Providing Safety Assurances for Systems with Unknown Dynamics

Hao Wang[1], Javier Borquez[1] and Somil Bansal[1]

***Abstract*—As autonomous systems become more complex and integral in our society, the need to accurately model and safely control these systems has increased significantly. In the past decade, there has been tremendous success in using deep learning techniques to model and control systems that are difficult to model using first principles. However, providing safety assurances for such systems remains difficult, partially due to the uncertainty in the learned model. In this work, we aim to provide safety assurances for systems whose dynamics are not readily derived from first principles and, hence, are more advantageous to be learned using deep learning techniques. Given the system of interest and safety constraints, we learn an ensemble model of the system dynamics from data. Leveraging ensemble uncertainty as a measure of uncertainty in the learned dynamics model, we compute a maximal robust control invariant set, starting from which the system is guaranteed to satisfy the safety constraints under the condition that realized model uncertainties are contained in the predefined set of admissible model uncertainty. We demonstrate the effectiveness of our method using a simulated case study with an inverted pendulum and a hardware experiment with a TurtleBot. The experiments show that our method robustifies the control actions of the system against model uncertainty and generates safe behaviors without being overly restrictive.**

I. Introduction

Autonomous systems are playing increasingly important roles in the functioning of modern society. However, traditional modeling techniques, such as using first principles, struggle to model these systems, given their increasing complexities. Recent advances in deep learning have enabled the modeling and control of systems with highly complex dynamics. While the methods have demonstrated strong control performance for a number of autonomous systems, they can lead to unsafe behaviors or even catastrophic failures due to the predictive uncertainty in the neural network models.

In this work, we are interested in providing safety assurances for systems whose dynamics are unknown and difficult to model using first principles. Despite success in safety analysis for models developed from first principles, it remains difficult to provide safety assurances for systems with unknown or uncertain dynamics. Many works have utilized safety analysis frameworks, such as Control Barrier Function (CBF) [2] and Hamilton-Jacobi (HJ) reachability analysis [5], to provide safety assurances to systems with unknown or uncertain dynamics. One popular line of works seeks to reduce model uncertainties to uncertainties in the CBF constraints, which are used to synthesize safety-critical controls [8, 20, 19]. However, these methods rely on prior knowledge of the system to construct the CBFs, and more critically assume the CBFs constructed for the nominal model is valid for the actual system. This assumption can be easily violated when the model uncertainty is severe, and it is difficult to determine when the assumption is no longer valid. Another family of approaches is to generate safety guarantees for systems under the *worst-case* model uncertainty using HJ reachability analysis. This line of work is rooted in differential game theory and its connection with the Hamilton-Jacobi-Isaacs partial differential equation (HJI PDE) [6, 10]. A time-dependent HJI PDE [18] is formulated to study the pursuer-evader game, a type of two-person zero-sum differential game integral to HJ reachability analysis, and this formulation has been applied to generate safety assurances for system under dynamics uncertainties. More specifically, the authors in [11, 14, 1] represent the model uncertainty as a Gaussian Process (GP) and utilize the predictive uncertainty of the GP to generate robust safety assurances online. However, the model uncertainties considered in the works depend only on the state and not the control input. Moreover, it is not immediately clear how similar techniques can be used for neural network-based dynamics models that does not predict uncertainties.

In its core, our method attempts to provide robust safety assurances for systems with unknown dynamics against model uncertainties. We first learn a nominal dynamics model of the system along with a measure of model uncertainty. Then, we compute robust safety assurances utilizing results from HJ reachability analysis. More specifically, given a set of undesirable states, our method computes the maximal robust control invariant set and a controller that renders the set control invariant for the model under bounded model uncertainty. A critical aspect of our work is that model uncertainty is both *state and control-dependent*. This control dependence of model uncertainty is often ignored in prior works for simplicity. Specifically, when the model uncertainty is control-independent, the players in the two-person game do not interact directly and renders the game much easier to solve. Instead, we take this interplay between the control and model uncertainty into account and systematically handle that under the HJ reachability framework. This leads to a significantly less conservative estimate of the safe set of the system. Furthermore, we pay special attention to learned neural network nominal models. Using an ensemble of neural networks with

[1]The authors are associated with the Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California. {haowwang, javierbo, somilban}@usc.edu

*This work is supported in part by the NSF CAREER Program under award 2240163, the DARPA ANSR program, a Ming Hsieh Department of Electrical and Computer Engineering Fellowship.

control-affine architectures as the nominal model, we harness the modeling power of neural networks and obtain a heuristic measure of model uncertainty from the ensemble, with which we provide robust safety assurances for the system.

Our method is shown to provide significant benefits as it explicitly incorporates state and control-dependent model uncertainty in safety analysis and, as a result, provides more robust, but not overly conservative controllers for the system. To summarize, the key contributions of this work are two-fold: 1) We propose a framework for providing robust safety assurances to systems with unknown dynamics by incorporating state and control-dependent model uncertainty in HJ reachability analysis. We also provide a closed-form solution to the two-player zero-sum differential game between the control and disturbance player, in order to accommodate the state and control-dependent nature of the model uncertainty, and 2) We provide a concrete instantiation of our framework by modeling the system with an ensemble of neural networks and solving a robust optimal control problem to safeguard the learned model against bounded model uncertainty.

