

Learning Discrete-Time Uncertain Nonlinear Systems With Probabilistic Safety and Stability Constraints

IMAN SALEHI ^{ID} (Member, IEEE), TYLER TAPLIN ^{ID} (Graduate Student Member, IEEE),
AND ASHWIN P. DANI ^{ID} (Senior Member, IEEE)

Department of Electrical and Computer Engineering at University of Connecticut, Storrs, CT 06269 USA

CORRESPONDING AUTHOR: ASHWIN P. DANI (e-mail: ashwin.dani@uconn.edu).

This work was supported in part by the Space Technology Research Institutes through NASA Space Technology Research Grants Program under Grant 80NSSC19K1076 and in part by NSF under Grant SMA-2134367.

ABSTRACT This paper presents a discrete-time dynamical system model learning method from demonstration while providing probabilistic guarantees on the safety and stability of the learned model. The controlled dynamic model of a discrete-time system with a zero-mean Gaussian process noise is approximated using an Extreme Learning Machine (ELM) whose parameters are learned subject to chance constraints derived using a discrete-time control barrier function and discrete-time control Lyapunov function in the presence of the ELM reconstruction error. To estimate the ELM parameters a quadratically constrained quadratic program (QCQP) is developed subject to the constraints that are only required to be evaluated at sampled points. Simulations validate that the system model learned using the proposed method can reproduce the demonstrations inside a prescribed safe set while converging to the desired goal location starting from various different initial conditions inside the safe set. Furthermore, it is shown that the learned model can adapt to changes in goal location during reproductions without violating the stability and safety constraints.

INDEX TERMS Discrete-time control barrier function, extreme learning machine, safe model learning.

I. INTRODUCTION

Advancements in data-driven dynamic model learning [1], [2], [3] provide new possibilities to learn dynamic models with applications to autonomous systems operating in remote, hazardous and/or confined environments. Behavior cloning (BC), one of the methods of imitation learning, learns a dynamic system model and/or a policy from expert demonstrations [4], [5]. Motion of autonomous systems or robots generated from expert demonstrations often exhibit stability and safety properties that should be preserved while learning the system model. BC presents many advantages such as increased sample efficiency compared to reinforcement learning and can be useful in many robotics and engineering applications [6], [7], [8], [9], [10], [11]. Dynamic system learning-based BC methods have incorporated stability [8], [9], [11], [12], [13] and safety [5], [10], [14] constraints in learning linear or nonlinear continuous-time system models. However, incorporating these constraints for uncertain discrete-time nonlinear

systems represented using neural networks (NN) with noise poses technical challenges for constrained learning of such system models.

This paper addresses the challenge by presenting a nonlinear dynamic system model learning method subject to probabilistic stability and safety constraints. The controlled dynamic model of a discrete-time system with a zero-mean Gaussian process noise is approximated using extreme learning machine (ELM), which is a computationally efficient approximation for the system model compared to NN [15]. Probabilistic constraints on the ELM parameter learning are derived using control Lyapunov function (CLF) and control Barrier function (CBF) theory for discrete-time systems. A chance-constrained quadratically constrained quadratic program (QCQP) optimization problem is then formulated to learn the ELM parameters with the probabilistic stability and safety constraints, which are further simplified to make the optimization problem tractable. The

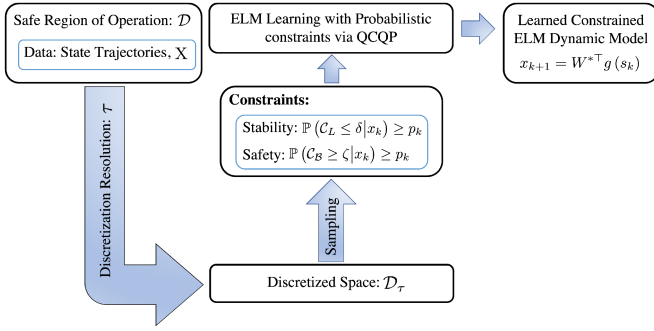


FIGURE 1. A block diagram that illustrates the information flow and procedure for the discrete-time dynamical system model learning method.

schematic view of the proposed approach is illustrated in Fig. 1.

The concept of safety is centered around the idea of constraining the behavior of the system model to a prescribed set, by ensuring forward invariance of the set with respect to the system model. Barrier functions (BF) are commonly used to certify the forward invariance of a closed set with respect to a system model. For the synthesis of safety-critical controller design CBFs have been used via Quadratic Programming (QP) [16], [17]. CLF and CBF are merged to synthesize stable and safe controllers by solving a QP in [18], [19]. CBF methods work for constraints on systems that have relative-degree more than one. In [20], a backstepping-inspired methodology is used to obtain a CBF that renders a closed set forward invariant when the system has relative degree more than one. However, demonstrating a backstepping-based CBF design for systems with relative degree greater than two in practice is challenging [21], [22]. An exponential control barrier function (ECBF) that can handle state-dependent constraints for nonlinear systems with any relative degree is introduced in [22]. In [23], a more general form of control barrier function termed high-order control barrier function (HOCBF) is proposed. HOCBFs are not restricted to exponential functions and are determined by set of class \mathcal{K} functions.

In [24] an adaptive CBF (aCBF) is proposed, which ensures the forward invariance of a closed set with respect to a nonlinear control-affine system with parametric uncertainties. In [25] the aCBF method is merged with an adaptive data-driven safety controller for contracting systems. Authors in [26] develop a method to learn a robustly safe controller along with learning the system parameters and corresponding uncertainty bounds. A metric based on the covariance of the parameter estimates is used to determine if the data is sufficient to update the parameters. These methods mainly focus on controller synthesis problem with a known dynamic system model or with model adaptation.

The controller synthesis methods assume the knowledge of barrier function. Methods for learning barrier functions and safe regions directly from sensor data are developed in [16], [27], and [28]. Although there is a lot of work on safe controller synthesis problem, very few methods are developed for

learning safe and stable closed-loop autonomous system using ELM dynamic system modeling that incorporates safety and stability constraints.

Classical linear system identification techniques utilize parametric model structure and guarantee the stability of learned models [2], [3]. Machine learning techniques, such as NN and Gaussian processes (GPs), have been used for nonlinear system identification [11], [29]. Model uncertainties, in the form of a NN approximation error, are accounted for in [14] along with stability and safety constraints using reciprocal BFs, where the uncertainty is bounded by a constant. In the context of Bayesian learning, GP state-space models are learned with stability constraints in [11]. In [30], a simultaneous continuous system model learning and control synthesis method is developed, where GPs are used for model learning, and probabilistic CBF and CLF are used to achieve safety and stability.

Compared to our previous work in [31], a controlled discrete-time nonlinear system with a zero-mean Gaussian noise is considered in this paper, resulting in a more generic formalism for learning closed-loop autonomous systems. Detailed proofs of theorems and lemmas are presented while including the ELM reconstruction error, and the method is evaluated on three different simulation examples. The contributions of the paper are summarized as follows.

- A learning method for discrete-time nonlinear controlled systems approximated using ELM with additive zero-mean Gaussian process noise is developed while verifying the probabilistic safety and stability properties of the learned system.
- A discrete-time zeroing control barrier function (DT-ZCBF) and discrete-time control Lyapunov function (DT-CLF) [32] are used to obtain safety and stability chance constraints that yield a computationally tractable system identification with probabilistic guarantees on the system's safety and stability, respectively.
- An ELM network is used to represent an uncertain discrete-time system, and the ELM reconstruction error is regarded as a bounded signal.
- The constrained ELM parameter learning problem is formulated as a QCQP, which contains finite number of decision variables with infinite constraints leading to a semi-infinite program. To make the optimization tractable, an equivalent optimization problem is derived that requires the constraints to be evaluated only at a finite number of sampled points [10], [33].
- The constrained model learning approach is demonstrated in simulation using three examples: (1) a 2 ° of freedom (DoF) planar robot whose joint positions are lower and upper bounded using a zeroing barrier function, (2) learning motion trajectories of one of the complex shapes from a publicly available dataset, (3) a wheeled mobile robot that travels along a certain trajectories in a confined space.

Notation: The sets of real numbers and integers are denoted by \mathbb{R} and \mathbb{Z} , respectively. The symbols \mathbb{R}^+ and \mathbb{Z}^+

denote the set of non-negative real numbers and non-negative integers, respectively. For ease of notation, the dependency of variables is abbreviated, for example, $x(k)$ is denoted by x_k , unless necessary for clarity. The standard Euclidean norm of a vector is denoted by $\|\cdot\|$ and $\|\cdot\|_F$ is the Frobenius norm. The weighted norm of a vector $x \in \mathbb{R}^n$ by a positive definite matrix A is denoted by $\|x\|_A = \sqrt{x^\top A x}$. The symbol $\lambda_{\max}\{\cdot\}$ denotes the maximum eigenvalue of a matrix. The symbol $[\cdot]_\tau$ denotes the function evaluation at a point that belongs to a state-space \mathcal{D}_τ . $g(s_k) \in \mathbb{R}^{n+1}$ denotes the sigmoid activation function computed using the estimated ELM input weight matrix U and the estimated internal slopes and biases a_p and b_p . $\epsilon(s_k) \in \mathbb{R}^n$ denotes the ELM model reconstruction error. The arguments of g and ϵ are dropped for brevity, unless necessary for clarity.

II. PRELIMINARIES

In this section, preliminaries of DT-CLF and DT-ZCBF are presented. Consider an unknown nonlinear discrete-time system of the form

$$x(k+1) = f(x(k), u(k)) + v(k), \quad (1)$$

where $f: \mathcal{D} \times \mathcal{U} \mapsto \mathcal{D}$ is a continuous function, $x(k) \in \mathcal{D} \subset \mathbb{R}^n$ is the state of the system at time step $k \in \mathbb{Z}^+$, and $u(k) = \pi(e(k)) \in \mathcal{U} \subset \mathbb{R}^m$ is the system control input with a policy $\pi(\cdot)$ that depends on the error $e(k) = x(k) - x^*$, $v(k)$ is independent and identically distributed zero-mean Gaussian process noise with covariance Σ , which is a symmetric and positive definite matrix, and x^* is the system equilibrium.

