Digital Chemical Engineering 7 (2023) 100084

Contents lists available at ScienceDirect

Digital Chemical Engineering

journal homepage: www.elsevier.com/locate/dche

Best practices and methods

Machine learning-based predictive control of nonlinear time-delay systems: Closed-loop stability and input delay compensation

Aisha Alnajdi [a,d], Atharva Suryavanshi [b], Mohammed S. Alhajeri [b,c], Fahim Abdullah [b], Panagiotis D. Christofides [a,b,*]

[a] Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095-1592, USA
[b] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA
[c] Department of Chemical Engineering, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait
[d] Department of Electrical Engineering, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait

ARTICLE INFO

Keywords:
Nonlinear time-delay systems
Recurrent neural networks
Long short-term memory
Machine learning
Process control
Model predictive control
Nonlinear systems

ABSTRACT

The purpose of this work is to study machine-learning-based model predictive control of nonlinear systems with time-delays. The proposed approach involves initially building a machine learning model (i.e., Long Short Term Memory (LSTM)) to capture the process dynamics in the absence of time delays. Then, an LSTM-based model predictive controller (MPC) is designed to stabilize the nonlinear system without time delays. Closed-loop stability results are then presented, establishing robustness of this LSTM-based MPC towards small time-delays in the states. To handle input delays, we design an LSTM-based MPC with an LSTM-based predictor that compensates for the effect of input delays. The predictor is used to predict future states using the process measurement, and then the predicted states are used to initialize the LSTM-based MPC. Stabilization of the time-delay system with both state and input delays around the steady state is achieved through the featured design. The approach is applied to a chemical process example, and its performance and robustness properties are evaluated via simulations.

1. Introduction

Machine learning algorithms have generated considerable interest in the field of control of nonlinear process systems. This is because of their ability to capture the system's dynamics and to model large-scale, complex, nonlinear systems. Moreover, the existence of large data sets, powerful computers, and the variety of machine learning training algorithms have contributed to the recent surge of machine learning being applied to numerous engineering applications. Although, historically, first-principles modeling approaches have been widely adapted in modeling chemical processes, they can be difficult and/or time-consuming to derive when dealing with large-scale, complex, nonlinear processes. In contrast, machine learning techniques have made a significant impact in the field of nonlinear control systems and have shown great success in modeling large-scale, complex, nonlinear processes (e.g., Wu et al. (2019a,b), Chen et al. (2012, 2020), Alhajeri et al. (2021), Wu et al. (2021a,b, 2022)). Researchers in the field have started adapting the science of machine learning since the 90's (Hoskins and Himmelblau, 1992; Vepa, 1993), when they started introducing the concept of machine learning to the field of chemical engineering, and many promising contributions and applications have been observed since then (Venkatasubramanian, 2019).

A wide variety of machine learning techniques are used for modeling nonlinear systems, and one of the powerful and efficient tools is the long short-term memory (LSTM) recurrent neural networks. LSTMs were introduced in Hochreiter and Schmidhuber (1997) and are a type of recurrent neural network (RNN) with a unique structure that allows it to model dynamical systems and overcome numerical issues commonly encountered in traditional RNNs. The incorporation of LSTM models into advanced model-based control strategies such as model predictive control (MPC) comes with notable success. For example, in Chen et al. (2012), a distributed MPC was designed and implemented using an LSTM model. Moreover, LSTMs were also used to design a decentralized MPC in Chen et al. (2020). LSTMs were proven to be efficient when it comes to dealing with noisy data and controlling nonlinear processes. For example, in Alhajeri et al. (2022), LSTMs were used to model a large-scale, chemical process using noisy, industrial data from Aspen Plus Dynamics, and its closed-loop performance under LSTM-based MPC was studied.

* Corresponding author at: Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA.
E-mail address: pdc@seas.ucla.edu (P.D. Christofides).

https://doi.org/10.1016/j.dche.2023.100084
Received 3 December 2022; Received in revised form 4 January 2023; Accepted 5 January 2023
Available online 9 January 2023

The unique structure of LSTM networks enables them to model systems that require long-time interval dependencies such as nonlinear time-delay systems, which require robustness considerations with regards to closed-loop stability and performance criteria. Time delays are a common phenomenon that occurs naturally in chemical processes. A common reason for state delays is transportation lag when materials flow through a pipe. These types of delays are usually reflected as state delays in the first-principles model of the process. Additionally, input delays are another common type of delays in chemical processes, typically caused by control actuator dynamics. Moreover, delays can also arise due to the approximation of complex reaction mechanisms and/or nonlinear higher-order dynamics. Therefore, investigating the stability, robustness, and performance properties of closed-loop time-delay systems is an important topic in the field of control systems.

A nonlinear time-delay system is usually represented by a differential difference equation (DDE). The behavior of DDEs is different from classical ordinary differential equations (ODEs) in several manners. One important difference between them is that, when solving an ODE, an initial condition is required, whereas, for a DDE, the history of states and inputs has to be stored; in other words, an initial history function has to be specified or computed for both states and manipulated inputs. As a matter of fact, time-delayed systems, even if the delay is small, constitute infinite-dimensional systems regardless of the dimension of the state vector of the system (see Hale and Lunel (2013) for more details on DDEs). Significant efforts have been undertaken to study nonlinear time-delay systems (DDEs) and investigate their stability properties. Most relevant to our present work, in Ellis and Christofides (2015), an economic MPC for nonlinear time-delay systems was designed using a first-principles model, with closed-loop stability results being derived using input-to-state stability theorems incorporated with Lyapunov–Razumikhin type arguments. In earlier works, Antoniades and Christofides (1999a,b) established fundamental results on feedback control and robustness of nonlinear time-delay systems with applications to process systems. For the case of controlling nonlinear systems with time-varying measurement delays, several works in the literature have discussed this case. For example, Liu et al. (2009) studied the stability of differential-functional equations with discrete and distributed delays. Additionally, Zhang et al. (2019) provided an overview of the stability of linear systems with time-varying delays and reviewed recent works that have been conducted in this topic.

Several common approaches to control nonlinear time-delay systems can be found in the literature. In the case of small time-delay values, earlier approaches include assigning the values of the time-delays to zero and proceeding with the control design using the resulting ODE systems. This control technique can be effective with acceptable closed-loop stability and performance when the nonlinear system suffers from small state delays. Yet, for larger values of time-delays, and in particular when input delays are present, it is necessary to utilize other approaches that are more involved and can compensate for the effect of input delays. Such approaches include using a predictor with the controller design, to predict future values of the state that can be used in the controller. In 1957, the classical Smith predictor was proposed, and it has become one of the most popular predictor structures used for linear time-delay systems (Smith, 1957). The Smith predictor has proven to be effective in many theoretical aspects and engineering applications. Moreover, various results are found in the literature, adapting the design of the Smith predictor to produce more variations of predictor designs in order to address different types of linear and nonlinear systems with input delay. For instance, Kravaris and Wright (1989), Henson and Seborg (1994) presented designs of predictors that can handle nonlinear systems with input delay through the use of conceptional insights from the Smith predictor approach.

In the context of data-based modeling for feedback controller design, machine learning techniques have been incorporated in MPC with resounding success due to their notable accuracy, efficiency, and ability to capture complex systems' dynamics, as successfully demonstrated via numerous chemical process applications in Ren et al. (2022). Specifically, Wu et al. (2019a,b) provide fundamental theoretical and practical insights of machine learning-based MPC and necessary stability analyses. The first conceptualization of MPC can be dated to 1978 (Richalet et al., 1978), and, since then, many contributions and applications have been made in both industry and academia. MPC has proven to be efficient in a diverse array of applications due to its ability to handle multiple inputs, outputs, and constraints by solving an optimization problem that minimizes a desired objective function of the inputs and outputs (Alhajeri et al., 2021; Wu et al., 2021a) subject to constraints while incorporating measurement feedback into the calculations. With a sufficiently accurate process model, MPC can also handle systems where only noisy data is available (Abdullah et al., 2022). The control actions of MPC are computed through repeatedly solving an optimization problem in a finite time horizon, with both state and input constraints. This guarantees the stability and boundedness of the trajectories of the nonlinear system at all times.

In light of the above considerations, in this article, we apply machine learning to develop a model—specifically an LSTM model—using data from the process model without the time delays and use this LSTM model to design a model predictive controller that renders the process model without the time delays stable. Subsequently, we established that the LSTM-based MPC ensures stability of the closed-loop system under sufficiently small state delays. Additionally, we design an LSTM-based predictor to compensate for input delays. Finally, an application of the LSTM-based MPC to a chemical reactor under both state and input delays is presented.

2. Preliminaries

2.1. Notation

A time-dependent vector is denoted by $x(t) \in \mathbb{R}^n$. x^T denotes the transpose of vector x. The Euclidean norm of a vector is denoted by $|\cdot|$, and the infinity norm of a function $\phi \in C([a,b],\mathbb{R}^n)$ is represented by $\|\cdot\|$, such that $\|\phi\| := \max_{a\le s\le b}|\phi(s)|$, where $C([a,b],\mathbb{R}^n)$ is the space of continuous functions mapping the interval $[a,b]$ to $\mathbb{R}^n$. Set subtraction is denoted by '\', such that $A\backslash B := \{x \in R^n | x \in A, x \notin B\}$. $S(\Delta)$ denotes the family of piecewise constant, right-continuous functions with period Δ.

2.2. Class of systems

The following family of differential difference equations (DDEs) describes the class of nonlinear time-delay systems considered in this work:

$$\dot{x}(t) = F(x,u) = f\big(x(t), x(t-d_1), u(t-d_2)\big) \tag{1}$$

$x(t)$ is the n-dimensional state vector, and $u(t)$ is the m-dimensional control input vector bounded by $u \in U$. The set U, defined as $U := \{|u_i| \le u_{\max,i}, i = 1,\dots,m\}$. The vector $f(\cdot)$ is a locally Lipschitz vector function of its arguments. Under the assumption that $f(0,0,0) = 0$, the origin is a steady-state of Eq. (1). $d_1 > 0$ is the value of the state delay and $d_2 > 0$ is the value of the input delay. Moreover, without loss of generality, the initial time is taken to be zero (i.e., $t_0 = 0$), and the initial data is denoted as ϕ_x, where $\phi_x \in C([-d_1,0],\mathbb{R}^n)$. Additionally, the symbol ϕ_u represents the initial input function, where $\phi_u \in C([-d_2,0],\mathbb{R}^m)$. Hence, ϕ_u is bounded and is assumed to be piecewise continuous over its domain. The system of Eq. (1) can be expressed as a perturbed form of the system without delays in the following form:

$$\dot{x}(t) = F(x,u,\xi) = f(x(t), x(t)+\xi_1(t), u(t)+\xi_2(t)) \tag{2a}$$

$$\xi_1(t) = x(t-d_1) - x(t) \tag{2b}$$

$$\xi_2(t) = u(t - d_2) - u(t) \tag{2c}$$

where $\xi^T := [\xi_1^T \quad \xi_2^T] \in D \times U \subset \mathbb{R}^{n+m}$ is the bounded perturbation vector, and D is an open neighborhood around the origin.

