Generalizable Physics-Informed Learning for Stochastic Safety-Critical Systems

Zhuoyuan Wang, Albert Chern, Yorie Nakahira

arXiv:2407.08868v5 [cs.LG] 7 Oct 2025

***Abstract*— Accurate estimation of long-term risk is essential for the design and analysis of stochastic dynamical systems. Existing risk quantification methods typically rely on extensive datasets involving risk events observed over extended time horizons, which can be prohibitively expensive to acquire. Motivated by this gap, we propose an efficient method for learning long-term risk probabilities using short-term samples with limited occurrence of risk events. Specifically, we establish that four distinct classes of long-term risk probabilities are characterized by specific partial differential equations (PDEs). Using this characterization, we introduce a physics-informed learning framework that combines empirical data with physics information to infer risk probabilities. We then analyze the theoretical properties of this framework in terms of generalization and convergence. Through numerical experiments, we demonstrate that our framework not only generalizes effectively beyond the sampled states and time horizons but also offers additional benefits such as improved sample efficiency, rapid online inference capabilities under changing system dynamics, and stable computation of probability gradients. These results highlight how embedding PDE constraints—which contain explicit gradient terms and inform how risk probabilities depend on state, time horizon, and system parameters—improves interpolation and generalization between/beyond the available data.**

I. Introduction

Accurate estimation of long-term risk is critical for the design and analysis of stochastic dynamical systems. Miscalculations of risk can lead to safety and performance issues over extended periods, and short-term risk is not sufficient to capture when a safe state inevitably reaches unsafe regions in the future. Existing risk quantification methods often require sufficient data on risk events and long-term trajectories. However, obtaining such data (*e.g.,* through real-world operations) can be prohibitively costly. Moreover, when the risk of interest pertains to rare events, the volume of data needed by purely data-driven approaches increases disproportionately, making reliable estimation a significant challenge [1].

Motivated by these challenges, we propose Physics-Informed Probability Estimator (PIPE), a physics-informed learning framework to estimate risk probabilities in an efficient and generalizable manner. Fig. 1 shows the overview diagram of the proposed PIPE framework. The framework consists of two key parts, namely the physics model part and the data-driven learning part. For the former part, we derive that the evolution of risk probabilities given stochastic system dynamics is governed by deterministic partial differential equations (PDEs) (Theorem 1-4). For the latter part, we integrate the physics model knowledge from the PDEs with sampled data to train a physics-informed neural network. We theoretically show that the proposed framework can generate risk probability estimates with bounded error, even when the data used for training only covers partial state space with short time horizons (Theorem 5). We also show convergences of the proposed method to ground truth risk probabilities under mild assumptions (Theorem 6). In addition, we experimentally show that the proposed framework is able to accurately predict risk probabilities and their gradients on the *entire* state-temporal domain, with only *sparse* data sample in the *sub-region* (Fig. 3 and Fig. 6), enabling huge sample complexity and online computation time reduction (Fig. 4 and Table I). Furthermore, we show that the proposed framework can serve as a surrogate model to learn risk probabilities of varying parameterized system dynamics, with data only on a few dynamics in the distribution (Fig. 5).

The rest of the paper is organized as follows. In section II, we list related work. In section III we give preliminaries. In section IV, we state the problem formulation. In section V, we derive the mathematical characterization of risk probability with the governing partial differential equations. In section VI, we present the physics-informed learning method for probability estimation and the corresponding theoretical analysis. In section VII, we show experiment results. Finally we conclude the paper in section VIII.

II. Related Work

A. Risk Probability Estimation

A wide range of techniques has been developed to estimate risk probabilities. These approaches vary in data availability, theoretical assumptions, and computational complexity. One line of work uses results from probability theory and dynamical systems. For example, probabilistic inequalities and martingale methods are used to bound the probability of deviating into an unsafe region [2], [3], [4], [5]. Large deviations theory provides asymptotic estimates of rare event probabilities by characterizing the decay rate of such probabilities [6]. While this line of work seeks to bound or approximate risk probabilities analytically, we characterize the exact risk probability (Theorem 1-4).

Zhuoyuan Wang and Yorie Nakahira are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, PA 15213 USA (e-mail: zhuoyuaw, ynakahir@andrew.cmu.edu).

Albert Chern is with the Department of Computer Science and Engineering, University of California San Diego, CA 92093 USA (e-mail: alchern@ucsd.edu).