II. Problem Formulation

In this work, we are interested in providing safety assurances for systems with unknown but deterministic *control-affine* dynamics, as many practical systems are control-affine [16], governed by ordinary differential equation

$$\frac{dx}{dt} = \dot{x} = f(x,u) = f_1(x) + f_2(x)u \tag{1}$$

where $x \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$ and $u \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ are the state and control of the system, and $f_1(x)$ and $f_2(x)$ are matrices of appropriate sizes. Given a set of states $\mathcal{F} \subset \mathcal{X}$, which we refer to as the *failure set*, that the system must avoid (e.g., obstacles for a mobile robot), our goal is ensuring the system f does not enter the failure set $\mathcal{F}$ for predefined time horizon $[0,T]$. More formally as stated in Prob. 1.

Problem 1 (Safety Problem for System f)**.** *Given a failure set $\mathcal{F}$ and time horizon $[0,T]$, obtain a safe set $\mathcal{S} \subseteq \mathcal{X}$, along with a state-feedback controller π, such that the system f starting within $\mathcal{S}$ implies that f does not enter $\mathcal{F}$ for $[0,T]$ using controls provided by π. A safe set is the maximal safe set, denoted by $\mathcal{S}^*$, if it contains all other safe sets as subsets.*

Because we do not have access to the system dynamics f, we further limit the scope of the problem to employing a model-based approach - first model the system of interest, then provide safety assurances on the system against any *bounded* model uncertainty $d \in \mathcal{D}$, where $\mathcal{D}$ is the set of admissible model uncertainty. We state the proposed safety assurance problem as follows:

Problem 2 (Safety Assurance under Model Uncertainty)**.** *Given a failure set $\mathcal{F}$ and time horizon $[0,T]$, obtain a set $\mathcal{R} \subseteq \mathcal{X}$, along with a state-feedback controller $\pi_{\mathcal{R}}$, such that under the condition that all realized model uncertainties are contained within $\mathcal{D}$, system f starts within $\mathcal{R}$ implies that f does not enter $\mathcal{F}$ for $[0,T]$ using controls from $\pi_{\mathcal{R}}$, or equivalently, $\mathcal{R} \subseteq \mathcal{S}^*$.*

Definition 1. *(Robust Safe Set) A solution to Problem 2, $\mathcal{R} \subseteq \mathcal{X}$, is referred to as a robust safe set. A robust safe set is the maximal robust safe set, denoted as $\mathcal{R}^*$, is the robust safe set that contains any robust safe set as subsets.*

III. Background

In this section, we provide a brief overview of Hamilton-Jacobi (HJ) reachability analysis, an approach that can help solve Prob. 2. Let g be a system described by dynamics $\dot{x} = g(x,u,d)$, where $x \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$, $u \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$, and $d \in \mathcal{D} \subseteq \mathbb{R}^{n_d}$ are the state, control, and disturbance of the system. Disturbance d, in the interest of this work, describes the model uncertainty. We use $\mathbf{u} : [0,T] \to \mathcal{U}$ and $\mathbf{d} : [0,T] \to \mathbb{R}^{n_d}$ to denote the control and disturbance signals. Furthermore, we denote the state trajectory starting from state x at time t evolved with control and disturbance signals $\mathbf{u}(\cdot)$ and $\mathbf{d}(\cdot)$ as $\xi_{x,t}^{\mathbf{u},\mathbf{d}}$. With a slight abuse of the notation, we use $\xi_{x,t}^{\mathbf{u},\mathbf{d}}(\tau)$ to denote the state at time $\tau \geq t$ along the trajectory $\xi_{x,t}^{\mathbf{u},\mathbf{d}}$.

Suppose $g(x,u,d)$ is uniformly continuous, bounded, and Lipschitz continuous in x for fixed u and d. We further assume that $\mathcal{U}$ and $\mathcal{D}$ are compact, and $\mathbf{u}(\cdot)$ as well as $\mathbf{d}(\cdot)$ are measurable. Let $l(x)$ be the signed distance function to $\mathcal{F}$. We can obtain $\mathcal{R}^*$ by solving the following robust optimal control problem (Prob. 3) with initial condition $x = x_0 \; \forall x_0 \in \mathcal{X}$ and $t = 0$. In robust control literature, Prob. 3 is posed as a two-player zero-sum differential game between the control u and disturbance d, who uses only nonanticipative strategies [18]. Let $\Gamma(t)$ be the set of nonanticipative strategies, and $\mathbb{U}(t)$ be the set of admissible control signals.

Problem 3 (Robust Safety Optimal Control Problem)**.**

$$\begin{aligned} \inf_{\mathbf{d}(\cdot)\in\Gamma(t)} \sup_{\mathbf{u}(\cdot)\in\mathbb{U}(t)} \quad & J(x,t,\mathbf{u},\mathbf{d}) = \min_{\tau\in[t,T]} l(\xi_{x,t}^{\mathbf{u},\mathbf{d}}(\tau)) \\ s.t. \quad & \dot{x} = g(x,u,d) \end{aligned}$$

Let us define the value function $V(x,t)$ to take on the optimal value of Prob. 3 at state x and time t:

$$\begin{aligned} V(x,t) &= \inf_{\mathbf{d}(\cdot)\in\Gamma(t)} \sup_{\mathbf{u}(\cdot)\in\mathbb{U}(t)} J(x,t,\mathbf{u},\mathbf{d}) \\ &= \inf_{\mathbf{d}(\cdot)\in\Gamma(t)} \sup_{\mathbf{u}(\cdot)\in\mathbb{U}(t)} \min_{\tau\in[t,T]} l(\xi_{x,t}^{\mathbf{u},\mathbf{d}}(\tau)) \end{aligned} \tag{2}$$