Remark 1. In this work, differential difference equations (DDEs) are used to describe the general class of nonlinear time-delay systems. A number of different methods to describe nonlinear time-delay systems exist in the literature, such as first order plus dead time and second order plus dead time models, which are specific and assume certain (linear) model structures. Therefore, nonlinear differential difference equations with constant delays were chosen as the class of systems in this paper to make the analysis more general. However, other works that describe nonlinear time-delay systems with functional differential equations can be found, and our results may be extended to such model structures as well. For example, describing systems with multiple state and input delays and systems with time-varying delays can be done using functional differential equations.

2.3. Stabilization via control Lyapunov function

Taking into consideration the ODE system in Eq. (2), we assume that there exists a locally Lipschitz feedback controller $\Phi(x) \in U$, such that the origin of the nominal system of Eq. (2) (i.e., with $\xi(t) \equiv 0$) is exponentially stable. Hence, the system of Eq. (2) is stabilizable, and there exists a continuously differentiable Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ such that the following inequalities hold:

$$c_1|x|^2 \leq V(x) \leq c_2|x|^2, \tag{3a}$$

$$\frac{\partial V(x)}{\partial x} F(x, \Phi(x), 0) \leq -c_3|x|^2, \tag{3b}$$

$$\left|\frac{\partial V(x)}{\partial x}\right| \leq c_4|x| \tag{3c}$$

given that c_i are positive constants, where $i = 1, 2, 3, 4$, for all $x \in \mathbb{R}^n \subset D$. Since the system $F(x, u, \xi)$ has a Lipschitz property, together with the bounded behavior of the input u and the perturbation ξ, there exist positive constants M, L_x, L'_x, L_ξ, and L'_ξ such that the following inequalities hold for all $x, x' \in D$ and $u \in U$:

$$|F(x, u, \xi)| \leq M \tag{4a}$$

$$|F(x, u, \xi) - F(x', u, 0)| \leq L_x|x - x'| + L_\xi|\xi| \tag{4b}$$

$$\left|\frac{\partial V(x)}{\partial x} F(x, u, \xi) - \frac{\partial V(x')}{\partial x} F(x', u, 0)\right| \leq L'_x|x - x'| + L'_\xi|\xi| \tag{4c}$$

Additionally, the closed loop stability region of the nonlinear system of Eq. (2) is characterized by the region Ω_ρ, where $\Omega_\rho := \{x \in D | V(x) \leq \rho\}$

2.4. Long short-term memory recurrent neural networks

An LSTM network is a special type of RNN that is composed of a number of gates and cells, which gives it a unique structure. LSTMs were introduced to overcome a number of limitations found in classical RNNs, primarily the vanishing gradient problem. This phenomenon occurs very often in classical RNNs, where the product of the gradients in the loss function get smaller in value as we proceed through the layers of the network, causing the loss function to have a value near zero for older data points. Hence, training the network becomes harder, and analyzing data over long time periods becomes a challenging task. Additionally, LSTMs are well known for modeling systems that require long time dependencies. The LSTM network is designed to predict future states of the process using the past state measurements and future control actions. Therefore, the input sequence of the LSTM network is denoted by $p \in \mathbb{R}^{(n+m)\times T}$, while the output of the LSTM network is denoted by $\hat{x} \in \mathbb{R}^{n\times T}$, where T is the number of time steps or repeating LSTM modules within one sampling period. Fig. 1 illustrates the LSTM structure.

The LSTM unit is expressed through the following equations:

$$g(k) = \sigma\left(\omega_g^p p(k) + \omega_g^h h(k-1) + b_g\right) \tag{5a}$$

$$i(k) = \sigma\left(\omega_i^p p(k) + \omega_i^h h(k-1) + b_i\right) \tag{5b}$$

$$c(k) = i(k)\tanh\left(\omega_c^p p(k) + \omega_c^h h(k-1) + b_c\right) + g(k)c(k-1) \tag{5c}$$

$$o(k) = \sigma\left(\omega_o^p p(k) + \omega_o^h h(k-1) + b_o\right) \tag{5d}$$

$$\hat{x}(k) = \omega_y h(k) + b_y \tag{5e}$$

$$h(k) = o(k)\tanh(c(k)) \tag{5f}$$

where the input sequence is denoted by $p(k)$, and the vector $\hat{x}(k) \in \mathbb{R}^{n\times T}$ represents the LSTM network output, with $k = 1, \ldots, T$. The weight matrix and bias vector for the output are ω_y and b_y, respectively. Additionally, $h(k)$ is the internal state, while $g(k), i(k)$, and $o(k)$ represent the outputs from the forget gate, the input gate, and the output gate, respectively. $\omega_g^p, \omega_g^h, \omega_i^p, \omega_i^h, \omega_o^p$, and ω_o^h are the weight matrices for the input vector p and the hidden state vector h within the forget gate, the input gate and the output gate, respectively. b_g, b_i, and b_o are the bias vectors for the forget gate, the input gate, and the output gate, respectively. Moreover, $c(k)$ represents the cell state, which is in charge of storing and passing the essential information through successive LSTM units. More precisely, the first term in Eq. (5c) is in charge of storing the new, important information coming from the input gate $i(k)$ in the cell state $c(k)$ that is to be passed to the next LSTM unit. In contrast, the second term in Eq. (5c) uses the forget gate $g(k)$ to compute the information that should be discarded from the previous state $c(k-1)$. Additionally, the weight matrices associated with the cell state are represented by ω_c^p and ω_c^h, where ω_c^p indicates the weight matrix for the input vector, and ω_c^h is the weight matrix for the hidden state vector. b_c represents the bias vector associated with the cell state. σ and $tanh$ are the nonlinear *sigmoid* and hyperbolic tangent activation functions, respectively. The LSTM input sequence is $p \in \mathbb{R}^{(n+m)\times T}$, which consists of the past state measurements x and the manipulated inputs u.

In this study, we develop an LSTM network model with the following form as a continuous-time nonlinear system:

$$\dot{\hat{x}} = F_{nn}(\hat{x}, u) := A\hat{x} + \Theta^T z \tag{6}$$

where $\hat{x} \in \mathbb{R}^n$ is the LSTM state vector, and $u \in \mathbb{R}^m$ is the manipulated input. $A \in \mathbb{R}^{n\times n}$ and $\Theta \in \mathbb{R}^{(n+m+1)\times n}$ are the weight matrices, and $z = [z_1 \ldots z_{n+m+1}]^T = [\sigma(\hat{x}_1) \ldots \sigma(\hat{x}_n)\ u_1 \ldots u_m\ 1]^T \in \mathbb{R}^{n+m+1}$ is a vector associated with the network states $\hat{x}$ and the manipulated input u.

Remark 2. Many applications in the control engineering field require machine learning models that can deal with sequential data. The inputs and outputs of a chemical process are often considered as time-series data. Sequential neural network models, such as recurrent neural networks (RNNs), gated recurrent units (GRUs) and long short-term memory networks (LSTMs), are types of machine learning models that are well-suited to model nonlinear dynamical systems (including the nonlinear time-delay systems considered in our work) using sequential time-series input–output data. The LSTM network is a strong candidate given its potential to overcome problems that occur in RNNs and its proven ability to model systems that require long time dependencies such as nonlinear time-delay systems, which require robustness considerations with regards to closed-loop stability and performance criteria.

2.5. Data generation and model training process

The goal is to design a stabilizing control law for the nonlinear time-delayed system of Eq. (1) under small-time delays or, in other words, small and bounded perturbations. In this work, we will be adapting a

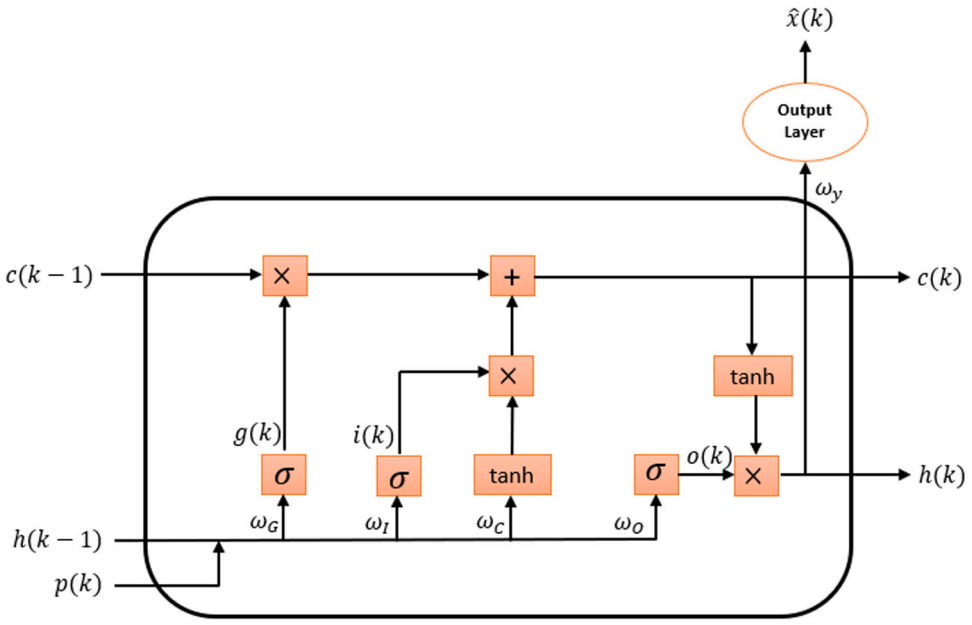

Fig. 1. LSTM structure.

machine learning method, specifically LSTM neural networks. The first step in the development of an LSTM model is to generate data. We follow the data generation technique described in Wu et al. (2019a), which is to run extensive open-loop simulations of the nonlinear system of Eq. (2) without the time delays. The aim is to capture the dynamics of the system for all $x \in \Omega_\rho$ and $u \in U$, sweeping over all possible combinations of initial conditions $x_0 \in \Omega_\rho$ and inputs $u \in U$. Moreover, we note that, in the data generation process as well as when simulating the system of Eq. (2) without time delays, the input $u \in U$ is applied in a sample-and-hold manner, where it is fed to the system of Eq. (2) without time delays as a piecewise constant function $u(t) = u(t_k), \forall t \in [t_k, t_{k+1})$, where $t_{k+1} := t_k + \Delta$ (Δ is the sampling period). Then, we integrate the system of Eq. (2) without time delays using the explicit Euler method with a sufficiently small integration time step $h_c \ll \Delta$. Therefore, we are able to create a set of time-series data for the state x within a chosen level set of the Lyapunov function, denoted as the operating region Ω_ρ. This procedure yields a large data set of diverse trajectories to be passed to the model training phase. Subsequently, we train the LSTM network using the generated data and the neural network library Keras. The architecture of the LSTM network of Eq. (6) is designed to be such that, given the current state measurement and future manipulated inputs, the LSTM is able to predict the future states for at least one sampling period ahead, i.e., $x(t) \in [t_k, t_k + \Delta)$. As a result, we obtain an LSTM model that has the ability to capture the dynamics of the system with sufficiently small modeling error. We point out that the gathered data set is divided into three main sets—the training, validation, and testing sets, each for its own purpose.

