Received: 15 April 2021 | Revised: 1 August 2021 | Accepted: 14 September 2021

DOI: 10.1002/aic.17456

AIChE JOURNAL

RESEARCH ARTICLE

Process Systems Engineering

Machine-learning-based construction of barrier functions and models for safe model predictive control

Scarlett Chen[1] | Zhe Wu[1] | Panagiotis D. Christofides[1,2]

[1]Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, California, USA

[2]Department of Electrical and Computer Engineering, University of California, Los Angeles, California, USA

Correspondence
Panagiotis D. Christofides, Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095, USA.
Email: pdc@seas.ucla.edu

Funding information
Department of Energy; National Science Foundation

Abstract

In this paper, we propose a control Lyapunov-barrier function-based model predictive control method utilizing a feed-forward neural network specified control barrier function (CBF) and a recurrent neural network (RNN) predictive model to stabilize nonlinear processes with input constraints, and to guarantee that safety requirements are met for all times. The nonlinear system is first modeled using RNN techniques, and a CBF is characterized by constructing a feed-forward neural network (FNN) model with unique structures and properties. The FNN model for the CBF is trained based on data samples collected from safe and unsafe operating regions, and the resulting FNN model is verified to demonstrate that the safety properties of the CBF are satisfied. Given sufficiently small bounded modeling errors for both the FNN and the RNN models, the proposed control system is able to guarantee closed-loop stability while preventing the closed-loop states from entering unsafe regions in state-space under sample-and-hold control action implementation. We provide the theoretical analysis for bounded unsafe sets in state-space, and demonstrate the effectiveness of the proposed control strategy using a nonlinear chemical process example with a bounded unsafe region.

KEYWORDS

control Lyapunov-barrier functions, neural networks, nonlinear model predictive control, process safety

1 | INTRODUCTION

The severity of potential hazards involved and the close interaction between human lives and environment make safety a top priority in any industrial plant operations. The catastrophic outcomes of these incidents alarm us of the importance of maintaining, designing, and implementing stricter and more robust process and operational safety measures.[1] One way to ensure this is by designing a comprehensive and robust process control system that not only maintains stable production and economic optimality, but also handles unexpected production scenarios that could lead to unsafe operating conditions and environmental hazards. In addition to configuring alarming thresholds on process variables, the interactions between multiple process variables in a large-scale complex plant and their impact on the operational safety of the system should also be considered. Hence, MPC has been proposed as an advanced control methodology to account for multivariable interactions, variable and safety constraints, and nonlinearities in industrial plants.

Amongst many MPC formulations, Lyapunov-based MPC (LMPC) ensures feasibility and stabilizability within an explicitly defined stability region using a Lyapunov-based stabilizing control law.[2,3] Previously, safety considerations have been incorporated in the design of LMPC algorithms to specify unsafe regions of operation in state space characterized by the relative safeness of process states,[4] as well as to ensure that unsafe regions are avoided at all times by utilizing a control Lyapunov-barrier function (CLBF).[5,6] CLBFs are developed from the combination of a control Lyapunov function (CLF) and a control barrier function (CBF), and can be used in control algorithms to account for both stability and safety, respectively.[7–9] Unsafe regions can be characterized by CBFs, which were proposed in many literature works[10–12] to ensure closed-loop safe performance. Amongst the many advanced control methods, CLBF has been included as part of the MPC formulation in Wu et al.[13] to account for input constraints, safety considerations, and the stability of the closed-loop system.

One challenge of implementing CLBF-based controllers is whether the unsafe operating region can be explicitly and accurately represented in closed form as a function of process states. While this may be possible to do for simple shapes or patterns of the unsafe region, it is practically difficult to express such a barrier function for real industrial processes with complex unsafe regions that cannot be readily described with common explicit functions. To this end, feed-forward neural networks (FNNs) can be used to model barrier functions based on data samples collected from the safe and the unsafe operating regions.

Neural networks (NNs) have a proven record of success in solving both classification and regression problems, whether it be via supervised or unsupervised approaches. According to the universal approximation theorem, NNs with sufficient number of neurons are able to approximate any nonlinear functions on compact subsets of the state space.[14,15] Many previous works have been developed to incorporate various machine learning modeling approaches with the design of MPC.[16–18] In this work, in addition to using a recurrent neural network (RNN) as the prediction model in the MPC, we also characterize the CLBF using an FNN model. Provided with extensive training data which are labeled, NN models can be constructed with strategically chosen architectures, activation and objective functions, and evaluation metrics, and ultimately trained with supervision to approximate the barrier function. The FNN-specified barrier function can be proven to satisfy all required conditions of a barrier function, and can then be applied to the CLBF-based controllers. In our study, we consider a CLBF-MPC, where the barrier function is found using FNN structures.

The remaining paper is organized as follows. Preliminaries on the class of systems considered, the stabilizability assumptions and safety considerations given by CLBF are described in Section 2. We introduce the structure and the development of the NN model in Section 3, along with proofs of its efficacy when applied in the CLBF-based controllers. In Section 4, the formulation of the CLBF-MPC with NN-specified BF is presented, where the proof for recursive feasibility of the optimization problem, as well as the boundedness and convergence of the closed-loop state while always avoiding the unsafe region is shown, given bounded modeling error of the NN-BF, sample-and-hold implementation of control actions, and a well-characterized set of initial conditions. Lastly, in Section 5, the control method proposed in this work is applied to a chemical process example to illustrate its effectiveness.

2 | PRELIMINARIES

2.1 | Notation

We use $|\cdot|$ to denote the Euclidean norm of a vector. $L_fV(x) := \frac{\partial V(x)}{\partial x} f(x)$ denotes the standard Lie derivative. Furthermore, a scalar continuous function $V : \mathbf{R}^n \rightarrow \mathbf{R}$ is proper if for all $k \in \mathbf{R}$, the set $\{x \in \mathbf{R}^n \,|\, V(x) \leq k\}$ is a compact set. x^T denotes the transpose of x. $\mathcal{B}_\beta(\varepsilon) := \{x \in \mathbf{R}^n \,| x - \varepsilon| < \beta|\}$ is an open ball around ε with radius of β, with positive real numbers β and ε. Set subtraction is denoted by "$\backslash$", that is, $A \backslash B := \{x \in \mathbf{R}^n \mid x \in A, x \notin B\}$. $\emptyset$ signifies the null set. Lastly, a function $f(\cdot)$ is of class $\mathcal{C}^1$ if it is continuously differentiable.

2.2 | Class of systems

The class of continuous-time nonlinear systems considered is described by the following state-space form:

$$\dot{x} = f(x) + g(x)u + h(x)w, \;\; x(t_0) = x_0, \tag{1}$$

where $x \in \mathbf{R}^n$ represents the state vector, $u \in \mathbf{R}^m$ represents the input vector, and $w \in \mathbf{W}$ is the bounded disturbance vector, where $\mathbf{W} := \left\{ w \in \mathbf{R}^l \,|\, |w| \leq \theta, \theta \geq 0 \right\}$. The input control actions are constrained by their lower and upper bounds, $u \in U := \{u_{\text{min}} \leq u \leq u_{\text{max}}\} \subset \mathbf{R}^m$. $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$ are vector and matrix functions of dimensions $n \times 1$, $n \times m$, and $n \times l$, respectively, and we assume that they are sufficiently smooth. Without loss of generality, we take the initial time t_0 to be zero, that is, $t_0 = 0$. It is assumed that $f(0) = 0$. Thus, the system of Equation (1) with $w(t) \equiv 0$ has a steady state at the origin. Additionally, it is assumed the feedback measurement of $x(t)$ is available at synchronous sampling times, t_k.

2.3 | Stabilizability assumptions expressed via Lyapunov-based control

For the nominal system of Equation (1) with $w(t) \equiv 0$, we assume that there exists a positive definite and proper CLF, V, that satisfies the small control property as well as the following conditions:

$$L_fV(x) < 0, \forall x \in \left\{ z \in \mathbf{R}^n \backslash \{0\} \,|\, L_gV(z) = 0 \right\}. \tag{2}$$

The small control property states that for every $\varepsilon > 0$, $\exists \delta > 0$, s.t. $\forall x \in \mathcal{B}_\delta(0)$, there exists u that satisfies $|u| < \varepsilon$ and $L_fV(x) + L_gV(x) \cdot u < 0$.[19] The existence of such CLF implies that there exists a stabilizing feedback control law $\Phi(x) \in U$ for the nominal system of Equation (1) such that Equation (2) holds for $u = \Phi(x) \in U$, and the origin of the closed-loop system is rendered asymptotically stable for all x in a neighborhood around the origin under $u = \Phi(x) \in U$. A candidate of a stabilizing feedback control law is shown in Lin and Sontag.[20] We can characterize a region ϕ_u where the time derivative of $V(x)$ is rendered negative using $u = \Phi(x) \in U$ as: $\phi_u = \left\{ x \in \mathbf{R}^n \mid \dot{V}(x) = L_fV(x) + L_gV(x) \cdot u < 0, u = \Phi(x) \in U \right\} \cup \{0\}$. Within this region ϕ_u, we define a level set of $V(x)$, $\Omega_b := \{x \in \phi_u \mid V(x) \leq b, b > 0\}$, which is a forward invariant set such that the closed-loop trajectory $x(t), t \geq 0$ of the nominal system of Equation (1) with $w(t) \equiv 0$ under $u = \Phi(x) \in U$ remains in Ω_b, for any initial condition $x_0 \in \Omega_b$.

2.4 | Process modeled using RNN

When first-principles models of a process are not available or may not be accurate, one way to model the process is to use data-based machine-learning methods. A RNN is an effective algorithm that is capable of modeling the dynamics of the nonlinear system of Equation (1), and its general formulation is shown as follows:

$$\dot{\hat{x}} = F_{nn}(\hat{x},u) := A\hat{x} + \Theta^T y, \tag{3}$$

where $\hat{x} \in \mathbf{R}^n$ is the state vector of the RNN model and $u \in \mathbf{R}^m$ is the manipulated input vector. $y = [y_1,\ldots,y_n,y_{n+1},\ldots,y_{n+m}] = [\sigma(\hat{x}_1),\ldots,\sigma(\hat{x}_n),u_1,\ldots,u_m] \in \mathbf{R}^{n+m}$ is a vector that contains both the network state $\hat{x}$ and the input u, where $\sigma(\cdot)$ is the nonlinear activation function. A is a diagonal coefficient matrix, that is, $A = diag\{-a_1,\ldots,-a_n\} \in \mathbf{R}^{n\times n}$, and $\Theta = [\theta_1,\ldots,\theta_n] \in \mathbf{R}^{(m+n)\times n}$ with $\theta_i = b_i\left[w_{i1},\ldots,w_{i(m+n)}\right]$, $i = 1,\ldots,n$. a_i and b_i are constants. w_{ij} represents the weight connecting the jth input to the ith neuron where $i = 1,\ldots,n$ and $j = 1,\ldots,(m+n)$. a_i is assumed to be positive for each state $\hat{x}_i$ to be bounded-input bounded-state stable. For the remainder of the manuscript, x will be used to denote the state of the nonlinear system of Equation (1), and $\hat{x}$ will be used to denote the state of the RNN model of Equation (3).

As the RNN model of Equation (3) is an input-affine system, it can be also written in the form that is similar to the general nonlinear system of Equation (1):

$$\dot{x} = \hat{f}(x) + \hat{g}(x)u, \tag{4}$$

where $\hat{f}(\cdot)$ and $\hat{g}(\cdot)$ can be derived from the coefficient matrices A and Θ in Equation (3) and are assumed to be sufficiently smooth. The construction of RNN models including procedures on data generation, model training and validation, as well as developing an ensemble of models have been outlined in Wu et al.[21] Note that $\hat{f}(\cdot)$ and $\hat{g}(\cdot)$ can be approximated via numerical methods. The modeling error of the RNN, $|\nu|$, needs to be below a certain threshold ν_m during training, and is bounded as follows: $|\nu| = |F(x,u,0) - F_{nn}(x,u)| \leq \gamma|x| \leq \nu_m$, where $\gamma > 0$. The bounded modeling error is a requirement to ensure that the nonlinear system of Equation (1) and the RNN model of Equation (3) have the same steady-state within the operating region considered, and is a requirement used in subsequent stability and safety proofs. Furthermore, we assume that there exists a CLF $\hat{V}$ and a Lyapunov-based stabilizing control law $u = \Phi_{nn}(x) \in U$ that renders the origin of the RNN modeled system of Equation (3) asymptotically stable.