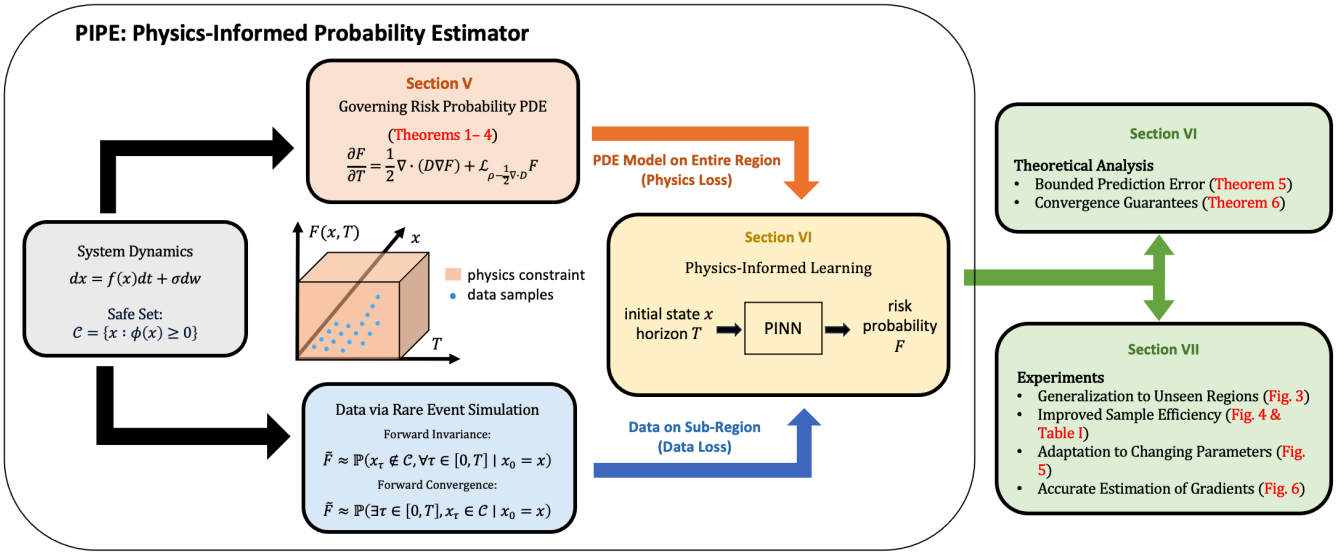


Fig. 1: The overview diagram of the proposed PIPE framework. The system takes form (1) with safe set defined as (2). For the physics model, we derive that the mapping between state-time pair and the risk probability satisfies a governing convection diffusion equation (Theorem 1-4). For training data, one can acquire the empirical risk probabilities with any existing rare event simulation method, without the need to cover the entire time-state space. The PIPE framework uses physics-informed neural networks to learn the risk probability by fitting the empirical training data and using the physics model as constraints. PIPE gives more accurate and sample efficient risk probability and gradient predictions than Monte Carlo or its variants, and can generalize to unseen regions in the state space and unknown parameters in the system dynamics thanks to its integration of data and physics models.

Another line of work uses Monte Carlo techniques to estimate risk or rare event probabilities [7]. Importance sampling methods are commonly used when sampling distributions are biased towards the failure events to efficiently obtain information about risk-sensitive states [8], [9]. The cross-entropy method iteratively optimizes the sampling distribution via cross-entropy minimization [10]. Reinforcement learning has also been employed to guide the sampling process [11]. Bayesian inference is used to steer samples toward near-optimal sampling distribution [12]. Subset simulation breaks the rare event into a sequence of more likely intermediate events, estimating the small failure probability as a product of conditional probabilities [13]. These data-driven approaches often selectively obtain risk event data for improved sample efficiency, and require data to be obtained from all states and time horizons of interest [7], [8], [13]. In contrast, our work addresses a different challenge of learning a surrogate model in a way that integrates data and physics information, and the proposed framework can also leverage these sampling techniques. Due to the use of physics information, the proposed method can infer long-term risk information from short-term safe event data (Theorem 5 and Fig. 3). This feature also contributes to improved sampled efficiency, particularly when the risk events are rare (Fig. 4).

B. Physics-informed Neural Networks

Physics-informed neural networks (PINNs) have emerged as a robust framework for integrating physical constraints—typically expressed as partial differential equations (PDEs)—directly into a neural network training [14], [15], [16]. By enforcing PDE constraints during training, PINNs ensure that the learned solutions are consistent with the structures arising from underlying physics. PINNs have demonstrated remarkable success in a wide range of applications, including stochastic control [17], power systems [18], fluid dynamics [19], and medical applications [20]. Various extensions of PINNs have emerged, including Bayesian PINNs for noisy data [21], PINNs with hard constraints for topology optimization [22], and parallel PINNs via domain decomposition for multi-scale problems [23]. Additionally, recent theoretical analyses address approximation errors [24], [25], [26], [27], generalization capabilities [28], [29], [30], [31], and convergence rates [32], [33], [34]. When the physics models are uncertain, it is incorporated in a framework with an additional inverse problem that identifies governing equations [14]. Our characterization of four types of long-term safety probabilities as the solutions to PDEs (Theorem 1-4) allows these quantities to be learned by physics-informed learning. To our knowledge, existing work has not previously employed PINNs to estimate long-term probabilities from short-term safe data while providing theoretical guarantees (such as Theorem 5 and Theorem 6). Our experiments demonstrate that embedding PDE constraints—specified using system parameters and with explicit gradient terms—facilitates stable interpolation across systems and gradient estimation. This results in additional advantages such as rapid online risk inference for varying system dynamics (Fig. 5) and stable computation of probability gradients (Fig. 6).

III. Preliminaries

Let $\mathbb{R}$, $\mathbb{R}_+$, $\mathbb{R}^n$, and $\mathbb{R}^{m\times n}$ be the set of real numbers, the set of non-negative real numbers, the set of n-dimensional real vectors, and the set of $m \times n$ real matrices. Let $x[k]$ be the k-th element of vector x. Let $x^+ = \max(x, 0)$ and $x \wedge y = \min\{x, y\}$ for $x, y \in \mathbb{R}$. Let $f : \mathcal{X} \to \mathcal{Y}$ represent that f is a mapping from space $\mathcal{X}$ to space $\mathcal{Y}$. Let $\mathbb{1}_{\mathcal{E}}$ be an indicator function, which takes 1 when condition $\mathcal{E}$ holds and 0 otherwise. Let $\mathbb{P}_x(\mathcal{E}) = \mathbb{P}(\mathcal{E}(X)|X_0 = x)$ denote the probability of event $\mathcal{E}(X)$ involving a stochastic process $X = \{X_t\}_{t\in\mathbb{R}_+}$ conditioned on $X_0 = x$. Let $\mathbb{E}_x[F(X)] = \mathbb{E}[F(X)|X_0 = x]$ denote the expectation of $F(X)$ (a functional of X) conditioned on $X_0 = x$. We use upper-case letters (*e.g.,* X) to denote random variables and lower-case letters (*e.g.,* x) to denote their specific realizations.

IV. Problem Statement

We present the settings and the scope of this paper in section IV-A and section IV-B, respectively.

A. System Description and design specifications

We consider a control system with stochastic noise of k-dimensional Brownian motion W_t starting from $W_0 = 0$. The system state, $X_t \in \mathbb{R}^n, t \in \mathbb{R}_+$, evolves according to the following stochastic differential equation (SDE)

$$dX_t = f(X_t)dt + \sigma(X_t)dW_t, \tag{1}$$

where $\sigma : \mathbb{R}^n \to \mathbb{R}^k$ characterizes the magnitudes of the noise term. We restrict ourselves to the settings where f, σ have sufficient regularity conditions such that the system has a well-defined solution. For example, the system is well-defined when f and σ are bounded and globally Lipschitz [35], which are common assumptions usually satisfied in practice. More generally, conditions required to have a unique solution can be found in [35, Chapter 1], [36, Chapter II.7] and references therein. The size of $\sigma(X_t)$ is determined from the uncertainties in the disturbance, unmodeled dynamics [5], and the prediction errors of the environmental variables [37], [38]. Examples of these cases include when the unmodeled dynamics are captured using statistical models such as Gaussian Processes [5][1] and when the motion of the environment variables are estimated using physics-based models such as Kalman filters [37], [38]. Note that dynamics (1) can capture not only autonomous systems, but also closed-loop systems with memoryless state feedback control (*e.g.,* PID and robust controllers).

