x, t)| \leq M_5(|X| + \|w(t)\|_2), \quad (44)$$

$$2bD \left| \int_0^1 (1+x)q_2(x, t)w(x, t)dx \right| \leq M_6(|X|^2 + \|w(t)\|_2^2), \quad (45)$$

$$\left| \dot{\tilde{D}}(t) \right| \leq \gamma M_7, \quad (46)$$

in conjunction with Young's inequality yields

$$\begin{aligned} \dot{W}(t) \leq & \frac{1}{N(t)} \left(-\eta(|X(t)|^2 + \|w(t)\|_2^2) \right. \\ & - \left(2b - \frac{C_2^2 DC_f^2}{2\lambda} \right) w^2(0, t) \\ & + \gamma M_6 M_7 \left(|X(t)|^2 + \|w(t)\|_2^2 \right) \\ & \left. + 2bw^2(1, t) \right), \end{aligned} \quad (47)$$

where $\eta := \min(D\lambda/2, 2b)$. Thus, given that

$$b > \frac{C_2^2 \bar{D} C_f^2}{4\lambda} =: b^*, \quad (48)$$

$$\gamma < \frac{\eta}{M_6 M_7} =: \gamma^*, \quad (49)$$

guarantees there exists $C > 0$ such that

$$\begin{aligned} \dot{W}(t) \leq & -\frac{C}{N(t)} (|X(t)|^2 + \|w(t)\|_2^2) \\ & + \frac{2b}{N(t)} (\kappa(P(x, t)) - \kappa(\hat{p}(x, t)))^2. \end{aligned} \quad (50)$$

Now, using the definition of $W(t)$ and Assumption 2, we have that

$$e^{W(t)/D - \frac{b}{\gamma \bar{D}} (\bar{D}(t))^2} - 1 \leq \max\{C_1, 2b\} (|X(t)|^2 + \|w(t)\|_2^2), \quad (51)$$

Using reverse Young's inequality (See Appendix A), we have

$$e^{W(t)/D} e^{-\frac{b}{\gamma \bar{D}} (\bar{D}(t))^2} \geq 2e^{W(t)/(2D)} - e^{\frac{b}{\gamma \bar{D}} \bar{D}(t)^2} \quad (52)$$

yielding

$$\begin{aligned} 2e^{W(t)/D} - e^{\frac{b}{\gamma \bar{D}} (\bar{D}(t))^2} - 1 \\ \leq \max\{C_1, 2b\} (|X(t)|^2 + \|w(t)\|_2^2), \end{aligned} \quad (53)$$

which then results in

$$-(|X(t)|^2 + \|w(t)\|_2^2) \leq \frac{1}{\max\{C_1, 2b\}} \left(-2e^{W(t)/D} + e^{\frac{b}{\gamma D}(\bar{D}(t))^2} + 1 \right). \quad (54)$$

Substituting into (50) yields

$$\begin{aligned} \dot{W}(t) \leq & -\frac{C}{\max\{C_1, 2b\}N(t)} 2e^{\frac{W(t)}{D}} \\ & + \left[\frac{C}{\max\{C_1, 2b\}N(t)} \left(1 + e^{\frac{b}{D\gamma}(\bar{\Delta D})^2} \right) \right. \\ & \left. + b(\kappa(P(x, t)) - \kappa(\hat{p}(x, t)))^2 \right]. \quad (55) \end{aligned}$$

Then, by Assumption 4, we have that for all $x \in [0, 1]$, $t \geq 0$

$$\begin{aligned} (\kappa(P(x, t)) - \kappa(\hat{p}(x, t)))^2 & \leq M_3^2 |P(x, t) - \hat{p}(x, t)|^2 \\ & \leq M_3^2 \epsilon^2. \quad (56) \end{aligned}$$