2.5 | Control barrier function

We assume that there exists an open set $\mathcal{D}$ in state-space that should be avoided during operations; for example, the operating conditions within this region may result in process safety risks. We also characterize a set of safe states, $\mathcal{X}_0 := \{x \in \mathbf{R}^n \backslash \mathcal{D}\}$ where $\{0\} \in \mathcal{X}_0$ and $\mathcal{X}_0 \cap \mathcal{D} = \emptyset$. The set of initial conditions to be considered in this study will be developed from $\mathcal{X}_0$.

Two types of unsafe regions are generally considered in literature—bounded and unbounded sets—the details of which can be found in Wu and Christofides.[6] We denote bounded unsafe set and unbounded unsafe set as $\mathcal{D}_b$ and $\mathcal{D}_u$, respectively. Due to the data-driven approach of constructing the CBF, there are relevant restrictions with collecting finite samples from compact sets of safe and unsafe data. Therefore, only bounded unsafe sets can be handled in this approach. Details of the limitations on the compactness of the unsafe set will be further presented in Section 3.2.2. We address process operational safety in the sense of not entering any unsafe sets. The formal definition of process operational safety is defined as follows:

Definition 1. Consider the nominal system of Equation (1) with $w(t) \equiv 0$ and input constraints $u \in U$. If there exists a control law $u = \Phi(x) \in U$ such that, for any initial state $x(t_0) = x_0 \in \mathcal{X}_0$, the origin of the closed-loop system of Equation (1) is rendered asymptotically stable, and the state trajectories of the system do not enter the unsafe region, that is, $x(t) \in \mathcal{X}_0$, $x(t) \notin \mathcal{D}$, $\forall t \geq 0$, then the control law $u = \Phi(x)$ maintains the process state within a safe operating region $\mathcal{X}_0$ for all times.

Following the definition of safe operation, the definition of a valid CBF is as follows[22]:

Definition 2. Given a set of unsafe points in state-space $\mathcal{D}$, a $\mathcal{C}^1$ function $B(x): \mathbf{R}^n \to \mathbf{R}$ is a CBF if it satisfies the following properties:

$$B(x) > 0, \quad \forall x \in \mathcal{D}, \tag{5a}$$

$$L_f B(x) \leq 0, \forall x \in \left\{z \in \mathbf{R}^n \backslash \mathcal{D} \,|\, L_g B(z) = 0\right\}, \tag{5b}$$

$$\mathcal{X}_B := \{x \in \mathbf{R}^n \,|\, B(x) \leq 0\} \neq \emptyset. \tag{5c}$$

Remark 1. In many chemical processes, unbounded unsafe sets represent unsafe operations where process variables exceed their safety envelopes, for example, when temperature is above a threshold that can lead to overheating, or when concentration is below a threshold which could lead to incomplete reaction. Bounded unsafe sets are more common in mechanical processes; for example, robotics navigation to avoid obstacles in its trajectory. In chemical processes, many mid-range operating conditions are suboptimal to achieving high yields of reactions. For example, low pressure steam could be used as a coolant at low temperature, or as a heat source at high temperature. However, if its temperature is in the middle ranges, then it is not fit for either purpose and might be discarded as waste.

Remark 2. For many industrial operations where the dynamics of the process is not well understood, it is difficult to model the intertwined relations between multitudes of variables. Although it is possible to specify certain operating envelopes within which individual process variables should operate within, the impact of these variables on other variables, and vice versa, may not be preassessed and therefore, cannot be explicitly described. It is common for plant operators to provide data points at or near which operation would be avoided. With these data points, we can use the approach discussed in this manuscript to model a CBF.

3 | CONSTRUCTION OF BARRIER FUNCTION USING NEURAL NETWORKS

3.1 | Neural network structure and training

In our study, we use a feed-forward artificial neural network (FNN) to synthesize the CBF $\hat{B}(x)$. A conventional FNN consists of an input layer, an output layer, and any number of hidden layers in between that can be customized depending on network complexity and computational need. Each layer undergoes nonlinear transformations, which consists of activation functions of a bias term plus the weighted sum of neurons in the previous layer. In turn, the results of these activation functions provide the values of the neurons in the current layer. The neurons in the first hidden layer are derived from the input layer, and the outputs are calculated based on the neurons in the last hidden layer. The input layer contains the state vector x of the nonlinear system of Equation (1) with a dimension of $\mathbf{R}^n$, and the single output in the output layer provides the predicted barrier function $\hat{B}(x)$ for the particular input data sample x.

Without having prior knowledge on an explicit formulation of the barrier function $B(x)$, training data for the NN will be collected for both the safe and unsafe regions with target values of $B(x)$ that satisfy the conditions of Equations (5a) and (5c) for each region respectively. We choose nonlinear activation functions that will best fit the dichotomous nature of the barrier function, which will aid in obtaining better prediction accuracy. Furthermore, we encode custom loss function and evaluation metric for the FNN in order to ensure that the condition of Equation (5b) is also satisfied. Since this approach is data-driven and dependent on the sampling of training data generation, we also provide formal proof for the verification of the FNN-learned barrier function $\hat{B}(x)$, proving that the $\hat{B}(x)$ indeed satisfies all conditions of Equations (5a)–(5c). The structure of a 2-hidden-layer FNN is presented in Figure 1 and in Equations (6a)–(6c):

$$\theta_j^{(1)} = g_1\left(\sum_{i=1}^{n} w_{ij}^{(1)} x_i + b_j^{(1)}\right), \tag{6a}$$

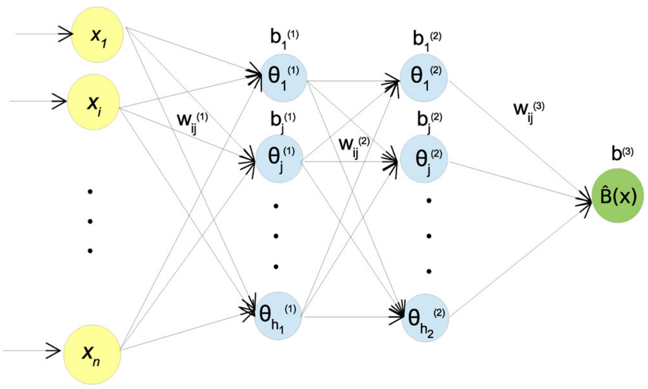

FIGURE 1 Structure of a 2-hidden-layer feedforward neural network with the state vector $x \in \mathbf{R}^n$ as inputs and the CBF $\hat{B}(x)$ as the output

$$\theta_j^{(2)} = g_2\left(\sum_{i=1}^{h_1} w_{ij}^{(2)} \theta_i^{(1)} + b_j^{(2)}\right), \tag{6b}$$

$$\hat{B} = g_3\left(\sum_{i=1}^{h_2} w_i^{(3)} \theta_i^{(2)} + b^{(3)}\right), \tag{6c}$$

with $\theta_j^{(1)}$ and $\theta_j^{(2)}$ representing neurons in the first and second hidden layer, respectively, where $j = 1,\ldots,h_l$ is the number of neurons in layer $l = 1$ and $l = 2$. The weight associated with the connections between neurons i and j, which are in consecutive layers (from $l-1$ to l), is denoted by $w_{ij}^{(l)}$, and $b_j^{(l)}$ represents the bias term added to the weighted sum for each neuron in hidden layers $l = 1,2$ and output layer $l = 3$. Upon receiving the information from the previous layer, neurons in the current layer l then computes an output via a nonlinear activation function, denoted g_l. There are many choices of activation functions, for example, sigmoid function, $g(z) = \frac{1}{1+e^{-z}}$, hyperbolic tangent sigmoid function $g(z) = \frac{2}{1+e^{-2z}} - 1$, and ReLu function, $g(z) = max(0,z)$; interested readers may refer to[23] for more details on the different activation functions and their characteristics. The two-hidden-layer representation in this section can be similarly extended to multiple hidden layers for better fitting suited for other applications.

To train the FNN, training data are generated by sampling points from the operating region of the system (i.e., $x \in \mathcal{X} \subset \mathbf{R}^n$ where $\mathcal{X}$ is a compact set). In order to ensure that the FNN developed from discrete data samples is able to meet the conditions of $B(x)$ in a continuous sense (for reasons that will be further explained in Section 3.2), the safe and the unsafe operating regions we consider need to be compact and connected sets within $\mathcal{X}$. Therefore, we first characterize a compact and connected set $\mathcal{H}$, that encloses the open set $\mathcal{D}$ such that a key condition used for designing CBF, as shown in Equation (12), is satisfied. These design guidelines are explained in detail in Section 4.1. Then, we use $\mathcal{H}'$, which encloses $\mathcal{H}$ with sufficient margin, to represent the unsafe region. Samples from the unsafe region $\mathcal{H}'$ and the safe region $\mathcal{X} \backslash \mathcal{H}'$ are collected by discretizing the respective regions with a fixed mesh size $(\delta x)_{\mathcal{H}'}$ and $(\delta x)_{\mathcal{X} \backslash \mathcal{H}'}$ respectively. We denote the finite sampled data set of the unsafe region as $S_{\mathcal{H}'}$, and the finite sampled data set of the safe region as $S_{\mathcal{I}}$. To achieve best training results, equal number of samples for each set is obtained, where N_D and N_I represents the number of sampled data points in the unsafe and the safe regions respectively.

Due to the dichotomous condition of $B(x)$ as specified by Equations (5a) and (5c) (depending on whether the particular point x belongs to the safe or the unsafe region in state-space), the activation functions of the two hidden layers and the output layer are all chosen to be the hyperbolic tangent sigmoid function (i.e., $tanh(z)$) due to the nature of $tanh(z)$ functions settling at 1 as z approaches $+\infty$, and -1 as z approaches $-\infty$, effectively polarizing the outputs and allowing the outputs of the FNN to take either relatively constant positive values, or relatively constant negative values. According to the requirement of conditions Equations (5a) and (5c), we can then label safe data points in $S_{\mathcal{I}}$ as having an output value $B(x)$ of -1, and unsafe data points in $S_{\mathcal{H}'}$ as having an output value $B(x)$ of $+1$. These labeled target output values can be then compared to the predicted output values given by the layers of neurons and *tanh* activation functions; more specifically, we use mean squared error (MSE) in the objective function to track the error between the target $B(x)$ and the predicted $\hat{B}(x)$ values. Minimizing the MSE between the target $B(x)$ and the predicted $\hat{B}(x)$ values will address the conditions of Equations (5a) and (5c). Furthermore, we add an additional term in the cost function, which uses the *ReLu* function and penalizes sample points that violate the condition of Equation (5b). To obtain an optimal set of weights and biases that will produce an output $\hat{B}(x)$ that meets all three conditions of Equations (5a)–(5c), we use an optimization algorithm to minimize the cost function, which has the following form:

$$\begin{aligned} Cost &= Cost_1 + Cost_2,\\ Cost_1 &= \alpha\frac{1}{N_s}\sum_{k=1}^{N_s}\left(\hat{B}_k - B_k\right)^2,\\ Cost_2 &= \beta\sum_{j=1}^{N_I} ReLu\left(L_{\hat{f}}\hat{B}_j + \tau_I\right), \end{aligned} \tag{7}$$

where $Cost_1$ represents the MSE between the target and the predicted outputs for all samples in the operating region, and $Cost_2$ represents the penalizing term to ensure that $L_{\hat{f}}\hat{B} \le 0$ for all $x \in S_{\mathcal{H}}$. $k = 1,\ldots,N_s$ represents the total number of samples in the training dataset, that is, $N_s = N_D + N_I$, and $j = 1,\ldots,N_I$ represents all sample points in the safe operating region. In $Cost_2$, τ_I is a small positive constant. Since *ReLu* is defined to take the maximum between its argument and 0, we penalize any occurrences of data samples producing $L_{\hat{f}}\hat{B}_j + \tau_I > 0$, thereby forcing $L_{\hat{f}}\hat{B}_j$ to be negative for all points in the safe region. Positive constants α and β are hyper-parameters representing the weights of the two terms in the cost function. During training, when $\sum_{j=1}^{N_I} ReLu\left(L_{\hat{f}}\hat{B}_j + \tau_I\right)$ has reached 0, then we have arrived at a predicted barrier function $\hat{B}(x)$ that satisfies the condition Equation (5b). In order to ensure the efficacy of the predicted barrier function $\hat{B}(x)$ at the end of the network training, we evaluate and monitor $Cost_1$ and $Cost_2$ separately during training, and implement stopping criteria that would require both $Cost_1$ and $Cost_2$ to reach below their respective thresholds to ensure bounded modeling error for $\hat{B}(x)$ as well as negative semi-definiteness of $L_{\hat{f}}\hat{B} \le 0$ for all $x \in S_{\mathcal{I}}$.