A. DISCRETE-TIME CONTROL LYAPUNOV FUNCTION

The classical stability analysis of continuous-time nonlinear dynamical systems using Lyapunov theory is extended to the discrete-time domain. It entails finding a positive definite function $V: \mathcal{D} \mapsto \mathbb{R}$, which decreases along the trajectories of the system given in (1). Refer to [32] and references therein for a comprehensive study on discrete-time Lyapunov theorems.

Definition 1 (DT-CLF): Given the discrete-time system (1), $V(x_k)$ is said to be a DT-CLF if it can be bounded by $\alpha_1(\|x_k\|) \leq V(x_k) \leq \alpha_2(\|x_k\|)$ and there exists a control input $u_k \in \mathcal{U}$ such that

$$\mathcal{C}_L(x_k, u_k) = \Delta V(x_k, u_k) + \beta(\|x_k\|) \leq 0, \quad \forall x_k \in \mathcal{D}, \quad (2)$$

where $\Delta V(x_k, u_k) := V(x_{k+1}) - V(x_k) = \Delta V_k$, α_1 , α_2 , and β belong to class \mathcal{K} function.

B. DISCRETE-TIME ZEROING CONTROL BARRIER FUNCTION

Safety properties can be achieved by constraining the solutions of the discrete-time dynamical system to a pre-specified closed set $\mathcal{S} = \{x_k \in \mathcal{D} : h(x_k) \geq 0\}$, where $h: \mathbb{R}^n \mapsto \mathbb{R}$ is a continuous function associated with a barrier function [32].

Definition 2 (DT-ZCBF): Given the discrete-time system (1) and the set \mathcal{S} , a continuously differentiable function $h: \mathbb{R}^n \mapsto \mathbb{R}$ is said to be a DT-ZCBF, if there exists a class \mathcal{K} function satisfying $\eta(r) < r$, $\forall r > 0$, a set \mathcal{D} with $\mathcal{S} \subseteq \mathcal{D} \subset \mathbb{R}^n$, and a control input $u_k \in \mathcal{U}$ such that

$\mathcal{C}_B(x_k, u_k) = \Delta h(x_k, u_k) + \eta(h(x_k)) \geq 0$, $\forall x_k \in \mathcal{D}$, (3)

where $\Delta h(x_k, u_k) := h(x_{k+1}) - h(x_k) = \Delta h_k$.

The set \mathcal{S} is *forward invariant* with respect to the discrete-time system in (1) if and only if there exists a DT-ZCBF as defined in Definition 2, which means for any initial condition $x_0 \in \mathcal{S}$, implies $x_k \in \mathcal{S}$, $\forall k \in \mathbb{Z}^+$ [34].

Assumption 1: The function $f: \mathcal{D} \times \mathcal{U} \mapsto \mathcal{D}$ is locally Lipschitz continuous and bounded in \mathcal{D} .

III. CHANCE-CONSTRAINED ELM LEARNING PROBLEM

Consider a discrete-time nonlinear dynamical system in (1), it is assumed that the unknown function $f(\cdot)$ can be parameterized as $f(x_k, \pi(e_k)) = \mathbf{W}^\top \mathbf{g}(s_k)$, where $\mathbf{W} \in \mathbb{R}^{(n_h+1) \times n}$ is the unknown ideal bounded constant output layer weight matrix, $\mathbf{g}(s_k) = [\psi(\mathbf{q}s_k)^\top, 1]^\top: \mathbb{R}^{2n+1} \mapsto \mathbb{R}^{n_h+1}$, $s_k = [x_k^\top, e_k^\top, 1]^\top \in \mathbb{R}^{2n+1}$, $\psi(\cdot) \in \mathbb{R}^{n_h}$ is the vector sigmoid activation function, n_h is the number of neurons in the hidden layer of the ELM. The vector sigmoid activation function $\psi(\cdot)$ is given by

$$\psi(\mathbf{q}s_k) = \left[\frac{1}{1 + e^{-(\mathbf{q}s_k)_1}}, \dots, \frac{1}{1 + e^{-(\mathbf{q}s_k)_{n_h}}} \right]^\top, \quad (4)$$

where $\mathbf{q} = [\mathbf{P}, \mathbf{b}_p] \in \mathbb{R}^{n_h \times (2n+1)}$, $\mathbf{P} = \text{diag}(\mathbf{a}_p)\mathbf{U}^\top \in \mathbb{R}^{n_h \times 2n}$, $\mathbf{a}_p \in \mathbb{R}^{n_h}$ and $\mathbf{b}_p \in \mathbb{R}^{n_h}$ are the ideal internal slopes and biases vectors, respectively and $\mathbf{U} \in \mathbb{R}^{2n \times n_h}$ is the ideal bounded input layer weight matrix. Other activation functions such as radial basis functions, tangent sigmoid can also be used [35], [36]. The nonlinear function $f(x_k, u_k)$, defined in (1), is modeled using an ELM given by

$$f(x_k, u_k) = \mathbf{W}^\top \mathbf{g}(s_k) + \epsilon(s_k), \quad (5)$$

where $\mathbf{W} \in \mathbb{R}^{(n_h+1) \times n}$ is the estimated ELM output weight matrix, $\epsilon(s_k) \in \mathbb{R}^n$ is the function reconstruction error, and $\mathbf{g}(s_k) = [\psi(\mathbf{q}s_k)^\top, 1]^\top$, where \mathbf{q} is constructed using the estimated ELM input weight matrix \mathbf{U} and the estimated internal slopes and biases a_p and b_p . The parameters a_p , b_p and \mathbf{U} of ELM are computed using intrinsic plasticity (IP) or batch intrinsic plasticity (BIP) algorithms. See [36] and [14] for preliminaries of ELM.

Assumption 2: The ideal weights of the ELM are bounded by known positive constants, i.e., $\|\mathbf{W}\|_F \leq \bar{W}$, $\|\mathbf{U}\|_F \leq \bar{U}$ [35].

Remark 1: The sigmoid function $\psi_i(\cdot) \in [0, 1]$ and hence its derivative $\psi_i(\cdot)(1 - \psi_i(\cdot))$ has upper and lower bounds given by $0 \leq \psi_i(\cdot)(1 - \psi_i(\cdot)) \leq 0.25$, $\forall i = 1, \dots, n_h$. Thus, $\|\psi(\cdot)\| \leq \sqrt{n_h}$ and $\|\psi(\cdot)(1 - \psi(\cdot))\| \leq 0.25\sqrt{n_h}$. Using Assumption 2, the ELM parameterization of $f(x_k, u_k)$ is Locally Lipschitz continuous.

Remark 2: The nonlinear function $f(x_k, u_k)$ is within real and positive constant $\bar{\epsilon}$ of the ELM range if there exists

a finite number of hidden neurons n_h and constant weights so that for all $x_k \in \mathcal{S}$ the approximation in (5) holds with $\|\epsilon(s_k)\| \leq \bar{\epsilon}$. The boundedness of the reconstruction error $\epsilon(s_k)$ follows from the *Universal Approximation Property* of single layer neural network [37] and the fact that for any activation function the reconstruction error goes to zero [15], [29], i.e., $\lim_{n_h \rightarrow \infty} \epsilon(s_k) = 0$.

Remark 3: Using Assumptions 1–2 and Remark 1, the following inequality holds: $\|\frac{\partial}{\partial s_k} \epsilon(s_k)\| \leq \bar{\epsilon}'$.

Let $X_{1:T_m}^{1:N} = \{x_1^j, \dots, x_{T_m}^j\}_{j=1}^{1:N}$ denote a set of N demonstrations with T_m time samples of the system's trajectory of (1), where each demonstration ends at a goal location x^* and $G_{0:T_m-1}^{1:N} = \{g(s_0^j), \dots, g(s_{T_m-1}^j)\}_{j=1}^{1:N}$ denote their sigmoid functions. The process noise v_k together with the model yields the following likelihood function

$$\begin{aligned} p(X|G, W) &= \prod_{j=1}^N \prod_{k=0}^{T_m-1} (2\pi)^{-\frac{n}{2}} |\Sigma|^{-\frac{1}{2}} \\ &\quad \times \exp\left(-\frac{1}{2} \|x_{k+1}^j - W^\top g(s_k^j) - \epsilon(s_k^j)\|_{\Sigma^{-1}}^2\right), \\ &= \prod_{j=1}^N \prod_{k=0}^{T_m-1} \mathcal{N}\left(W^\top g(s_k^j) + \epsilon(s_k^j), \Sigma\right) \end{aligned} \quad (6)$$

where the indexes for X and G are dropped for brevity. The Gaussian distribution of the process noise induces a distribution over the DT-CLF and DT-ZCBF constraints in (2) and (3), which can be used to obtain probabilistic stability and safety guarantees in the constrained parameter learning of the ELM, respectively.

III. PROBLEM FORMULATION

Consider a set of state trajectories $X_{1:T_m}^{1:N}$ and their corresponding sigmoid functions $G_{0:T_m-1}^{1:N}$ generated by a model of a controlled system, where there exists $u_k \in \mathcal{U}$ that stabilizes the system's trajectories to a desired goal location while keeping them in a confined space. The constrained model learning problem is then formulated as a parameter learning problem of an ELM approximation of $f(x_k, u_k)$ given in (5), such that the system in (1) satisfy probabilistic safety and stability properties.