Then, $\mathcal{R}^*$ can be characterized using $V(x,t)$:

$$\mathcal{R}^* = \{x \in \mathcal{X} | V(x,0) > 0\} \tag{3}$$

HJ reachability analysis provides a tractable means to compute the value function $V(x,t)$. It has been shown that $V(x,t)$ is the unique viscosity solution of the Hamilton-Jacobi-Isaacs Variational Inequality (HJI-VI) [17, 18]:

$$\begin{aligned} \min\{D_t V + H(x,t,\nabla V), l(x) - V(x,t)\} = 0 \\ V(x,T) = l(x), \quad \text{for } t \in [0,T] \end{aligned} \tag{4}$$

$H(x,t,\nabla V) = \max_u \min_d \langle \nabla V, g(x,u,d) \rangle$ is the Hamiltonian. $D_t V$ and ∇V denote the temporal derivative and the

spatial gradients of $V(x,t)$. It is important to note that HJ reachability analysis also provides a state-feedback controller $\pi_{\mathcal{R}^*}$ that renders $\mathcal{R}^*$ control-invariant:

$$\pi_{\mathcal{R}^*}(x,t) = \arg\max_{u\in\mathcal{U}} \min_{d\in\mathcal{D}} \langle \nabla V(x,t), g(x,u,d)\rangle \tag{5}$$

IV. Safety Assurances for Learned Dynamics

At the heart of our framework is solving a robust optimal control problem (Prob. 3) to provide safety assurances for system f, by incorporating the worst-case model uncertainty in the safety analysis. We obtain the maximal robust safe set $\mathcal{R}^*$ that safeguards the system against model uncertainty by solving the HJI-VI (4) for $V(x,t)$. In this section, we first introduce the notion of *uncertain model* and use it to set up the robust optimal control problem. Then, we show $\mathcal{R}^*$ does in fact confer safety assurance to f. Finally, we present a concrete instantiation of our framework with f modeled by an ensemble of neural networks, and we discuss a method to quantify the model uncertainty from the ensemble.

A. Model Representation and Hamiltonian Formulation

Given a deterministic, continuous-time, control-affine system $f(x,u)$, we learn a nominal model $\bar{f}(x,u) = \bar{f}_1(x) + \bar{f}_2(x)u$ of f. We assume there are *bounded, state-dependent* model uncertainties $d_1(x) \in \mathcal{D}_1(x) \subseteq \mathbb{R}^{n_x}$ and $d_2(x) \in \mathcal{D}_2(x) \subseteq \mathbb{R}^{n_x\times n_u}$, arising from errors of model approximation and unmodeled dynamics additive to $\bar{f}_1(x)$ and $\bar{f}_2(x)$. Hence, the *uncertain* model $\hat{f}(x,u,d_1,d_2)$ can be written as

$$\hat{f}(x,u,d_1,d_2) = \bar{f}_1(x) + d_1(x) + (\bar{f}_2(x) + d_2(x))u \tag{6}$$

We refer to $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$ as the *model uncertainty bounds* on $\bar{f}_1(x)$ and $\bar{f}_2(x)$ at state x, respectively. The Hamiltonian $H(x,t,\nabla V)$ formulated using $\hat{f}(x,u,d_1,d_2)$ is given in (7), where u^*, d_1^*, and d_2^* are the solutions to the maximin game in the RHS of (7a).

$$\begin{aligned} H(x,t,\nabla V(x,t)) &= \max_{u\in\mathcal{U}} \min_{d_1\in\mathcal{D}_1(x)} \min_{d_2\in\mathcal{D}_2(x)} \Big\langle \nabla V(x,t), \\ &\quad \bar{f}_1(x) + d_1 + \left(\bar{f}_2(x) + d_2\right) u \Big\rangle \qquad \text{(7a)} \\ &= \left\langle \nabla V(x,t), \bar{f}_1(x) + d_1^* + \left(\bar{f}_2(x) + d_2^*\right) u^* \right\rangle \qquad \text{(7b)} \end{aligned}$$

The maximin game in (7a) does not generally have a closed-form solution since u and d_2 are multiplied together. We make the following assumptions to enable tractable computation of (7a): 1) $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$ are hypercubes containing the origin in their respective spaces $\mathbb{R}^{n_x}$ and $\mathbb{R}^{n_x\times n_u}$, and 2) the set of admissible controls $\mathcal{U}$ is a hypercube containing the origin in $\mathbb{R}^{n_u}$. Under the assumptions, we can obtain closed-form solutions for u, d_1 and d_2.

Note 1. *The total model uncertainty is given by $d(x,u) = d_1(x) + d_2(x)u$, and hence the representation of d is state and control-dependent.*

Let $d_{1i}^*(x)$ and $\nabla V_i(x,t)$ denote the ith component of $d_1^*(x) \in \mathbb{R}^{n_x}$ and $\nabla V(x,t) \in \mathbb{R}^{n_x}$, respectively. Since $\mathcal{D}_1(x)$ is a hypercube containing the origin of $\mathbb{R}^{n_x}$, we can write the uncertainty bound on the ith component of $\bar{f}_1(x)$ by an interval $\left[\underline{d_{1i}(x)}, \overline{d_{1i}(x)}\right]$. Then, $d_1^*(x)$ is given by

$$d_{1i}(x) = \begin{cases} \overline{d_{1i}(x)} \text{ if } \nabla V_i(x,t) < 0 \\ \underline{d_{1i}(x)} \text{ if } \nabla V_i(x,t) \geq 0 \end{cases} \tag{8}$$