In this paper, we aim to study the following two aspects related to safety for the system: forward invariance and recovery. The forward invariance refers to the property that the system state always stays in the safe region, while recovery refers to the property that the system state recovers to the safe region even if it originates from the unsafe region.

[1]In such settings, the value of $\sigma(X_t)$ can be determined from the output of the Gaussian Process.

Definition 1 (Safe Set)**.** The safe region is defined using a set $\mathcal{C}$ that is characterized by a super level set of some function $\phi(x)$, *i.e.,*

$$\mathcal{C} = \{x \in \mathbb{R}^n : \phi(x) \geq 0\}, \tag{2}$$

We call $\mathcal{C}$ the *safe set*, and $\phi(x)$ the barrier function. Additionally, we denote the boundary of the safe set and unsafe set by

$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : \phi(x) = 0\}, \tag{3}$$
$$\mathcal{C}^c = \{x \in \mathbb{R}^n : \phi(x) < 0\}. \tag{4}$$

We assume that $\phi(x) : \mathbb{R}^n \to \mathbb{R}$ is a first-order differentiable function whose gradient does not vanish at $\partial\mathcal{C}$.

We assume that the dynamics and uncertainty in the system and the environment are all captured in (1) so that the safe set can be defined using a static function $\phi(x)$ instead of a time-varying one.

1) Forward invariance: When the initial state starts from a safe region, *i.e.,* $X_0 = x \in \mathcal{C}$, we study the quantities associated with forward invariance: the safety margin from $\partial\mathcal{C}$ and the first exit time from $\mathcal{C}$.

Properties associated invariance to a set have been studied from the perspective of forward invariance. A forward invariant set with respect to a system is often defined to satisfy $X_0 \in \mathcal{M} \Rightarrow X_t \in \mathcal{M}, \forall t \in \mathbb{R}_+$ in a deterministic framework [39, Section 4.2]. However, in a stochastic system, it may not be always possible to ensure $X_t \in \mathcal{M}$ at all time because the probability of exiting $\mathcal{M}$ can accumulate over time. Therefore, we use a modified definition, formally defined below, where $X_t \in \mathcal{M}$ is only required to hold in a finite time interval T.

Definition 2 (Forward Invariance)**.** The state of system (1) is said to remain forward invariant with respect to a set $\mathcal{M}$ during a time interval $[0, T]$ if, given $X_0 = x \in \mathcal{M}$,

$$X_t \in \mathcal{M}, \quad \forall t \in (0, T]. \tag{5}$$

We study the probability of forward invariance to the safe set through characterizing the distributions of the following two random variables:

$$\Phi_x(T) := \inf\{\phi(X_t) \in \mathbb{R} : t \in [0, T], X_0 = x\}, \tag{6}$$
$$\Gamma_x(\ell) := \inf\{t \in \mathbb{R}_+ : \phi(X_t) \leq \ell, X_0 = x\}, \tag{7}$$

where $x \in \mathcal{C}$, and $\ell \in \mathbb{R}$ is the safety margin. The value of $\Phi_x(T)$ informs the worst-case safety margin from the boundary $\partial\mathcal{C}$ during $[0, T]$, while the value of $\Gamma_x(0)$ is the first exit time from the safe set.

The probability of staying within the safe set $\mathcal{C}$ during a time interval $[0, T]$ given $X_0 = x \in \mathcal{C}$ can be computed using the distributions of (6) or (7) as

$$\begin{aligned}\mathbb{P}_x(X_t \in \mathcal{C}, \forall t \in [0, T]) &= \mathbb{P}\left(\Phi_x(T) \geq 0\right) \\ &= 1 - \mathbb{P}\left(\Gamma_x(0) \leq T\right).\end{aligned} \tag{8}$$

2) Recovery: When the initial state is outside of the safe region, *i.e.,* $X_0 = x \notin \mathcal{C}$, we study the quantities associated with recovery: the distance from $\mathcal{C}$ and the recovery time.

Definition 3 (Forward Convergent Set)**.** A set $\mathcal{M}$ is said to be a forward convergent set with respect to system (1) if

$$X_0 \notin \mathcal{M} \Rightarrow \exists \tau \in (0, \infty) \text{ s.t. } X_\tau \in \mathcal{M}. \tag{9}$$

That is, for any initial state that does not belong to set $\mathcal{M}$, the state re-enters the set in finite time. Equivalently, even if the state does not belong to set $\mathcal{M}$ at some time, it will enter $\mathcal{M}$ in finite time. Note that a forward convergent set does not require the state to stay in the set indefinitely after re-entry.

We study forward convergence to the safe set through characterizing the distribution of the following two random variables:

$$\Theta_x(T) := \sup\{\phi(X_t) \in \mathbb{R} : t \in [0, T], X_0 = x\}, \tag{10}$$
$$\Psi_x(\ell) := \inf\{t \in \mathbb{R}_+ : \phi(X_t) \geq \ell, X_0 = x\}, \tag{11}$$

where $x \notin \mathcal{C}$. The value of $\Theta_x(T)$ informs whether the state re-enters the safe set within a time horizon T, and $\Psi_x(0)$ characterizes the duration required for the state to re-enter the safe region for the first time.

The probability of having re-entered the safe set $\mathcal{C}$ by time T given $X_0 = x \notin \mathcal{C}$ can be computed using the distributions of (10) or (11) as

$$\begin{aligned} \mathbb{P}_x\left(\exists t \in [0, T] \text{ s.t. } X_t \in \mathcal{C}\right) &= \mathbb{P}\left(\Theta_x(T) \geq 0\right) \\ &= 1 - \mathbb{P}\left(\Psi_x(0) \leq T\right), \end{aligned} \tag{12}$$

given the fact that system (1) is time-invariant. More generally, the distributions of $\Phi_x(T)$, $\Theta_x(T)$, $\Gamma_x(\ell)$, and $\Psi_x(\ell)$ contain much richer information than the probability of forward invariance or recovery. Thus, knowing their exact distributions allows uncertainty and risk to be rigorously quantified in autonomous control systems.