Given a safe set \mathcal{S} defined in Section II, the ELM parameterized model in (5), the initial state $x_0 \in \mathcal{S}$, and the likelihood function $p(X|G, W)$ in (6), the ELM parameter learning problem can be formulated as a chance constrained regularized maximum log-likelihood estimation as follows

$$W^* = \arg \min_{W \in \mathbb{R}^{(n_h+1) \times n}} -\ln p(X|G, W) + \mu_W \text{tr}(W^\top W) \quad (7a)$$

$$\text{s.t. } \mathbb{P}(\mathcal{C}_B \geq \zeta | x_k) \geq p_k, \quad \forall x_k \in \mathcal{D}, \quad (7b)$$

$$\mathbb{P}(\mathcal{C}_L \leq \delta | x_k) \geq p_k, \quad \forall x_k \in \mathcal{D}, \quad (7c)$$

and $\forall k \in \mathbb{Z}^+$, where

$$\begin{aligned} &-\ln p(X|G, W) \\ &= \frac{1}{2} \sum_{j=1}^N \sum_{k=0}^{T_m-1} \|x_{k+1}^j - W^\top g(s_k^j) - \epsilon(s_k^j)\|_{\Sigma^{-1}}^2, \end{aligned} \quad (8)$$

$\mu_W \in \mathbb{R}^+$ is the regularization parameter, and W^* denotes the optimal solution for W , which is used in (5) to represent the learned system. $p_k \in (0, 1)$ is a user-specified risk tolerance, and $\zeta, \delta \in \mathbb{R}$.

The negative log-likelihood function in (8) has the ELM reconstruction error term, which is not measurable in practice. To circumvent this issue and to make (8) implementable, we only keep the terms with the decision variable W , in the objective function. For deriving the cost function consider the following expression

$$\frac{1}{2} \sum_{j=1}^N \sum_{k=0}^{T_m-1} \|x_{k+1}^j - W^\top g(s_k^j)\|_{\Sigma^{-1}}^2 + 2g(s_k^j)^\top W \Sigma^{-1} \epsilon(s_k^j). \quad (9)$$

Using the Young's inequality, the second term in (9) can be upper bounded so we have

$$2g(s_k^j)^\top W \Sigma^{-1} \epsilon(s_k^j) \leq \|W^\top g(s_k^j)\|_{\Sigma^{-1}}^2 + \|\epsilon(s_k^j)\|_{\Sigma^{-1}}^2, \quad (10)$$

where $\|\epsilon(s_k)\|_{\Sigma^{-1}}^2$ can be ignored since it is not associated with the decision variable W . Hence, the cost function $J(W)$ to be implemented takes the following form

$$\begin{aligned} J(W) &= \frac{1}{2} \sum_{j=1}^N \sum_{k=0}^{T_m-1} \|x_{k+1}^j - W^\top g(s_k^j)\|_{\Sigma^{-1}}^2 + \|W^\top g(s_k^j)\|_{\Sigma^{-1}}^2 \\ &\quad + \mu_W \text{tr}(W^\top W) \end{aligned} \quad (11)$$

IV. CHANCE-CONSTRAINED OPTIMIZATION

A. ENCODING PROBABILISTIC SAFETY CONSTRAINTS

DT-ZCBF defines a forward invariant set such that solutions of the nonlinear dynamical system that start in that set remain in that set for all time. A hyperellipsoid is used to represent a DT-ZCBF, $h(x_k) : \mathbb{R}^n \mapsto \mathbb{R}$, and it is given by

$$h(x_k) = 1 - (x_k - \bar{x})^\top A(x_k - \bar{x}), \quad (12)$$

where $A \in \mathbb{R}^{n \times n}$ is a symmetric positive definite matrix, and \bar{x} is the vector that denotes the center of gravity of the hyperellipsoid.

To account for the uncertainty in safety constraint satisfaction, the DT-ZCBF is rewritten as

$$\mathbb{P}(\mathcal{C}_B \geq \zeta | x_k) = 1 - \mathcal{F}_{\mathcal{C}_B}(\zeta) \geq p_k \quad (13)$$

where $\mathcal{F}_{\mathcal{C}_B}(\zeta)$ is given by

$$\mathcal{F}_{\mathcal{C}_B}(\zeta) \triangleq \Phi\left(\frac{\zeta - \mathbb{E}[\mathcal{C}_B]}{\sqrt{\text{Var}[\mathcal{C}_B]}}\right) = \frac{1}{2} + \frac{1}{2} \text{erf}\left(\frac{\zeta - \mathbb{E}[\mathcal{C}_B]}{\sqrt{2\text{Var}[\mathcal{C}_B]}}\right) \quad (14)$$

and $\Phi(\cdot)$ is a cumulative distribution function of the standard Gaussian. Note that as the variance tends to zero, i.e., $\text{Var}[\mathcal{C}_B] \rightarrow 0$, the probability $\mathbb{P}(\mathcal{C}_B \geq \zeta | x_k)$ tends to one when $\zeta < \mathbb{E}[\mathcal{C}_B]$. As the uncertainty in the system dynamics decreases to zero the safety assurance can be guaranteed with probability one. Substituting (14) into (13), the probabilistic safety constraint is given by

$$\zeta - \mathbb{E}[\mathcal{C}_B] \leq -c(p_k)\sqrt{\text{Var}[\mathcal{C}_B]}, \quad (15)$$

where $c(p_k) = \sqrt{2} \text{erf}^{-1}(2p_k - 1)$.

Lemma 1 is presented next to compute the parameters required for the cumulative distribution in (15), i.e., $\mathbb{E}[\mathcal{C}_B]$ and $\text{Var}[\mathcal{C}_B]$.

Lemma 1: Given the controlled system in (1) modeled using (5), the process noise v_k induces a distribution over the DT-ZCBF constraint in (3) with the following parameters:

$$\mathbb{E}[\mathcal{C}_B] = 1 - \left\| (W^\top g + \epsilon - \bar{x}) \right\|_A^2 - \text{tr}(A\Sigma) - (1 - \gamma)h(x_k), \quad (16)$$

$$\text{Var}[\mathcal{C}_B] = 4 \left\| A(W^\top g + \epsilon - \bar{x}) \right\|_\Sigma^2 + 2\text{tr}(A\Sigma A\Sigma), \quad (17)$$

where the class \mathcal{K} function $\eta(h(x_k))$ in Definition 2 is selected as $\eta(h(x_k)) = \gamma h(x_k)$, and γ is a positive constant.

Proof: Substituting (12) in (3) for $h(x_{k+1})$ and using (1) and (5) yields

$$\mathcal{C}_B = 1 - \left\| W^\top g + \epsilon + v_k - \bar{x} \right\|_A^2 - (1 - \gamma)h(x_k). \quad (18)$$

Taking the expectation of \mathcal{C}_B we have

$$\begin{aligned} \mathbb{E} \left[1 - \left\| W^\top g + \epsilon + v_k - \bar{x} \right\|_A^2 - (1 - \gamma)h(x_k) \right] \\ = 1 - (W^\top g + \epsilon - \bar{x})^\top A \\ \times (W^\top g + \epsilon - \bar{x}) - \text{tr}(A\Sigma) - (1 - \gamma)h(x_k), \end{aligned} \quad (19)$$

and the variance is

$$\begin{aligned} \text{Var}[\mathcal{C}_B] &= \mathbb{E} \left[(h(x_{k+1}) - \mathbb{E}[h(x_{k+1})])^2 \right] \\ &= \mathbb{E} \left[(-2g^\top W A v_k - 2\epsilon^\top A v_k - v_k^\top A v_k + 2\bar{x}^\top A v_k \right. \\ &\quad \left. + \text{tr}(A\Sigma))^2 \right]. \end{aligned} \quad (20)$$

Utilizing properties such as the odd moments of multivariate Gaussian distribution are zero and the trace is invariant under circular permutations, (20) can be written as

$$\begin{aligned} \text{Var}[\mathcal{C}_B] &= 4(g^\top W A \Sigma A W^\top g + \epsilon^\top A \Sigma A \epsilon + \bar{x}^\top A \Sigma A \bar{x} \\ &\quad + 2g^\top W A \Sigma A \epsilon - 2g^\top W A \Sigma A \bar{x} - 2\epsilon^\top A \Sigma A \bar{x}) \\ &\quad + \mathbb{E} \left[(v_k^\top A v_k)^2 \right] - \text{tr}^2(A\Sigma). \end{aligned} \quad (21)$$

After substituting $\mathbb{E}[(v_k^\top A v_k)^2] = \text{tr}^2(A\Sigma) + 2\text{tr}(A\Sigma A\Sigma)$ in (21), the result in (17) is obtained. ■

B. ENCODING PROBABILISTIC STABILITY CONSTRAINTS

In this subsection, a probabilistic stability constraint is derived using Definition 1 for the uncertain discrete-time system given

in (5) and a Lyapunov candidate $V(x_k) = (x_k - x^*)^\top P(x_k - x^*)$, where $P \in \mathbb{R}^{n \times n}$ such that $P = P^\top > 0$.

To account for the uncertainty in stability constraint satisfaction, the DT-CLF in (2) can be rewritten as

$$\mathbb{P}(\mathcal{C}_L \leq \delta | x_k) = \mathcal{F}_{\mathcal{C}_L}(\delta) \geq p_k, \quad (22)$$

where

$$\mathcal{F}_{\mathcal{C}_L}(\delta) = \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\delta - \mathbb{E}[\mathcal{C}_L]}{\sqrt{2\text{Var}[\mathcal{C}_L]}} \right). \quad (23)$$

Substituting (23) into (22), the probabilistic stability constraint is given by

$$\delta - \mathbb{E}[\mathcal{C}_L] \geq c(p_k)\sqrt{\text{Var}[\mathcal{C}_L]}, \quad (24)$$

where $\mathbb{E}[\mathcal{C}_L]$ and $\text{Var}[\mathcal{C}_L]$ are given in Lemma 2, which is presented next.

Lemma 2: Given the controlled system in (1) modeled using (5), the process noise v_k induces a distribution over the DT-CLF constraint in (2) with the following parameters:

$$\mathbb{E}[\mathcal{C}_L] = \left\| (W^\top g + \epsilon - x^*) \right\|_P^2 + \text{tr}(P\Sigma) - (1 - \rho)V(x_k), \quad (25)$$

$$\text{Var}[\mathcal{C}_L] = 4 \left\| P(W^\top g + \epsilon - x^*) \right\|_\Sigma^2 + 2\text{tr}(P\Sigma P\Sigma), \quad (26)$$

where the class \mathcal{K} function $\beta(\|x_k\|)$ in Definition 1 is defined as $\beta(\|x_k\|) = \rho V(x_k)$ and $0 < \rho < 1$.