Following [24, Theorem C.3], we have that there exists functions $\beta_1 \in \mathcal{KL}$ and $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ such that

$$\begin{aligned} W(t) & \leq \beta_1(W(0), t) + \alpha_1(\bar{\Delta D}) \\ & \quad + \alpha_2 \left(\sup_{0 \leq \tau \leq t} (\kappa(P(x, t)) - \kappa(\hat{p}(x, t)))^2 \right) \\ & \leq \beta_1(W(0), t) + \alpha_1(\bar{\Delta D}) + \alpha_3(\epsilon), \quad (57) \end{aligned}$$

when ϵ and $\bar{\Delta D}$ is small enough relative to all possible values of W (i.e. $\epsilon + (\bar{\Delta D}) \leq \alpha_4(W(t)) \leq \alpha_5(\bar{X} + \bar{U} + \bar{D})$ where $\alpha_4, \alpha_5 \in \mathcal{K}_\infty$). To obtain an estimate on Γ , note that from [9, Eqn. (47), (48)], we have that

$$\Gamma(t) \leq \left(D \left(r_1 + \frac{r_2}{b} \right) + 1 + \frac{\gamma D}{b} \right) \left(e^{\frac{W(t)}{D}} - 1 \right), \quad (58)$$

$$W(t) \leq D \left(c_1 + 2b \left(s_1 + \frac{s_2}{D} \right) + \frac{b}{\gamma D} \right) \Gamma(t). \quad (59)$$

Therefore, substituting in the stability estimate in (57), we obtain the exists of class K^∞ functions $\alpha_1^*, \alpha_2^*, \alpha_3^*$ such that

$$\Gamma(t) \leq \alpha_1^*(\Gamma(0)) + \alpha_2^*(\epsilon) + \alpha_3^*(\bar{\Delta D}), \quad (60)$$

when $\Gamma(0) \leq (\alpha_1^*)^{-1}(\bar{\Gamma} - \alpha_2^*(\epsilon) - \alpha_3^*(\bar{\Delta D}))$ where $\bar{\Gamma} = \bar{X} + \bar{U} + \bar{D}$.

To show obtain a bound on $X(t)$, note that we have

$$|X(t)|^2 \leq V(t) \leq \left(e^{\frac{W(t)}{D}} - 1 \right). \quad (61)$$

Using the stability estimate on $W(t)$ in (57), along with the inequalities (54), (61) we have that there exists $\beta_2 \in \mathcal{KL}$, $\alpha_6, \alpha_7 \in \mathcal{K}_\infty$ such that

$$|X(t)|^2 \leq \beta_2(|X(0)|^2, t) + \alpha_6(\bar{\Delta D}) + \alpha_7(\epsilon). \quad (62)$$

Similarly, to show a stability estimate on $U(t)$, notice that (44) and Assumption 4 implies that there exists constants $r_1, r_2, s_1, s_2 > 0$ such that

$$\|u(t)\|_2^2 \leq r_1 |X(t)|^2 + r_2 \|w(t)\|_2^2, \quad (63)$$

$$\|w(t)\|_2^2 \leq s_1 |X(t)|^2 + s_2 \|u(t)\|_2^2. \quad (64)$$

Thus, using (63), (64), along with the definition of $W(t)$, we have that

$$\begin{aligned} \int_{t-D}^t U(s)^2 ds & = D \|u(t)\|_2^2 \leq D(r_1 |X(t)|^2 + r_2 \|w(t)\|_2^2) \\ & \leq D \left(r_1 + \frac{r_2}{b} \right) \left(e^{\frac{W(t)}{D}} - 1 \right). \quad (65) \end{aligned}$$

Then, repeating the argument above with the stability estimate in (57) along with the inequalities (54), (65), we have there exists $\beta_3 \in \mathcal{KL}$, $\alpha_8, \alpha_9 \in \mathcal{K}_\infty$ such that

$$\|u(t)\|_{L^2[0, D]}^2 \leq \beta_3(\|u(0)\|_{L^2[-D, 0]^2}, t) + \alpha_8(\bar{\Delta D}) + \alpha_9(\epsilon). \quad (66)$$

Letting $\beta_1^*(\cdot, \cdot) = \max\{\beta_2(\cdot, \cdot), \beta_3(\cdot, \cdot)\}$, $\alpha_4^*(\cdot) = \max\{\alpha_6(\cdot), \alpha_8(\cdot)\}$ and $\alpha_5^*(\cdot) = \max\{\alpha_7(\cdot), \alpha_9(\cdot)\}$ completes the result. \blacksquare

V. NUMERICAL EXPERIMENTS

For the simulation analysis, we provide all code, numerical parameters, and datasets publicly on Github (<https://github.com/lukebhan/Neural-Operator-Delay-Adaptive-Predictor-Feedback>).