Remark 3. The LSTM model can be trained using data from the delayed process. However, training on the basis of the ODE process model without delays allows us to further evaluate the robustness of this model with respect to applying it in an MPC that is used to control the delayed process, despite the training data being from the system without delays.

Remark 4. Referring to the nonlinear system of Eq. (2), the existence of perturbation or disturbances in the system indicates that we may be dealing with noisy data in the training phase, which can affect the accuracy of the developed LSTM model. In some cases, the LSTM model can fail to make the correct predictions due to the existence of perturbation in the data set. Additionally, some studies indicate that the perturbed data makes the LSTM model more robust and enhances the performance of the model (see Bishop (1995) for more details on this topic). It is important to highlight that, in this article, we generated data and trained the LSTM model based on the nominal system of Eq. (2) (i.e., with $\xi = 0$). Using machine learning techniques to control perturbed nonlinear systems is a topic that requires further studies. However, in further sections, we will show that the LSTM model developed based on the nominal system of Eq. (2), when incorporated in a Lyapunov-based model predictive controller (LMPC), is able to stabilize the perturbed nonlinear system of Eq. (2), under the condition of bounded perturbation, and eventually under sufficiently small state delays.

3. Robustness of LSTM-based LMPC to small time state delays

In this section, we will focus on the closed loop stability analysis of the perturbed nonlinear system of Eq. (2), taking into consideration sufficiently small state delays only (i.e., $d_2 = 0$ and, hence, $\xi_2 = 0$). However, the stabilization of the perturbed system of Eq. (2) in the presence of both state and input delays will be achieved using a predictor feedback LSTM-based LMPC methodology in Section 5. Additionally, knowing that the state delays are represented through the perturbation term $\xi_1(t) = x(t - d_1) - x(t)$, the upper bound of the perturbation can be written as follows:

$$|\xi(t)| = |\xi_1(t)| = |x(t - d_1) - x(t)| \leq d_1 \|x_d(t)\| \tag{7}$$

where $\|x_d(t)\|$ is the max-norm of $x_d(t) \in C([-d_1, 0], \mathbb{R}^n)$ (i.e., $\|x_d(t)\| = \max_{\theta \in [-d_1, 0]} |x(t - \theta)|$).

3.1. Stabilization of LSTM models via control Lyapunov function

Taking into consideration the LSTM model developed in Eq. (6), we assume that there exists a locally Lipschitz feedback controller $\Phi_{nn}(x) \in U$ such that exponential stability of the LSTM model is attained at the origin. This implies that there exists a continuously differentiable Lyapunov function $\hat{V} : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ such that the following inequalities hold:

$$\hat{c}_1 |x|^2 \leq \hat{V}(x) \leq \hat{c}_2 |x|^2 \tag{8a}$$

$$\frac{\partial \hat{V}(x)}{\partial x} F_{nn}(x, \Phi_{nn}(x)) \leq -\hat{c}_3 |x|^2 \tag{8b}$$

$$\left|\frac{\partial \hat{V}(x)}{\partial x}\right| \leq \hat{c}_4|x| \tag{8c}$$

given that $\hat{c}_i$ are positive constants, for all $x \in \mathbb{R}^n \subset \hat{D}$ where, $i = 1,2,3,4$, and $\hat{D}$ is an open neighborhood around the origin. The LSTM model of Eq. (6) has a stability region denoted as $\Omega_{\hat{\rho}}$, characterized as a compact set embedded in $\hat{D}$ as follows: $\Omega_{\hat{\rho}} := \{x \in \hat{D} \mid \hat{V}(x) \leq \hat{\rho}\}$, where $\hat{\rho} > 0$. In addition, there exist positive constants M_{nn} and L_{nn} such that the following inequalities hold for all $x, x' \in \Omega_{\hat{\rho}}$ and $u \in U$:

$$|F_{nn}(x,u)| \leq M_{nn} \tag{9a}$$

$$\left|\frac{\partial \hat{V}(x)}{\partial x}F_{nn}(x,u) - \frac{\partial \hat{V}(x')}{\partial x}F_{nn}(x',u)\right| \leq L_{nn}|x-x'| \tag{9b}$$

The proposition shown below illustrates that the feedback control input $u = \Phi_{nn}(x) \in U$ can stabilize the nominal system of Eq. (2), under a sufficiently small modeling error.

Proposition 1 (*c.f Proposition 2 in (Wu et al., 2019a)*). *Consider the LSTM model of Eq. (6) that satisfies the stabilizability criteria of Eq. (8) and is exponentially stable around the origin under the control law $u = \Phi_{nn}(x) \in U$ for all $x \in \Omega_{\hat{\rho}}$. Then, the origin of the perturbed nonlinear system of Eq. (2) is exponentially stable around the origin for all $x \in \Omega_{\hat{\rho}}$ under the condition that there exists a positive real number γ, where $\gamma < \hat{c}_3/\hat{c}_4$. Additionally, γ is the upper bound of the modeling error between the nominal system of Eq. (2) with $\xi = 0$ and the LSTM model (i.e., $\nu = |F(x,u,0) - F_{nn}(x,u)| \leq \gamma|x|$ for all $x \in \Omega_{\hat{\rho}}$).*

Proof. We proceed by following the proof of Proposition 2 in Wu et al. (2019a). The goal is to prove that the nominal system of Eq. (2) is exponentially stable around the origin for all $x \in \Omega_{\hat{\rho}}$. This can be achieved by showing that $\dot{\hat{V}}(x)$ is negative for the nominal system of Eq. (2) under the stabilizing control law $u = \Phi_{nn}(x) \in U$ for all $x \in \Omega_{\hat{\rho}}$. Using the inequalities in Eqs. (8b) and (8c), $\dot{\hat{V}}(x)$ can be computed as follows:

$$\begin{aligned}\dot{\hat{V}} &= \frac{\partial \hat{V}(x)}{\partial x}F(x,\Phi_{nn}(x),0)\\ &= \frac{\partial \hat{V}}{\partial x}F_{nn}(x,\Phi_{nn}(x)) + F(x,\Phi_{nn}(x),0) - F_{nn}(x,\Phi_{nn}(x))\\ &\leq -\hat{c}_3|x|^2 + \hat{c}_4|x|(F(x,\Phi_{nn}(x),0) - F_{nn}(x,\Phi_{nn}(x)))\\ &= -\hat{c}_3|x|^2 + \gamma\hat{c}_4|x|^2\end{aligned} \tag{10}$$

By letting $\gamma < \frac{\hat{c}_3}{\hat{c}_4}$, we achieve $\dot{\hat{V}}(x) \leq -\tilde{c}_3|x|^2 \leq 0$ where $\tilde{c}_3 = -\hat{c}_3 + \gamma\hat{c}_4 \geq 0$ and, consequently, closed-loop stability of the nominal system of Eq. (2) around the origin under the control law $\Phi_{nn}(x) \in U$ for all $x \in \Omega_{\hat{\rho}}$. □

3.2. Sample-and-hold implementation of Lyapunov-based controller

It is important to highlight that the LSTM-based LMPC is designed using the LSTM model generated in Eq. (6), where the control actions are executed in sample-and-hold fashion. In order to study the robustness of the LSTM-based LMPC to small time-delays in the states, we note that, in the next two propositions, we will consider state delays only (i.e., $\xi_2 = 0$). Given that the state delays are sufficiently small, this implies that the perturbation ξ_1 is bounded. The following proposition shows that the error between the state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ and the predicted state by the LSTM model Eq. (6) is bounded.

Proposition 2 (*c.f Proposition 3 in (Wu et al., 2019a)*). *Consider the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ in the presence of bounded disturbances (i.e., $|\xi(t)| = |\xi_1(t)| \leq d_1\|x_d(t)\|, \|x_d(t)\| = \max_{\theta\in[-d_1,0]}|x(t-\theta)|$) and the LSTM model of Eq. (6) with the same initial condition $x_0 = \hat{x}_0 \in \Omega_{\hat{\rho}}$. There exists a class $\mathcal{K}$ function $f_\xi(\cdot)$ and a positive constant κ such that the following inequalities hold for all $x, \hat{x} \in \Omega_{\hat{\rho}}$ and $|\xi(t)| = |\xi_1(t)| \in D \subset \mathbb{R}^n$:*

$$|x(t) - \hat{x}(t)| \leq f_\xi(t) := \frac{L_\xi\, d_1\|x_d(t)\| + \nu_m}{L_x}\left(e^{L_x t} - 1\right) \tag{11a}$$

$$\hat{V}(x) \leq \hat{V}(\hat{x}) + \frac{\hat{c}_4\sqrt{\hat{\rho}}}{\sqrt{\hat{c}_1}}|x-\hat{x}| + \kappa|x-\hat{x}|^2 \tag{11b}$$

Proof. Let $e(t) = x(t) - \hat{x}(t)$ represent the error vector between the state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ and the state of the LSTM model of Eq. (6). The following bound can be found for the time-derivative of $e(t)$:

$$\begin{aligned}|\dot{e}(t)| &= |F(x,u,\xi) - F_{nn}(\hat{x},u)|\\ &\leq |F(x,u,\xi) - F(\hat{x},u,0)| + |F(\hat{x},u,0) - F_{nn}(\hat{x},u)|\end{aligned} \tag{12}$$