Proof: Substituting $V(x_k) = (x_k - x^*)^\top P(x_k - x^*)$ in (2) for $V(x_{k+1})$ and using (1) and (5) yields

$$\mathcal{C}_L = \left\| W^\top g + \epsilon + v_k - x^* \right\|_P^2 - (1 - \rho)V(x_k). \quad (27)$$

Taking the expectation of \mathcal{C}_L we have

$$\begin{aligned} \mathbb{E} \left[\left\| W^\top g + \epsilon + v_k - x^* \right\|_P^2 - (1 - \rho)V(x_k) \right] \\ = (W^\top g + \epsilon - x^*)^\top P(W^\top g + \epsilon - x^*) \\ + \text{tr}(P\Sigma) - (1 - \rho)V(x_k), \end{aligned} \quad (28)$$

and the variance is

$$\begin{aligned} \text{Var}[\mathcal{C}_L] &= \mathbb{E} \left[(V(x_{k+1}) - \mathbb{E}[V(x_{k+1})])^2 \right] \\ &= \mathbb{E} \left[(2g^\top W P v_k + 2\epsilon^\top P v_k + v_k^\top A v_k - 2x^{*T} P v_k \right. \\ &\quad \left. - \text{tr}(P\Sigma))^2 \right]. \end{aligned} \quad (29)$$

Utilizing properties that odd moments of multivariate Gaussian distribution are zero, and that the trace, when the matrices are of suitable dimensions, is invariant under circular permutations, (29) can be written as

$$\begin{aligned} \text{Var}[\mathcal{C}_L] &= 4(g^\top W P \Sigma P W^\top g + \epsilon^\top P \Sigma P \epsilon + x^{*T} P \Sigma P x^* \\ &\quad + 2g^\top W P \Sigma P \epsilon - 2g^\top W P \Sigma P x^* - 2\epsilon^\top P \Sigma P x^*) \\ &\quad + \mathbb{E} \left[(\epsilon^\top P \epsilon)^2 \right] - \text{tr}^2(P\Sigma) \end{aligned} \quad (30)$$

After substituting $\mathbb{E}[(\epsilon^\top P \epsilon)^2] = \text{tr}^2(P\Sigma) + 2\text{tr}(P\Sigma P\Sigma)$ in (30), the result in (26) is obtained. ■

Assumption 3: The variances $\text{Var}[\mathcal{C}_B] \geq 1$ and $\text{Var}[\mathcal{C}_L] \geq 1$.

Remark 4: Functions $\sqrt{\text{Var}[\mathcal{C}_B]}$ and $\sqrt{\text{Var}[\mathcal{C}_L]}$ are not Lipschitz continuous on $[0, 1]$ as the slope of the tangent line to their corresponding arguments become steeper as they approach to zero. Therefore, Assumption 3 is made to satisfy the constraint's Lipschitz continuity requirement.

Remark 5: If Assumption 3 does not hold, i.e., $\text{Var}[\mathcal{C}_B] < 1$ and $\text{Var}[\mathcal{C}_L] < 1$, the right hand side (RHS) of the constraints in (15) and (24) can be modified by introducing a constant $\xi > 1$ such that $\sqrt{\xi^2 \text{Var}[\mathcal{C}_B]} \geq 1$ and $\sqrt{\xi^2 \text{Var}[\mathcal{C}_L]} \geq 1$. For example, for the safety constraint in (15) we get

$$\zeta - \mathbb{E}[\mathcal{C}_B] \leq \frac{-c(p_k)}{\xi} \sqrt{\xi^2 \text{Var}[\mathcal{C}_B]}, \quad (31)$$

for which the right hand side of (31) can be lower bounded by $\zeta - \mathbb{E}[\mathcal{C}_B] \leq -\xi c(p_k) \text{Var}[\mathcal{C}_B]$. Scaling by the selected ξ is equivalent to solving the optimization problem with a tighter constraint.

C. ENCODING PROBABILISTIC SAFETY AND STABILITY CONSTRAINTS VIA QCQP

In this section, the chance-constrained ELM learning problem is formulated as a QCQP.

Proposition 1: Given the controlled system in (1) modeled using (5) and Assumption 3, the ELM parameter learning problem with probabilistic safety and stability constraints given in (7b)-(7c) can be formulated as a QCQP with the cost function in (11) and the following constraints:

$$W^* = \arg \min_W J(W) \quad (32a)$$

$$\text{s.t.} \quad \|A^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 \leq \Gamma_{\mathcal{C}_B}, \quad (32b)$$

$$\|P^{\frac{1}{2}} (W^\top g + \epsilon - x^*)\|^2 \leq \Gamma_{\mathcal{C}_L}, \quad (32c)$$

where

$$\begin{aligned} \mathcal{A} &\triangleq A + 4c(p_k)A^\top \Sigma A, \quad \mathcal{P} \triangleq P + 4c(p_k)P^\top \Sigma P, \\ \Gamma_{\mathcal{C}_B} &\triangleq 1 - \zeta - \text{tr}(A\Sigma) - (1 - \gamma)h(x_k) - 2c(p_k)\text{tr}(A\Sigma A\Sigma), \\ \Gamma_{\mathcal{C}_L} &\triangleq \delta - \text{tr}(P\Sigma) + (1 - \rho)V(x_k) - 2c(p_k)\text{tr}(P\Sigma P\Sigma). \end{aligned}$$

Proof: Using Assumption 3, the RHS of (15) can be lower bounded, and the constraint can be rewritten as

$$\zeta - \mathbb{E}[\mathcal{C}_B] \leq -c(p_k)\text{Var}[\mathcal{C}_B]. \quad (33)$$

Substituting $\mathbb{E}[\mathcal{C}_B]$ and $\text{Var}[\mathcal{C}_B]$ from (16) and (17) into (33) we have

$$\begin{aligned} &\|A^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 + 4c(p_k) \left\| \Sigma^{\frac{1}{2}} A (W^\top g + \epsilon - \bar{x}) \right\|^2 \leq \\ &1 - \zeta - \text{tr}(A\Sigma) - (1 - \gamma)h(x_k) - 2c(p_k)\text{tr}(A\Sigma A\Sigma), \end{aligned} \quad (34)$$

where the RHS of (34) can be substituted by $\Gamma_{\mathcal{C}_B}$. Expanding and rearranging (34) we get

$$(W^\top g + \epsilon - \bar{x})^\top (A + 4c(p_k)A^\top \Sigma A) (W^\top g + \epsilon - \bar{x}) \leq \Gamma_{\mathcal{C}_B}, \quad (35)$$

which after substituting \mathcal{A} for the middle term in (35) the result in (32b) is obtained.

Similarly, using Assumption 3 to upper bound the RHS of (24) we have

$$\delta - \mathbb{E}[\mathcal{C}_L] \geq c(p_k)\text{Var}[\mathcal{C}_L], \quad (36)$$

and when $\mathbb{E}[\mathcal{C}_L]$ and $\text{Var}[\mathcal{C}_L]$ from (25) and (26) are substituted into (36) we get

$$\begin{aligned} &\left\| P^{\frac{1}{2}} (W^\top g + \epsilon - x^*) \right\|^2 + 4c(p_k) \left\| \Sigma^{\frac{1}{2}} P (W^\top g + \epsilon - x^*) \right\|^2 \leq \\ &\delta - \text{tr}(P\Sigma) + (1 - \rho)V(x_k) - 2c(p_k)\text{tr}(P\Sigma P\Sigma), \end{aligned} \quad (37)$$

where the RHS of (37) can be substituted by $\Gamma_{\mathcal{C}_L}$. Expanding and rearranging (37) we get

$$\begin{aligned} &(W^\top g + \epsilon - x^*)^\top (P + 4c(p_k)P^\top \Sigma P) \\ &\times (W^\top g + \epsilon - x^*) \leq \Gamma_{\mathcal{C}_L}, \end{aligned} \quad (38)$$

which after substituting \mathcal{P} for the middle term in (38) the result in (32c) is obtained. ■

The constraints (32b) and (32c) for the optimization problem must be satisfied for all $x_k \in \mathcal{D}$, which leads to a semi-infinite program. To circumvent the challenge of having an uncountably infinite set, Theorem 1 is formulated that proves only enforcing a modified constraints for a finite number of sampled points is sufficient for obtaining a solution for the optimization problem in (32a)-(32c).

Let $\mathcal{D}_\tau \subset \mathcal{D}$ be a discretization of the state space \mathcal{D} with the closest point in \mathcal{D}_τ to $x_k \in \mathcal{D}$ denoted by $\|x_k - [x_k]_\tau\| \leq \frac{\tau}{2}$, where τ is the discretization resolution. Thus, sampling x_k from \mathcal{D}_τ , we have $\|s_k - [s_k]_\tau\| \leq \frac{\tau}{\sqrt{2}}$.