B. Objective and scope of this paper

In this paper, we build upon analysis tools from stochastic processes to answer the following question:

- When the stochastic system dynamics (1) is given, what are the *exact distributions* of (6), (7), (10), and (11)?
- With the characterization of the exact distribution of (6), (7), (10), and (11), how do we efficiently calculate them?

We will answer the first question in section V and the second question in section VI. The distribution of (6) and (7) will allow us to study the property of invariance from a variety of perspectives, such as the average failure time, the tail distributions of safety or loss of safety, and the mean and tail of the safety margin. The distribution of (10) and (11) will allow us to study the property of recovery from a variety of perspectives, such as the average recovery time, tail distribution of recovery, and the mean and tail distribution of recovery vs. crashes.

Remark 1. The efficient calculation of (6), (7), (10), and (11) can potentially be applied to chance-constrained safe predictive control [40], safe neural network-based control [41], and reinforcement learning with long-term safety requirements [42].

V. Characterization of Probabilities

In this section, we study the properties associated with safety (forward invariance) and recovery (forward convergence). The former is quantified using the distribution of the safety margin (6) and first exit time (7) in section V-A. The latter is quantified using the distribution of the distance to the safe set (10) and the recovery time (11) in section V-B.

Definition 4 (Infinitesimal Generator)**.** The infinitesimal generator A of a stochastic process $\{Y_t \in \mathbb{R}^n\}_{t \in \mathbb{R}_+}$ is

$$AF(y) = \lim_{h \to 0} \frac{\mathbb{E}_y\left[F(Y_h)\right] - F(y)}{h} \tag{13}$$

whose domain is the set of all functions $F : \mathbb{R}^n \to \mathbb{R}$ such that the limit of (13) exists for all $y \in \mathbb{R}^n$.

Applying the infinitesimal generator to ϕ yields

$$A\phi = \mathcal{L}_f \phi + \frac{1}{2} \operatorname{tr}\left(\sigma\sigma^\intercal \operatorname{Hess} \phi\right) \tag{14}$$
$$=: D_\phi(x) \tag{15}$$

We define $D_\phi(x)$ to be the value of (14) evaluated at x. Here, $\mathcal{L}_v$ is the Lie derivative along a vector or matrix field v. The infinitesimal generator can be interpreted as the stochastic counterpart of the Lie derivative: It gives the derivative along the flow of dX_t in expectation, *i.e.,* $\mathbb{E}[d\phi(X_t)] = A\phi(X_t)dt$.

The random variables of our interests, (6), (7), (10) and (11), are functions of $\phi(X_t)$. Since the stochastic dynamics of $d\phi(X_t)$ are driven by X_t, we consider the augmented state space

$$Z_t := \begin{bmatrix} \phi(X_t) \\ X_t \end{bmatrix} \in \mathbb{R}^{n+1}. \tag{16}$$

From Itô's lemma, dynamics of $\phi(X_t)$ satisfies

$$d\phi(X_t) = A\phi(X_t)dt + \mathcal{L}_\sigma \phi(X_t) dW_t, \tag{17}$$

where $\mathcal{L}_\sigma$ is the Lie derivative along σ. Combining (1) and (17), the dynamics of Z_t are given by the SDE

$$dZ_t = \rho(Z_t)\, dt + \zeta(Z_t)\, dW_t, \tag{18}$$

where the drift and diffusion parameters are

$$\rho(z) = \begin{bmatrix} D_\phi(x) \\ f(x) \end{bmatrix}, \qquad \zeta(z) = \begin{bmatrix} \mathcal{L}_\sigma \phi(x) \\ \sigma(x) \end{bmatrix}. \tag{19}$$

Since risk probabilities are complementary to safety probabilities, we will consider four safety-related probabilities of interest in section V-A and section V-B. We will call them risk probabilities for simplicity.

We then present four theorems on characterization of different risk probabilities. The proof of all theorems can be found in the appendix of the paper.

A. Probability of Forward Invariance

When the state initiates inside the safe set, the distribution of the safety margin, $\Phi_x(T)$ in (6), is given below.

Theorem 1. *Consider system* (1) *with the initial state* $X_0 = x$. *Let* $z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. *Then, the complementary cumulative distribution function (CCDF) of* $\Phi_x(T)$,

$$F(z,T;\ell) = \mathbb{P}(\Phi_x(T) \geq \ell), \quad \ell \in \mathbb{R}, \tag{20.A}$$

is the solution to the initial-boundary-value problem of the convection-diffusion equation on the super-level set $\{z \in \mathbb{R}^n : z[1] \geq \ell\}$

$$\begin{cases} \frac{\partial F}{\partial T} = \frac{1}{2}\nabla\cdot(D\nabla F) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D}F, & z[1] \geq \ell, T > 0, \\ F(z,T;\ell) = 0, & z[1] < \ell, T > 0, \\ F(z,0;\ell) = \mathbb{1}_{\{z[1]\geq\ell\}}(z), & z \in \mathbb{R}^{n+1}, \end{cases} \tag{20.B}$$

where $D := \zeta\zeta^\intercal$. *Here, the spatial derivatives* ∇ *and* $\mathcal{L}$ *apply to the variable* $z \in \mathbb{R}^{n+1}$.[2]

Theorem 1 converts the problem of characterizing the distribution function into a deterministic convection-diffusion problem. When we set $\ell = 0$, it gives the exact probability that $\mathcal{C}$ is a forward invariant set with respect to (1) during the time interval $[0, T]$. For arbitrary $\ell > 0$, it can be used to compute the probability of maintaining a safety margin of ℓ during $[0, T]$.

Remark 2. In Theorem 1 and the other theorems in this section, we use CDFs and CCDFs with a slight abuse of notation. The presence or absence of equality in the inequality conditions are chosen so that the obtained probability can be used to compute (8) and (12). Also, note that if a random variable Y has a PDF, then $\mathbb{P}(Y \leq \ell) = \mathbb{P}(Y < \ell)$, so the presence or absence of equality in the probability density function does not affect the computed probabilities.

Moreover, the distribution of the first exit time from the safe set, $\Gamma_x(0)$ in (7), and the first exit time from an arbitrary super level set $\{x \in \mathbb{R}^n : \phi(x) \geq \ell\}$, is given below.