3.2 | Effectiveness of NN-based barrier function

The definition given in Definition 2 presents the properties and characteristics of an adequate barrier function. In this section, we will show how FNN-based barrier function can be verified to satisfy Definition 2 and be applied to continuous nonlinear systems of Equation (1).

3.2.1 | Continuity and differentiability

By Definition 2, the barrier function is a continuously differentiable function, thus we need to show that $\hat{B}(x)$ and $\dot{\hat{B}}(x)$ are continuous. By the universal approximation theorem, feed-forward artificial neural networks are able to model any continuous nonlinear functions on compact subsets of the state space $\mathbf{R}^n$ with sufficient number of neurons.[15] Furthermore, $\hat{B}(x)$ is the output of a series of nonlinear activation functions of inputs, weights and biases. We choose activation functions that are Lipschitz continuous in the compact subset within which the FNN training data is collected, such as *tanh*. All hidden layers and output layer of the FNN model for approximating $B(x)$ use *tanh* as the activation function, therefore making $\hat{B}(x)$ also Lipschitz continuous.

3.2.2 | Verification

Minimizing the cost function of Equation (7) aims to minimize the error between the values of $B(x)$ and $\hat{B}(x)$ as well as to penalize violations of the decrease condition $L_{\hat{f}}\hat{B}(x) \le 0, \forall x \in S_{\mathcal{I}}$, but does not enforce the conditions of Equations (5a)–(5c) in a continuous sense. Therefore, we must verify that these conditions hold over the compact subsets for which the respective data samples are collected from. Many verification techniques can be used, such as the satisfiability modulo theories (SMT) algorithm in Chang et al.[24] and the Lipschitz method in Jin et al.[25] and Richards.[26] More specifically, the work in Bobiti and Lazar[27] has shown the verification of the decrease condition for a candidate Lyapunov function on a finite sampling of a bounded set of initial conditions. The following theorem is adapted from the work in Bobiti and Lazar,[27] in which the full proof of the theorem is presented in details.

Theorem 1. *Let S_s be a finite set sampled from a compact set $S \subset \mathbf{R}^n$ such that for all $x \in S$, there exists at least a pair $(x_s, \delta x_s) \in S_s \times \mathbf{R}_+$ s.t. $|x - x_s| \le \delta x_s$. If for all $x_s \in S_s$ it holds that $F(x_s) \le -L_F \cdot \delta x_s$ (or respectively $F(x_s) < -L_F \cdot \delta x_s$), where $L_F > 0$ is the Lipschitz constant for function F, then $F(x) \le 0$ (respectively $F(x) < 0$) holds for all $x \in S$.*[27]

Therefore, by Theorem 1, we can show that $L_{\hat{f}}\hat{B}(x) \leq 0, \forall x \in \mathcal{X}\backslash\mathcal{H}'$ by checking the tightened condition $L_{\hat{f}}\hat{B}(x) \leq -L' \cdot \delta x_{\mathcal{X}\backslash\mathcal{H}'}, \forall x \in S_{\mathcal{I}}$, where the sampled finite set $S_{\mathcal{I}}$ is a discretization of the compact set $\mathcal{X}\backslash\mathcal{H}'$, $L' > 0$ is the Lipschitz constant for $L_{\hat{f}}\hat{B}(x)$, and $\delta x_{\mathcal{X}\backslash\mathcal{H}'} > 0$ is the discretization mesh size in the safe region $\mathcal{X}\backslash\mathcal{H}'$. Similarly, we can show that $\hat{B}(x) \leq 0, \forall x \in \mathcal{X}\backslash\mathcal{H}'$ by showing that $\hat{B}(x) \leq -L \cdot \delta x_{\mathcal{X}\backslash\mathcal{H}'} \forall x \in S_{\mathcal{I}}$, where L is the Lipschitz constant for $\hat{B}$. Once this tightened condition is verified, it is sufficient to show that the condition of Equation (5c) is satisfied. Lastly, we show that the condition of Equation (5a) is satisfied by verifying that $-\hat{B}(x) < -L \cdot \delta x_{\mathcal{H}'}, \forall x \in S_{\mathcal{H}'}$ (where the sampled finite set $S_{\mathcal{H}'}$ is a discretization of the compact set $\mathcal{H}'$), which means $-\hat{B}(x) < 0 \forall x \in \mathcal{H}'$, and equivalently $\hat{B}(x) > 0 \forall x \in \mathcal{H}'$.

3.2.3 | Characterization of unsafe data

It is generally difficult to describe the exact unsafe operating conditions of nonlinear processes as the actual unsafe set $\mathcal{D}$ can be open and not connected. For example, unsafe sets are not connected if there are multiple clusters of unsafe operating regions located within close proximity such that navigating around them would be nearly impossible. Therefore, in order to proceed with designing an adequate CBF, we first characterize a compact, connected set, denoted as $\mathcal{H}$, to embed the unsafe set $\mathcal{D}$. This approach is similarly applied in the design of constrained CLBF $W_c(x)$ proposed in Wu et al.,[13] where an explicit form of the CBF was constructed. In our study, we use a similar compact and connected set $\mathcal{H}$, such that $\mathcal{D} \subset \mathcal{H}$, to characterize the set of unsafe states considered.

To obtain a FNN model for the CBF, we need to supply the model with training data samples from the unsafe and the safe operating regions in state-space. As there always exists inherent modeling error in the approximation of the CBF, a contingency margin should be considered when generating these training data. More specifically, we use a larger compact set, $\mathcal{H}'$ where $\mathcal{H} \subset \mathcal{H}'$, to distinguish the different labels assigned to safe and unsafe data samples. Data samples obtained from a discretization of the region $\mathcal{H}'$ will be labeled as "unsafe," and data samples obtained from a discretization of the set $\mathcal{X}\backslash\mathcal{H}'$ will be labeled as "safe." Upon verification of the trained model with regards to the definition of CBF (Equations (5a)–(5c)) and with regards to the classification accuracy, we ensure that the resulting unsafe region as predicted by the FNN-modeled CBF, denoted as $\hat{\mathcal{H}}$, should be as close to $\mathcal{H}'$ as possible and always be a superset of the compact unsafe region $\mathcal{H}$. We leave sufficient margin between $\mathcal{H}$ and $\mathcal{H}'$ so that, with bounded modeling error in the FNN model for CBF (Equations (6a)–(6c)) and in the RNN model for the nonlinear process (Equation (3)), it is guaranteed that the closed-loop state will not enter $\mathcal{H}$ given any initial condition $x_0 \in \mathcal{X}\backslash\mathcal{H}'$.

Remark 3. Despite rigorous training and extensive validation, there may still exist modeling error in the testing phase or in the implementation of the NN model that we cannot eliminate completely. Without knowing an explicit analytical form of $B(x)$, it is difficult to quantify such modeling error as well. We assume that the contingency margin that we leave when characterizing the set of unsafe points for which training data will be generated from is able to account for the inherent modeling error of the FNN-modeled CBF $\hat{B}(x)$. Hence, while the FNN output $\hat{B}$ aims to characterize the unsafe region boundary as close to $\mathcal{H}'$ as possible, in the presence of modeling error, the predicted $\hat{B}(x)$ will satisfy all conditions on CBF and CLBF with respect to the actual unsafe closed and compact set $\mathcal{H}$.

Remark 4. To verify that the FNN-modeled barrier function $\hat{B}(x)$ satisfies the properties of a CBF in a continuous sense, the finite sets of safe and unsafe data used to build the FNN must be sampled from a compact (i.e., closed and bounded) safe set $\mathcal{X}\backslash\mathcal{H}'$, and a compact unsafe set $\mathcal{H}'$, respectively, as shown in Section 3.2.2. It should be noted that the unsafe set $\mathcal{H}'$ is a set characterized by the user to enclose the compact set $\mathcal{H}$ to account for the error margin in the neural network model. Moreover, the compact set $\mathcal{H}$ is a set characterized by the user to enclose the actual unsafe region $\mathcal{D}$. In this study, we focus on bounded unsafe sets, $\mathcal{D}_b$. Bounded unsafe sets in the middle of the operating region could obstruct the state trajectories, and are therefore the more difficult case to handle. In the case of unbounded unsafe sets, $\mathcal{D}_u$, they must be first approximated by a sufficiently large compact set within a reasonable physical range, $\mathcal{D}_{\tilde{b}}$. Based on this approximated unsafe set, we can then characterize the compact set $\mathcal{H}' \supset \mathcal{D}_{\tilde{b}}$ from which we will collect finite samples of unsafe data used for training the FNN, and subsequently, the analysis and design of CLBF will follow that of the bounded unsafe set.

4 | STABILIZATION AND SAFETY VIA CONTROL LYAPUNOV-BARRIER FUNCTION

A CLBF was proposed in Romdlony and Jayawardhana,[5] which is a weighted average of a CLF and a CBF, where it was shown that if a CLBF exists for the system of Equation (1) with $w(t) \equiv 0$, there exists a controller $u = \Phi(x)$ that will maintain the closed-loop state with $x_0 \in \mathcal{X}_0$ within a level set of the CLBF and outside of $\mathcal{D}$ at all times. The work in Wu et al.[6,13] extends the analysis to constrained CLBFs, accounting for physical constraints on manipulated inputs $u \in U$. In the recent work in Wu and Christofides,[28] a constrained CLBF-MPC is analyzed where the MPC uses a prediction model built from an ensemble of RNN models, and the stability and safety properties of this approach were guaranteed using a control law $u = \Phi_{nn}(x) \in U$. The CLF needs to meet the conditions outlined in Section 2.3 and the CBF needs to meet the conditions of Equations (5a)–(5c). As we have shown in Section 3.2, upon successful verification of $\hat{B}(x)$ against the conditions of Equations (5a)–(5c), it is a valid CBF which CLBF-based controllers can take in. Therefore, the theoretical results shown in[28] can be similarly applied to a CLBF constructed with a FNN-specified CBF $\hat{B}(x)$, where closed-loop stability and safe operation can be achieved under the CLBF-based control law $u = \Phi_{nn}(x) \in U$ for the RNN system of Equation (3).

The definition of a constrained CLBF constructed using the FNN-CBF $\hat{B}$, denoted as $W_{nn}(x)$ with respect to the RNN model of Equation (3) is as follows:

Definition 3. Given a set of unsafe points in state-space $\mathcal{D}$, a proper, lower-bounded and $\mathcal{C}^1$ function $W_{nn}(x): \mathbf{R}^n \to \mathbf{R}$ is a constrained CLBF if $W_{nn}(x)$ has a minimum at the origin and also satisfies the following properties:

$$W_{nn}(x) > \rho,\ \forall x \in \mathcal{D} \subset \phi_{uc}, \tag{8a}$$

$$\begin{gathered} L_{\hat{f}} W_{nn}(x) < 0, \\ \forall x \in \left\{ z \in \phi_{uc} \backslash (\mathcal{D} \cup \{0\} \cup \mathcal{X}_e) \mid L_{\hat{g}} W_{nn}(z) = 0 \right\}, \end{gathered} \tag{8b}$$

$$\mathcal{U}_\rho := \{ x \in \phi_{uc} \mid W_{nn}(x) \le \rho \} \ne \emptyset, \tag{8c}$$

$$\overline{\phi_{uc} \backslash (\mathcal{D} \cup \mathcal{U}_\rho)} \cap \overline{\mathcal{D}} = \emptyset, \tag{8d}$$

where $\rho \in \mathbf{R}$ is a constant, $\mathcal{X}_e := \{ x \in \phi_{uc} \backslash (\mathcal{D} \cup \{0\}) \mid \partial W_{nn}(x)/\partial x = 0 \}$ is a set of states for the RNN model of Equation (4) where $L_{\hat{f}} W_{nn}(x) = 0$ (for $x \ne 0$) due to $\partial W_{nn}(x)/\partial x = 0$. $\hat{f}$ and $\hat{g}$ are from the RNN model in Equation (4). Under a stabilizing control law $u = \Phi_{nn}(x) \in U$, ϕ_{uc} is defined to be the union of the set where the time-derivative of $W_{nn}(x)$ is negative with constrained inputs, the origin, and the set $\mathcal{X}_e$: $\phi_{uc} = \left\{ x \in \mathbf{R}^n \mid \dot{W}_{nn}(x(t), \Phi_{nn}(x)) = L_{\hat{f}} W_{nn} + L_{\hat{g}} W_{nn} \cdot u < -\alpha_W |W_{nn}(x) - W_{nn}(0)| u = \Phi_{nn}(x) \in U \right\} \cup \{0\} \cup \mathcal{X}_e$, and α_W is a positive real number used to characterize the set ϕ_{uc}. A control law $u = \Phi_{nn}(x) \in U$ that renders the origin exponentially stable within ϕ_{uc} is assumed to exist for the RNN system of Equation (3) in the sense that there exists a $\mathcal{C}^1$ constrained CLBF $W_{nn}(x)$. The CLBF function satisfies the following conditions $\forall x \in \phi_{uc}$ and has a minimum at the origin:

$$\hat{c}_1 |x|^2 \le W_{nn}(x) - \rho_0 \le \hat{c}_2 |x|^2, \tag{9a}$$

$$\frac{\partial W_{nn}(x)}{\partial x} F_{nn}(x, \Phi_{nn}(x)) \le -\hat{c}_3 |x|^2, \forall x \in \phi_{uc} \backslash \mathcal{B}_\delta(x_e) \tag{9b}$$

$$\left| \frac{\partial W_{nn}(x)}{\partial x} \right| \le \hat{c}_4 \, |x| \tag{9c}$$

where $\hat{c}_j(\cdot)$, $j = 1,2,3,4$ are positive real numbers, $W_{nn}(0) = \rho_0$ is the global minimum value of $W_{nn}(x)$, and $\mathcal{B}_\delta(x_e)$ is a small neighborhood around $x_e \in \mathcal{X}_e$. $F_{nn}(x,u)$ is the RNN system of Equation (3).