We now derive $u^*(x)$, the optimal safety controller given in (9). Let us denote the jth component of $u^* \in \mathbb{R}^{n_u}$ and the jth column of $\bar{f}_2(x)$ by u_j^* and $\bar{f}_{2j}(x)$. Since $\mathcal{U}$ is a hypercube in $\mathbb{R}^{n_u}$, the bound on the jth component of u is given by an interval $\left[\underline{u_j}, \overline{u_j}\right]$. Again, since $\mathcal{D}_2(x)$ is a hypercube in $\mathbb{R}^{n_x\times n_u}$, the model uncertainty bound on the ij^{th} component of $\bar{f}_2(x)$ is an interval $\left[\underline{d_{2ij}(x)}, \overline{d_{2ij}(x)}\right]$. Let $d_2^+(x) \in \mathbb{R}^{n_x\times n_u}$ and $d_2^-(x) \in \mathbb{R}^{n_x\times n_u}$ be the "best effort" $d_2(x)$ that intuitively try to decrease the Hamiltonian for positive or negative u, respectively. We denote the jth column of $d_2^+(x)$ and $d_2^-(x)$ by $d_{2j}^+(x)$ and $d_{2j}^-(x)$. More precisely, the ith component of $d_{2j}^+(x) \in \mathbb{R}^{n_x}$ is given by $\overline{d_{2ij}}$ when $\nabla V_i < 0$, $\underline{d_{2ij}}$ when $\nabla V_i \geq 0$. The ith component of $d_{2j}^-(x) \in \mathbb{R}^{n_x}$ is given by $\overline{d_{2ij}}$ when $\nabla V_i \geq 0$, $\underline{d_{2ij}}$ when $\nabla V_i < 0$. u^* is then given in (9).

$$u_j^*(x) = \begin{cases} \overline{u_j}, \text{if } \nabla V(x,t)^\top \left(\bar{f}_{2j}(x) + d_{2j}^+(x)\right) > 0 \\ \underline{u_j}, \text{if } \nabla V(x,t)^\top \left(\bar{f}_{2j}(x) + d_{2j}^-(x)\right) < 0 \\ 0, \text{otherwise} \end{cases} \tag{9}$$

Lastly, we provide $d_2^*(x)$. Let us denote the ij^{th} component of $d_2^*(x) \in \mathbb{R}^{n_x\times n_u}$ by $d_{2ij}^*(x)$. $d_2^*(x)$ is given below in (10).

$$d_{2ij}^*(x) = \begin{cases} \overline{d_{2ij}(x)} \text{ if } u_j^*(x)\nabla V_i(x,t) < 0 \\ \underline{d_{2ij}(x)} \text{ if } u_j^*(x)\nabla V_i(x,t) \geq 0 \end{cases} \tag{10}$$

We solve the HJI-VI (4), with the Hamiltonian (7b) formulated for $\hat{f}$, to obtain $V(x,t)$ and the maximal robust safe set $\mathcal{R}^*$ as well as its corresponding safety controller $\pi_{\mathcal{R}^*}$. Following directly from the definition of the value function (2), $\mathcal{R}^*$ and $\pi_{\mathcal{R}^*}$ confer safety assurances to system f, if for any state $x \in \mathcal{X}$, the realized model uncertainties at state x are contained within $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$. This result is formalized in Lemma. 1.

Lemma 1. *Given a control-affine system $f(x,u) = f_1(x) + f_2(x)u$ and its nominal model $\bar{f}(x,u) = \bar{f}_1(x) + \bar{f}_2(x)u$, let $\delta_1(x) = f_1(x) - \bar{f}_1(x)$ and $\delta_2(x) = f_2(x) - \bar{f}_2(x)$ be the realized model uncertainties. If $\delta_1(x) \in \mathcal{D}_1(x)$ and $\delta_2(x) \in \mathcal{D}_2(x)$ $\forall x \in \mathcal{X}$, then $\mathcal{R}^* \subseteq \mathcal{S}^*$.*

B. Ensemble Dynamics Representation

Although our framework is agnostic to how the nominal model $\bar{f}$ or the model uncertainties d_1 and d_2 are obtained, in this subsection, we introduce one method to jointly obtain $\bar{f}$, d_1, d_2, and their corresponding bounds $\mathcal{D}_1$ and $\mathcal{D}_2$.

In this work, we employ an ensemble of neural networks to obtain uncertainty in learned dynamics, a popular approach in literature to quantify uncertainty in deep networks

[15, 9]. More specifically, given a deterministic, continuous-time, control-affine dynamical system $f(x, u)$, we learn an ensemble of M fully connected feed-forward neural networks with control-affine architectures, and we use the ensemble as the nominal model $\bar{f}$. More explicitly, the ensemble is given by

$$E = \{NN^k(x, u) = NN_1^k(x) + NN_2^k(x)u\}_{k=1}^{N_m} \tag{11}$$

where NN_1^k and NN_2^k are neural networks. With a slight abuse of notation, we denote the prediction of the ensemble E by

$$\begin{aligned} E(x, u) &= \frac{1}{N_m}\sum_{k=1}^{N_m} NN^k(x, u) \\ &= \left(\frac{1}{N_m}\sum_{k=1}^{N_m} NN_1^k(x)\right) + \left(\frac{1}{N_m}\sum_{k=1}^{N_m} NN_2^k(x)\right) u \\ &= \bar{f}_1(x) + \bar{f}_2(x)u = \bar{f}(x, u) \end{aligned} \tag{12}$$

We sometimes refer to $E(x, u)$ as the *mean dynamics*, as we are taking the mean prediction among the neural networks within the ensemble.