Using Eq. (4b), for all $x, \hat{x} \in \Omega_{\hat{\rho}}$ and $|\xi(t)| = |\xi_1(t)| \in D \subset \mathbb{R}^n$, we can bound the first term in Eq. (12) as follows:

$$\begin{aligned}|F(x,u,\xi) - F(\hat{x},u,0)| &\leq L_x|x(t)-\hat{x}(t)| + L_\xi|\xi|\\ &\leq L_x|x(t)-\hat{x}(t)| + L_\xi d_1\|x_d(t)\|\end{aligned} \tag{13}$$

We observe that the second term of Eq. (13) is equivalent to the modeling error, which is upper bounded by ν_m for all $\hat{x} \in \Omega_{\hat{\rho}}$. Therefore, the bound of the modeling error and the bound of Eq. (13) can be used to further bound $\dot{e}(t)$ as follows:

$$\begin{aligned}|\dot{e}(t)| &\leq L_x|x(t)-\hat{x}(t)| + L_\xi|d_1|\|x_d(t)\| + \nu_m\\ &\leq L_x|e(t)| + L_\xi d_1\|x_d(t)\| + \nu_m\end{aligned} \tag{14}$$

Integrating the inequality of Eq. (14) from zero initial conditions (i.e., $e(0) = 0$), the following upper bound for the error vector can be obtained for all $x, \hat{x} \in \Omega_{\hat{\rho}}$ and $|\xi(t)| = |\xi_1(t)| \in D \subset \mathbb{R}^n$:

$$|e(t)| = |x(t)-\hat{x}(t)| \leq \frac{L_{\xi_1} d_1\|x_d(t)\| + \nu_m}{L_x}\left(e^{L_x t} - 1\right) \tag{15}$$

Using the Taylor series expansion of $\hat{V}(x)$ around $\hat{x}$, we derive Eq. (11b) as follows, for all $x, \hat{x} \in \Omega_{\hat{\rho}}$:

$$\hat{V}(x) \leq \hat{V}(\hat{x}) + \frac{\partial \hat{V}(\hat{x})}{\partial x}|x-\hat{x}| + \kappa|x-\hat{x}|^2 \tag{16}$$

where κ is a positive real number. Additionally, we use Eqs. (8a) and (8b) to further upper bound $\hat{V}(x)$ as follows:

$$\hat{V}(x) \leq \hat{V}(\hat{x}) + \frac{\hat{c}_4\sqrt{\hat{\rho}}}{\sqrt{\hat{c}_1}}|x-\hat{x}| + \kappa|x-\hat{x}|^2 \quad \square \tag{17}$$

The following proposition shows that the closed-loop state trajectory $x(t)$ of the perturbed system of Eq. (2) with $\xi_2 = 0$ is bounded in $\Omega_{\hat{\rho}}$ for all times and can be driven to a small neighborhood around the origin, $\Omega_{\rho_{\min}}$, under the controller $\Phi_{nn}(x) \in U$ executed in sample-and-hold fashion.

Proposition 3 (*c.f Proposition 4 in (Wu et al., 2019a)*). *Consider the nonlinear system of Eq. (2) with $\xi_2 = 0$ under the controller $\Phi_{nn}(\hat{x}) \in U$ that meets the conditions of Eq. (8) and stabilizes the LSTM model of Eq. (6). The controller is executed in sample-and-hold, i.e., $\Phi_{nn}(\hat{x}(t_k)), \forall t \in [t_k, t_{k+1})$, where $t_{k+1} := t_k + \Delta$. Then, there exist $\epsilon_w > 0$, $\Delta > 0$ and $\hat{\rho} > \rho_{\min} > \rho_{nn} > \rho_s$ that satisfy*

$$-\frac{\hat{c}_3}{\hat{c}_2}\rho_s + L_{nn}M_{nn}\Delta \leq -\epsilon_s \tag{18a}$$

$$-\frac{\tilde{c}_3}{\hat{c}_2}\rho_s + L'_x M_F \Delta \leq -\epsilon_w \tag{18b}$$

and

$$\rho_{nn} := \max\{\hat{V}\left(\hat{x}(t+\Delta)\right) \mid \hat{x}(t) \in \Omega_{\rho_s}, u \in U\} \tag{19a}$$

$$\rho_{\min} \geq \rho_{nn} + \frac{\hat{c}_4\sqrt{\hat{\rho}}}{\sqrt{\hat{c}_1}}f_\xi(\Delta) + \kappa\left(f_\xi(\Delta)\right)^2 \tag{19b}$$

such that for any $x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$, the following inequality holds:

$$\hat{V}(x(t)) \leq \hat{V}(x(t_k)), \ \forall t \in [t_k, t_{k+1}) \tag{20}$$

and the state $x(t)$ of the perturbed nonlinear system of Eq. (2) is bounded in $\Omega_{\hat{\rho}}$ for all times and ultimately bounded in $\Omega_{\rho_{\min}}$.

Proof. First, we need to show that $\hat{V}(x)$ is decreasing under the controller $u(t) = \Phi_{nn}(x(t_k))$ for $t \in [t_k, t_{k+1})$. Consider $x(t_k) = \hat{x}(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$, where $x(t_k)$ is the state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$, and $\hat{x}(t_k)$ is the state of the LSTM model in Eq. (6). The time-derivative of $\hat{V}(x)$ for all $t \in [t_k, t_{k+1})$ is computed as follows:

$$\begin{aligned}\dot{\hat{V}}(\hat{x}(t)) &= \frac{\partial \hat{V}(\hat{x}(t))}{\partial \hat{x}} F_{nn}\big(\hat{x}(t), \Phi_{nn}(\hat{x}(t_k))\big) \\ &= \frac{\partial \hat{V}(\hat{x}(t_k))}{\partial \hat{x}} F_{nn}\big(\hat{x}(t_k), \Phi_{nn}(\hat{x}(t_k))\big) \\ &\quad + \frac{\partial \hat{V}(\hat{x}(t))}{\partial \hat{x}} F_{nn}\big(\hat{x}(t), \Phi_{nn}(\hat{x}(t_k))\big) \\ &\quad - \frac{\partial \hat{V}(\hat{x}(t_k))}{\partial \hat{x}} F_{nn}\big(\hat{x}(t_k), \Phi_{nn}(\hat{x}(t_k))\big)\end{aligned} \tag{21}$$

Using the inequalities in Eqs. (8a) and (8b), we obtain the following:

$$\begin{aligned}\dot{\hat{V}}(\hat{x}(t)) \leq &-\frac{\hat{c}_3}{\hat{c}_2}\rho_s + \frac{\partial \hat{V}(\hat{x}(t))}{\partial \hat{x}} F_{nn}(\hat{x}(t), \Phi_{nn}(\hat{x}(t_k))) \\ &- \frac{\partial \hat{V}(\hat{x}(t_k))}{\partial \hat{x}} F_{nn}(\hat{x}(t_k), \Phi_{nn}(\hat{x}(t_k)))\end{aligned} \tag{22}$$

Using the Lipschitz inequalities in Eq. (9), we can further bound $\dot{\hat{V}}(\hat{x}(t))$ by the following:

$$\begin{aligned}\dot{\hat{V}}(\hat{x}(t)) &\leq -\frac{\hat{c}_3}{\hat{c}_2}\rho_s + L_{nn}|\hat{x}(t) - \hat{x}(t_k)| \\ &\leq -\frac{\hat{c}_3}{\hat{c}_2}\rho_s + L_{nn} M_{nn} \Delta\end{aligned} \tag{23}$$

Hence, if Eq. (18a) is satisfied, the following inequality holds for all $\hat{x}(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$ and $t \in [t_k, t_{k+1})$:

$$\dot{\hat{V}}(x(t)) \leq -\epsilon_s \tag{24}$$

Integrating the above inequality over $t \in [t_k, t_{k+1})$ yields the desired result, $\hat{V}(\hat{x}(t_{k+1})) \leq \hat{V}(\hat{x}(t_k)) - \epsilon_s \Delta$. Hence, we have shown that, if Eq. (18a) holds, then $\dot{\hat{V}}(x(t))$ is negative for any $\hat{x}(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$. This implies that the closed-loop state of the LSTM model of Eq. (6) under the sample-and-hold implementation of the controller $u = \Phi_{nn}(\hat{x})$ is bounded within the region $\Omega_{\hat{\rho}}$ and moves toward the origin. Additionally, the region $\Omega_{\rho_{nn}}$ in Eq. (19a) is introduced for the case where $x(t_k) = \hat{x}(t_k) \in \Omega_{\rho_s}$. In this case, Eq. (24) may not hold, and the state $\hat{x}(t_k)$ may leave the region Ω_{ρ_s} within one sampling period. Therefore, $\Omega_{\rho_{nn}}$ is designed to guarantee that the closed-loop state $\hat{x}(t_k)$ of the LSTM model will be bounded in the region $\Omega_{\rho_{nn}}$ within one sampling period, for all $t \in [t_k, t_{k+1})$, $u \in U$ and $\hat{x}(t_k) \in \Omega_{\rho_s}$, because, if $\hat{x}(t_{k+1})$ leaves Ω_{ρ_s}, the controller $u = \Phi_{nn}\big(x(t_{k+1})\big)$ reactivates, such that Eq. (24) will be satisfied again at $t = t_{k+1}$, and the state will be driven back toward Ω_{ρ_s} over the next sampling period. Thus far, we can conclude that the state of the LSTM system of Eq. (6) is ultimately bounded in $\Omega_{\rho_{nn}}$ for all $x_0 \in \Omega_{\hat{\rho}}$.