Theorem 2. *Consider system* (1) *with the initial state* $X_0 = x$. *Let* $z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. *Then, the cumulative distribution function (CDF) of the first exit time* $\Gamma_x(\ell)$,

$$G(z,T;\ell) = \mathbb{P}(\Gamma_x(\ell) \leq T), \tag{21.A}$$

is the solution to

$$\begin{cases} \frac{\partial G}{\partial T} = \frac{1}{2}\nabla\cdot(D\nabla G) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D}G, & z[1] \geq \ell, T > 0, \\ G(z,T;l) = 1, & z[1] < \ell, T > 0. \\ G(z,0;l) = \mathbb{1}_{\{z[1]<\ell\}}(z), & z \in \mathbb{R}^{n+1}, \end{cases} \tag{21.B}$$

where $D = \zeta\zeta^\intercal$.

Similarly, Theorem 2 gives the distribution of the first exit time from an arbitrary super level set of barrier function as the solution to a deterministic convection-diffusion equation. When we set $\ell = 0$, it can also be used to compute the exact probability of staying within the safe set during any time interval.

[2]This rule applies to Theorem 2 - 4 as well.

B. Probability of Forward Convergence

When the state initiates outside the safe set, the distribution of the distance from the safe set, $\Theta_x(T)$ in (10), is given below.

Theorem 3. *Consider system* (1) *with the initial state* $X_0 = x$. *Let* $z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. *Then, the CDF of* $\Theta_x(T)$,

$$Q(z,T;\ell) = \mathbb{P}\left(\Theta_x(T) < \ell\right), \quad \ell \in \mathbb{R}, \tag{22.A}$$

is the solution to

$$\begin{cases} \frac{\partial Q}{\partial T} = \frac{1}{2}\nabla\cdot(D\nabla Q) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D}Q, & z[1] < \ell, T > 0, \\ Q(z,T;\ell) = 0, & z[1] \geq \ell, T > 0, \\ Q(z,0;\ell) = \mathbb{1}_{\{z[1]<\ell\}}(z), & z \in \mathbb{R}^{n+1}, \end{cases} \tag{22.B}$$

where $D = \zeta\zeta^\intercal$.

Theorem 3 gives the distribution of safety distance $\Theta_x(T)$ as a deterministic convection-diffusion problem. When we set $\ell = 0$, it provides the exact probability that a stochastic process (1) initiating outside the safe set enters the safe set during the time interval $[0, T]$. When we set $\ell < 0$, it can be used to compute the probability of coming close to $|\ell|$-distance from the safe set along the time interval $[0, T]$.

Finally, the distribution of the recovery time from the unsafe set, $\Psi_x(0)$ in (11), and the entry time to an arbitrary super level set of the barrier function, $\Psi_x(\ell)$, is given below.

Theorem 4. *Consider system* (1) *with the initial state* $X_0 = x$. *Let* $z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. *Then, the CDF of the recovery time* $\Psi_x(\ell)$,

$$N(z,T;\ell) = \mathbb{P}(\Psi_x(\ell) \leq T), \tag{23.A}$$

is the solution to

$$\begin{cases} \frac{\partial N}{\partial t} = \frac{1}{2}\nabla\cdot(D\nabla N) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D}N, & z[1] < \ell, T > 0, \\ N(z,T;l) = 1, & z[1] \geq \ell, T > 0, \\ N(z,0;l) = \mathbb{1}_{\{z[1]\geq\ell\}}(z), & z \in \mathbb{R}^{n+1}, \end{cases} \tag{23.B}$$

where $D = \zeta\zeta^\intercal$.

Theorem 4 gives the distribution of the re-entry time as a solution to a deterministic convection-diffusion equation. When we set $\ell = 0$, it can be used to compute the exact probability of entering the safe region during any time intervals. When we set $\ell < 0$, it can be used to study the first time to reach $|\ell|$-close to the safe set.

Till here, we have characterized all probabilities of interest into the solution of certain PDEs. However, solving such PDEs with sampling-based methods [7] can be computationally expensive. In the following section, we will present our proposed physics-informed learning framework for efficient evaluation of risk probabilities based on both sampled data and the PDE model we derived earlier.

VI. Physics-informed Learning

In this section, we present our proposed physics-informed learning framework to efficiently solve the associating PDEs for risk probabilities of interest.

A. Physics-informed Learning Framework

For simplicity of notation, we use one single notation F to denote the probabilities of interest *i.e.,* equations (20.A), (21.A), (22.A) and (23.A), and use W_F to denote the associated PDEs (20.B), (21.B), (22.B) and (23.B). For instance, for complementary cumulative distribution of $\Phi_x(T)$ defined in (20.A), we have

$$W_F := \frac{\partial F}{\partial T} - \frac{1}{2}\nabla\cdot(D\nabla F) + \mathcal{L}_{\rho-\frac{1}{2}\nabla\cdot D}F. \tag{24}$$

While the PDE provides a way to get the actual risk probability of the system, to solve a PDE using numerical techniques is not easy in general, especially when the coefficients are time varying as in the case of (24). Monte Carlo (MC) methods provide another way to solve this problem. Assume the dynamics of the system is given, one can simulate the system for an initial condition multiple times to get an empirical estimate of the risk probability by calculating the ratio of unsafe trajectories over all trajectories. However, MC requires huge number of trajectories to get accurate estimation, and the evaluation of the risk probability can only be conducted at a single point at a time.

To leverage the advantages of PDE and MC and to overcome their drawbacks, we propose to use physics-informed neural networks (PINNs) to learn the mapping from the state and time horizon to the risk probability value F. Fig. 2 shows the architecture of the PINN. The PINN takes the state-time pair (x,T) and the system parameter λ as the input, and outputs the risk probability prediction $\hat{F}$, the state and time derivatives $\frac{\partial\hat{F}}{\partial x}$ and $\frac{\partial\hat{F}}{\partial T}$, and the Hessian $\frac{\partial^2\hat{F}}{\partial x^2}$, which come naturally from the automatic differentiation in deep learning frameworks such as PyTorch [43] and TensorFlow [44]. Unlike standard PINN, we add the system parameter λ as an input to achieve adaptations on varying system parameters. Assume the PINN is parameterized by $\boldsymbol{\theta}$, the loss function is defined as

$$\mathcal{L}(\boldsymbol{\theta}) = \omega_p\mathcal{L}_p(\boldsymbol{\theta}) + \omega_d\mathcal{L}_d(\boldsymbol{\theta}), \tag{25}$$

where

$$\begin{aligned}\mathcal{L}_p(\boldsymbol{\theta}) &= \frac{1}{|\mathcal{P}|}\sum_{(x,T)\in\mathcal{P}}\|W_{\hat{F}_{\boldsymbol{\theta}}}(x,T)\|_2^2,\\ \mathcal{L}_d(\boldsymbol{\theta}) &= \frac{1}{|\mathcal{D}|}\sum_{(x,T)\in\mathcal{D}}\|\hat{F}_{\boldsymbol{\theta}}(x,T) - \bar{F}(x,T)\|_2^2.\end{aligned} \tag{26}$$