Theorem 1: If Assumptions 2-3 hold, then the optimization problem in (32a)-(32c) can be solved with the following modified conditions

$$\|(W^\top [g]_\tau - \bar{x})\|_{\mathcal{A}}^2 - [\Gamma_{\mathcal{C}_B}]_\tau \leq -\tau (\chi_{\mathcal{C}_B} + \Upsilon_{\mathcal{C}_B} + \mathfrak{J}_{\mathcal{C}_B}) - \Theta_{\mathcal{C}_B} \bar{\epsilon}, \quad (39)$$

$$\|(W^\top [g]_\tau - x^*)\|_{\mathcal{P}}^2 - [\Gamma_{\mathcal{C}_L}]_\tau \leq -\tau (\chi_{\mathcal{C}_L} + \Upsilon_{\mathcal{C}_L} + \mathfrak{J}_{\mathcal{C}_L}) - \Theta_{\mathcal{C}_L} \bar{\epsilon}, \quad (40)$$

$\forall [x]_\tau \in \mathcal{D}_\tau$, where

$$\chi_{\mathcal{C}_B} \triangleq \left(\lambda_{\max}\{\mathcal{M}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{A}\} \|\bar{x}\| \right) \bar{g}, \quad (41a)$$

$$\Upsilon_{\mathcal{C}_B} \triangleq (1 - \gamma) \lambda_{\max}\{A\} (\|x_k\| + \|\bar{x}\|), \quad (41b)$$

$$\mathfrak{J}_{\mathcal{C}_B} \triangleq \left((\|\bar{x}\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{A}\} \bar{\epsilon}', \quad (41c)$$

$$\Theta_{\mathcal{C}_B} \triangleq 2 \lambda_{\max}\{\mathcal{A}\} \left(\bar{W} \sqrt{n_h + 1} + \|\bar{x}\| + \frac{\bar{\epsilon}}{2} \right), \quad (41d)$$

$$\chi_{C_L} \triangleq \left(\lambda_{\max}\{\mathcal{H}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{P}\} \|x^*\| \right) \bar{g}, \quad (41e)$$

$$\Upsilon_{C_L} \triangleq (1 - \rho) \lambda_{\max}\{\mathcal{P}\} (\|x_k\| + \|x^*\|), \quad (41f)$$

$$\mathfrak{J}_{C_L} \triangleq \left((\|x^*\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{P}\} \bar{\epsilon}', \quad (41g)$$

$$\Theta_{C_L} \triangleq 2 \lambda_{\max}\{\mathcal{P}\} \left(\bar{W} \sqrt{n_h + 1} + \|x^*\| + \frac{\bar{\epsilon}}{2} \right), \quad (41h)$$

$\mathcal{M} \triangleq W A W^\top$, $\mathcal{H} \triangleq W P W^\top$, $\bar{g} \triangleq \frac{\bar{a}_p \bar{U} \sqrt{n_h}}{2\sqrt{2}}$, and $\bar{a}_p = \|\text{diag}(a_p)\|_F$ then the safety and stability conditions in (32b) and (32c) are satisfied $\forall x_k \in \mathcal{D}$.

Proof: The modified safety constraint in (39) can be written as

$$\|\mathcal{A}^{\frac{1}{2}} (W^\top [g]_\tau - \bar{x})\|^2 - [\Gamma_{C_B}]_\tau + \tau (\chi_{C_B} + \Upsilon_{C_B} + \mathfrak{J}_{C_B}) \leq 0, \quad (42)$$

$\forall x_k \in \mathcal{D}_\tau$. If $\|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \Gamma_{C_B}$ is a lower bound on the left hand side of (42) then the safety constraint in (32b) is satisfied. Consider

$$\mathcal{O}_B = \|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \|\mathcal{A}^{\frac{1}{2}} (W^\top [g]_\tau + [\epsilon]_\tau - \bar{x})\|^2 - (\Gamma_{C_B} - [\Gamma_{C_B}]_\tau), \quad (43)$$

which by using the triangle inequality \mathcal{O}_B can be upper bounded as

$$\mathcal{O}_B \leq \left| \|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \|\mathcal{A}^{\frac{1}{2}} (W^\top [g]_\tau + [\epsilon]_\tau - \bar{x})\|^2 \right| + |\Gamma_{C_B} - [\Gamma_{C_B}]_\tau|. \quad (44)$$

After expanding each term, using Assumption 2 and Remark 1, we obtain an upper bound on the RHS of (44) given by

$$\begin{aligned} & \left(\lambda_{\max}\{\mathcal{M}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{A}\} \|\bar{x}\| \right) \frac{\bar{a}_p \bar{U} \sqrt{n_h}}{2\sqrt{2}} \tau + \\ & \left((\|\bar{x}\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \frac{\bar{a}_p \bar{U} \sqrt{n_h}}{2\sqrt{2}} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{A}\} \bar{\epsilon}' \tau + \\ & (1 - \gamma) \lambda_{\max}\{\mathcal{A}\} (\|x_k\| + \|\bar{x}\|) \tau. \end{aligned} \quad (45)$$

Using (43) and the upper bound obtained in (45) yields

$$\begin{aligned} & \|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \Gamma_{C_B} \\ & \leq \|\mathcal{A}^{\frac{1}{2}} (W^\top [g]_\tau + [\epsilon]_\tau - \bar{x})\|^2 \\ & \quad - [\Gamma_{C_B}]_\tau + \left(\lambda_{\max}\{\mathcal{M}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{A}\} \|\bar{x}\| \right) \bar{g} \tau \\ & \quad + \left((\|\bar{x}\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{A}\} \bar{\epsilon}' \tau + (1 - \gamma) \times \\ & \quad \lambda_{\max}\{\mathcal{A}\} (\|x_k\| + \|\bar{x}\|) \tau. \end{aligned} \quad (46)$$

The first term on the RHS of (46) contains $[\epsilon]_\tau$. Hence, it is not measurable. However, expanding the norm squared and upper bounding the terms that include $[\epsilon]_\tau$ results in

$$\|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \Gamma_{C_B} \leq \|\mathcal{A}^{\frac{1}{2}} (W^\top [g]_\tau - \bar{x})\|^2 -$$

$$\begin{aligned} & [\Gamma_{C_B}]_\tau + 2 \lambda_{\max}\{\mathcal{A}\} \left(\bar{W} \sqrt{n_h + 1} + \|\bar{x}\| + \frac{\bar{\epsilon}}{2} \right) \bar{\epsilon} \\ & + \left(\lambda_{\max}\{\mathcal{M}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{A}\} \|\bar{x}\| \right) \bar{g} \tau \\ & + \left((\|\bar{x}\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{A}\} \bar{\epsilon}' \tau + (1 - \gamma) \times \\ & \lambda_{\max}\{\mathcal{A}\} (\|x_k\| + \|\bar{x}\|) \tau, \quad \forall x_k \in \mathcal{D} \text{ and } \forall [x_k]_\tau \in \mathcal{D}_\tau. \end{aligned} \quad (47)$$

From (42), (47), and (41a)–(41d), it is concluded that $\|\mathcal{A}^{\frac{1}{2}} (W^\top g + \epsilon - \bar{x})\|^2 - \Gamma_{C_B} \leq 0, \forall x_k \in \mathcal{D}$.

In a similar fashion, the modified stability constraint in (40) can be written as

$$\begin{aligned} & \|\mathcal{P}^{\frac{1}{2}} (W^\top [g]_\tau - x^*)\|^2 - [\Gamma_{C_L}]_\tau \\ & + \tau (\chi_{C_L} + \Upsilon_{C_L} + \mathfrak{J}_{C_L}) \leq 0, \end{aligned} \quad (48)$$

$\forall x_k \in \mathcal{D}_\tau$. If $\|\mathcal{P}^{\frac{1}{2}} (W^\top g + \epsilon - x^*)\|^2 - \Gamma_{C_L}$ is a lower bound on the left hand side of (48) then the safety constraint in (32c) is satisfied. Consider

$$\begin{aligned} \mathcal{O}_L &= \|\mathcal{P}^{\frac{1}{2}} (W^\top g + \epsilon - x^*)\|^2 - \|\mathcal{P}^{\frac{1}{2}} (W^\top [g]_\tau \\ & + [\epsilon]_\tau - x^*)\|^2 - (\Gamma_{C_L} - [\Gamma_{C_L}]_\tau), \end{aligned} \quad (49)$$

which by using the triangle inequality can be upper bounded as

$$\begin{aligned} \mathcal{O}_L &\leq \left| \|\mathcal{P}^{\frac{1}{2}} (W^\top g + \epsilon - x^*)\|^2 - \|\mathcal{P}^{\frac{1}{2}} (W^\top [g]_\tau \right. \\ & \quad \left. + [\epsilon]_\tau - x^*)\|^2 \right| + |\Gamma_{C_L} - [\Gamma_{C_L}]_\tau|. \end{aligned} \quad (50)$$

After expanding each term and using Assumption 2 and Remark 1, the RHS of (50) can be upper bounded by

$$\begin{aligned} & \left(\lambda_{\max}\{\mathcal{H}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{P}\} \|x^*\| \right) \frac{\bar{a}_p \bar{U} \sqrt{n_h}}{2\sqrt{2}} \tau + \\ & \left((\|x^*\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \frac{\bar{a}_p \bar{U} \sqrt{n_h}}{2\sqrt{2}} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{P}\} \bar{\epsilon}' \tau + \\ & (1 - \gamma) \lambda_{\max}\{\mathcal{P}\} (\|x_k\| + \|x^*\|) \tau. \end{aligned} \quad (51)$$

Using (49) and the upper bound derived in (51) yields

$$\begin{aligned} & \|\mathcal{P}^{\frac{1}{2}} (W^\top g + \epsilon - x^*)\|^2 - \Gamma_{C_L} \leq \|\mathcal{P}^{\frac{1}{2}} \\ & \quad \times (W^\top [g]_\tau + [\epsilon]_\tau - x^*)\|^2 - [\Gamma_{C_L}]_\tau \\ & \quad + \left(\lambda_{\max}\{\mathcal{H}\} \sqrt{n_h + 1} + \bar{W} \lambda_{\max}\{\mathcal{P}\} \|x^*\| \right) \bar{g} \tau \\ & \quad + \left((\|x^*\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g} \bar{W} \tau \right) \lambda_{\max}\{\mathcal{P}\} \bar{\epsilon}' \tau + (1 - \rho) \\ & \quad \times \lambda_{\max}\{\mathcal{P}\} (\|x_k\| + \|x^*\|) \tau. \end{aligned} \quad (52)$$