The next step is to show that the controller $u = \Phi_{nn}(x) \in U$, applied in sample-and-hold fashion, is able to bound the states of the perturbed nonlinear system of Eq. (2) with sufficiently small state delays and with $\xi_2 = 0$, in some neighborhood around the origin. Therefore, we need to show that $\hat{V}(x)$ for the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ is decreasing under the controller $u(t) = \Phi_{nn}(x(t_k))$ for $t \in [t_k, t_{k+1})$ and $x(t_k) = \hat{x}(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$. The time-derivative of $\hat{V}(x(t))$ is calculated as:

$$\begin{aligned}\dot{\hat{V}}(x(t)) &= \frac{\partial \hat{V}(x(t))}{\partial x} F(x(t), \Phi_{nn}(x(t_k)), \xi) \\ &= \frac{\partial \hat{V}(x(t_k))}{\partial x} F(x(t_k), \Phi_{nn}(x(t_k)), 0) \\ &\quad + \frac{\partial \hat{V}(x(t))}{\partial x} F(x(t), \Phi_{nn}(x(t_k)), \xi) \\ &\quad - \frac{\partial \hat{V}(x(t_k))}{\partial x} F(x(t_k), \Phi_{nn}(x(t_k)), 0)\end{aligned} \tag{25}$$

where the first term can be further bounded using the inequality in Eq. (10) as follows:

$$\begin{aligned}\dot{\hat{V}}(x(t)) \leq &-\frac{\tilde{c}_3}{\hat{c}_2}\rho_s + \frac{\partial \hat{V}(x(t))}{\partial x} F(x(t), \Phi_{nn}(x(t_k)), \xi) \\ &- \frac{\partial \hat{V}(x(t_k))}{\partial x} F(x(t_k), \Phi_{nn}(x(t_k)), 0)\end{aligned} \tag{26}$$

Since f and Φ_{nn} in the perturbed nonlinear system, $F(x(t), \Phi_{nn}(x(t_k)), \xi)$, are locally Lipschitz vector functions, there exists a $\gamma_1^* \in \mathcal{K}$ such that:

$$\begin{aligned}|\xi(t)| = |\xi_1(t)| &= \left| \int_{t-d_1}^{t} f\big(x(s), x(t-s), \Phi_{nn}(x(s))\big)\, \mathrm{d}s \right| \\ &\leq d_1 \gamma_1^*(\|x_d(t)\|)\end{aligned} \tag{27}$$

where $\|x_d(t)\| = \max_{s \in [-2d_1, 0]} |x(t+s)|$. Applying the inequality of Eq. (27) and the Lipschitz condition of Eq. (4), we obtain the following bound for $\dot{\hat{V}}(x(t))$:

$$\begin{aligned}\dot{\hat{V}}(x(t)) &\leq -\frac{\tilde{c}_3}{\hat{c}_2}\rho_s + L'_x|x(t) - x(t_k)| + L'_\xi|\xi| \\ &\leq -\frac{\tilde{c}_3}{\hat{c}_2}\rho_s + L'_x M_F \Delta + L'_\xi d_1 \gamma_1^*(\|x_d(t)\|)\end{aligned} \tag{28}$$

Hence, if Eq. (18b) is satisfied, the following inequality holds for all $x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$ and for all $t \in [t_k, t_{k+1})$:

$$\dot{\hat{V}}(x(t)) \leq -\epsilon_w \tag{29}$$

By integrating the above inequality over $t \in [t_k, t_{k+1})$, it is shown that Eq. (20) holds, and $\dot{\hat{V}}(x(t))$ is negative for all $x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_s}$. Hence, the state of the closed-loop perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ is bounded within the region $\Omega_{\hat{\rho}}$ for all times and can be driven towards the origin in every sampling period under the control law $u = \Phi_{nn}(x)$. For the case where $x(t_k) \in \Omega_{\rho_s}$, we recall Eq. (11a), where the error between the state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ and the state of the LSTM model of Eq. (6) is bounded by the term f_ξ. We introduce a compact set $\Omega_{\rho_{\min}} \supset \Omega_{\rho_{nn}}$ satisfying Eq. (19b). This ensures that, if the state of the LSTM model of Eq. (6) is bounded in $\Omega_{\rho_{nn}}$, then the state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ will be bounded within $\Omega_{\rho_{\min}}$ during one sampling period. If the state $x(t)$ enters $\Omega_{\rho_{\min}} \backslash \Omega_{\rho_s}$, it is shown that Eq. (29) is satisfied, which implies that the state will be driven towards the origin.

Finally, we conclude that the closed loop state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$ is always bounded in $\Omega_{\hat{\rho}}$ and ultimately bounded within a small region around the neighborhood (i.e., $\Omega_{\rho_{\min}}$) under the control law $u = \Phi_{nn}(x) \in U$, provided that the assumptions in Proposition 3 are satisfied. □

4. LSTM-based model predictive control

The trained LSTM model is then incorporated into a Lyapunov-based MPC (LMPC), where it will be used to evaluate future states in the MPC algorithm. The resulting LSTM-based LMPC computes the optimal control actions by solving the following optimization problem (Wu et al., 2019a,b):

$$\mathcal{J} = \min_{u \in S(\Delta)} \int_{t_k}^{t_{k+N}} L_{MPC}(\tilde{x}(t), u(t)) dt \tag{30a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = F_{nn}(\tilde{x}(t), u(t)) \tag{30b}$$

$$u(t) \in U, \ \forall \ t \in [t_k, t_{k+N}) \tag{30c}$$

$$\tilde{x}(t_k) = x(t_k) \tag{30d}$$

$$\dot{\hat{V}}(x(t_k), u) \leq \dot{\hat{V}}(x(t_k), \boldsymbol{\Phi}_{nn}(x(t_k))),$$

$$\text{if } x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_{nn}} \tag{30e}$$

$$\hat{V}(\tilde{x}(t)) \leq \rho_{nn}, \ \forall \ t \in [t_k, t_{k+N}), \ \text{if } x(t_k) \in \Omega_{\rho_{nn}} \tag{30f}$$

In the LMPC formulation, $\tilde{x}(t)$ is the predicted state trajectory, and the number of sampling periods in the prediction horizon is denoted by N. The LSTM-based LMPC computes the optimal control action, $u^*(t)$, over the whole prediction horizon $t \in [t_k, t_{k+N})$. The controller sends the optimal control action $u^*(t_k)$ computed for the first sampling period within the prediction horizon to be applied to the process, after which the resultant real-time state from the process $x(t_k)$ is sent back to the LSTM-based LMPC to resolve the optimal input trajectory at the next sampling time. The cost function of the optimization problem is shown in Eq. (30a), and it minimizes the time-integral of $L_{MPC}(\tilde{x}(t), u(t))$ over the prediction horizon. The first constraint of the optimization problem is that of Eq. (30b), which uses the LSTM model to predict the states. The second constraint is Eq. (30c), which limits the inputs that may be applied, over the whole prediction horizon. Eq. (30d) is the state measurement at $t = t_k$, which is the initial condition to integrate $\tilde{x}(t)$ from when integrating Eq. (30b). The closed loop trajectory converges towards the steady state value if $x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_{nn}}$ due to the constraint of Eq. (30e). Otherwise, if $x(t_k)$ enters the region $\Omega_{\rho_{nn}}$, the constraint in Eq. (30f) ensures that the states predicted by the LSTM model remain trapped inside the region $\Omega_{\rho_{nn}}$ for the whole prediction horizon. Additionally, if the assumptions in Propositions 2 and 3 are satisfied with small time-delays in the states of the system, stability results derived in Wu et al. (2019a) show that using the LSTM-based LMPC of Eq. (30) guarantees that the closed-loop state of the perturbed nonlinear system of Eq. (2) with $\xi_2 = 0$, under the control law $u = \boldsymbol{\Phi}_{nn} \in U$, is bounded within the stability region $\Omega_{\hat{\rho}}$ and ultimately bounded within a small region around the origin, $\Omega_{\rho_{\min}}$, for all $t \geq 0$ and any initial state $x_0 \in \Omega_{\hat{\rho}}$.

5. Predictor feedback LSTM-based LMPC methodology

The LSTM-based LMPC is proven to be robust for systems that have sufficiently small state delays. Robustness can be enhanced by tuning some parameters in the LSTM-based LMPC controller (Ellis and Christofides, 2015), such as the weights in the cost function of Eq. (30a) or the parameter ρ_{nn}. Hence, the parameters can be chosen to ensure a margin of robustness of the closed-loop system in the presence of state delays. On the other hand, input delays are more challenging, and require further modifications in the controller structure.

In this section, we will present a predictor feedback LSTM-based LMPC methodology, and how the predictor is incorporated within the closed-loop system to compensate for the effect of input delays. As the name indicates, it is an LSTM-based predictor, in the sense that it uses an LSTM model to predict the evolution of the future states of the process up to a future time equal to the input delay. Specifically, at sampling time t_k, the predictor is used to predict the future state at time $t_k + d_2$, utilizing past state values and the input trajectory that has been calculated previously over t_k to $t_k + d_2$. Additionally, an LMPC formulation with the shifted timescale, $\bar{t}_k = k\Delta + d_2$, is used to calculate the future input trajectory from $\bar{t}_k$ to $\bar{t}_{k+N}$:

$$\mathcal{J} = \min_{u \in S(\Delta)} \int_{\bar{t}_k}^{\bar{t}_{k+N}} L_{MPC}(\tilde{x}(t), u(t)) \, dt \tag{31a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = F_{nn}(\tilde{x}(t), u(t)) \tag{31b}$$

$$u(t) \in U, \ \forall \ t \in [\bar{t}_k, \bar{t}_{k+N}) \tag{31c}$$

$$\tilde{x}(\bar{t}_k) = \bar{x}(\bar{t}_k) \tag{31d}$$

$$\dot{\hat{V}}(x(\bar{t}_k), u) \leq \dot{\hat{V}}(x(\bar{t}_k), \boldsymbol{\Phi}_{nn}(x(\bar{t}_k))),$$

$$\text{if } x(\bar{t}_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_{nn}} \tag{31e}$$

$$\hat{V}(\tilde{x}(t)) \leq \rho_{nn}, \ \forall \ t \in [\bar{t}_k, \bar{t}_{k+N}), \ \text{if } x(\bar{t}_k) \in \Omega_{\rho_{nn}} \tag{31f}$$

With respect to the LSTM-based predictor initialization, it is important to clarify the following: specifically, we need to assume initial data for both the states and inputs. For states, we assume the initial state data, from time $-d_1$ to 0, to be equal to the value of the states at time $t_k = 0$. The inputs are assumed to be at their steady state values from time 0 to d_2.