Given the setup of the nominal model $\bar{f}(x, u)$ in (12), we would like the model uncertainties d_1 and d_2 to intuitively quantify the *variations of outputs* of the sub-nets NN_1^k and NN_2^k within the ensemble. We bound $d_1(x)$ and $d_2(x)$ using constant multiples of standard deviation of $\{NN_1^k(x)\}_{k=1}^{N_m}$ and $\{NN_2^k(x)\}_{k=1}^{N_m}$. Let us denote the extents of the ith and ij^{th} dimensions of $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$ by $\mathcal{D}_{1i}(x)$ and $\mathcal{D}_{2ij}(x)$, respectively. By assumption, $\mathcal{D}_1(x) \subseteq \mathbb{R}^{n_x}$ and $\mathcal{D}_2(x) \subseteq \mathbb{R}^{n_x \times n_u}$ are hypercubes in their respective spaces, and therefore $\mathcal{D}_{1i}(x)$ and $\mathcal{D}_{2ij}(x)$ are real intervals. More precisely, the intervals are given by

$$\mathcal{D}_{1i}(x) = [-\alpha\sigma_{1i}(x), \alpha\sigma_{1i}(x)] \tag{13a}$$

$$\mathcal{D}_{2ij}(x) = [-\gamma\sigma_{2ij}(x), \gamma\sigma_{2ij}(x)] \tag{13b}$$

where $\sigma_{1i}(x) = \texttt{StdDev}\left(\{NN_{1i}^k(x)\}_{k=1}^{N_m}\right)$, $\sigma_{2ij}(x) = \texttt{StdDev}\left(\{NN_{2ij}^k(x)\}_{k=1}^{N_m}\right)$, $\texttt{StdDev}$ is a short hand for "standard deviation", $NN_{1i}^k(x)$ and $NN_{2ij}^k(x)$ denote the ith and ij^{th} outputs of the kth NN_1 and NN_2 sub-nets, respectively.

Note 2. *α and γ are tunable parameters that determine conservativeness of the resulting safe set $\mathcal{S}$. As α and γ increase, the model uncertainty bounds $\mathcal{D}_1(x)$ and $\mathcal{D}_2(x)$ increases $\forall x \in \mathcal{X}$. Accordingly, the safe set $\mathcal{S}$ shrinks. For all the experiments in this work, we use $\alpha = \gamma = 3$.*

V. Experiments

A. Inverted Pendulum

In this example, we simulate an inverted pendulum with state $x = [\theta, \dot{\theta}]^\top$ and control u, and its dynamics is given by $\dot{x} = \left[\dot{\theta}, \ddot{\theta}\right]^\top = \left[\dot{\theta}, \frac{-b\dot{\theta}+\frac{1}{2}mgl\sin\theta-u}{\frac{ml^2}{3}}\right]^\top$, where l, m, g, and b represents the length and mass of the pendulum, acceleration of gravity, and the friction coefficient, respectively. For the purposes of this case study, the analytical expression of the dynamics is assumed to be unknown.

We first train an ensemble dynamics model, consisting of 5 fully connected feed-forward neural networks with 3 hidden layers and 256 neurons per hidden layer, using dataset $\{(x_i, u_i), \dot{x}_i\}_{i=1}^M$ parsed from trajectory rollouts with random controls. In this experiment, we want the pendulum to avoid deviating from its unstable equilibrium for more than 0.6π radians. Equivalently, the failure set is given by $\mathcal{F} = \{[\theta, \dot{\theta}]^\top | \theta > 0.6\pi\} \cup \{[\theta, \dot{\theta}]^\top | \theta < -0.6\pi\}$.

Next, we compute the safe set $\mathcal{S}$, or equivalently complement of the backward reachable tube (BRT) of the failure set $\mathcal{F}$, for a time horizon of 0.7 seconds, using the learned ensemble dynamics model along with model uncertainties. We also consider three baselines: 1) computing the safe set using only the mean dynamics $E(x, u)$ (Baseline 1 Mean Dynamics), 2) computing the safe set with $\mathcal{D}_1$ and $\mathcal{D}_2$ calculated using split conformal prediction with 5000 calibration samples and marginal coverage of 95% [3] (Baseline 2 Conformal Prediction), and 3) computing the safe set with $d_2(x)u$ approximated as $d_3(x) \in \mathbb{R}^{n_x}$ (Baseline 3 Partial Game). The purpose of Baseline 3 is to remove the interaction of the control player u and the disturbance player d_2, and renders the model uncertainty *control-independent*, as the action of the disturbance player d_3 no longer depends on that of the control player u. Furthermore, the Hamiltonian computation (7a) decouples into three independent optimizations with respect to u, d_1, and d_3. The bound on the ith component of d_3 is given by $\mathcal{D}_3(x)_i = [-a(x), a(x)]$, where $a(x) = \sum_{j=1}^{n_u} \max\{|\underline{u_j}|, |\overline{u_j}|\} \times \max\{|\underline{d_{2ij}(x)}|, |\overline{d_{2ij}(x)}|\}$.