In addition, in the nonlinear system of Equation (1), we assumed that functions f, g, and h are sufficiently smooth, by continuity, there exist positive constants L_x, L_w, $L_x^{'}, L_w^{'}$, M, such that for all $x, x' \in \mathcal{U}_\rho$, $w \in W$, and $u \in U$, the following conditions will hold:

$$|F(x,u,w)| \le M, \tag{10a}$$

$$|F(x,u,w) - F(x',u,0)| \le L_x |x - x'| + L_w |w|, \tag{10b}$$

$$\left| \frac{\partial W_{nn}(x)}{\partial x} F(x,u,w) - \frac{\partial W_{nn}(x')}{\partial x} F(x',u,0) \right| \le L_x' |x - x'| + | \le L_w' |w_m|. \tag{10c}$$

In Wu and Christofides,[6] an exemplar stabilizing control law $\Phi_{nn}(x)$ is shown. The Lyapunov function $V(x)$ can be replaced with the CLBF $W_{nn}(x)$ within the Lyapunov-based control law that is presented in the form of the universal Sontag controller.[20]

4.1 | Design of constrained CLBF

We first design CLF and CBF separately to meet their respective conditions, and we follow the practical design guidelines presented in Wu et al.[13] to construct the CLBF. We present the design method for choosing the CLF, the CBF, and the corresponding weights in this section, and show that the $\hat{B}(x)$ is able to meet all the conditions on CBF, through which $W_{nn}(x)$ is able to meet all its required properties of Equations (8a)–(8d) and (9a)–(9c) and has a global minimum at the origin.

Proposition 1. *Given an open set $\mathcal{D}$ of unsafe states for the system of Equation (1) with $w(t) \equiv 0$, assume that there exists a $\mathcal{C}^1$ CLF $V: \mathbf{R}^n \to \mathbf{R}_+$, and a $\mathcal{C}^1$ CBF $\hat{B}: \mathbf{R}^n \to \mathbf{R}$, such that the following conditions hold:*

$$c_1 |x|^2 \le V(x) \le c_2 |x|^2, \forall x \in \mathbf{R}^n, c_2 > c_1 > 0, \tag{11}$$

$$\mathcal{D} \subset \mathcal{H} \subset \mathcal{H}' \subset \phi_{uc}, 0 \notin \mathcal{H}, 0 \notin \mathcal{H}', \tag{12}$$

$$\hat{B}(x) = -\eta < 0, \forall x \in \mathbf{R}^n \backslash \mathcal{H}'; \hat{B}(x) > 0, \forall x \in \mathcal{H}', \tag{13}$$

where $\mathcal{H}$ is a compact and connected set within ϕ_{uc}, and $\mathcal{H}'$ is a compact and connected set within ϕ_{uc} that encloses $\mathcal{H}$ with sufficient margin to account for modeling errors in $\hat{B}(x)$, $\hat{f}$, and $\hat{g}$. Define $W_{nn}(x)$ to have the form $W_{nn}(x) := V(x) + \mu \hat{B}(x) + \nu$, where

$$\left| \frac{\partial W_{nn}(x)}{\partial x} \right| \le \hat{c}_4 \, |x|, \tag{14}$$

$$\begin{gathered} L_{\hat{f}} W_{nn}(x) < 0, \\ \forall x \in \left\{ z \in \phi_{uc} \backslash (\mathcal{D} \cup \{0\} \cup \mathcal{X}_e) \mid L_{\hat{g}} W_{nn}(z) = 0 \right\} \end{gathered} \tag{15}$$

$$\mu > \frac{c_2 c_3 - c_1 c_4}{\eta}, \tag{16a}$$

$$\nu = \rho - c_1 c_4, \tag{16b}$$

$$c_3 := \max_{x\in\partial\mathcal{H}'} |x|^2, \tag{16c}$$

$$c_4 := \min_{x\in\partial\mathcal{D}} |x|^2, \tag{16d}$$

then the control law $\Phi_{nn}(x)$ (Lyapunov-based Sontag controller with $W_{nn}(x)$ replacing $V(x)$) guarantees that the closed-loop state is bounded in $\phi_{uc}\backslash\mathcal{H}$ and does not enter the unsafe region $\mathcal{H}$ for all times, for any initial state $x_0\in\phi_{uc}\backslash\mathcal{D}_{\mathcal{H}'}$, where $\mathcal{D}_{\mathcal{H}'} := \{x\in\mathcal{H}' \mid W_{nn}(x) > \rho\}$.

> *Proof*. By the construction of the FNN model for the CBF, $\hat{B}(x)$ meets the condition of Equation (13) despite modeling error due to the characterization of $\mathcal{H}'\supset\mathcal{H}$, where the margin between $\mathcal{H}'$ and $\mathcal{H}$ accounts for the modeling error of $\hat{B}(x)$, and of $\hat{f}$ and $\hat{g}$ of the RNN model of Equation (4). It was proven in Wu et al.[13] and Wu and Christofides[28] that a constrained CLBF designed following these guidelines satisfies the properties of Equations (8a)–(8d) and (9c); the proofs will be omitted here.

In addition, we also need to prove that the constrained CLBF $W_{nn}(x)$ designed using a CLF $V(x)$ and a CBF $\hat{B}(x)$ satisfies the additional properties of Equations (9a) and (9b), which are required for $u=\Phi_{nn}(x)\in U$ to render the origin of the RNN system of Equation (3) exponentially stable. In order to make sure Equation (9a) holds, both $|V(x)-V(0)|$ and $|\hat{B}(x)-\hat{B}(0)|$ need to be bounded. From Equation (11), we know that $c_1|x|^2 \le V(x)-V(0)\le c_2|x|^2, \forall x\in\mathbf{R}^n$ since $V(0)=0$. Based on the construction and the training objectives of the FNN-modeled CBF, we also know that $|\hat{B}(x)-\hat{B}(0)|\le 2$ within a sufficiently small bounded error that includes modeling inaccuracies and numerical error in the $\hat{B}$ predictions. Therefore, the resulting CLBF, $W_{nn}(x)-W_{nn}(0)$, which is a linear combination of the bounded $V(x)$ and $\hat{B}(x)$, is also bounded by its respective lower and upper bounds as shown in Equation (9a).

The condition of Equation (9b) holds due to the definition of ϕ_{uc} as well as the boundedness of $|W_{nn}(x)-W_{nn}(0)|$, where $\hat{c}_3=\alpha_W\hat{c}_2$. Furthermore, $V(x)$ has a global minimum at the origin: $V(0)=0$ and $V(x)>0$ for all $x\in\mathbf{R}^n\backslash\{0\}$. With a sufficiently small bounded numerical error and modeling error, $\hat{B}(x)=-1$ for all $x\in\phi_{uc}\backslash\mathcal{H}'$, where $\{0\}\in\phi_{uc}\backslash\mathcal{H}'$, and $\hat{B}(x)=+1$ for all $x\in\mathcal{H}'$. Therefore, $\hat{B}(x)$ also has a global minimum at the origin within bounded numerical error. Since $W_{nn}(x)$ is a weighted average of $V(x)$ and $\hat{B}(x)$, the global minimum of $W_{nn}(x)$ is also at the origin. Therefore, we have demonstrated, a CLBF $W_{nn}(x)$ and a controller $u=\Phi_{nn}(x)\in U$ exist that together satisfy all conditions of Equations (8a)–(8d) and (9a)–(9c), and will guarantee exponential stability for all $x_0\in\phi_{uc}\backslash\mathcal{D}_{\mathcal{H}'}$.

In the rest of our paper, we will focus on initial conditions in $\mathcal{U}_\rho$, which is a forward invariant set of $W_{nn}(x)$ as defined in Equation (8c). Furthermore, closed-loop stability and safety for the RNN system of Equation (3) are analyzed with respect to bounded unsafe sets similar to Theorem 1 in Wu and Christofides.[6] Specifically, in the presence of bounded unsafe sets, there exist stationary points $x_e\in\mathcal{X}_e$ in state-space other than the origin that can be treated as saddle points. When states reach these stationary points, the continuous control law of $u=\Phi_{nn}(x)\in U$ is unable to drive the states away from them. We design discontinuous control actions $u=\bar{u}(x)\in U$, $\bar{u}(x)\ne\Phi_{nn}(x)$, to drive the states away from these saddle points in the direction of decreasing $W_{nn}(x)$. The theorem below provides the sufficient conditions under which the controller $u=\Phi_{nn}(x)\in U$ designed based on the CLBF $W_{nn}(x)$ is able to fulfill stability and safety for the closed-loop RNN system of Equation (3).

> **Theorem 2.** *Consider a constrained CLBF $W_{nn}(x):\mathbf{R}^n\to\mathbf{R}$ built using $\hat{B}(x)$, that has a minimum at the origin and satisfies the conditions of Equations (8a)–(8d), exists for the RNN system of Equation (3). The controller $u=\Phi_{nn}(x)\in U$ that satisfies Equations (9a)–(9c) guarantees that the closed-loop state stays within $\mathcal{U}_\rho$ for all times for any $x_0\in\mathcal{U}_\rho$. In the presence of a bounded unsafe region in state-space, the origin can be rendered exponentially stable under $u=\Phi_{nn}(x)\in U$ (if x is not near a saddle point x_e) and under discontinuous control actions $u=\bar{u}(x)\in U$ that decrease $W_{nn}(x)$ (if x is near a saddle point x_e) for all $x_0\in\mathcal{U}_\rho$.*

> *Proof*. It has been proven in Wu et al.[6,13,28] that the universal Sontag controller[19] with the CLBF $W_{nn}(x)$ replacing the Lyapunov function $V(x)$ gives a valid $u=\Phi_{nn}(x)\in U$ that ensures $\dot{W}_{nn}(x)\le 0$ for all $x\in\mathcal{U}_\rho$, therefore ensuring that for $x_0\in\mathcal{U}_\rho$, x is bounded in $\mathcal{U}_\rho$ for all times. Furthermore, since $\mathcal{U}_\rho$ is a level set of $W_{nn}(x)$ in ϕ_{uc} (ϕ_{uc} is a set within which Eqs. (9a)–(9c) is met), the origin is rendered exponentially stable under $u=\Phi_{nn}(x)\in U$. In the presence of bounded unsafe regions, the saddle points at which $\dot{W}_{nn}(x)=0$ can be handled by discontinuous control actions $u=\bar{u}(x)\in U, \bar{u}(x)\ne\Phi_{nn}(x)$ that decrease $W_{nn}(x)$. The detailed proofs for handling bounded unsafe sets can be referenced from Theorem 1 in Wu and Christofides,[6] and will be omitted here.