For numerical validation, we consider the biological system in [11] consisting of two proteins. The activator protein that promotes expression of itself and a repressor protein that represses the expression of the activator. Such biological clocks play fundamental roles in cell physiology [34] and control of such systems is of valuable interest to synthetic biologists for the design of new medicines. The system dynamics manifest as

$$\dot{x}_1 = -x_1 + f_1(x_1, x_2) + U(t - D), \quad (67)$$

$$\dot{x}_2 = -\frac{x_2}{2} + f_2(x_1), \quad (68)$$

where x_1 is the concentration of the activator protein, x_2 is the concentration of the repressor protein and f_1, f_2 are Hill functions given by

$$f_1(x_1, x_2) = \frac{K_1 x_1^2 + K_a}{1 + x_1^2 + x_2^2}, \quad (69)$$

$$f_2(x_1) = \frac{K_2 x_1^2 + K_b}{1 + x_1^2}, \quad (70)$$

with $K_1 = K_2 = 300$, $K_a = 0.04$, $K_b = 0.004$. The input U controls the activator protein concentration with a delay due to the time to pass through an inlet pipe. Additionally, we assume the pipe is equipped with a UV spectrometer such that we can measure the presence of the activator protein along the pipe and thus the distributed input $u(\cdot, t)$ is known. The goal is to stabilize the system to the unstable target equilibrium (x_1^*, x_2^*) .

A control law satisfying Assumption 2 is given by

$$\kappa(X(t)) = -f_1(x_1, x_2) + f_1(x_1^*, x_2^*), \quad (71)$$

where $(x_1^*, x_2^*) = (0.0939, 5.2525)$ is an unstable equilibrium point of the system. The associated Lyapunov