For the number of training samples $M = 300$, we visualize the recovered safe sets for our method, all the baselines, and the ground truth in Fig. 1. The ground truth safe set, computed using the analytical expression of the dynamics, is shaded in green. The safe set recovered using our method is entirely contained within the ground truth safe set, indicating satisfaction of the safety constraint. On the other hand, Baseline 1 fails to satisfy the safety constraint, since the recovered safe set is not contained within the ground truth safe set. The safe set from Baseline 2 is not visualized in Fig. 1 because its recovered safe set is empty due to model uncertainty bounds $\mathcal{D}_1$ and $\mathcal{D}_2$ being too conservative. Specifically, conformal prediction provides *state-independent* uncertainty bounds, which are dictated by the worst-case modeling errors across all states, leading to overly conservative behaviors. Baseline 3 is less conservative than Baseline 2, as its model uncertainty bounds are state-dependent, but it is more conservative than our methods since its model uncertainty $d_3(x)$ is control-independent and its bound $\mathcal{D}_3(x)$ is an overapproximation of that of our method.

We also perform an ablation study over the number of training samples M to further highlight the benefit of using state and control-dependent model uncertainty. The percent safe set

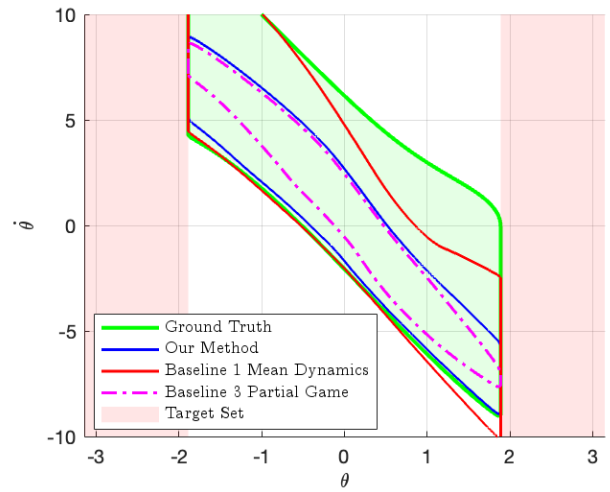

Fig. 1. Recovered safe sets (states starting from which the inverted pendulum stays within 0.6π of the upright for 0.7 seconds) for our method, the Mean Dynamics baseline (Baseline 1), the Partial Game baseline (Baseline 3) in the inverted pendulum experiment, with the ensemble trained with 300 training samples. Note that Baseline 2 (conformal prediction) is not visualized, because its safe set is empty.

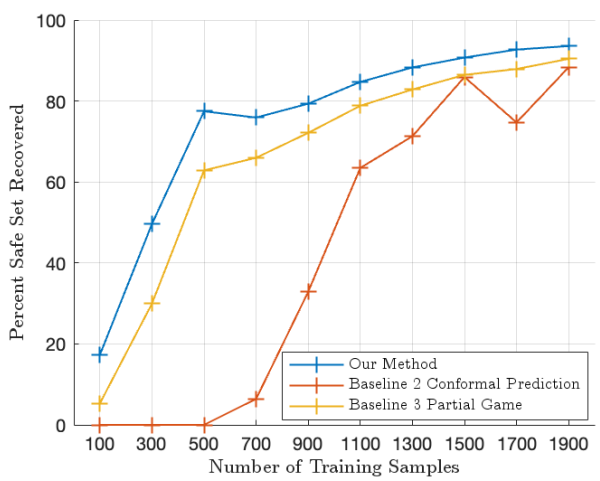

Fig. 2. Changes in percent safe set recovered with our method, the Conformal Prediction baseline (Baseline 2), and the Partial Game baseline (Baseline 3), over the number of training samples.

recovered, as a function of M, is charted in Fig. 2. Across all experimented M, our method consistently outperforms the baselines, indicating that the state and control-dependent model uncertainty representation leads to less conservative behaviors. Our model uncertainty representation can reflect local variations of model uncertainties, allowing the safe set to expand or shrink according to local model uncertainty level. Furthermore, the control-dependent nature of our model uncertainty representation, which taken into consideration of the interaction between the control and the model uncertainty, also helps reduce the conservativeness.

B. Turtlebot Hardware Experiment

We apply the proposed approach on a real hardware testbed, TurtleBot 4, which we refer to as the *vehicle*, in this experiment. We are interested in providing safety assurances for the vehicle *carrying a payload*, and we seek to demonstrate the importance of incorporating model uncertainty in safety analysis and the benefit of model learning.

Let $\mathcal{A} \subseteq \mathbb{R}^2$ be a rectangular experimental space, centered at $[0,0]^\top$ with side lengths l_x and l_y. The failure set $\mathcal{F}$ is hence given by $\mathcal{F} = \{x \in \mathcal{X} | |p_x| > \frac{l_x}{2}, |p_y| > \frac{l_y}{2}\}$, where p_x and p_y are the x, y positions of the center of mass of the vehicle. We model the vehicle as a 4-dimensional system with state $[p_x, p_y, \theta, \omega]^\top$, where ω is the angular velocity, and control u, an angular velocity setpoint. We operate the vehicle with a constant forward velocity.

We first collect 120 state and control trajectories using a random control at each time step. Each trajectory is roughly 5 seconds and yields about 100 training samples $([x, u], \dot{x})$. Then, we train an ensemble dynamics model containing 5 fully connected feed-forward neural networks, each with 3 hidden layers and 512 neurons per layer.