The first term on the RHS of (52) contains $[\epsilon]_\tau$. Hence, it is not measurable. However, after expanding the norm squared, the terms associated with $[\epsilon]_\tau$ can be upper bounded, and in

doing so we obtain

$$\begin{aligned}
& \|\mathcal{P}^{\frac{1}{2}}(W^\top g + \epsilon - x^*)\|^2 - \Gamma_{C_L} \leq \|\mathcal{P}^{\frac{1}{2}}(W^\top [g]_\tau - x^*)\|^2 \\
& - [\Gamma_{C_L}]_\tau + 2\lambda_{\max}\{\mathcal{P}\} \left(\bar{W}\sqrt{n_h + 1} + \|x^*\| + \frac{\bar{\epsilon}}{2} \right) \bar{\epsilon} \\
& + \left(\lambda_{\max}\{\mathcal{H}\} \sqrt{n_h + 1} + \bar{W}\lambda_{\max}\{\mathcal{P}\} \|x^*\| \right) \bar{g}\tau \\
& + \left((\|x^*\| + \bar{\epsilon}) \frac{2}{\sqrt{2}} + \bar{g}\bar{W}\tau \right) \lambda_{\max}\{\mathcal{P}\} \bar{\epsilon}'\tau + (1 - \gamma) \\
& \times \lambda_{\max}\{A\} (\|x_k\| + \|x^*\|) \tau, \quad \forall x_k \in \mathcal{D} \text{ and } \forall [x_k]_\tau \in \mathcal{D}_\tau.
\end{aligned} \tag{53}$$

From (48), (53), and (41e)-(41h), it is concluded that $\|\mathcal{P}^{\frac{1}{2}}(W^\top g + \epsilon - x^*)\|^2 - \Gamma_{C_L} \leq 0, \forall x_k \in \mathcal{D}$. ■

Remark 6: Theorem 1 proves that enforcing a modified stability and Lyapunov constraints for a finite number of sampled points is sufficient to obtain an optimal solution of the QCQP defined in (11).

A relaxation variable ϑ can also be introduced in the modified Lyapunov constraint given in (40) to maintain the feasibility of the optimization problem by softening the Lyapunov constraint. The modified probabilistic discrete-time control barrier and Lyapunov functions-based QCQP (PDCBFLF-QCQP) is given by:

PDCBFLF-QCQP

$$W^*, \vartheta^* = \arg \min_{W, \vartheta} J(W) + \mathfrak{c}\vartheta^2 \tag{54a}$$

$$\begin{aligned}
\text{s.t. } & \|(W^\top [g]_\tau - \bar{x})\|_{\mathcal{A}}^2 - [\Gamma_{C_B}]_\tau \\
& \leq -\tau (\chi_{C_B} + \Upsilon_{C_B} + \mathfrak{I}_{C_B}) - \Theta_{C_B} \bar{\epsilon}
\end{aligned} \tag{54b}$$

$$\begin{aligned}
& \|(W^\top [g]_\tau - x^*)\|_{\mathcal{P}}^2 - [\Gamma_{C_L}]_\tau \\
& \leq -\tau (\chi_{C_L} + \Upsilon_{C_L} + \mathfrak{I}_{C_L}) - \Theta_{C_L} \bar{\epsilon} + \vartheta
\end{aligned} \tag{54c}$$

where \mathfrak{c} is a positive constant and $J(W)$ is given in (11).

Remark 7: In practice, the unconstrained learning for ELM can be used to obtain the Frobenius norm of the output layer weight matrix W stated in Assumption 2 to be included as a constraint for the optimization problem in (54a). Moreover, the unconstrained learning can also be used to obtain $\bar{\epsilon}$ and $\bar{\epsilon}'$ used in constraints (54b)–(54c).

V. NUMERICAL EVALUATIONS

In this section three numerical examples to evaluate the proposed method are presented. For each example, a set of point-to-point demonstrations of a motion governed by the controlled difference equation in (1) is considered. Each demonstration consisting of the state trajectories is assumed to be the solution of the underlying discrete-time dynamical system. The objective is to learn the parameters of the ELM network with theoretical guarantees such that the solutions of

the learned dynamics remain inside a prescribed safe region while converging to the desired goal location. An ellipse, which is a commonly used approximation for representing the region of operation, is selected such that it encloses all the demonstrations. Once the model is learned, the parameters are used to forward propagate the dynamics from various initial conditions and to validate that the trajectories remain inside the ellipse at all times without crossing the boundaries and converge to the desired goal location. The optimization problem is solved using the CVX package in MATLAB 2020b. Random initialization process in the ELM algorithm may lead to having saturated or constant neurons, which are not desired while learning the model [38]. To circumvent this problem, a BIP learning rule is used to estimate the slopes and biases of the ELM [38]. The active sampling strategy introduced in [14] is also used to select informative points for the constraint computations of the optimization problem in (54a)–(54c). For all the three examples, the risk tolerance of $p_k = 0.8$, for v_k the covariance of $\Sigma = 0.02I$, and regularization parameter $\mu_W = 0.01$ are chosen. The subsequent results are generated using an ELM network with hidden neurons $n_h = 25$. A 100 Monte Carlo simulation runs are conducted to test the model on bound violations during the trajectory reproduction from various initial conditions.

To motivate our first example, consider a scenario in a manufacturing industry where humans and robots must work collaboratively and close to one another. In such settings, robots must maintain their joint movements bounded when moving in a confined space to ensure the safety of humans and the manufacturing operation is carried out effectively. Robot manipulators are modeled using Euler-Lagrange (EL) dynamics. To demonstrate the applicability of our proposed method to such cases, we consider, for simplicity, a two-link robot planar manipulator with lengths $L_1 = 1$ [m] and $L_2 = 1$ [m] and masses $m_1 = 1$ [kg] and $m_2 = 1$ [kg], whose dynamics are described by the EL equation: $M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = u$, where $M(q) \in \mathbb{R}^{2 \times 2}$ denotes the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{2 \times 2}$ denotes the centripetal-Coriolis matrix, $G(q) \in \mathbb{R}^2$ denotes the gravity vector, $u \in \mathbb{R}^2$ represents the control input vector, and $q, \dot{q} \in \mathbb{R}^2$ denote the joint angles and angular velocity, respectively. A nonlinear black-box model of the EL-dynamics in the following form $q_{k+1} = f(q_k, u_k)$ is considered.

To generate the state trajectories data q_k for training the ELM model, a PID set-point controller is designed to regulate the joint positions to a desired value $q = [\pi, -\frac{\pi}{4}]^T$. Through empirical investigations, a set of seven trajectories with randomly selected initial conditions are collected and their PID gains are tuned individually. The joint position satisfies the following bounds: $1.10 \leq q(1) \leq 3.49$ and $-1.90 \leq q(2) \leq 0.31$, where $q(1)$ and $q(2)$ denote the joint's first and second dimension. To avoid over sampling for the constraint computations, an invariant region in the form of an ellipse inscribed within the joint positions bounds is selected, i.e., $h(q_k) = 1 - (q_k - \bar{q})^\top A(q_k - \bar{q})$, where $\bar{q} \in \mathbb{R}^2$ denotes the

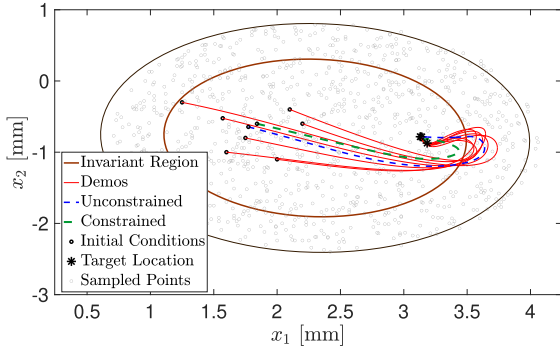


FIGURE 2. Illustration of the ELM model learning using probabilistic safety and stability constraints on joint positions data of a 2 DoF planar robot.

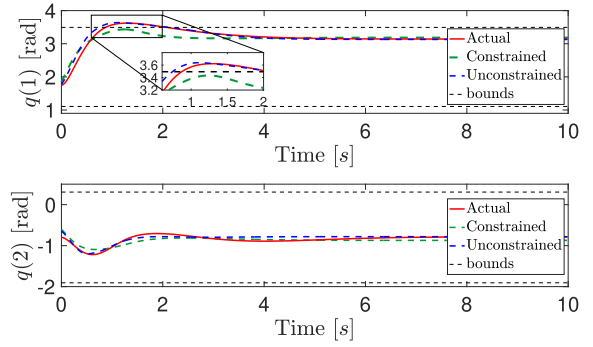


FIGURE 3. Evolution of the joint angles for the planar robot simulation using the learned ELM parameters.

hyperellipsoid's center of gravity,

$$A = \begin{bmatrix} \frac{\cos^2 \alpha}{\iota_1^2} + \frac{\sin^2 \alpha}{\iota_2^2} & \cos \alpha \sin \alpha \left(\frac{1}{\iota_1^2} - \frac{1}{\iota_2^2} \right) \\ \cos \alpha \sin \alpha \left(\frac{1}{\iota_1^2} - \frac{1}{\iota_2^2} \right) & \frac{\sin^2 \alpha}{\iota_1^2} + \frac{\cos^2 \alpha}{\iota_2^2} \end{bmatrix}, \quad (55)$$

ι_1 and ι_2 are the major and minor axes of the ellipse, respectively and α is the orientation of the ellipse such that it encloses the demonstrations data. Other parameters $\gamma = 0.8$, $\rho = 0.9$, $\zeta = 0.8$, and $\delta = 0.9$ are selected empirically. In Fig. 2 the results of the ELM parameter learning are shown when the system parameters are subject to probabilistic safety and stability constraints given in (54b) and (54c). Using the DT-ZCBF implies asymptotic stability of set \mathcal{S} , shown as an invariant region in Fig. 2. Hence, if any bounded disturbances push the state outside the invariant region, the set \mathcal{S} is asymptotically reached. For this reason, when training the system model, the points are sampled from a more extensive set than the selected ellipse [19]. The robot joint position trajectory is produced at a different initial condition selected at random from a uniform distribution between the initial condition used during training and a point that is 0.5 mm distant apart. It can be seen from Fig. 2, the reproduced trajectory remains inside the desired invariant region for all time and converges to the desired set-point. Same results are observed for all the Monte Carlo runs. Fig. 3 shows the evolution of the joint angles using the learned ELM parameters plotted as function of time. The reproduced joint position trajectories never cross the set boundaries in either direction and asymptotically converge to a point very close to the desired position.