5 | CLBF-BASED MPC USING FNN CBF AND RNN PREDICTION MODEL

In this work, we propose a CLBF-based MPC which is designed based on a CLBF-based controller that ensures simultaneous closed-loop stability and process safety for the nonlinear system of Equation (1). The CLBF-based controller $u=\Phi_{nn}(x)\in U$, which uses a $W_{nn}(x)$ incorporating an FNN-modeled CBF ($\hat{B}(x)$), is designed based on the $\hat{f}$ and $\hat{g}$ of the RNN system of Equation (4). Then, the CLBF-MPC is developed to optimize process performance while driving the process states to a small ball around the origin. So far, we have shown that a valid CLBF $W_{nn}(x)$ can be constructed using $\hat{B}(x)$, from which the controller $u=\Phi_{nn}(x)\in U$ exponentially stabilizes the origin of the RNN system of Equation (3) while keeping closed-loop states in a safe region of operation $\mathcal{U}_\rho$.

The control actions of the CLBF-MPC are implemented in a sample-and-hold manner to the original nonlinear system of Equation (1), that is, for any $t\in[t_k,t_{k+1})$, $u(t)=u(t_k)$, where $t_{k+1}:=t_k+\Delta$. Note that Δ is the sampling period of the MPC. Due to the presence of bounded disturbances in the nonlinear system of Equation (1), as well as the modeling mismatch between the RNN system of Equation (3) and the first-principles system of Equation (1), we must investigate the safety and stability properties of the system with these considerations in mind.

In Proposition 1 of Wu and Christofides,[28] given that the modeling error of the RNN model of Equation (3) is bounded by $|\nu|=|F(x,u,0)-F_{nn}(x,u)|\le\gamma|x|\le\nu_m$, and the nonlinear system of Equation (1) has bounded disturbances $|w|\le w_m$, the boundedness of the state error $|x-\hat{x}|$ and the difference between $|W_c(x)-W_c(\hat{x}))|$ was shown, where W_c is a CLBF that uses an explicitly defined CBF $B(x)$. More specifically, $|x(t)-\hat{x}(t)|\le f_w(t):=\frac{L_w w_m+\nu_m}{L_x}\left(e^{L_x t}-1\right)$, and $W_c(x)\le W_c(\hat{x})+\kappa|x-\hat{x}|^2+\frac{\hat{c}_4\sqrt{\rho-\rho_0}}{\sqrt{\hat{c}_1}}|x-\hat{x}|$, where $\kappa>0$. Since $W_{nn}(x)$ can be also shown to be continuous and bounded on a compact set and behaves the same as $W_c(x)$, the same proofs apply on $W_{nn}(x)$, and we can conclude that $W_{nn}(x)\le W_{nn}(\hat{x})+\frac{\hat{c}_4\sqrt{\rho-\rho_0}}{\sqrt{\hat{c}_1}}f_w(t)+\kappa f_w(t)^2$, where $\hat{c}_1$, and $\hat{c}_4$ are positive real constants in Equations (9a)–(9c) for $W_{nn}(x)$.

All subsequent proofs on the stability and safety of the nominal system of Equation (1) under the CLBF-based control law designed based on $W_{nn}(x)$ follow the same proofs in Wu and Christofides[28] (Propositions 2 and 3), with W_{nn} replacing W_c. This is shown for bounded unsafe regions, where the CLBF-based control law designed using the RNN model of Equation (3) can also guarantee closed-loop exponential stability and safety for the nominal system of Equation (1). We will show that the combination of the CLBF-based control law $u=\Phi_{nn}(x)\in U$ along with discontinuous control actions that yield decreasing $W_{nn}(x)$ will provide exponential stability and safety in the case of bounded unsafe sets.

We consider the nominal system of Equation (1) with a bounded unsafe set, where saddle points $x_e\in\mathcal{X}_e$ are present in $\mathcal{U}_\rho$. We provide sufficient conditions, under which the continuous CLBF-based control actions $u=\Phi_{nn}(x)\in U$ and the control actions $u=\bar{u}(x)\in U$ designed in a discontinuous manner, can ensure closed-loop stability and safety. The proof for the following adapted proposition can be found in Wu and Christofides.[28]

Proposition 2. *If the RNN model is developed such that for all $x\in\mathcal{U}_\rho$ and $u\in U$, the modeling error is constrained by $|\nu|=|F(x,u,0)-F_{nn}(x,u)|\le\gamma|x|$, where γ is a positive real number that satisfies $\gamma<\hat{c}_3/\hat{c}_4$, and furthermore, Equation (17) is satisfied under discontinuous control actions $u=\bar{u}(x)\in U$ when $x(t_k)=\hat{x}(t_k)\in\mathcal{B}_\delta(x_e)$,*

$$W_{nn}(\hat{x}(t))<W_{nn}(\hat{x}(t_k))-f_e(t-t_k),\forall t>t_k \tag{17}$$

where

$$f_e(t-t_k):=\frac{\hat{c}_4\sqrt{\rho-\rho_0}}{\sqrt{\hat{c}_1}}f_w(t-t_k)-\kappa f_w(t-t_k)^2$$

and $f_w(t)$ is the upper bound on the state error $|x(t)-\hat{x}(t)|\le f_w(t)$, then the stability and safety properties outlined in Theorem 2 also apply to the nominal system of Equation (1) with a bounded unsafe region $\mathcal{D}_b$ under $u=\Phi_{nn}(x)\in U$ and $u=\bar{u}(x)\in U$.[28]

In the presence of bounded disturbances (i.e., $|w(t)|\le w_m$), now we show that the nonlinear system of Equation (1) can be rendered exponentially stable and maintained within the safe region $\mathcal{U}_\rho$. Under the sample-and-hold implementation of the control actions, the state of the closed-loop system of Equation (1) is always bounded in $\mathcal{U}_\rho$, and converges to a small neighborhood $\mathcal{U}_{\rho_{min}}$. Given that the set of initial conditions $\mathcal{U}_\rho$ for which exponential stability and safety of the RNN system of Equation (3) can be guaranteed under the CLBF-based control laws is characterized using $W_{nn}(x)$, the following proposition has been adapted from Proposition 4 in Wu and Christofides.[28]

Proposition 3. *Consider the nonlinear system of Equation (1) under the CLBF-based controller $u=\Phi_{nn}(x)\in U$ (under sample-and-hold implementation), which is built using a valid W_{nn} following Proposition 1 and satisfies Equations (9a)–(9c). If Equation (17) is satisfied under the controller $u=\bar{u}(x)\in U$ in a sample-and-hold fashion for $x\in\mathcal{B}_\delta(x_e)$, and there exist $\varepsilon_w>0$, $\Delta>0$ and $\rho_s<\rho_{nn}<\rho_{min}<\rho$ that satisfy*

$$-\frac{\widetilde{c}_3}{\hat{c}_2}(\rho_s-\rho_0)+L_x^{'}M\Delta+L_w^{'}w_m\le-\varepsilon_w \tag{18}$$

and

$$\rho_{nn}:=\max\{W_{nn}(\hat{x}(t+\Delta))\mid u\in U,\hat{x}(t)\in\mathcal{U}_{\rho_s}\} \tag{19a}$$

$$\rho_{nn}+f_e(\Delta)\le\rho_{min} \tag{19b}$$

where $f_e(t)$ is given by Equation (17), then for any $x(t_k)\in\mathcal{U}_\rho\backslash\mathcal{U}_{\rho_s}$, $W_{nn}(x(t))$ is guaranteed to decrease within every sampling period, and can be bounded in $\mathcal{U}_\rho$ for all times and ultimately bounded in $\mathcal{U}_{\rho_{min}}$.

5.1 | Formulation of CLBF-MPC

The following optimization problem represents the CLBF-MPC design:

$$\mathcal{J}=\min_{u\in S(\Delta)}\int_{t_k}^{t_{k+N}}L(\widetilde{x}(t),u(t))dt, \tag{20a}$$

$$\text{s.t.}\quad \dot{\widetilde{x}}(t)=F_{nn}(\widetilde{x}(t),u(t)), \tag{20b}$$

$$\tilde{x}(t_k) = x(t_k), \tag{20c}$$

$$u(t) \in U, \forall t \in [t_k, t_{k+N}), \tag{20d}$$

$$\begin{aligned} &\dot{W}_{nn}(x(t_k), u(t_k)) \leq \dot{W}_{nn}(x(t_k), \Phi_{nn}(t_k)), \\ &\text{if } x(t_k) \notin \mathcal{B}_\delta(x_e) \text{ and } W_{nn}(x(t_k)) > \rho_{nn}, \end{aligned} \tag{20e}$$

$$W_{nn}(\tilde{x}(t)) \leq \rho_{nn}, \forall t \in [t_k, t_{k+N}), \text{if } W_{nn}(x(t_k)) \leq \rho_{nn}, \tag{20f}$$

$$\begin{aligned} &W_{nn}(\tilde{x}(t)) < W_{nn}(x(t_k)) - f_e(t - t_k), \forall t \in (t_k, t_{k+N}), \\ &\text{if } x(t_k) \in \mathcal{B}_\delta(x_e), \end{aligned} \tag{20g}$$

where $\tilde{x}(t)$ is the predicted state trajectory, N is the number of sampling periods in the prediction horizon, $S(\Delta)$ represents the set of piece-wise constant functions with sampling period Δ. The CLBF-MPC optimization problem has an objective function of Equation (20a), which is the integral of $L(\tilde{x}(t), u(t))$ over the prediction horizon typically in a quadratic form, that is, $L(\tilde{x}(t), u(t)) = \tilde{x}^T Q \tilde{x} + u^T R u$, where Q, R are positive definite weighting matrices, and the minimum of this objective function is achieved at the origin. In Equation (20b), the predicted state trajectory $\tilde{x}(t)$, $t \in [t_k, t_{k+N})$ are calculated using the RNN model F_{nn} of Equation (3). $\dot{W}_{nn}(x, u)$ represents $\frac{\partial W_{nn}(x)}{\partial x}\left(\hat{f}(x) + \hat{g}(x)u\right)$, where $\hat{f}$ and $\hat{g}$ are the approximated nonlinear functions of the RNN model of Equation (4). The input constraints of Equation (20d) are applied over the entire prediction horizon. We assume that the measured states of the closed-loop system are available at each sampling time. For the predicated state trajectory of Equation (20b), the initial condition is obtained from the feedback measurement of Equation (20c) at $t = t_k$. To ensure closed-loop stability and process operational safety, the constraints of Equations (20e)–(20g) are utilized. When $x(t_k) \notin \mathcal{B}_\delta(x_e)$ and $W_{nn}(x(t_k)) > \rho_{nn}$, the constraint of Equation (20e) forces $W_{nn}(\tilde{x})$ to decrease along at a rate less than or equal to that under the CLBF-based control law $u = \Phi_{nn}(x) \in U$. If $W_{nn}(x(t_k)) \leq \rho_{nn}$, the constraint of Equation (20f) maintains the predicted state of the RNN system of Equation (3) within $\mathcal{U}_{\rho_{nn}}$ such that the closed-loop state of the nonlinear system of Equation (1) is bounded in $\mathcal{U}_{\rho_{min}}$. Furthermore, if $x(t_k) \in \mathcal{B}_\delta(x_e)$, the constraint of Equation (20g) decreases $W_{nn}(x)$ over the predicted state trajectory such that the closed-loop state can escape from saddle points x_e within a finite number of sampling periods. Once the state leaves $\mathcal{B}_\delta(x_e)$, it will be driven to smaller level sets of $W_{nn}(x)$ under the constraint of Equation (20e), therefore guaranteeing that the state does not go back to $\mathcal{B}_\delta(x_e)$ afterwards. After solving the optimal solution $u^*(t)$, the control action at the first time instant, $u^*(t_k)$, is applied over the next sampling period in a sample-and-hold manner. The horizon will be moved forward one sampling period, and the above process is repeated.

The following theorem and proof will show that safety and stability can be established for the closed-loop nonlinear system of Equation (1) using the CLBF-based MPC.