Subsequently, at sampling time t_k or, in other words, $\bar{t}_k - d_2$, the predictor receives the state measurement $x(\bar{t}_k - d_2)$ $\left(\text{or } x(t_k)\right)$ along with past input measurements from time $\bar{t}_k - d_2$ to $\bar{t}_k$ as its inputs. The LSTM-based predictor is then used to predict the future state value $\bar{x}(\bar{t}_k)$. Subsequently, the output of the predictor is sent to the LSTM-based LMPC to initialize it and compute the optimal input trajectory along the whole prediction horizon. The computed control action $u^*(\bar{t}_k | \bar{t}_k - d_2)$ is then sent to the process to be applied from time $\bar{t}_k$ to $\bar{t}_{k+1}$, which yields the output, i.e., the process state $x(\bar{t}_{k+1} - d_2)$ $\left(\text{or } x(t_{k+1})\right)$. Fig. 2 illustrates the proposed LSTM-based LMPC framework. Additionally, the LSTM-predictor used in this study is a closed-loop predictor (i.e., at each sampling time, a new measurement $x(t_k)$ is sent to it from the process). Hence, unlike open-loop predictors, closed-loop ones play an effective role when trying to control processes with open-loop unstable equilibrium points (Ellis and Christofides, 2015). Fig. 3 shows the LSTM-based predictor block in the feedback loop of the closed-loop system.

We summarize the implementation of the LSTM based predictor in the following algorithm:

Algorithm 1. LSTM-based predictor, MPC feedback implementation.

1. At sampling time t_k (i.e., $\bar{t}_k - d_2$), the predictor receives $x(\bar{t}_k - d_2)$ and past input measurements from time $\bar{t}_k - d_2$ to $\bar{t}_k$.
2. The predictor predicts the future state $\bar{x}(\bar{t}_k)$.
3. The LSTM-based LMPC is then initialized with the predicted state $\bar{x}(\bar{t}_k)$, and the optimal control input trajectory is computed.
4. The computed control action, $u^*(\bar{t}_k | \bar{t}_k - d_2)$, is then applied to the process from $\bar{t}_k$ to $\bar{t}_{k+1}$.
5. Set $k \longleftarrow k + 1$ and go to step 1.

6. Application to a chemical process example

To illustrate the use of the LSTM-based LMPC and the LSTM-based predictor for stabilizing a nonlinear system in the presence of small time-delays, we consider the chemical reactor in Ellis and Christofides (2015). In a well-mixed, non-isothermal continuous stirred tank reactor (CSTR), the irreversible, exothermic, and elementary second order reaction transforming a reactant A to a desired product B ($A \rightarrow B$) takes place. Fig. 4 shows the process flow diagram of the CSTR.

The inlet stream enters the reactor with a flow rate $\lambda\psi$, feed concentration C_{A0}, and a feed temperature T_f. The outlet stream of the reactor is split into two streams. The first stream is the product of the reactor, with a volumetric flow rate $\lambda\psi$, concentration C_A, and temperature T. The second stream is a recycle stream with flow rate $(1-\lambda)\psi$ that is carried back to the reactor. Specifically, the unprocessed portion of chemical A is reused through the recycle steam, where it carries a splitting fraction $(1-\lambda)$ of the outlet stream back to the reactor. This recycle stream causes a transportation lag. Hence, a time-delay of value d_1 appears in the dynamics of the process. C_{A0} and Q are the feed concentration and the heat rate, respectively, which are the manipulated inputs of the CSTR. The control actuators' dynamics and their operation with dead-times cause an input delay of value d_2, which appear in the process dynamics.

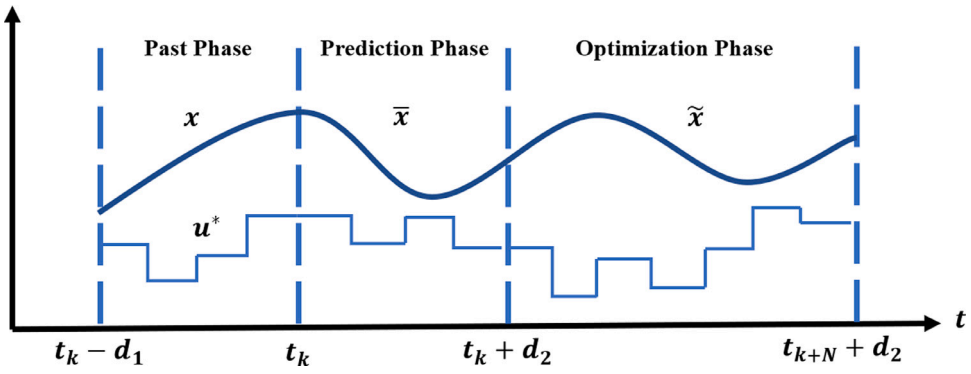

Fig. 2. Time phases of the states and the control action of the predictor-based control system.

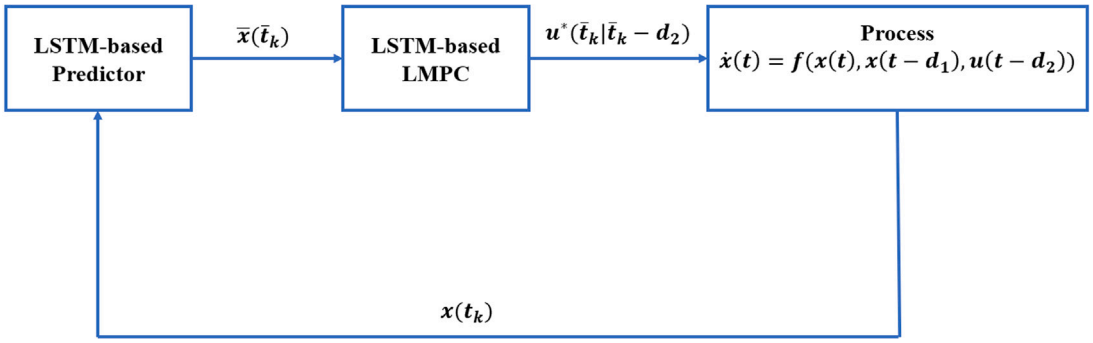

Fig. 3. Flow diagram of the closed-loop system with the predictor block.

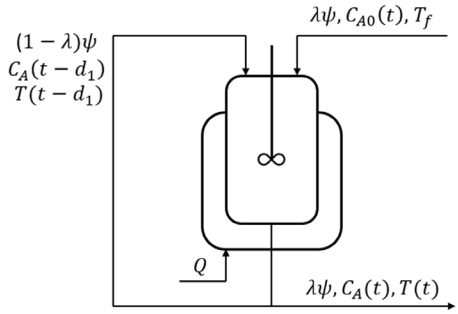

Fig. 4. Process flow diagram of the CSTR with the recycle stream.

The first-principles model of the second-order CSTR with a recycle stream is described by the following material and energy balance equations:

$$\dot{C}_A(t) = \frac{(1-\lambda)\psi}{V_R} C_A(t-d_1) + \frac{\lambda\psi}{V_R} C_{A0}(t-d_2) - \frac{\psi}{V_R} C_A(t) - k_0 \exp\left(\frac{-E}{RT(t)}\right) C_A^2(t) \tag{32a}$$

$$\dot{T}(t) = \frac{(1-\lambda)\psi}{V_R} T(t-d_1) + \frac{\lambda\psi}{V_R} T_f - \frac{\psi}{V_R} T(t) - \frac{\Delta H k_0}{\rho_L C_p} \exp\left(\frac{-E}{RT(t)}\right) C_A^2(t) + \frac{Q(t-d_2)}{V_R \rho_L C_p} \tag{32b}$$

The vector $x^T = [C_A \quad T]$ represents the state vector, where C_A is the concentration of reactant A, and T is the temperature of the reactor. The notations and the parameter values are illustrated in Table 1. The inputs of the reactor are bounded as follows: $C_{A0} \in [0.5, 7.5]$ kmol/m^3 and $Q \in [-8 \times 10^4, 8 \times 10^4]$ kJ/h. Additionally, the LSTM-based MPC is designed to drive the system to the steady state, $x_s = (C_{As}, T_s) = (2.96 \text{ kmol/m}^3, 320 \text{ K})$, which is open-loop asymptotically stable. This is achieved under the input values $C_{A0s} = 4$ kmol/m^3 and $Q_s = 12.2 \times 10^3$ kJh^{-1}. By converting both the state and input variables to deviation variables, the steady state of the system is shifted to the origin. For the development of the LSTM model, we follow the technique illustrated in Section 2.5, where 10^5 data points are generated through extensive open-loop simulations of the nonlinear system of Eq. (32) (without the delays) using the explicit Euler method with a sufficiently small time step of $h_c = 10^{-4}$ h and sampling period $\Delta = 0.01$ h. The data set is then split into 80,000 points for training and 20,000 points for validation. Then, using the machine learning library Keras, we construct the LSTM model of Eq. (6), consisting of 600 LSTM units. The loss function was chosen to be the mean squared error (MSE). Additionally, during each epoch of the training, both the training and the validation loss were calculated simultaneously. Moreover, early stopping was used, in which the stopping criterion was defined on the basis of the validation data loss and chosen as 2×10^{-6}, i.e., training would terminate once the validation loss was below 2×10^{-6}. Once the early stopping criterion was satisfied, the training was stopped and the final values of the training and validation loss were reported to be 5.4×10^{-5} and 1.4×10^{-6}, respectively; both considered to be sufficiently small. The trained LSTM model is then incorporated into a Lyapunov-based MPC (LMPC). The model's performance is then evaluated not using a test set but rather with respect to its ability to lead to an LMPC that gives satisfactory closed-loop performance and is capable of handling the delays in states and inputs, as this is the fundamental criterion when designing an MPC. The LSTM-based LMPC predicts optimal control laws for a prediction horizon of $N = 3$. Moreover, the Lyapunov function of the CSTR system is defined as