Here, $\bar{F}$ is the training data, $\hat{F}_{\boldsymbol{\theta}}$ is the prediction from the PINN, $\mathcal{P}$ and $\mathcal{D}$ are the training point sets for the physics model and external data, respectively. The loss function $\mathcal{L}$ consists of two parts, the physics model loss $\mathcal{L}_p$ and data loss $\mathcal{L}_d$. The physics model loss $\mathcal{L}_p$ measures the satisfaction of the PDE constraints for the learned output. It calculates the actual PDE equation value $W_{\hat{F}_{\boldsymbol{\theta}}}$, which is supposed to be 0, and use its 2-norm as the loss. The data loss $\mathcal{L}_d$ measures the accuracy of the prediction of PINN on the training data. It calculates the mean square error between the PINN prediction and the training data point as the loss. The overall loss function $\mathcal{L}$ is the weighted sum of the physics model loss and data loss with weighting coefficients ω_p and ω_d. In practice, we suggest setting a larger weight ω_d when sufficient data are available, and a smaller weight otherwise. Usually one can set both coefficients ω_p and ω_d close to 1 and increase the weight ω_p or ω_d if the physics model or data is more reliable.

The resulting PIPE framework combines MC data and the governing PDE into a PINN to learn the risk probability. The advantages of the PIPE framework include fast inference at test time, accurate estimation, and ability to generalize from the combination of data and model.

B. Performance Analysis

Here we provide performance analysis of PIPE. We first show that for standard neural networks (NNs) without physics model constraints, it is fundamentally difficult to estimate the risk probability of a longer time horizon than those generated from sampled trajectories. We then show that with the PINN, we are able to estimate the risk probability at any state for any time horizon with bounded error. Let Ω be the state space, $\tau = [0, T_H]$ be the time domain, $\Sigma = (\partial\Omega\times[0,T_H])\cup(\Omega\times\{0\})$ be the boundary of the space-time domain. Denote $D := \Omega\times\tau$ for notation simplicity and denote $\bar{D}$ be the interior of D.

Corollary 1. *Suppose that $D\in\mathbb{R}^{d+1}$ is a bounded domain, $u\in C^0(\bar{D})\cap C^2(D)$ is the solution to the PDE of interest, and $\tilde{u}(x,T),(x,T)\in\Sigma$ is the boundary condition. Let Σ_s be a strict sub-region in Σ, and D_s be a strict sub-region in D. Consider a neural network $F_{\boldsymbol{\theta}}$ that is parameterized by $\boldsymbol{\theta}$ and has sufficient representation capabilities. For $\forall M>0$, there can exist $\bar{\boldsymbol{\theta}}$ that satisfies both of the following conditions simultaneously:*

1) $\sup_{(x,T)\in\Sigma_s}|F_{\bar{\boldsymbol{\theta}}}(x,T) - \tilde{u}(x,T)| < \delta_1$
2) $\sup_{(x,T)\in D_s}|F_{\bar{\boldsymbol{\theta}}}(x,T) - u(x,T)| < \delta_2$

and

$$\sup_{(x,T)\in D}|F_{\bar{\boldsymbol{\theta}}}(x,T) - u(x,T)| \geq M. \tag{27}$$

Proof. See Appendix C. ■

Theorem 5. *Suppose that $D\in\mathbb{R}^{d+1}$ is a bounded domain, $u\in C^0(\bar{D})\cap C^2(D)$ is the solution to the PDE of interest, and $\tilde{u}(x,T),(x,T)\in\Sigma$ is the boundary condition. Let $F_{\boldsymbol{\theta}}$ denote a PINN parameterized by $\boldsymbol{\theta}$. If the following conditions holds:*

1) $\mathbb{E}_{\mathbf{Y}}\left[|F_{\boldsymbol{\theta}}(\mathbf{Y}) - \tilde{u}(\mathbf{Y})|\right] < \delta_1$, *where $\mathbf{Y}$ is uniformly sampled from Σ*
2) $\mathbb{E}_{\mathbf{X}}\left[|W_{F_{\boldsymbol{\theta}}}(\mathbf{X})|\right] < \delta_2$, *where $\mathbf{X}$ is uniformly sampled from D*
3) $F_{\boldsymbol{\theta}}$, $W_{F_{\boldsymbol{\theta}}}$, u are $\frac{l}{2}$ Lipshitz continuous on D.

Then the error of $F_{\boldsymbol{\theta}}$ over D is bounded by

$$\sup_{(x,T)\in D}|F_{\boldsymbol{\theta}}(x,T) - u(x,T)| \leq \tilde{\delta}_1 + C\frac{\tilde{\delta}_2}{\sigma^2} \tag{28}$$

where C is a constant depending on D, Σ and W, and

$$\begin{aligned}\tilde{\delta}_1 &= \max\left\{\frac{2\delta_1|\Sigma|}{R_\Sigma|\Sigma|}, 2l\cdot\left(\frac{\delta_1|\Sigma|\cdot\Gamma(d/2+1)}{lR_\Sigma\cdot\pi^{d/2}}\right)^{\frac{1}{d+1}}\right\},\\ \tilde{\delta}_2 &= \max\left\{\frac{2\delta_2|D|}{R_D|D|}, 2l\cdot\left(\frac{\delta_2|D|\cdot\Gamma((d+1)/2+1)}{lR_D\cdot\pi^{(d+1)/2}}\right)^{\frac{1}{d+2}}\right\},\end{aligned} \tag{29}$$