Advanced driver-assistance systems (ADAS) are technologies that assist drivers by providing sensory information as warnings or reducing their control efforts. Most modern-day vehicles are equipped with ADAS, an example of safety-critical control applications. The architecture of autonomous driving technology, which can also be viewed as ADAS but with lesser driver engagements, consists of sensing, perception, planning, and operation [39]. Motion planning that computes moving paths of a vehicle while operating within a

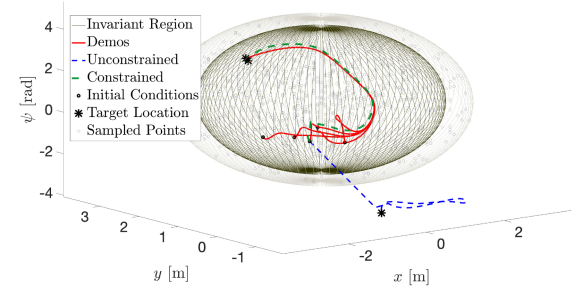


FIGURE 4. Trajectories of the wheeled mobile robot learned using a constrained learning method subject to probabilistic safety and stability constraints. The boundary of the safe set is shown in a dark gray ellipsoid, and random samples are selected from a more extensive set shown in the light gray ellipsoid.

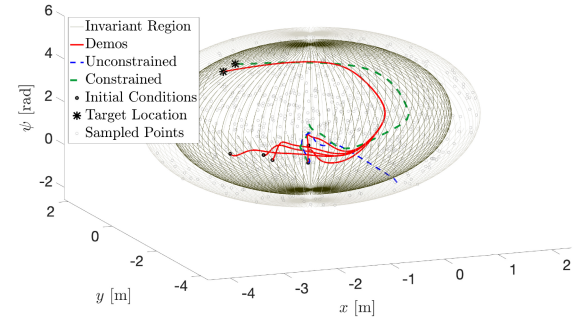


FIGURE 5. Illustration of the model's ability to the change of goal location when it is learned using the proposed method. The boundary of the safe set is shown in a dark gray ellipsoid.

confined region is an example of where the proposed method can be utilized. Consider the following dynamic model of the unicycle-like mobile robot.

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} v \cos(\psi) - a\omega \sin(\psi) \\ v \sin(\psi) + a\omega \cos(\psi) \\ \omega \end{bmatrix}, \quad (56)$$

where p_x and p_y denote the 2D position, ψ is the orientation, and v , ω denote the linear and angular velocities

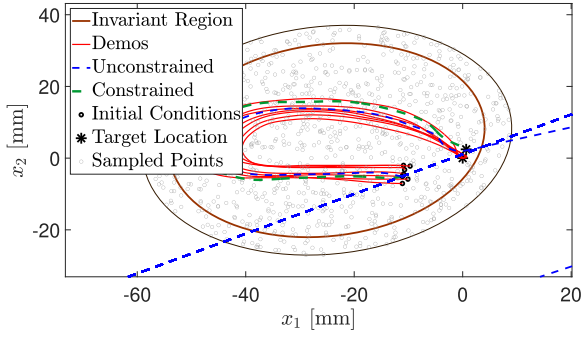


FIGURE 6. Model *BendedLine*, learned with probabilistic safety and stability constraints for ellipse sets.

of the robot, respectively. Moreover, a represents the center of the wheel to the position of interest (p_x, p_y) , which for this example a constant $a = 0.12$ [m] is chosen. This model can be written as a nonlinear system given in (1) after discretization, where $x_k = [p_x(k), p_y(k), \psi(k)]^T \in \mathbb{R}^3$ is the state vector, and $u_k \in \mathbb{R}^2$ is the system control input. For the system in (56), an adaptive controller similar to [40] is first designed that tracks a desired trajectory. Through empirical investigation, five trajectories with randomly selected initial conditions are collected to be used as a demonstration data for training the ELM network. The bounds on the states of (56) are selected as $p_x \in (-2, 1.07)$ [m], $p_y \in (-0.59, 1.61)$ [m], and $\psi = (-0.23\pi, 1.11\pi)$ [rad]. The invariant region is given by $h(x_k) = 1 - (x_k - \bar{x})^T A_{\text{wmr}}(x_k - \bar{x})$, where $\bar{x} \in \mathbb{R}^3$ denotes the ellipsoid's center of gravity, and $A_{\text{wmr}} = \text{diag}(\varrho_1^{-2}, \varrho_2^{-2}, \varrho_3^{-2})$, where ϱ_1, ϱ_2 , and ϱ_3 are the semi-axes of the ellipsoid whose values are chosen such the ellipsoid encloses the demonstration data. Along with the states x_k two additional inputs in the form of the error, i.e., $e_x = p_x - p_x^*$ and $e_y = p_y - p_y^*$ are provided as inputs to the ELM network, where p_x^* and p_y^* are the target locations that can be obtained from the demonstration data. The design parameters for this simulation are selected empirically as $\gamma = 0.95$ and $\rho = 0.90$, $\zeta = 0.5$, $\delta = 1$. Fig. 4 shows the significance of constrained model learning methods. As shown in Fig. 4, the green trajectory is generated using a model that is learned subject to probabilistic safety and stability constraints, whereas the blue trajectory that goes outside the state constraints is a result of the model that is learned without any constraints. Moreover, the robustness of the described method to the change of location is tested, and it is shown in Fig. 5. The change of location is determined according to the guidelines presented in [41]. The constrained trajectories are bound to stay inside a user defined safe region while reaching the target.

The purpose of imitation learning is to efficiently learn motion dynamics by showing an example. However, its application is not limited to physical systems. The LASA human handwriting dataset [41] consists of 7 demonstrations where the handwriting motions were collected from a pen input using a Tablet PC. For the third example, the proposed method is

tested on the *BendedLine* shape from dataset. An ellipse is chosen for the invariant region that encloses the BendedLine shape's seven demonstrations. The state trajectories in all 30 shapes in the LASA dataset converge to an origin equilibrium, i.e., $x^* = 0$, causing redundancy in the input vector to the ELM network, that is $s_k = [x_k^T, x_k^T, 1]^T$. Thus, to circumvent the data redundancy issue for this particular dataset and be able to compute the constraints derived in (54b) and (54c), the state trajectory for the selected shape is translated by a constant before training. Once the model is learned, the state trajectories are translated back to zero. The optimization design parameters for this simulation are selected empirically as follows: $\gamma = 0.85$ and $\rho = 0.75$, $\zeta = 0.1$, $\delta = 1$. Fig. 6 shows that the trajectories generated by the learned model remain inside a safe set, and converge to the system equilibrium. Since the constraints are derived for one-step-ahead prediction, no barrier violation occurs during the Monte Carlo runs if the initial condition is sufficiently close to the initial condition of the demonstration trajectories. The average CPU time for training the ELM network for all the simulations is computed over five independent runs with a 1000 number of sampling points is 3.4 seconds.

VI. DISCUSSION AND FUTURE WORK

This paper presents a dynamical system model learning method while providing theoretical guarantees on satisfying the safety and stability properties of the underlying nonlinear discrete-time dynamical system. The first two simulations are examples of physical systems where motion dynamics are modeled using an ELM network whose parameters are learned such that the state remains inside a predefined safe region of operation while converging to the desired goal location. The third simulation shows another example of using the imitation learning method to learn hand-drawn shapes collected from a pen input using a Tablet PC while providing theoretical guarantees on convergence to the goal location and staying within a predefined safe operating region. The scope of the work is that of learning the dynamic model to represent the trajectories of a controlled dynamical system, e.g., the ones considered in the simulation examples. The constrained learned model can be used in many ways, for instance, to produce reference trajectories for an autonomous systems from different initial states to the final states to be followed by designing a controller or to forecast trajectories of a human motion similar to that of the third simulation example.

While designing identification algorithms for nonlinear discrete-time systems from data is fundamental and has significant use in many applications, studying the sample complexity of the proposed learning scheme is of the same importance. The sample complexity of the proposed learning scheme can be analyzed via the following two avenues of research which will be pursued as a future work.

- 1) Investigations on obtaining an optimal number of demonstrations required to learn the state trajectories of constrained controlled dynamical systems. The sample-complexity properties of incremental gain

stability-constrained iterative learning algorithms is analyzed in [42] and some insights are provided into the relationship between system stability and sample-complexity. An algorithm is introduced in [43] that learns the parameters of a stable nonlinear dynamical system from a single trajectory with optimal sample complexity (offset Rademacher complexity) and nearly-linear run time.

- 2) While the method presented in this paper uses the discretization resolution τ to make the semi-infinite program tractable, authors in [7] uses the notion of $(\alpha\text{-}\beta)$ -grid to obtain modified constraints with an upper bound on the sample-point density in the region of attraction. Inspired by the method in [7] a study will be conducted to determine the minimum number of samples required to compute CBF and CLF so that the safety and stability of the underlying systems are satisfied point-wise in a closed set. In a related work [44], algorithms are proposed to learn how much data is required so that the certificate function (the Lyapunov or contraction metric) is valid at a prescribed fraction of the relevant state space.

VII. CONCLUSION

A discrete-time nonlinear system model learning method is presented which uses the ELM with probabilistic DT-ZCBF and DT-CLF constraints. The system process noise is modeled as a normally distributed Gaussian noise, which induces a distribution over the safety and stability constraints. Therefore, the probabilistic form of the ZBF and Lyapunov constraints are derived such that safety and stability are guaranteed within user-defined risk tolerance. The constrained ELM learning problem is formulated as a QCQP. A theorem is developed to make the QCQP implementable, which proves that with a modification to the constraints, the quadratic program with an infinite number of constraints can be relaxed to a finite number of constraints. The applicability of the proposed model learning method is demonstrated using three robotics engineering examples. The safety set is modeled as an hyperellipsoid in this work, which is a common representation for a region of operation or an obstacle. In future the safe set representation will be generalized to other sets by modifying the constraints.