The safe set and the safety controller are computed for our method and the mean dynamics baseline until convergence (i.e., the time horizon is $[0, \infty]$). For visualization purposes, we project the safe set to the x, y plane at some fixed θ and ω. In Fig. 3, the safe sets projected at $\theta = 0.7$ and $\omega = 0$ are shown. We apply mean dynamics baseline's safety controller π_m^*, from two states x_1 and x_2, within the mean dynamics baseline safe set $\mathcal{S}_m$. π_m^* is unable to keep the vehicle inside $\mathcal{A}$, since $\mathcal{S}_m$ and π_m^* do not take into consideration the model uncertainty and, as a result, are overly optimistic. On the other hand, we roll out the vehicle from 2 nearby states, x_3 and x_4, within our method's safe set $\mathcal{S}$ with our method's safety controller π^*, and the vehicle stays within $\mathcal{A}$, indicating that our method is able to obtain a better estimate of the actual safe set of the system.

We now highlight the benefit of model learning with a safety filtering experiment. When the vehicle is traveling at a constant forward velocity, it is common to model the vehicle with a three-dimensional Dubins Car (Dubins3D) with the dynamics $\left[\dot{p_x}, \dot{p_y}, \dot{\theta}\right]^\top = [V\cos(\theta), V\sin(\theta), u]^\top$. However, since the vehicle carries a payload, which introduces factors that could render the Dubins3D model inaccurate, a learned model can more accurately represent the system. We compute the safe set $\mathcal{S}_d$ and safety controller π_d^* using the Dubins3D model to convergence. Then, we use π_d^* to *filter* a nominal controller $\pi(x) = 0$ in a least restrictive fashion [7]. The filtered controller $\hat{\pi}_d$ keeps the vehicle moving with the current heading unless it is at risk of exiting $\mathcal{S}_d$, in which case π_d^* takes over (i.e. $\hat{\pi}_d(x) = \pi_d^*(x)$). We similarly filter π with π^*, and filtered controller is denoted as $\hat{\pi}$.

Starting from a state $x \in \mathcal{S} \cap \mathcal{S}_d$, we roll out the vehicle with $\hat{\pi}_d$ and $\hat{\pi}$. The state trajectories projected to the $x - y$ plane are shown in Fig. 4. The vehicle eventually exits $\mathcal{A}$ under $\hat{\pi}_d$, indicating that π_d^* does not actually render $\mathcal{S}_d$ control invariant

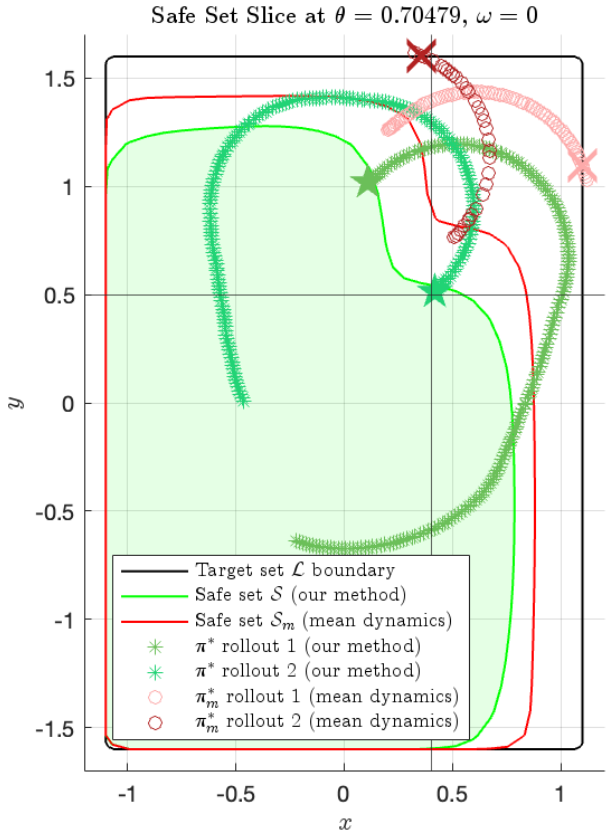

Fig. 3. Comparison between projected safe sets $\mathcal{S}$ and $\mathcal{S}_m$ along with TurtleBot rollout trajectories using safety controllers π^* and π^*_m. The initial state of rollouts using π^* are marked with pentagons. All rollouts start with heading $\theta = 0.7$ and angular velocity $\omega = 0$. The states at which the TurtleBot enters failure set $\mathcal{F}$ are marked with crosses.

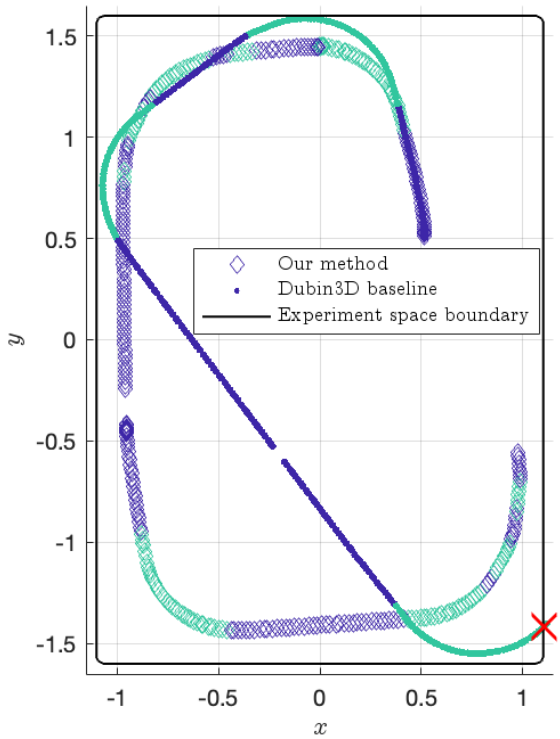

Fig. 4. Filtered controller $\hat{\pi}$ and $\hat{\pi}_d$ rollout trajectories. The blue markers indicate the states at which the nominal controller π is on. Whereas the mint markers indicates the states where safety controllers π^* and π^*_d intervene. The state at which the vehicle exits the experiment space under $\hat{\pi}_d$ is marked with a red cross.