$$V(x) = (x - x_s)^T P (x - x_s) \tag{33}$$

where the matrix P is given by

$$P = \begin{bmatrix} 500 & 20 \\ 20 & 1 \end{bmatrix}$$

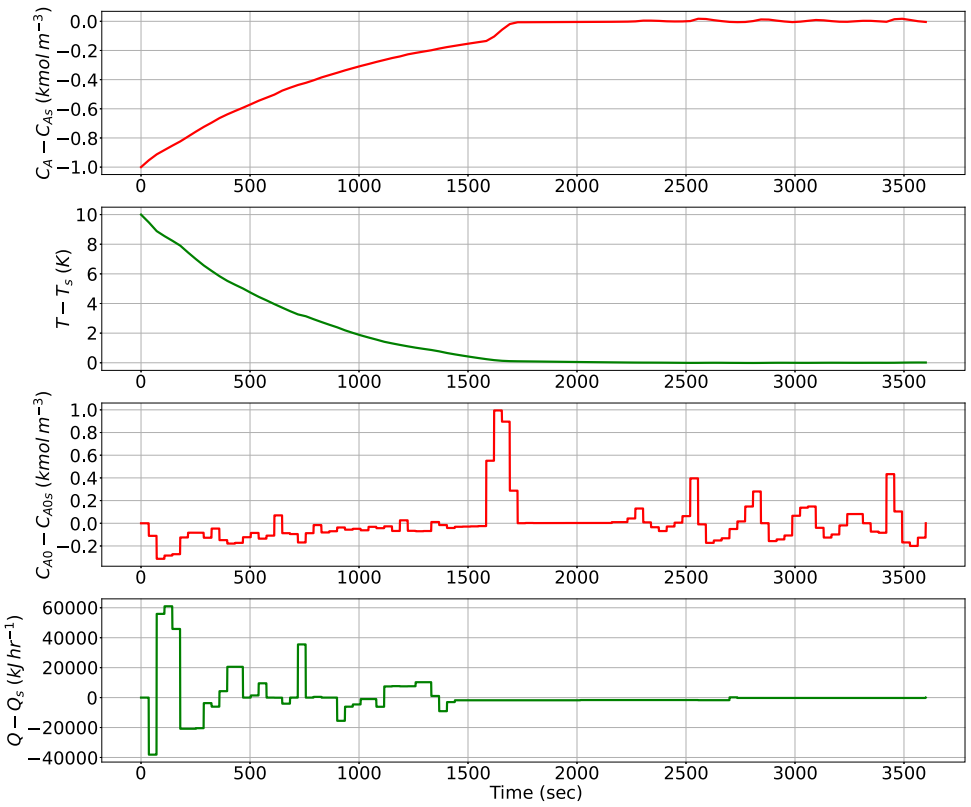

Fig. 5. Closed-loop state and input trajectories under LSTM-based LMPC with time delays: $d_1 = 0.01$ h and $d_2 = 0.01$ h.

Table 1
Notation and parameter values of the CSTR with recycle stream.

Concentration of chemical A	C_A
Reactor temperature	T
Feed concentration	C_{A0}
Heat removal rate from the reactor	Q
Splitting fraction	$\lambda = 0.7$
Reaction rate constant	$k_0 = 1 \times 10^9$ m^3 kmol^{-1} h^{-1}
Feed temperature	$T_f = 300$ K
Density	$\rho_L = 1 \times 10^3$
Heat Capacity	$C_p = 4.18$ kJ kg^{-1} K^{-1}
Reactor volume	$V_R = 1$ m^3
Flow rate	$\psi = 6$ m^3 h^{-1}
Heat of reaction	$\Delta H = -7.8 \times 10^4$ kJ kmol^{-1}
Activation Energy	$E/R = 5.7 \times 10^4/8.314$ K

In the following subsections, we will show the results of the designed LSTM-based LMPC and its ability to stabilize the system. Moreover, we will show simulation results to demonstrate the predictor feedback LSTM-based LMPC methodology and its ability to compensate for the effect of input time-delays.

6.1. LSTM-based LMPC closed-loop simulation results

We first conduct closed-loop simulations for the CSTR under the LSTM-based LMPC with different values of input time delays as they are known to have a significant impact on the state trajectories and stability. Hence, in these simulations, the value of the state delay was fixed at $d_1 = 0.01$ h for all simulations results. Fig. 5 shows the closed-loop trajectories of the CSTR under the LSTM-based LMPC with time delays of $d_1 = 0.01$ h and $d_2 = 0.01$ h. We observe that the trajectories converge to the steady-state values and are stabilized by the controller. This shows that the designed controller is robust to small time-delays, and that closed-loop stability is achieved. In this particular application, when the value of the input delay, d_2, is higher than 0.01 h, stability is lost, and we notice oscillations and fluctuations in the closed-loop trajectories as seen in Figs. 6 and 7, which correspond to input time-delays of $d_2 = 0.02$ h, and $d_2 = 0.03$ h, respectively. Moreover, from the trajectories of Figs. 6 and 7, we observe that increasing the value of the input delay d_2 increases the amplitude of the oscillations around the steady state as well. Hence, the closed-loop trajectories oscillate around the steady state, causing the system to become unstable.

6.2. Predictor feedback LSTM-based LMPC closed-loop simulation results

In this section, we will demonstrate the results of closed-loop simulations using the predictor feedback LSTM-based LMPC design. This method was proposed to overcome the performance deterioration that arises due to larger time delays, particularly to compensate for the effect of larger input time-delays (i.e., $d_2 > 0.01$ h). The scheme was applied for the cases in Section 6.1 where we found oscillations in the closed loop stability under the LSTM-based LMPC. Figs. 8 and 9 show the closed-loop trajectories of the CSTR under the predictor feedback LSTM-based LMPC with input time delays of $d_2 = 0.02$ h and $d_2 = 0.03$ h, respectively, while maintaining $d_1 = 0.01$ h in all simulations. From the results, we observe, under larger values of input time-delays, significant improvement in the closed-loop performance under the proposed control system. This is achieved through the use of an LSTM-based predictor together with the LSTM-based LMPC in the feedback loop. As the trajectories converge to their steady state values without oscillations, the process is considered to be stabilized.

7. Conclusion

In this work, we considered a nonlinear time-delay system expressed using nonlinear differential difference equations, and we approximated it with a perturbed nonlinear system with bounded perturbations. First, we introduced Long Short Term Memory Recurrent Neural Networks (LSTMs) to model the system dynamics in the absence of time delays

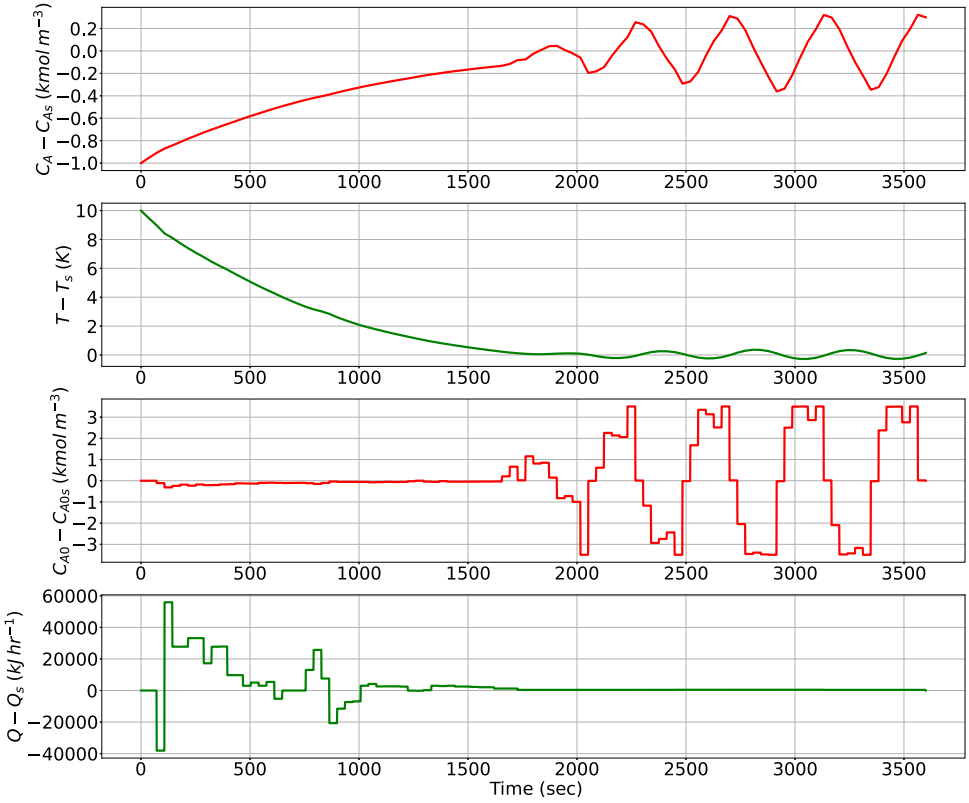

Fig. 6. Closed-loop state and input trajectories under LSTM-based LMPC with time delays $d_1 = 0.01$ h and $d_2 = 0.02$ h.

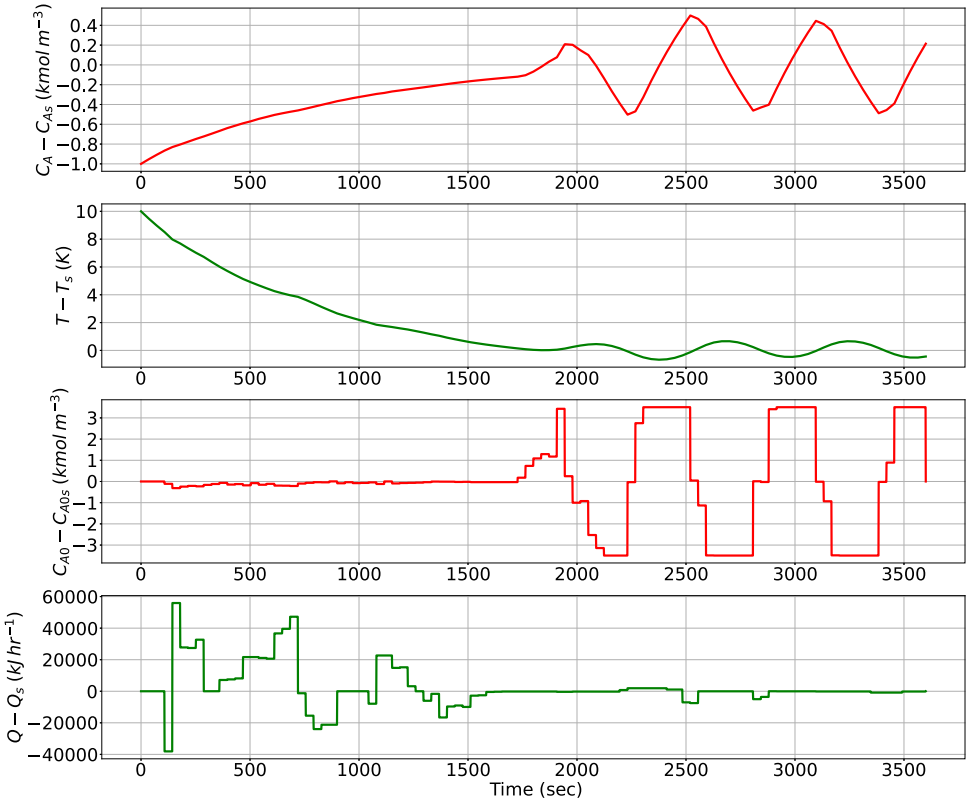

Fig. 7. The closed-loop trajectories of the CSTR under LSTM-based LMPC with time delays: $d_1 = 0.01$ h and $d_2 = 0.03$ h.