REFERENCES

- [1] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data by sparse identification of nonlinear dynamical systems," *Proc. Nat. Acad. Sci.*, vol. 113, no. 15, pp. 3932–3937, 2016.
- [2] A. Chiuso and G. Pillonetto, "System identification: A machine learning perspective," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 2, pp. 281–304, 2019.
- [3] M. Sznajder, "Control oriented learning in the era of Big Data," *IEEE Control Syst. Lett.*, vol. 5, no. 6, pp. 1855–1867, Dec. 2021.
- [4] T. Osa et al., "An algorithmic perspective on imitation learning," *Found. Trends Robot.*, vol. 7, no. 1-2, pp. 1–179, 2018.
- [5] H. Yin, P. Seiler, M. Jin, and M. Arcak, "Imitation learning with stability and safety guarantees," *IEEE Control Syst. Lett.*, vol. 6, pp. 409–414, 2022.
- [6] J. M. Maciejowski, "Guaranteed stability with subspace methods," *Syst. Control Lett.*, vol. 26, no. 2, pp. 153–156, 1995.
- [7] M. Khosravi and R. S. Smith, "Nonlinear system identification with prior knowledge on the region of attraction," *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 1091–1096, Jul. 2021.
- [8] S. M. Khansari-Zadeh and A. Billard, "Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions," *Robot. Auton. Syst.*, vol. 62, no. 6, pp. 752–765, 2014.
- [9] H. Ravichandar, I. Salehi, and A. P. Dani, "Learning partially contracting dynamical systems from demonstrations," in *Conf. Robot Learn.*, 2017, pp. 369–378.
- [10] I. Salehi, G. Yao, and A. P. Dani, "Active sampling based safe identification of dynamical systems using extreme learning machines and barrier certificates," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 22–28.
- [11] J. Umlauf and S. Hirche, "Learning stochastically stable Gaussian process state-space models," *IFAC J. Syst. Control*, vol. 12, 2020, Art. no. 100079.
- [12] S. Singh, S. M. Richards, V. Sindhwan, J.-J. E. Slotine, and M. Pavone, "Learning stabilizable nonlinear dynamics with contraction-based regularization," *Int. J. Robot. Res.*, vol. 40, no. 10-11, pp. 1123–1150, 2021.
- [13] H. Ravichandar and A. Dani, "Learning position and orientation dynamics from demonstrations via contraction analysis," *Auton. Robots*, vol. 43, no. 4, pp. 897–912, 2019.
- [14] I. Salehi, G. Rotithor, G. Yao, and A. P. Dani, "Dynamical system learning using extreme learning machines with safety and stability guarantees," *Int. J. Adaptive Control Signal Process.*, vol. 35, no. 6, pp. 894–914, 2021.
- [15] G.-B. Huang et al., "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
- [16] F. Berkenkamp, R. Moriconi, A. P. Schoellig, and A. Krause, "Safe learning of regions of attraction for uncertain, nonlinear systems with Gaussian processes," in *Proc. IEEE Conf. Decis. Control*, 2016, pp. 4661–4666.
- [17] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.
- [18] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using control Lyapunov-barrier function," *Automatica*, vol. 66, pp. 39–47, 2016.
- [19] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54–61, 2015.
- [20] S.-C. Hsu, X. Xu, and A. D. Ames, "Control barrier function based quadratic programs with application to bipedal robotic walking," in *Proc. Amer. Control Conf.*, 2015, pp. 4542–4548.
- [21] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. 18th Eur. Control Conf.*, 2019, pp. 3420–3431.
- [22] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *Proc. Amer. Control Conf.*, 2016, pp. 322–328.
- [23] W. Xiao and C. Belta, "Control barrier functions for systems with high relative degree," in *Proc. IEEE Conf. Decis. Control*, 2019, pp. 474–479.
- [24] A. J. Taylor and A. D. Ames, "Adaptive safety with control barrier functions," in *Proc. Amer. Control Conf.*, 2020, pp. 1399–1405.
- [25] B. T. Lopez, J. J. E. Slotine, and J. P. How, "Robust adaptive control barrier functions: An adaptive and data-driven approach to safety," *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 1031–1036, Jul. 2021.
- [26] M.-A. Beaudoin and B. Boulet, "Structured learning of safety guarantees for the control of uncertain dynamical systems," *IEEE Trans. Intell. Veh.*, early access, Feb. 4, 2022, doi: [10.1109/ITV.2022.3148212](https://doi.org/10.1109/ITV.2022.3148212).
- [27] K. Long, C. Qian, J. Cortés, and N. Atanasov, "Learning barrier functions with memory for robust safe navigation," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 4931–4938, Jul. 2021.
- [28] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 2460–2465.
- [29] E. B. Kosmatopoulos, M. M. Polycarpou, M. A. Christodoulou, and P. A. Ioannou, "High-order neural network structures for identification of dynamical systems," *IEEE Trans. Neural Netw.*, vol. 6, no. 2, pp. 422–431, Mar. 1995.
- [30] M. J. Khojasteh, V. Dhiman, M. Franceschetti, and N. Atanasov, "Probabilistic safety constraints for learned high relative degree system dynamics," in *Proc. Learn. Dyn. Control*, 2020, pp. 781–792.

- [31] I. Salehi, T. Taplin, and A. P. Dani, "Chance-constrained system identification of nonlinear discrete systems with safety and stability guarantees," in *Proc. Amer. Control Conf.*, 2022, pp. 1497–1502.
- [32] A. Agrawal and K. Sreenath, "Discrete control barrier functions for safety-critical control of discrete systems with application to bipedal robot navigation," in *Proc. Robot.: Sci. Syst.*, 2017, vol. 13.
- [33] M. Srinivasan, M. Abate, G. Nilsson, and S. Coogan, "Extent-compatible control barrier functions," *Syst. Control Lett.*, vol. 150, 2021, Art. no. 104895.
- [34] M. Ahmadi, A. Singletary, J. W. Burdick, and A. D. Ames, "Safe policy synthesis in multi-agent POMDPs via discrete-time barrier functions," in *Proc. IEEE Conf. Decis. Control*, 2019, pp. 4797–4803.
- [35] F. L. Lewis, J. Campos, and R. Selmic, *Neuro-Fuzzy Control of Industrial Systems With Actuator Nonlinearities*. Philadelphia, PA, USA: Soc. Ind. Appl. Math., 2002.
- [36] G. Huang, G.-B. Huang, S. Song, and K. You, "Trends in extreme learning machines: A review," *Neural Netw.*, vol. 61, pp. 32–48, 2015.
- [37] K. Hornik, et al., "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.
- [38] K. Neumann and J. J. Steil, "Optimizing extreme learning machines via ridge regression and batch intrinsic plasticity," *Neurocomputing*, vol. 102, pp. 23–30, 2013.
- [39] R. Okuda, Y. Kajiwara, and K. Terashima, "A survey of technical trend of ADAS and autonomous driving," in *Proc. Tech. Papers Int. Symp. VLSI Des., Automat. Test*, 2014, pp. 1–4.
- [40] F. N. Martins, W. C. Celeste, R. Carelli, M. Sarcinelli-Filho, and T. F. Bastos-Filho, "An adaptive dynamic controller for autonomous mobile robot trajectory tracking," *Control Eng. Pract.*, vol. 16, no. 11, pp. 1354–1363, 2008.
- [41] A. Lemme, Y. Meirovitch, S. M. Khansari-Zadeh, T. Flash, A. Billard, and J. J. Steil, "Open-source benchmarking for learned reaching motion generation in robotics," *Paladyn, J. Behav. Robot.*, vol. 6, pp. 30–41, 2015.
- [42] S. Tu, A. Robey, T. Zhang, and N. Matni, "On the sample complexity of stability constrained imitation learning," in *Proc. Learn. Dyn. Control Conf.*, 2022, pp. 180–191.
- [43] D. Foster, T. Sarkar, and A. Rakhlin, "Learning nonlinear dynamical systems from a single trajectory," in *Proc. Learn. Dyn. Control*, 2020, pp. 851–861.
- [44] N. Boffi, S. Tu, N. Matni, J.-J. Slotine, and V. Sindhvani, "Learning stability certificates from data," in *Proc. Conf. Robot Learn.*, 2021, pp. 1341–1350.



2020 IFAC Applications Paper Award Finalist and a recipient of the 2020 IEEE L-CSS Outstanding Reviewer Award.

IMAN SALEHI (Member, IEEE) received the M.E. degree from the Department of Electrical and Computer Engineering, University of Hartford, West Hartford, CT, USA, in 2015. He was a full-time lecturer for three semesters at the University of Hartford. He is currently working toward the Ph.D. degree with the Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT, USA. His research interests include learning for control, human–robot interaction, and system identification. He was a co-recipient of the



TYLER TAPLIN (Graduate Student Member, IEEE) received the B.S. degree in physics and mathematics from the Rose-Hulman Institute of Technology, Terre Haute, IN, USA. He is currently working toward the Ph.D. degree with the Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT, USA.



ASHWIN P. DANI (Senior Member, IEEE) received the Ph.D. degree from the University of Florida, Gainesville, FL, USA, in 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT, USA. He was a Postdoctoral Research Associate with the University of Illinois at Urbana-Champaign, Urbana, IL, USA. His research interests include estimation and control, learning for control, vision-based estimation and control, and human–robot collaboration.

He is an Associate Editor for IEEE TRANSACTIONS ON MECHATRONICS and a Member of the Conference Editorial Board for IEEE Control System Society. He was a co-recipient of the 2020 IFAC Applications Paper Award Finalist, 2016 FUSION Student Paper Award - 1st runner up, 2015 ASME Dynamics, Systems and Controls Conference Robotics Paper Award, and 2012 Technology Innovator Award from UF.