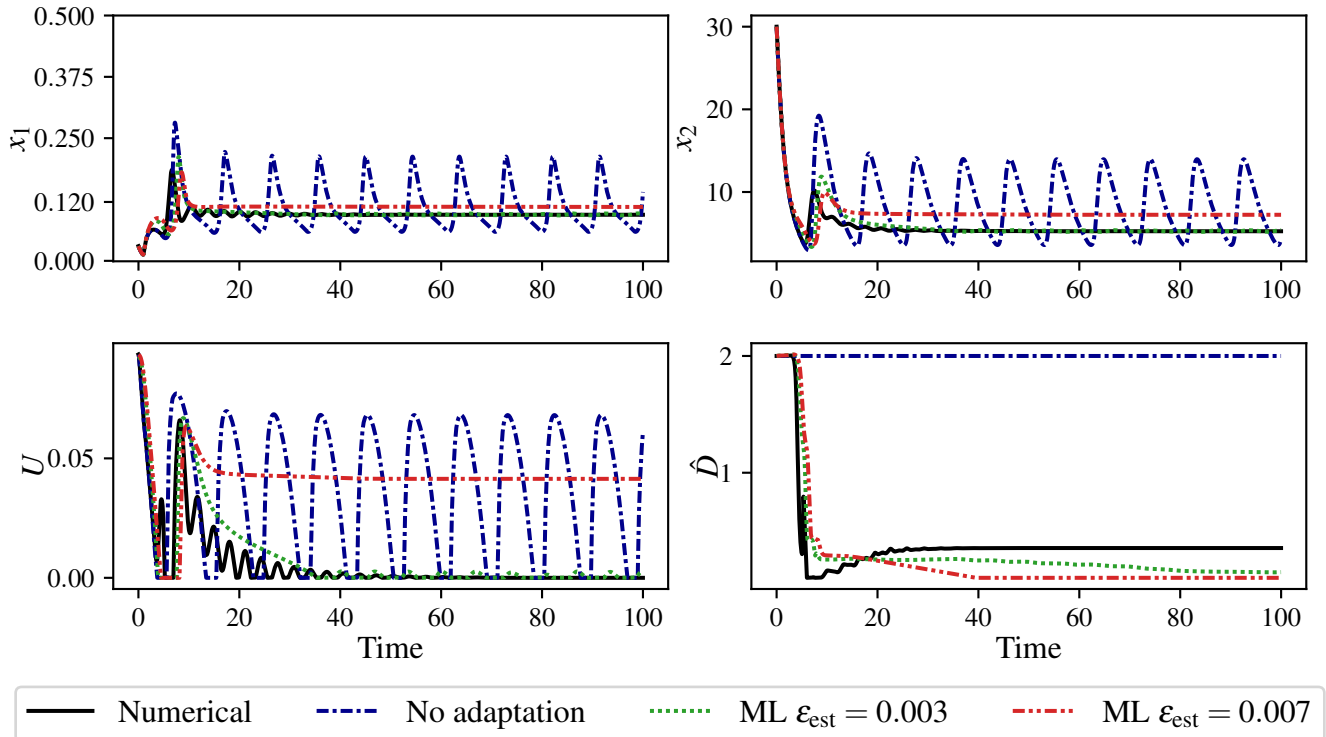


Fig. 1. Simulation of the plant (1) with various approximate predictors. The initial state is $X(0) = [0.03, 30]$, delay is $D = 1$, $\hat{D}(0) = 2$, $\gamma = 1000$, $b = 1$. The black line indicates the numerical predictor, the blue-line is the predictor without delay-adaptation, the red line indicates the DeepONet predictor with higher error and the green line indicates the DeepONet predictor trained to optimality.

function is given by $V = (X - X^*)^T(X - X^*)$ and note the system experiences a limit cycle [9, Figure 2] when an open-loop control law $U(\cdot) = 0$ is used.

To train a neural operator approximate predictor, we generate a dataset consisting of 5000 instances where each instance is sampled from simulating the system with a numerical implementation of the predictor [18] with various initial conditions and initial delay estimations (See code for parameters; Dataset generation took 60 minutes). We then train two different neural operator architectures (DeepONet, FNO) requiring approximately 10 minutes on a Nvidia 3090Ti GPU. Both architectures performed similarly, but the speedup of DeepONet was larger as given in Table I.

We present the simulation results of the dynamical system in Figure 1 with the DeepONet predictor. To highlight the effect of ϵ , we consider two approximate predictors where one has the training stopped early and thus contains a large ϵ -error and the other is trained to optimality. In the case of early stopping, the system converges around the equilibrium, but with a larger radius than the predictor trained to optimality. Furthermore, despite the smaller error, the predictor trained to optimality does not fully converge to the setpoint, but converges to the point $(0.094, 5.39)$ (Recall $(x_1^*, x_2^*) = (0.0939, 5.2525)$) due to the uniform approximation error ϵ . In both cases, we see that the effect of the training error ϵ directly impacts the radius of convergence around the target equilibrium setpoint as expected given in Theorem 3. Furthermore, note that the delays do not converge to D ,

TABLE I
COMPUTATION TIME FOR VARIOUS APPROXIMATE PREDICTORS
AVERAGED OVER 1000 SAMPLES (SECONDS).

Step size (dx)	Numerical ↓	DeepONet ↓	FNO ↓	DeepONet Speedup ↑	FNO Speedup ↑
0.01	1.601	0.496	1.331	3.22x	1.20x
0.005	3.295	0.587	1.440	5.61x	2.29x
0.001	18.197	1.212	2.108	15.01x	8.63x

but close enough to ensure stability. This is a well-known phenomena of adaptive control where the global goal of the scheme is *not parameter convergence*, but is instead stability [24].

Lastly, we investigated the speedup of the neural operator approximate predictors. At all discretizations, the ML predictor is faster than the numerical approach with the largest speedup of 15x under the smallest discretization size ($dx = 0.001$). We note that the discretization size must be less than 0.005 numerical stabilization (Using an Euler approximation scheme). This is less than the 100x speedup achieved in [6, Table 1], but is attributed to the fact the dynamics f are much simpler to simulate in this example compared to the robotic manipulator in [6]. If one applied the same methodology with more expensive forward dynamics f , one should expect to see even larger speedup over the numerical method under a neural operator approximate predictor.

VI. CONCLUSION

In this paper, we presented the first result studying neural operator approximate predictors when the delay is *unknown* and therefore estimated online. In particular, we formalized and proved the existence of an arbitrary close neural operator approximation of the predictor operator which was then used in to prove semi-global ϵ -practical asymptotic stability via a Lyapunov-Kasovskii functional. We emphasize that our theoretical result is applicable to any *black-box* approximate predictor satisfying a uniform error bound. Lastly, we concluded by training a neural operator predictor to compensate for a input-lag in a biological protein activator/repressor system achieving speedups on the magnitude of 15x. In the future, we hope to explore the robustness of neural operator approximate predictors for more challenging delay problems such as state-dependent and unknown noisy delays.