under the actual vehicle dynamics and is overly optimistic. However, $\hat{\pi}$ does keep the vehicle inside $\mathcal{A}$ for the entire experiment. For both trajectories, the mint-colored markers indicate the states where the safety controller π^* or π^*_d takes over and commands the vehicle to turn maximally to stay within $\mathcal{A}$. Since $\mathcal{S}_d$ is more optimistic than $\mathcal{S}$, π^*_d intervenes closer to the boundary of $\mathcal{A}$ than π^* would, and the resulting trajectory is uncomfortably close to exiting $\mathcal{A}$. In contrast, there is a healthy margin for the trajectory filtered by π^*.

VI. Conclusion

In this work, we proposed a framework for generating robust safety assurances for systems with unknown dynamics. Further, we provide a concrete instantiation of our framework with ensemble neural network models as the nominal model and safeguard the system against the worst-case model uncertainty. Though our method is shown to provide robust safety assurances in the experiments, it faces several challenges, which we look to address in future works. First of all, our method does not scale well to higher-dimensional systems. We will investigate the possibility of incorporating learning-based reachability computation tools [4, 12] into our framework. Second, the proposed model uncertainty estimation approach might not provide model uncertainty bounds that accurately reflect the distribution of realized model uncertainties. We will address this challenge by examining other uncertainty estimation approaches, such as [15] and [13].

References

[1] Anayo K Akametalu, Jaime F Fisac, Jeremy H Gillula, Shahab Kaynama, Melanie N Zeilinger, and Claire J Tomlin. Reachability-based safe learning with gaussian processes. In *53rd IEEE Conference on Decision and Control*, pages 1424–1431. IEEE, 2014.

[2] Aaron D. Ames, Xiangru Xu, Jessy W. Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017. doi: 10.1109/TAC.2016.2638961.

[3] Anastasios N Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *arXiv preprint arXiv:2107.07511*, 2021.

[4] Somil Bansal and Claire J. Tomlin. Deepreach: A deep learning approach to high-dimensional reachability.

In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1817–1824, 2021. doi: 10.1109/ICRA48506.2021.9561949.

[5] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J. Tomlin. Hamilton-jacobi reachability: A brief overview and recent advances. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2242–2253, 2017. doi: 10.1109/CDC.2017.8263977.

[6] Emmanuel Nicholas Barron, Lawrence Craig Evans, and Robert Jensen. Viscosity solutions of isaacs' equations and differential games with lipschitz controls. *Journal of Differential Equations*, 53(2):213–233, 1984.

[7] Javier Borquez, Kaustav Chakraborty, Hao Wang, and Somil Bansal. On safety and liveness filtering using hamilton-jacobi reachability analysis. *arXiv preprint arXiv:2312.15347*, 2023.

[8] Fernando Castañeda, Jason J. Choi, Bike Zhang, Claire J. Tomlin, and Koushil Sreenath. Pointwise feasibility of gaussian process-based safety-critical control under model uncertainty. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6762–6769, 2021. doi: 10.1109/CDC45484.2021.9683743.

[9] Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

[10] Lawrence C Evans and Panagiotis E Souganidis. Differential games and representation formulas for solutions of hamilton-jacobi-isaacs equations. *Indiana University mathematics journal*, 33(5):773–797, 1984.

[11] Jaime F. Fisac, Anayo K. Akametalu, Melanie N. Zeilinger, Shahab Kaynama, Jeremy Gillula, and Claire J. Tomlin. A general safety framework for learning-based control in uncertain robotic systems. *IEEE Transactions on Automatic Control*, 64(7):2737–2752, 2019. doi: 10.1109/TAC.2018.2876389.

[12] Jaime F. Fisac, Neil F. Lugovoy, Vicenç Rubies-Royo, Shromona Ghosh, and Claire J. Tomlin. Bridging hamilton-jacobi safety analysis and reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8550–8556, 2019. doi: 10.1109/ICRA.2019.8794107.

[13] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

[14] Sylvia Herbert, Jason J Choi, Suvansh Sanjeev, Marsalis Gibson, Koushil Sreenath, and Claire J Tomlin. Scalable learning of safety guarantees for autonomous systems using hamilton-jacobi reachability. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5914–5920. IEEE, 2021.

[15] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[16] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[17] John Lygeros. On reachability and minimum cost optimal control. *Automatica*, 40(6):917–927, 2004.

[18] Ian Mitchell, Alex Bayen, and Claire J. Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control (TAC)*, 50(7):947–957, 2005.

[19] Luyao Niu, Hongchao Zhang, and Andrew Clark. Safety-critical control synthesis for unknown sampled-data systems via control barrier functions. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6806–6813. IEEE, 2021.

[20] Andrew Taylor, Andrew Singletary, Yisong Yue, and Aaron Ames. Learning for safety-critical control with control barrier functions. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pages 708–717. PMLR, 10–11 Jun 2020. URL https://proceedings.mlr.press/v120/taylor20a.html.