Theorem 3. *Consider the system of Equation (1) with a constrained CLBF W_{nn} built using a NN-BF $\hat{B}(x)$ following the procedures in Section 3. The constrained NN-based CLBF $W_{nn}(x)$ satisfies Equations (8a)–(8d) and has a minimum at the origin. Given any initial state $x_0 \in \mathcal{U}_\rho$, it is guaranteed that the CLBF-MPC optimization problem of Equations (20a)–(20g) can be solved with recursive feasibility for all times. Additionally, under the sample-and-hold implementation of CLBF-MPC based on an RNN prediction model that satisfies $|\nu| = |F(x,u,0) - F_{nn}(x,u)| \leq \gamma|x| \leq \nu_m$ and the conditions in Proposition 3, it is guaranteed that for any $x_0 \in \mathcal{U}_\rho$, the state is bounded in $\mathcal{U}_\rho$, $\forall t \geq 0$, and ultimately converges to $\mathcal{U}_{\rho_{min}}$ as $t \to \infty$.*

Proof. Part 1: The optimization problem of Equations (20a)–(20g) for the CLBF-MPC has a feasible solution at all times since the CLBF-MPC constraints of Equations 20d-20g can be satisfied by the sample-and-hold control laws $u = \bar{u}(x) \in U$, $\forall x \in \mathcal{B}_\delta(x_e)$ and $u = \Phi_{nn}(x) \in U$, $\forall x \in \mathcal{U}_\rho \backslash \mathcal{B}_\delta(x_e)$. This has been demonstrated in Propositions 2 and 3 with detailed proofs outlined in Wu and Christofides.[28] More specifically, the control laws $u = \bar{u}(x) \in U$, $\forall x \in \mathcal{B}_\delta(x_e)$ and $u = \Phi_{nn}(x) \in U$, $\forall x \in \mathcal{U}_\rho \backslash \mathcal{B}_\delta(x_e)$ are already constrained by $u \in U$, therefore the input constraint of Equation (20d) can be met over the prediction horizon. By letting $u(t_k) = \Phi_{nn}(x(t_k))$ when $x(t_k) \in \mathcal{U}_\rho \backslash (\mathcal{B}_\delta(x_e) \cup \mathcal{U}_{\rho_{nn}})$, Equation (20e) is also satisfied. It has been shown in Proposition 3 that once the closed-loop state is inside $\mathcal{U}_{\rho_s}$ under the control law $u = \Phi_{nn}(x) \in U$, it will not leave $\mathcal{U}_{\rho_{nn}}$ for any $u \in U$ within one sampling period. Thus, the CLBF-based control law $u(t) = \Phi_{nn}(x(t_{k+i})) \in U$, $\forall t \in [t_{k+i}, t_{k+i+1})$ with $i = 0, \ldots, N-1$ provides a feasible trajectory of control actions that meet the constraint of Equation (20f). Lastly, as the controller $u = \bar{u}(x) \in U$ satisfies Equation (17), the control action $u(t) = \bar{u}(x(t_{k+i})) \in U$, $\forall t \in [t_{k+i}, t_{k+i+1})$ with $i = 0, \ldots, N-1$ will satisfy the constraint of Equation (20g) and drive the state away from saddle points if $x(t_k) \in \mathcal{B}_\delta(x_e)$. The proof for recursive feasibility of the optimization problem of Equations (20a)–(20g) is complete.

Part 2: Now we will prove that the optimized solution of Equations (20a)–(20g) will guarantee simultaneous safety and stability for the closed-loop nonlinear system of Equation (1). For any $x_0 \in \mathcal{U}_\rho \backslash \mathcal{U}_{\rho_{nn}}$, the constraint of Equation (20e) ensures that the optimized CLBF-MPC control action u^* will drive the closed-loop state of the RNN system towards the origin and into $\mathcal{U}_{\rho_{nn}}$ within finite sampling periods. After the state enters $\mathcal{U}_{\rho_{nn}}$, the constraint of Equation (20f) ensures the boundedness of the closed-loop state of the RNN model in $\mathcal{U}_{\rho_{nn}}$ for the remaining time. With the impact of the RNN modeling error, bounded disturbances, and sample-and-hold implementation of control actions, it has been shown in Proposition 3 that when the closed-loop state of the RNN system is bounded in $\mathcal{U}_{\rho_{nn}}$, the actual state of the nonlinear system of Equation (1) is ultimately bounded in $\mathcal{U}_{\rho_{min}}$. Furthermore, since the safe operating region $\mathcal{U}_\rho$ has no intersection with the unsafe region $\mathcal{D}$, the closed-loop state will be bounded in $\mathcal{U}_\rho$ for any $x_0 \in \mathcal{U}_\rho$, and thus will not enter $\mathcal{D}$ at all times.

In addition, avoiding convergence to saddle points needs to be considered. Saddle points are points in state-space at which the CLBF W_{nn} has a local minima. Starting from an initial condition $x_0 \in \mathcal{U}_\rho \backslash \mathcal{U}_{\rho_{nn}}$, the constraint of Equation (20e) pulls the state towards the origin. When the closed-loop state reaches a neighborhood around the saddle point where $x(t_k) \in \mathcal{B}_\delta(x_e)$, the constraint of Equation (20g) will drive the state away from the neighborhood of saddle point in a direction of decreasing $W_{nn}(x)$. Once the state escapes $\mathcal{B}_\delta(x_e)$, then the constraints of Equations (20e)–(20f) will ensure operational safety and closed-loop stability, and the closed-loop state ultimately converges to the origin and is bounded in $\mathcal{U}_{\rho_{min}}$. Therefore, the presence of saddle points have been addressed, and closed-loop stability and safety under the CLBF-MPC for the nonlinear system of Equation (1) with bounded unsafe sets have been proven.

6 | APPLICATION TO A CHEMICAL PROCESS EXAMPLE

In this section, we apply the proposed CLBF-MPC on a chemical process example. The process considered consists of a well-mixed, nonisothermal continuous stirred tank reactor (CSTR) where an irreversible first-order exothermic reaction $A \rightarrow B$ takes place. There is a heating jacket installed on the reactor to supply and remove heat. The material and energy balances of this CSTR system is as follows:

$$\frac{dC_A}{dt} = \frac{F}{V_L}(C_{A0} - C_A) - k_0 e^{-E/RT} C_A, \tag{21a}$$

$$\frac{dT}{dt} = \frac{F}{V_L}(T_0 - T) - \frac{\Delta H k_0}{\rho_L C_p} e^{-E/RT} C_A + \frac{Q}{\rho_L C_p V_L}, \tag{21b}$$

where T is the temperature in the reactor, C_A represents the concentration of reactant A, Q is the heat rate, and V_L is the volume of the reacting liquid in the reactor. The reactor feed contains the reactant A at a concentration C_{A0}, temperature T_0, and volumetric flow rate F. ρ_L, C_p, k_0, E, and ΔH are the liquid density, heat capacity, reaction pre-exponential factor, activation energy and the enthalpy of the reaction, respectively. Process parameter values can be found in Wu et al.[13] The control objective is to operate the CSTR at the steady-state point $(C_{As}, T_s) = (0.57\,\text{kmol/m}^3, 395.3\,\text{K})$ and maintain the state in a safe region by manipulating the inlet concentration of species A, $\Delta C_{A0} = C_{A0} - C_{A0_s}$, and the heat input rate $\Delta Q = Q - Q_s$. The input constraints for ΔQ and ΔC_{A0} are $|\Delta Q| \leq 0.0167\,\text{kJ/min}$ and $|\Delta C_{A0}| \leq 1\,\text{kmol/m}^3$, respectively.

Deviation variables are used such that the equilibrium point of the system is at the origin of the state-space. $x^T = [C_A - C_{As}\; T - T_s]$, $u^T = [\Delta C_{A0}\; \Delta Q]$ represent the state vector and the manipulate input vector in deviation variable forms, respectively. As the focus of the current work is on the machine-learning construction of CBF and its application on an RNN-MPC, we do not consider bounded disturbances. Further simulations can be run with added disturbances to assess the robustness of the proposed control system.

We construct a Control Lyapunov Function using the standard quadratic form $V(x) = x^T P x$ with $P = \begin{bmatrix} 9.35 & 0.41 \\ 0.41 & 0.02 \end{bmatrix}$. The P matrix of the control Lyapunov function is determined via extensive closed-loop simulations of the process. With the goal of finding the largest stability and safety region in state space, we carry out an iterative search where we start with an initial guess of the P matrix, then find the region in state space within which the time derivative of CLBF can be rendered negative under the Sontag control law, and characterize the largest forward invariant set within this region to be considered as the stability and safety region. We define the unsafe region, $\mathcal{D}$, as a region embedded fully within the closed-loop system stability region. The unsafe region is located in the middle of the stability region such that the state trajectory will intersect the unsafe region on its converging route towards the origin. Such a bounded unsafe set poses both theoretical as well as implementation challenges for CLBF-MPC as the controller has to drive the state around the unsafe region and to the steady-state.

6.1 | Development of the RNN model for the CSTR process

We follow similar procedures of data generation, training and validation as outlined in Wu and Christofides[28] to obtain an RNN model for the nonlinear process of Equation (1). To generate training data sufficiently large to represent the entire operating region, open-loop simulations are run for finite sampling steps starting at various initial conditions within the safe and stabilizable set $\mathcal{U}_\rho$ with various control actions $u \in U$. The RNN model constructed takes the form of a long–short-term-memory network, which is a special kind of RNN known for its superior performance in remembering longer-intervaled temporal relationships. The RNN model uses one input layer, one hidden layer consisting of 20 recurrent units, and one output layer. State measurements $x(t_k)$ and the control actions $u(t_k)$ are the inputs to the RNN model, and the RNN model has the outputs of the predicted state trajectory over one sampling period $\hat{x}(t)$ for $t \in [t_k, t_{k+1}]$. The number of recurrent units in the hidden layer corresponds with the number of internal states within each sampling period. In our simulations, the time progression of states are simulated using an Euler integration method at an integration time step of h_c, and the sampling period of MPC is $\Delta = 100h_c$. In order to predict the states at the end of each sampling period, we could choose to have a maximum of 100 internal states, with a time interval of h_c between each internal state. In order to provide the RNN with sufficient neurons to achieve adequate accuracy and to also reduce computational effort, we have conducted a grid search between various numbers of internal states, and have chosen the design of 20 internal states with a time interval of $5h_c$ between each internal state. An early stopping criterion of achieving a validation MSE of below 1×10^{-6} is implemented to avoid over-fitting and to ensure that the modeling error is rendered sufficiently small. After 65 epochs, early stopping is triggered and the obtained RNN model achieves a training MSE of 4.17×10^{-6} and a validation MSE of 9.03×10^{-7}.

6.2 | Development of the FNN model for barrier function

In this example, we define the unsafe region as follows: $\mathcal{D} := \left\{x \in \mathbf{R}^2 \mid F(x) = \frac{(x_1+0.22)^2}{1} + \frac{(x_2-4.6)^2}{1\times 10^4} < 2\times 10^{-4}\right\}$. $\mathcal{H}$ is defined as $\mathcal{H} := \left\{x \in \mathbf{R}^2 \mid F(x) < 2.5\times 10^{-4}\right\}$ such that it satisfies $\mathcal{D}\subset\mathcal{H}\subset\phi_{uc}$ in Proposition 1. We define the unsafe region to be an ellipse as an illustrative example of a challenging case of bounded unsafe set embedded in the operating region. In practice, the bounded unsafe set can be of any bounded form in state-space, and may not be easily described explicitly. For example, operating at certain mid-ranges of temperature and concentration could lead to material corrosion, incomplete reactions, or generation of byproducts from side reactions. There are also circumstances where specific ranges of operation are suboptimal to efficiency and productivity. To generate training data for the FNN model, we specify $\mathcal{H}' := \left\{x \in \mathbf{R}^2 \mid F(x) < 5.6\times 10^{-4}\right\}$. The set of initial conditions considered $\mathcal{U}_\rho$ is characterized with $\rho = 0$ as per Equation (8c). The CLBF $W_{nn}(x)$ is constructed with the following parameters: $c_1 = 0.001, c_2 = 10, c_3 = 48.269, c_4 = 16.85, \nu = \rho - c_1 c_4 = -1.685\times 10^{-2}$, and $\mu = 5000$. The safe region, $\mathcal{U}_\rho \backslash \mathcal{H}'$, and the unsafe region $\mathcal{H}'$, are discretized into 18,000 data samples, respectively. The data samples are assigned a target label of "+1" if they belong in the unsafe region, and "−1" if they belong in the safe region. The FNN model is constructed with 2 hidden layers of 12 and 10 neurons respectively. The inputs to the FNN model are the state measurement vector, and the output of the FNN model gives the predicted class of the data point in state-space indicating whether it is located inside the safe or the unsafe region. Both hidden layers use an activation function of *tanh*, and the cost function of Equation (7) has the following weighting parameters: $\alpha = 1.1, \beta = 0.005$. The validation metric examines the magnitude of $Cost_2$ of Equation (7), and early stopping is triggered if $Cost_2$ has reached 0. After 700 epochs of training, the MSE ($Cost_1$) is 0.0155, and $Cost_2$ has a cumulative value of 0.4233. The classification accuracy over the testing dataset is 99.5%. The predicted $\hat{B}$ values are shown in Figure 2, and the misclassified data points in the testing set are shown in Figure 3. $Cost_2$ has not reached 0 as required by the algorithm within the specified number of epochs; however, the classification accuracy has reached an acceptable level to cease training. Then, the model can be assessed in terms of its