REFERENCES

- [1] N. Bekiaris-Liberis and M. Krstic, "Compensation of state-dependent input delay for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 2, pp. 275–289, 2013.
- [2] —, *Nonlinear Control Under Nonconstant Delays*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2013. [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/1.9781611972856>
- [3] —, "Robustness of nonlinear predictor feedback laws to time- and state-dependent delay perturbations," *Automatica*, vol. 49, no. 6, pp. 1576–1590, 2013. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109813001398>
- [4] —, "Predictor-feedback stabilization of multi-input nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 516–531, 2017.
- [5] A. Bertino, P. Naseradinmousavi, and M. Krstić, "Delay-adaptive control of a 7-DOF robot manipulator: Design and experiments," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 6, pp. 2506–2521, 2022.
- [6] L. Bhan, P. Qin, M. Krstic, and Y. Shi, "Neural operators for predictor feedback control of nonlinear delay systems," 2024. [Online]. Available: <https://arxiv.org/abs/2411.18964>
- [7] L. Bhan, Y. Shi, and M. Krstic, "Neural operators for bypassing gain and control computations in PDE backstepping," *IEEE Transactions on Automatic Control*, 2023.
- [8] —, "Adaptive control of reaction–diffusion PDEs via neural operator-approximated gain kernels," *Systems & Control Letters*, vol. 195, p. 105968, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167691124002561>
- [9] D. Bresch-Pietri and M. Krstic, "Delay-Adaptive Control for Nonlinear Systems," *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1203–1218, 2014.
- [10] T. Chen and H. Chen, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 911–917, 1995.
- [11] D. Del Vecchio, "Design and analysis of an activator-repressor clock in *e. coli*," in *2007 American Control Conference*, 2007, pp. 1589–1594.
- [12] N. Espitia and W. Perruquetti, "Predictor-feedback prescribed-time stabilization of LTI systems with input delay," *IEEE Transactions on Automatic Control*, vol. 67, no. 6, pp. 2784–2799, 2022.
- [13] M. A. Henson and D. E. Seborg, "Time delay compensation for nonlinear processes," *Industrial & engineering chemistry research*, vol. 33, no. 6, pp. 1493–1500, 1994.
- [14] F. IKHOUANE and M. KRSTIĆ, "Adaptive backstepping with parameter projection: Robustness and asymptotic performance," *Automatica*, vol. 34, no. 4, pp. 429–435, 1998. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109897001842>
- [15] M. Jankovic, "Control Lyapunov-Razumikhin functions and robust stabilization of time delay systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 7, pp. 1048–1060, 2001.
- [16] —, "Control of nonlinear systems with time delay," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, vol. 5, 2003, pp. 4545–4550 Vol.5.
- [17] I. Karafyllis, "Finite-time global stabilization by means of time-varying distributed delay feedback," *SIAM Journal on Control and Optimization*, vol. 45, no. 1, pp. 320–342, 2006. [Online]. Available: <https://doi.org/10.1137/040616383>
- [18] I. Karafyllis and M. Krstic, *Predictor Feedback for Delay Systems: Implementations and Approximations*. Birkhäuser Cham, 2017.
- [19] H. K. Khalil, *Nonlinear systems*. Upper Saddle River, N.J.: Prentice Hall, 2002.
- [20] N. B. Kovachki, S. Lanthaler, and H. Mhaskar, "Data Complexity Estimates for Operator Learning," 2024.
- [21] C. Kravaris and R. A. Wright, "Deadline compensation for nonlinear processes," *AIChE Journal*, vol. 35, no. 9, pp. 1535–1542, 1989. [Online]. Available: <https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.690350914>
- [22] M. Krstic, "Input Delay Compensation for Forward Complete and Strict-Feedforward Nonlinear Systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 2, pp. 287–303, 2010.
- [23] M. Krstic, L. Bhan, and Y. Shi, "Neural operators of backstepping controller and observer gain functions for reaction–diffusion PDEs," *Automatica*, vol. 164, p. 111649, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0005109824001420>
- [24] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*, 1st ed. USA: John Wiley & Sons, Inc., 1995.
- [25] M. Lamarque, L. Bhan, Y. Shi, and M. Krstic, "Adaptive neural-operator backstepping control of a benchmark hyperbolic PDE," 2024. [Online]. Available: <https://arxiv.org/abs/2401.07862>
- [26] M. Lamarque, L. Bhan, R. Vazquez, and M. Krstic, "Gain scheduling with a neural operator for a transport PDE with nonlinear recirculation," *IEEE Transactions on Automatic Control*, pp. 1–8, 2025.
- [27] S. Lanthaler, Z. Li, and A. M. Stuart, "Nonlocality and Nonlinearity Implies Universality in Operator Learning," 2024. [Online]. Available: <https://arxiv.org/abs/2304.13221>
- [28] S. Lanthaler, A. M. Stuart, and M. Trautner, "Discretization Error of Fourier Neural Operators," 2024.
- [29] Z. Li, N. B. Kovachki, K. Azizadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier Neural Operator for Parametric Partial Differential Equations," in *International Conference on Learning Representations*, 2021. [Online]. Available: <https://openreview.net/forum?id=c8P9NQVtmnO>
- [30] L. Lu, P. Jin, G. Pang, Z. Zhang, and G. E. Karniadakis, "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators," *Nature Machine Intelligence*, vol. 3, no. 3, pp. 218–229, Mar 2021. [Online]. Available: <https://doi.org/10.1038/s42256-021-00302-5>
- [31] F. Mazenc and P.-A. Bliman, "Backstepping design for time-delay nonlinear systems," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, vol. 5, 2003, pp. 4551–4556 Vol.5.
- [32] T. G. Molnar, A. K. Kiss, A. D. Ames, and G. Orosz, "Safety-critical control with input delay in dynamic environment," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 4, pp. 1507–1520, 2023.
- [33] A. Mukherjee and A. Roy, "Size Lowerbounds for Deep Operator Networks," *Transactions on Machine Learning Research*, 2024. [Online]. Available: <https://openreview.net/forum?id=RwmWODTNFE>
- [34] J. R. Pomeroy, E. D. Sontag, and J. E. Ferrell, "Building a cell cycle oscillator: hysteresis and bistability in the activation of *cdc2*," *Nature Cell Biology*, vol. 5, no. 4, pp. 346–351, Apr 2003. [Online]. Available: <https://doi.org/10.1038/ncb954>
- [35] ProofWiki contributors, "Reverse young's inequality for products," https://proofwiki.org/wiki/Reverse_Young%27s_Inequality_for_Products, 2025, accessed: 2025-08-20.
- [36] J. Qi, J. Zhang, and M. Krstic, "Neural operators for PDE backstepping control of first-order hyperbolic PIDE with recycle and delay," *Systems & Control Letters*, vol. 185, p. 105714, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167691124000021>
- [37] O. J. Smith, "Closer control of loops with dead time," *Chemical engineering progress*, vol. 53, pp. 217–219, 1957.
- [38] S. Wang, M. Diagne, and M. Krstić, "Deep learning of delay-