and used this LSTM model to construct an LSTM-based MPC. In the presence of small state delays, we established closed-loop stability under the LSTM-based MPC. Subsequently, the LSTM network was used to develop a closed-loop LSTM-based predictor, that compensated

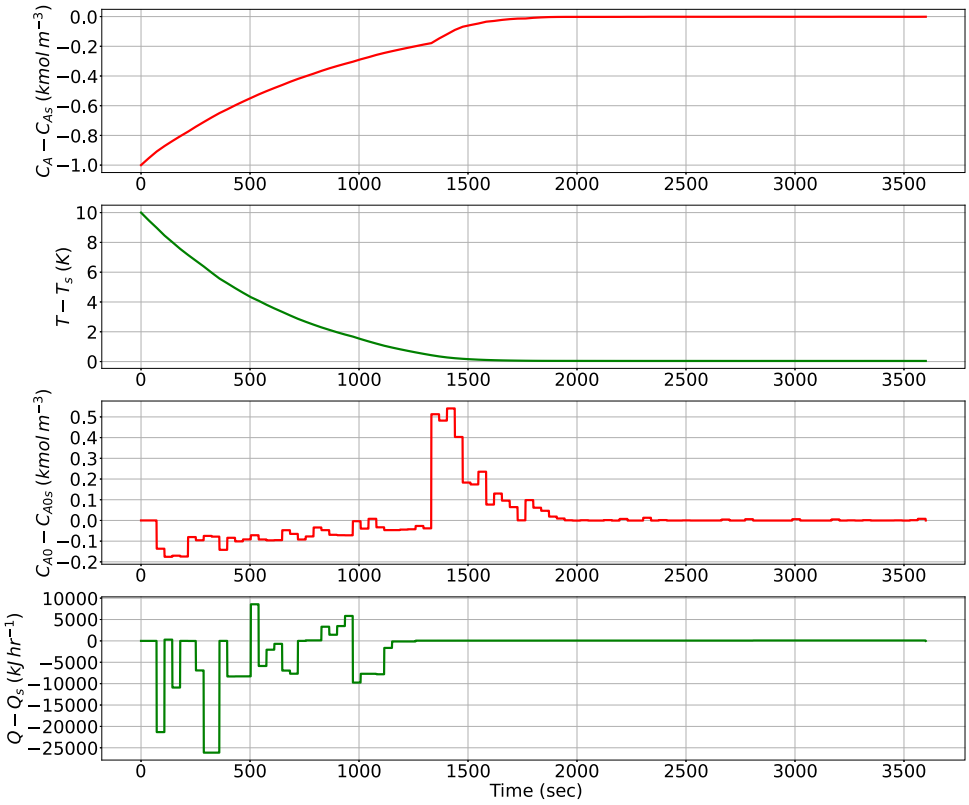

Fig. 8. Closed-loop state and input trajectories under the predictor feedback LSTM-based LMPC, where the time delays: $d_1 = 0.01$ h and $d_2 = 0.02$ h.

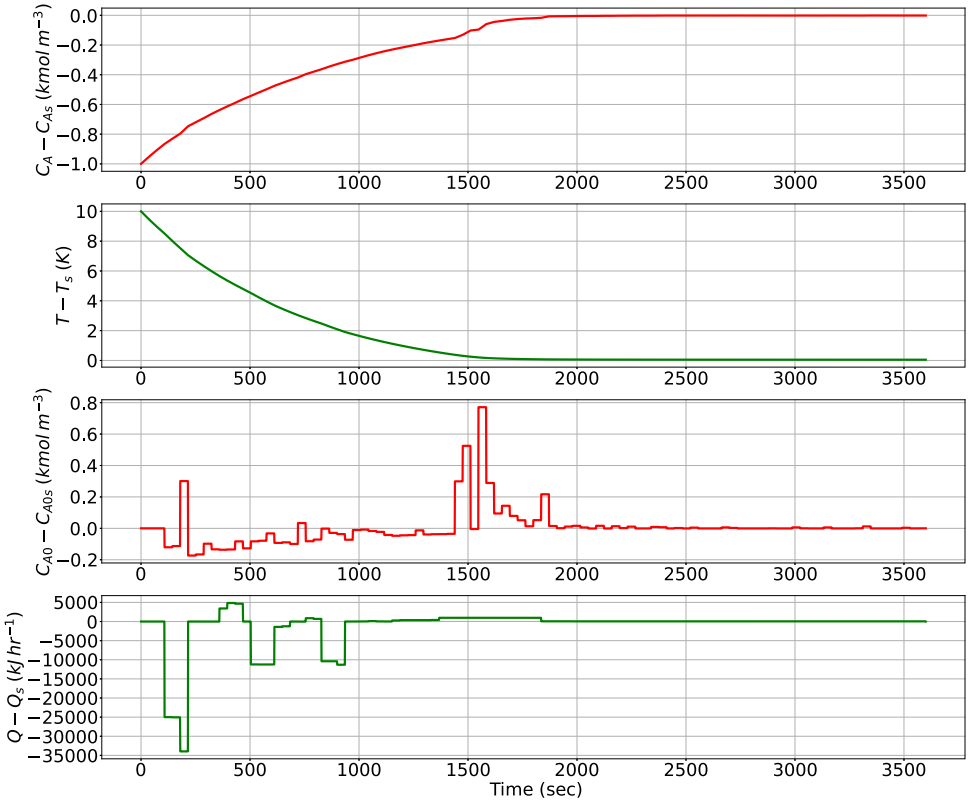

Fig. 9. Closed-loop state and input trajectories under the predictor feedback LSTM-based LMPC, where the time delays: $d_1 = 0.01$ h and $d_2 = 0.03$ h.

for the effect of input time-delays. Finally, we applied the proposed control schemes to a chemical reactor example with time-delays and demonstrated their ability to stabilize the closed-loop system under small and large state and input time-delays.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Financial support from the National Science Foundation, USA and the Department of Energy, USA is gratefully acknowledged. The first and the third authors acknowledge Kuwait University support via the KU-scholarship program.

References

Abdullah, F., Wu, Z., Christofides, P.D., 2022. Handling noisy data in sparse model identification using subsampling and co-teaching. Comput. Chem. Eng. 157, 107628.

Alhajeri, M.S., Abdullah, F., Wu, Z., Christofides, P.D., 2022. Physics-informed machine learning modeling for predictive control using noisy data. Chem. Eng. Res. Des. 186, 34–49.

Alhajeri, M.S., Wu, Z., Rincon, D., Albalawi, F., Christofides, P.D., 2021. Machine-learning-based state estimation and predictive control of nonlinear processes. Chem. Eng. Res. Des. 167, 268–280.

Antoniades, C., Christofides, P.D., 1999a. Feedback control of nonlinear differential difference equation systems. Chem. Eng. Sci. 54, 5677–5709.

Antoniades, C., Christofides, P.D., 1999b. Robust control of nonlinear time-delay systems. Int. J. Appl. Math. Comput. Sci. 9, 811–837.

Bishop, C.M., 1995. Training with noise is equivalent to Tikhonov regularization. Neural Comput. 7, 108–116.

Chen, X., Heidarinejad, M., Liu, J., Christofides, P.D., 2012. Distributed economic MPC: Application to a nonlinear chemical process network. J. Process Control 22, 689–699.

Chen, S., Wu, Z., Christofides, P.D., 2020. Decentralized machine-learning-based predictive control of nonlinear processes. Chem. Eng. Res. Des. 162, 45–60.

Ellis, M., Christofides, P.D., 2015. Economic model predictive control of nonlinear time-delay systems: Closed-loop stability and delay compensation. AIChE J. 61, 4152–4165.

Hale, J.K., Lunel, S.M.V., 2013. Introduction to Functional Differential Equations, Vol. 99. Springer Science & Business Media.

Henson, M.A., Seborg, D.E., 1994. Time delay compensation for nonlinear processes. Ind. Eng. Chem. Res. 33, 1493–1500.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.

Hoskins, J., Himmelblau, D., 1992. Process control via artificial neural networks and reinforcement learning. Comput. Chem. Eng. 16, 241–251.

Kravaris, C., Wright, R.A., 1989. Deadtime compensation for nonlinear processes. AIChE J. 35, 1535–1542.

Liu, J., Munoz de la Pena, D., Christofides, P.D., Davis, J.F., 2009. Lyapunov-based model predictive control of nonlinear systems subject to time-varying measurement delays. Internat. J. Adapt. Control Signal Process. 23, 788–807.

Ren, Y.M., Alhajeri, M.S., Luo, J., Chen, S., Abdullah, F., Wu, Z., Christofides, P.D., 2022. A tutorial review of neural network modeling approaches for model predictive control. Comput. Chem. Eng. 165, 107956.

Richalet, J., Rault, A., Testud, J., Papon, J., 1978. Model predictive heuristic control: Applications to industrial processes. Automatica 14, 413–428.

Smith, O.J., 1957. Closed control of loop with dead time. Chem. Eng. Progr. 53, 217–219.

Venkatasubramanian, V., 2019. The promise of artificial intelligence in chemical engineering: Is it here, finally? AIChE J. 65, 466–478.

Vepa, R., 1993. A review of techniques for machine learning of real-time control strategies. Intell. Syst. Eng. 2, 77–90.

Wu, Z., Alnajdi, A., Gu, Q., Christofides, P.D., 2022. Statistical machine-learning–based predictive control of uncertain nonlinear processes. AIChE J. 68, e17642.

Wu, Z., Luo, J., Rincon, D., Christofides, P.D., 2021a. Machine learning-based predictive control using noisy data: Evaluating performance and robustness via a large-scale process simulator. Chem. Eng. Res. Des. 168, 275–287.

Wu, Z., Rincon, D., Gu, Q., Christofides, P.D., 2021b. Statistical machine learning in model predictive control of nonlinear processes. Mathematics 9, 1912.

Wu, Z., Tran, A., Rincon, D., Christofides, P.D., 2019a. Machine learning-based predictive control of nonlinear processes. Part I: Theory. AIChE J. 65, e16729.

Wu, Z., Tran, A., Rincon, D., Christofides, P.D., 2019b. Machine learning-based predictive control of nonlinear processes. Part II: Computational implementation. AIChE J. 65, e16734.

Zhang, X.-M., Han, Q.-L., Seuret, A., Gouaisbaut, F., He, Y., 2019. Overview of recent advances in stability of linear systems with time-varying delays. IET Control Theory Appl. 13, 1–16.