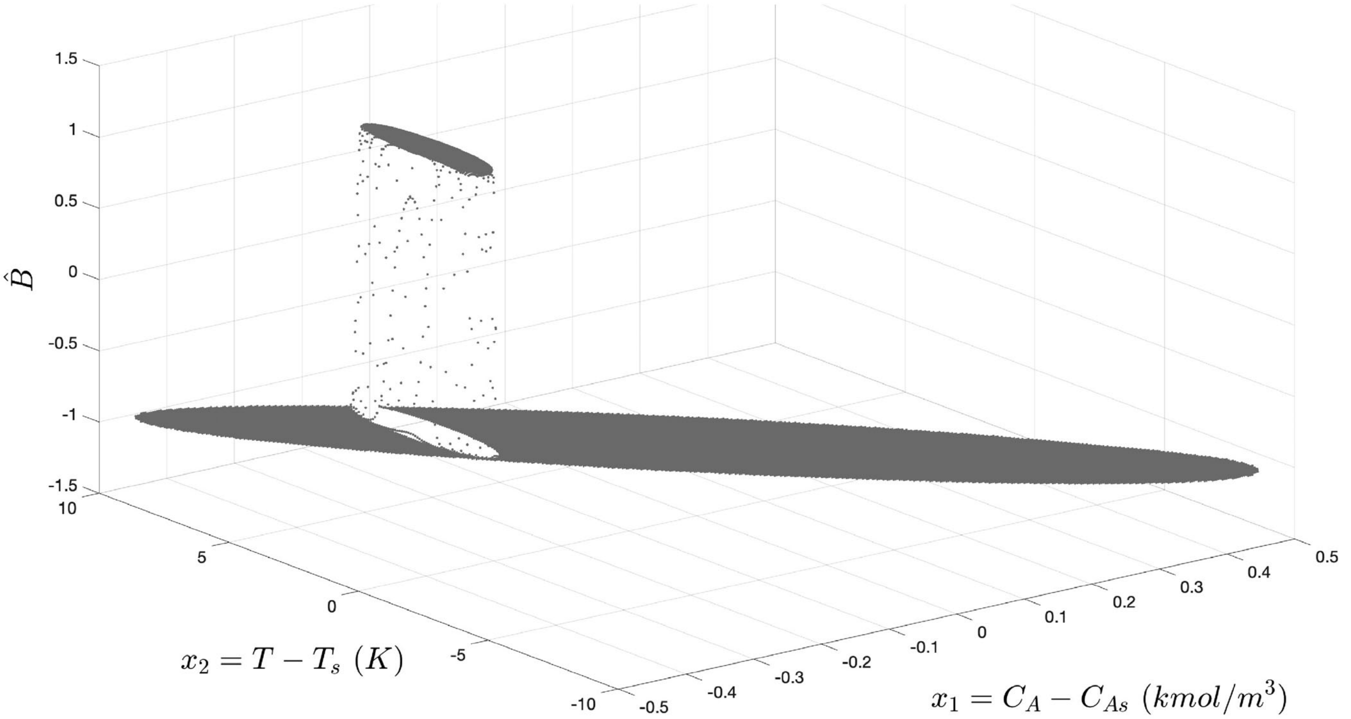

FIGURE 2 FNN-predicted barrier function $\hat{B}(x)$ for all data points in the training and the testing datasets

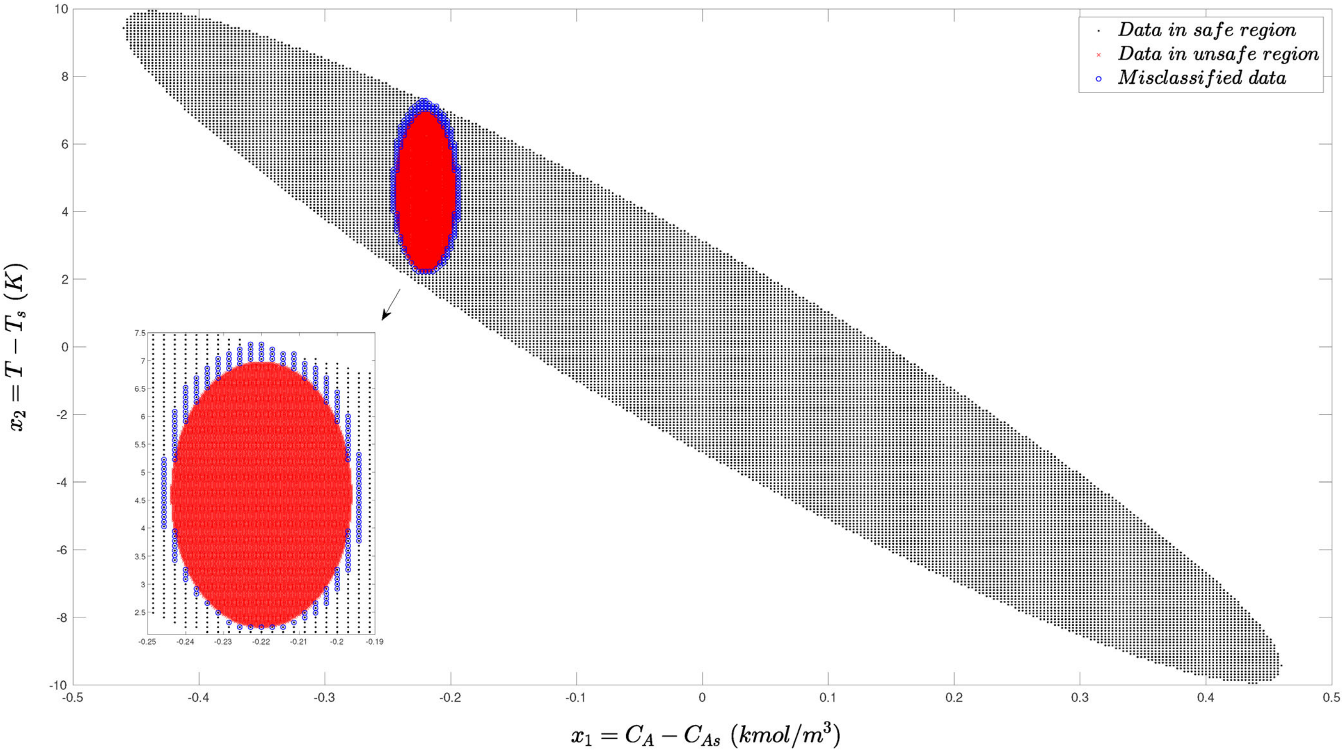

FIGURE 3 State values in the safe (black) and unsafe (red) operating regions, with misclassified data points (blue circles) showing that all inaccuracies are safe points misclassified as unsafe points

misclassification rate, as well as the conditions on the values of $\hat{B}(x)$ and $L_{\hat{f}}\hat{B}(x)$ as specified by Equations (5a)–(5c). There are 171 out of 36,000 data points being misclassifed, all of them are safe data classified as being unsafe. This does not cause any problems as the controller will simply be prompted to act sooner due to this misclassification when the closed-loop state approaches the boundary of the unsafe region. This also means that the predicated unsafe region given by the FNN-modeled CBF, denoted as $\hat{\mathcal{H}}$, is larger than $\mathcal{H}'$ as specified by the training data samples, therefore more conservative than what was intended. Moreover, to verify that the FNN model satisfies the safety conditions of CBF of Equations (5a)–(5c), we verify that the tighter conditions hold for all discretized data points in their respective regions. It is shown that all predicted $\hat{B}(x) > 0.0197$ for all discretized x points in $\hat{\mathcal{H}}$, and the predicted $\hat{B}(x) < -0.0241$ for all discretized x points in $\mathcal{U}_\rho \backslash \hat{\mathcal{H}}$. Since $\hat{\mathcal{H}}$ is a superset of $\mathcal{D}$, it is proven that conditions Equations (5a) and (5c) hold, respectively. We also examine $L_{\hat{f}}\hat{B}(x)$ values for all discretized x points outside of the unsafe region and where $L_{\hat{g}}\hat{B}(x) = 0$. Although both the FNN and the RNN models can be expressed in continuous forms, for simplicity, we use numerical approximation to calculate $L_{\hat{f}}\hat{B}(x)$ and $L_{\hat{g}}\hat{B}(x)$ respectively. Due to the dichotomous nature of $\hat{B}$ having nearly constant values close to $+1$ or -1, all discretized points in $x \in \mathcal{U}_\rho \backslash \mathcal{D}$ such that $L_{\hat{g}}\hat{B}(x) = 0$ have $L_{\hat{f}}\hat{B}(x) = 0$. Although we cannot conclude that $\forall x \in \left\{ z \in \mathbf{R}^n \backslash \mathcal{D} \mid L_{\hat{g}}\hat{B}(x) = 0 \right\}$, $L_{\hat{f}}\hat{B}(x) \leq 0$ holds, we know that the FNN model with a high accuracy can achieve $L_{\hat{f}}\hat{B}(x) = 0$ within the discretized region. Therefore, we proceed with this FNN model for CBF and apply it in the CLBF-MPC to assess its closed-loop performance.

> *Remark* 5. When training the FNN model, one may find that the weighting parameters α and β need to be chosen based on a grid-search approach as these two parameters indicate the trade-off between classification accuracy and enforcing $L_{\hat{f}}\hat{B}(x) \leq 0$ for all discretized data points in the safe region. In our simulations, striving for high classification accuracy while minimizing $Cost_2$ yielded a good model with its safety requirements met. In the case that the verification against the safety requirements of Equations (5a)–(5c) are not met, the FNN needs to be retrained with more weighting on $Cost_2$ and more epochs.

6.3 | Closed-loop simulations

To demonstrate that the closed-loop state trajectory does not reach the unsafe region $\mathcal{D}$ when being driven towards the origin, we choose

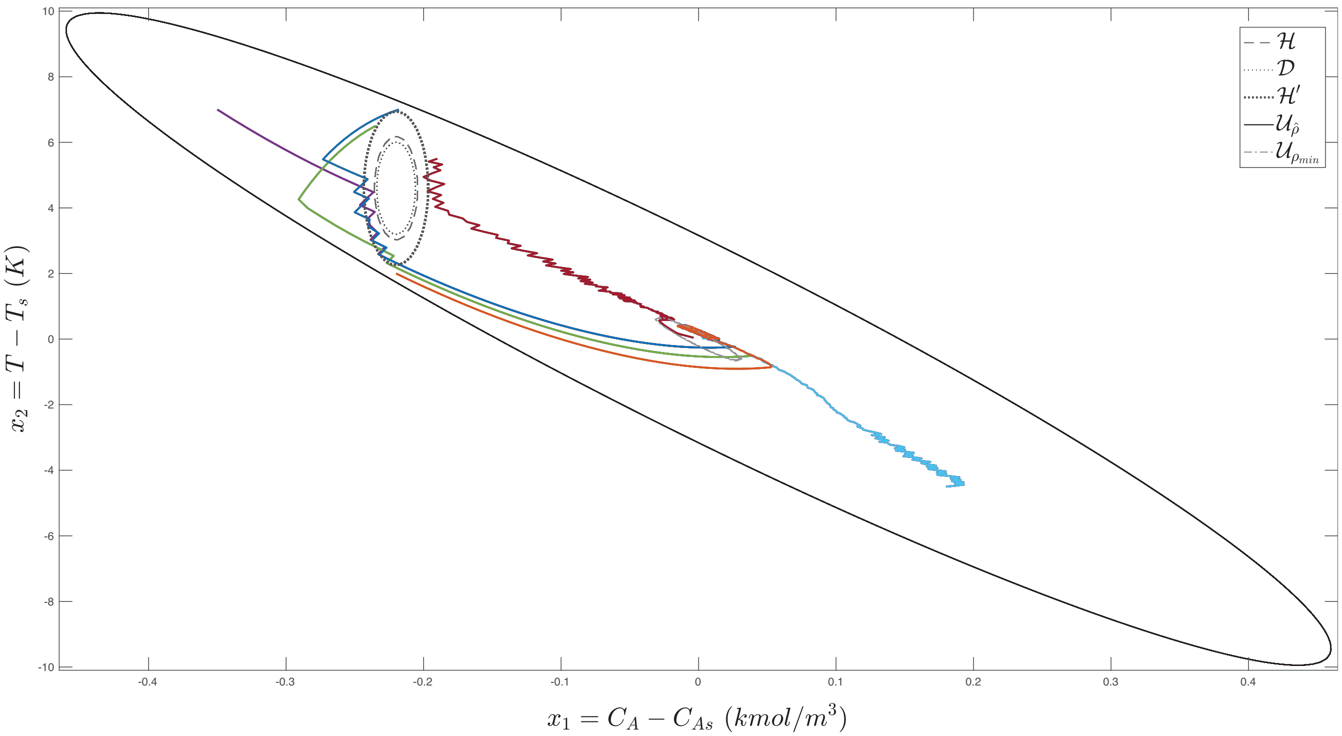

FIGURE 4 State trajectories originated from six different initial conditions in the safe operating regions under the closed-loop control of the CLBF-MPC using the RNN predictive model and the FNN-based control barrier function