- compensated backstepping for reaction-diffusion PDEs,” *IEEE Transactions on Automatic Control*, pp. 1–8, 2025.
- [39] Y. Xiao, Y. Yuan, B. Luo, and X. Xu, “Neural operators for robust output regulation of hyperbolic PDEs,” *Neural Networks*, vol. 179, p. 106620, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0893608024005446>
- [40] X. Xu, L. Liu, and G. Feng, “Stability and stabilization of infinite delay systems: A Lyapunov-based approach,” *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4509–4524, 2020.
- [41] Y. Zhang, R. Zhong, and H. Yu, “Neural operators for boundary stabilization of stop-and-go traffic,” in *Proceedings of the 6th Annual Learning for Dynamics & Control Conference*, ser. Proceedings of Machine Learning Research, A. Abate, M. Cannon, K. Margellos, and A. Papachristodoulou, Eds., vol. 242. PMLR, 15–17 Jul 2024, pp. 554–565. [Online]. Available: <https://proceedings.mlr.press/v242/zhang24c.html>

APPENDIX A. REVERSE YOUNG’S INEQUALITY

Lemma 3: Given $a, b \geq 0$ and p, q such that

$$\frac{1}{p} - \frac{1}{q} = 1, \quad (72)$$

then

$$\frac{a^p}{p} - \frac{b^{-q}}{q} \leq ab. \quad (73)$$

Proof: See [35]. ■