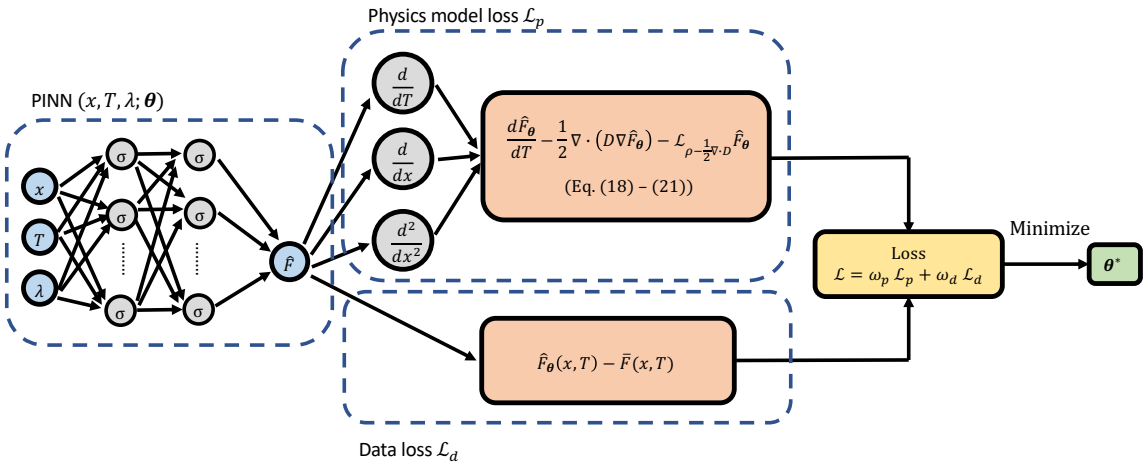


Fig. 2: The training scheme of the physics-informed neural network.

with $R_{(\cdot)}$ being the regularity of $(\cdot)$, $\|(\cdot)\|$ is the Lebesgue measure of a set $(\cdot)$ and Γ is the Gamma function.

Proof. See Appendix C. ■

Corollary 1 says that standard NN can give arbitrarily inaccurate prediction due to insufficient physical constraints. This explains why risk estimation problems cannot be handled solely on fitting sampled data. Theorem 5 says that when the PDE constraint is imposed on the full space-time domain, the prediction of the PINN has bounded error.

Also, we present the following theorem based on [45] to show that the PINN converges to the solution of the safety-related PDE when the number of training data increases under some additional regularity assumptions.

Assumption 1. *The system dynamics f, g and σ in (1) are bounded continuous functions. Also, for any $x \in \mathcal{X}$, $eig(\sigma(x)) \neq 0$ and $\sigma^{11}(x) > 0$, i.e., all eigenvalues of σ is non-zero and the first row first column entry of the noise magnitude is positive.*

Assumption 2. *Let $D \in \mathbb{R}^{d+1}$ and $\Gamma \in \mathbb{R}^d$ be the state-time space of interest and a subset of its boundary, respectively. Let μ_r and μ_b be probability distributions defined on D and Γ. Let ρ_r be the probability density of μ_r with respect to $(d+1)$-dimensional Lebesgue measure on D. Let ρ_b be the probability density of μ_b with respect to d-dimensional Hausdorff measure on Γ. We assume the following conditions.*

1) *D is at least of class $C^{0,1}$.*
2) *ρ_r and ρ_b are supported on D and Γ, respectively. Also, $\inf_D \rho_r > 0$ and $\inf_\Gamma \rho_b > 0$.*
3) *For $\epsilon > 0$, there exists partitions of D and Γ, $\left\{D_j^\epsilon\right\}_{j=1}^{K_r}$ and $\left\{\Gamma_j^\epsilon\right\}_{j=1}^{K_b}$ that depend on ϵ such that for each j, there are cubes $H_\epsilon(\mathbf{z}_{j,r})$ and $H_\epsilon(\mathbf{z}_{j,b})$ of side length ϵ centered at $\mathbf{z}_{j,r} \in D_j^\epsilon$ and $\mathbf{z}_{j,b} \in \Gamma_j^\epsilon$, respectively, satisfying $D_j^\epsilon \subset H_\epsilon(\mathbf{z}_{j,r})$ and $\Gamma_j^\epsilon \subset H_\epsilon(\mathbf{z}_{j,b})$.*
4) *There exists positive constants c_r, c_b such that $\forall \epsilon > 0$, the partitions from the above satisfy $c_r\epsilon^d \leq \mu_r\left(D_j^\epsilon\right)$ and $c_b\epsilon^{d-1} \leq \mu_b\left(\Gamma_j^\epsilon\right)$ for all j. There exists positive constants C_r, C_b such that $\forall x_r \in D$ and $\forall x_b \in \Gamma, \mu_r(B_\epsilon(x_r) \cap D) \leq C_r\epsilon^d$ and $\mu_b(B_\epsilon(x_b) \cap \Gamma) \leq C_b\epsilon^{d-1}$ where $B_\epsilon(x)$ is a closed ball of radius ϵ centered at x. Here C_r, c_r depend only on (D, μ_r) and C_b, c_b depend only on (Γ, μ_b).*
5) *When $d = 1$, we assume that all boundary points are available. Thus, no random sample is needed on the boundary.*
6) *For $x' \in \partial D$, there exists a closed ball $\bar{B}$ in $\mathbb{R}^d$ such that $\bar{B} \cap \bar{D} = \{x'\}$.*

Note that Assumption 1 is not strict for most systems.[3] Assumption 2 guarantees that random samples drawn from probability distributions can fill up both the interior of the domain D and the boundary ∂D. These are mild assumptions and can be satisfied in many practical cases. For example, let $D = (0,1)^d$, then the uniform probability distributions on both D and ∂D satisfy Assumption 2.

Theorem 6. *Suppose Assumption 1 holds, and $D \in \mathbb{R}^{d+1}$ is a bounded domain that satisfies Assumption 2, $u \in C^0(\bar{D}) \cap C^2(D)$ is the solution to the PDE of interest. Let m_r be the number of training data in D. Let F_{m_r} be the physics-informed neural networks that minimize the losses (25) with m_r training data. Assume for any m_r, architecture-wise the physics-informed neural network has enough representation power to characterize u, then*

$$\lim_{m_r \to \infty} F_{m_r} = u. \tag{30}$$

Proof. See Appendix D. ■

Till here, we have shown error bounds and convergence properties of the PIPE framework. For approximation error of PINNs in terms of neural network sizes (number of layers and number of neurons per layer), previous works on universal approximation theorem have shown that sufficiently large PINNs can approximate PDEs uniformly [24], [25], [26], [27].

[3] the noise magnitude requirement can be easily satisfied by rearranging the sequence of the state if the first state has zero noise.