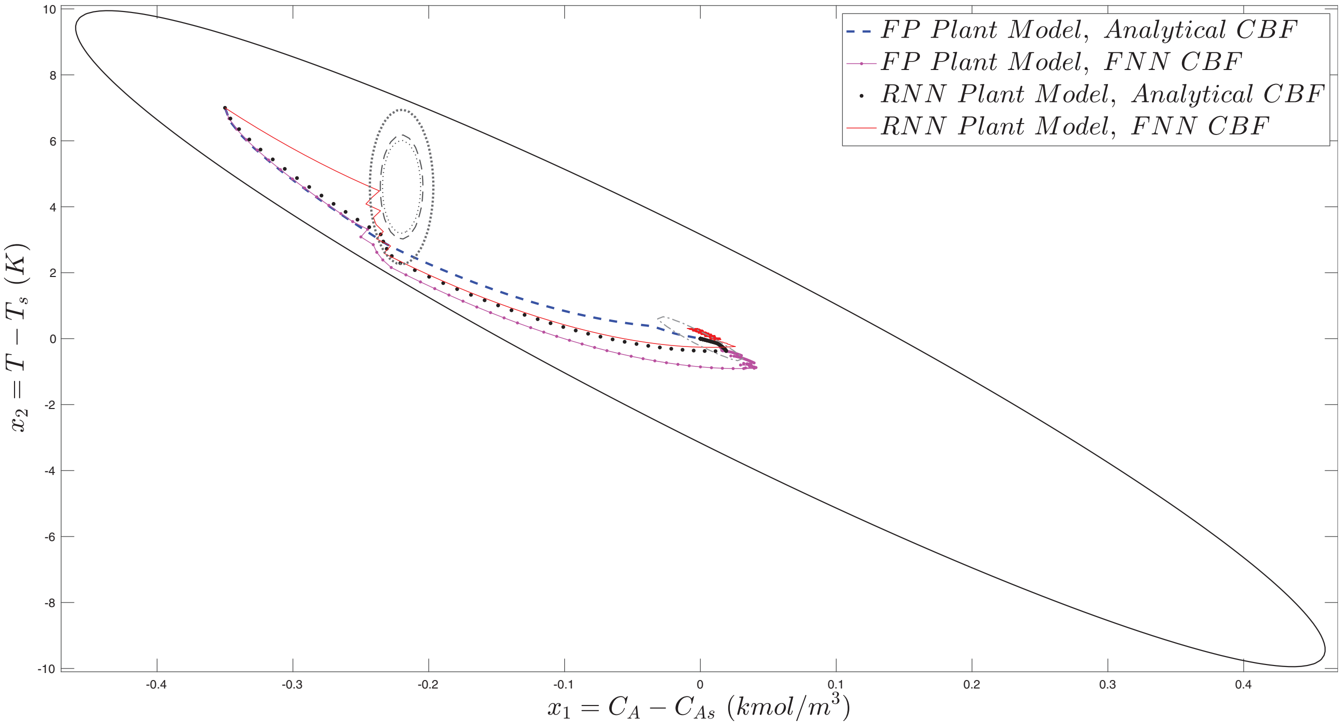

FIGURE 5 Closed-loop state trajectories under the CLBF-MPC using different combinations of first-principles (FP) process model or RNN process model, and analytical control barrier function (CBF) or FNN-based CBF

various initial conditions within $\mathcal{U}_\rho$ to start the simulation. It is demonstrated that the stabilization of the closed-loop system can be achieved when the simulation starts at an initial condition $(x_1,x_2) = (0.18, -4.5)$, which is on the opposite side far from the unsafe region. More initial conditions near the unsafe region within the ellipse $\mathcal{U}_\rho$ are selected, from which the closed-loop state would have encountered the unsafe region if a conventional tracking controller were to be implemented. It is demonstrated that although the state enters the region $\mathcal{H}'$ due to inevitable modeling error within the FNN model for the CBF and the RNN model for the nonlinear process, the state never reaches the border of $\mathcal{D}$. Note that $\mathcal{D}$ represents the actual unsafe set in state-space from physical law, $\mathcal{H}$ is the closed and compact set that encloses $\mathcal{D}$, $\mathcal{H}'$ is the set within which training data collected are deemed as "unsafe." In addition, we use $\hat{\mathcal{H}}$ to denote the unsafe set predicted by $\hat{B}(x)$, which as shown in the previous section, encloses the unsafe set given by the training dataset $\mathcal{H}'$. All trajectories demonstrate that the states can successfully converge to the terminal set $\mathcal{U}_{\rho_{min}}$ while not entering the unsafe region $\mathcal{D}$, as shown in Figure 4. We also compare the closed-loop performance of the proposed machine-learning-based CLBF-MPC with other CLBF-MPC's with various levels of machine-learning implemented as part of the formulation. The trajectories are shown in Figure 5. As shown, all trajectories successfully avoided the unsafe region, bounded in $\mathcal{U}_{\rho}$, and ultimately converged to $\mathcal{U}_{\rho_{min}}$. The trajectories using the analytical CBF, which is designed based on the region $\mathcal{H}$, enter and trespass the $\mathcal{H}'$ region (as they should in order to converge to the origin faster) while not entering the $\mathcal{H}$ region. The trajectories using the FNN-modeled CBF do oscillate around the boundary of the $\mathcal{H}'$ region due to modeling error, but remain distanced from the $\mathcal{H}$ region with the conservative contingency margin considered.

7 | CONCLUSION

In this work, we have demonstrated that nonlinear systems subject to input constraints could be stabilized by a CLBF-MPC while not entering unsafe regions where the barrier function was constructed using an FNN model and the predictive model within MPC was obtained using an RNN model. A CBF was first characterized by building an FNN model with unique structures and properties, and was then trained and validated using discretized data collected from a conservative rendition of unsafe and safe regions. Given sufficiently small bounded modeling errors with the two NN models, the proposed CLBF-MPC was able to meet its control objective of ensuring simultaneous stability and safety for all initial conditions within a subset of the stability region under sample-and-hold control action implementation. The effectiveness of the machine-learning-based CLBF-MPC was demonstrated using a nonlinear chemical process example with a bounded unsafe set.

ACKNOWLEDGMENTS

Financial support from the National Science Foundation and the Department of Energy is gratefully acknowledged.

AUTHOR CONTRIBUTION

Scarlett Chen: Conceptualization (equal); data curation (equal); formal analysis (equal); methodology (equal); software (equal); writing–original draft (equal). **Zhe Wu:** Conceptualization (supporting); formal analysis (supporting); methodology (supporting); writing–original draft (supporting). **Panagiotis Christofides:** Conceptualization (supporting); project administration (lead); supervision (lead), writing review & editing (lead).

DATA AVAILABILITY STATEMENT

Our source data are available upon request.

ORCID

Panagiotis D. Christofides https://orcid.org/0000-0002-8772-4348

REFERENCES

1. Crowl DA, Louvar JF. Chemical process safety-fundamentals with applications. *Process Safety Progress*. 2011;30:408-409.
2. Mhaskar P, El-Farra NH, Christofides PD. Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control. *Systems Control Lett*. 2006;55:650-659.
3. Muñoz de la Peña D, Christofides PD. Lyapunov-based model predictive control of nonlinear systems subject to data losses. *IEEE Trans Autom Control*. 2008;53:2076-2089.
4. Albalawi F, Durand H, Christofides PD. Process operational safety using model predictive control based on a process safeness index. *Comput Chem Eng*. 2017;104:76-88.
5. Romdlony MZ, Jayawardhana B. Stabilization with guaranteed safety using control Lyapunov-barrier function. *Automatica*. 2016; 66:39-47.
6. Wu Z, Christofides PD. Handling bounded and unbounded unsafe sets in control Lyapunov-barrier function-based model predictive control of nonlinear processes. *Chem Eng Res Des*. 2019;143:140-149.
7. Jankovic M. Combining control Lyapunov and barrier functions for constrained stabilization of nonlinear systems. *Proceedings of the American Control Conference*, IEEE; 2017:1916-1922.
8. Niu B, Zhao J. Barrier Lyapunov functions for the output tracking control of constrained nonlinear switched systems. *Syst Control Lett*. 2013;62:963-971.
9. Tee KP, Ge SS, Tay EH. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*. 2009;45: 918-927.
10. Ames AD, Grizzle JW, Tabuada P. Control barrier function based quadratic programs with application to adaptive cruise control. *Proceedings of the 53rd IEEE Conference on Decision and Control*. IEEE; 2014: 6271-6278.
11. Prajna S, Jadbabaie A. Safety verification of hybrid systems using barrier certificates. *Proceedings of the 7th International Workshop, Hybrid Systems: Computation and Control*. Vol 2993, Berlin, Heidelberg: Springer; 2004:477-492.
12. Xu X, Tabuada P, Grizzle JW, Ames AD. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*. 2015;48(27):54-61.
13. Wu Z, Albalawi F, Zhang Z, Zhang J, Durand H, Christofides PD. Control Lyapunov-barrier function-based model predictive control of nonlinear systems. *Automatica*. 2019;109:108508.
14. Kosmatopoulos EB, Polycarpou MM, Christodoulou MA, Ioannou PA. High-order neural network structures for identification of dynamical systems. *IEEE Trans Neural Netw*. 1995;6:422-431.
15. Sontag ED. Neural nets as systems models and controllers. *Proceedings of the Seventh Yale Workshop on Adaptive and Learning Systems*. Yale University; 1992:73-79.
16. Draeger A, Engell S, Ranke H. Model predictive control using neural networks. *IEEE Control Syst Magaz*. 1995;15:61-66.
17. Hewing L, Wabersich KP, Menner M, Zeilinger MN. Learning-based model predictive control: toward safe learning in control. *Annu Rev Control Robot Auton Syst*. 2020;3:269-296.
18. Wong WC, Chee E, Li J, Wang X. Recurrent neural network-based model predictive control for continuous pharmaceutical manufacturing. *Mathematics*. 2018;6:242.

19. Sontag ED. A “universal” construction of Artstein's theorem on nonlinear stabilization. *Syst Control Lett*. 1989;13:117-123.
20. Lin Y, Sontag ED. A universal formula for stabilization with bounded controls. *Syst Control Lett*. 1991;16:393-397.
21. Wu Z, Tran A, Rincon D, Christofides PD. Machine learning-based predictive control of nonlinear processes. Part I: theory. *AIChE J*. 2019;65:e16729.
22. Wieland P, Allgöwer F. Constructive safety using control barrier functions. *IFAC Proc Vol*. 2007;40:462-467.
23. Sibi P, Jones SA, Siddarth P. Analysis of different activation functions using back propagation neural networks. *J Theor Appl Inf Technol*. 2013;47:1264-1268.
24. Y. C. Chang, N. Roohi, and S. Gao. *Neural Lyapunov control*. arXiv preprint arXiv:2005.00611; 2020.
25. W. Jin, Z. Wang, Z. Yang, and S. Mou. *Neural certificates for safe control policies*. arXiv preprint arXiv:2006.08465; 2020.
26. Richards SM, Berkenkamp F, Krause A. The Lyapunov neural network: adaptive stability certification for safe learning of dynamical systems. *Proceedings of the Conference on Robot Learning*, PMLR; 2018:466-476.
27. Bobiti R, Lazar M. A sampling approach to finding Lyapunov functions for nonlinear discrete-time systems. *In Proceedings of the 2016 European Control Conference (ECC)*, IEEE; 2016:561-566.
28. Wu Z, Christofides PD. Control Lyapunov-barrier function-based predictive control of nonlinear processes using machine learning modeling. *Comput Chem Eng*. 2020;134:106706.

How to cite this article: Chen S, Wu Z, Christofides PD. Machine-learning-based construction of barrier functions and models for safe model predictive control. *AIChE J*. 2022;68(6):e17456. doi:10.1002/aic.17456