VII. Experiments

We conduct four experiments to illustrate the efficacy of the proposed method. The system dynamics of interest is (1) with $X \in \mathbb{R}$, $f(X) \equiv \lambda$, $g(X) \equiv 0$ and $\sigma(X) \equiv 1$. The system dynamics become

$$dX_t = \lambda\, dt + dW_t. \tag{31}$$

The safe set is defined as (2) with $\phi(x) = x - 2$. The state-time region of interest is $\Omega \times \tau = [-10, 2] \times [0, 10]$. For risk probability, we consider the recovery probability of the system from initial state $x_0 \notin \mathcal{C}$ outside the safe set. Specifically, from section V we know that the risk probability F is characterized by the solution of the following convection diffusion equation

$$\frac{\partial F}{\partial T}(x,T) = \lambda \frac{\partial F}{\partial x}(x,T) + \frac{1}{2}\,\mathrm{tr}\left(\frac{\partial^2 F}{\partial x^2}(x,T)\right), \tag{32}$$

with initial condition $F(x,0) = \mathbb{1}(x \geq 2)$ and boundary condition $F(2,T) = 1$. We choose this system because we have the analytical solution of (32) as ground truth for comparison, as given by

$$F(x,T) = \int_0^T \frac{(2-x)}{\sqrt{2\pi t^3}} \exp\left(-\frac{((2-x)-\lambda t)^2}{2t}\right) dt. \tag{33}$$

The empirical data of the risk probability is acquired by running MC with the system dynamics (1) with initial state $x = x_0$ multiple times, and calculate ratio of trajectories with recovered state, *i.e.,*

$$\bar{F}(x,T) = \mathbb{P}(\exists t \in [0,T], x_t \in \mathcal{C} \mid x_0 = x) = \frac{N_{\text{recovery}}}{N}, \tag{34}$$

where N_{recovery} is the number of trajectories that the state recovers to the safe set during time horizon $[0,T]$, and N is the total number of sample trajectories and is a tunable parameter that affects the accuracy of the estimated risk probability. Specifically, larger N gives more accurate estimation.

In all experiments, we use PINN with 3 hidden layers and 32 neurons per layer. The activation function is chosen as hyperbolic tangent function ($\tanh$). We use Adam optimizer [46] for training with initial learning rate set as 0.001. The PINN parameters $\boldsymbol{\theta}$ is initialized via Glorot uniform initialization. The weights in the loss function (25) are set to be $\omega_p = \omega_d = 1$. We train the PINN for 60000 epochs in all experiments. The simulation is constructed based on the DeepXDE framework [47]. Experiment details, simulation results on higher-dimensional systems, and applications to stochastic safe control can be found in the Appendix. Code is available at https://github.com/jacobwang925/PIPE-L4DC.

A. Generalization to unseen regions

In this experiment, we test the generalization ability of PIPE to unseen regions of the state-time space. We consider system (31) with $\lambda = 1$. We train the PINN with data only on the sub-region of the state-time space $\Omega \times \tau = [-10, -4] \times [0, 8]$, but test the trained PINN on the full state-time region $\Omega \times \tau = [-10, 2] \times [0, 10]$. The training data is acquired through MC with sample trajectory number $N = 1000$, and is down-sampled to $dx = 0.4$ and $dT = 1$. For comparison,

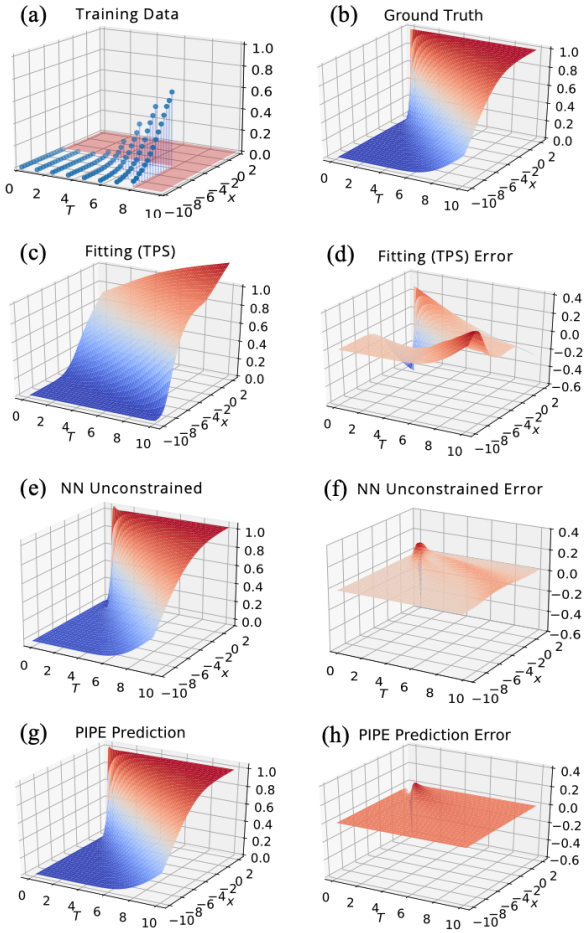


Fig. 3: Settings and results of the risk probability generalization task. Note that shaded area in (a) in the spatial-temporal space are not covered by training data. MC with fitting (TPS), neural network without PDE constraint and the proposed PIPE framework are compared. The average absolute error of prediction is 9.2×10^{-2} for TPS, and 1.5×10^{-2} for neural network without PDE constraints, 0.3×10^{-2} for PIPE.

we consider MC with fitting and neural networks without the PDE constraint. Among all fitting methods (*e.g.,* cubic spline and polynomial fitting), thin plate spline (TPS) performs the best and is used for presentation. Fig. 3 visualizes the training data samples and shows the results. The spline fitting and the standard neural network do not include any physical model constraint, thus fail to accurately generalize to unseen regions in the state space. In contrast, PIPE can infer the risk probability value very accurately in the entire state-temporal space due to the global enforcement of physics loss, which constrains the prediction to satisfy the PDE models we derived in section V.

B. Efficient estimation of risk probability

In this experiment, we show that PIPE will give more efficient estimations of risk probability in terms of accuracy and sample number compared to MC and its variants. We consider system (31) with $\lambda = 1$. The training data is sampled on the state-time space $\Omega \times \tau = [-10, 2] \times [0, 10]$ with $dx = 0.2$

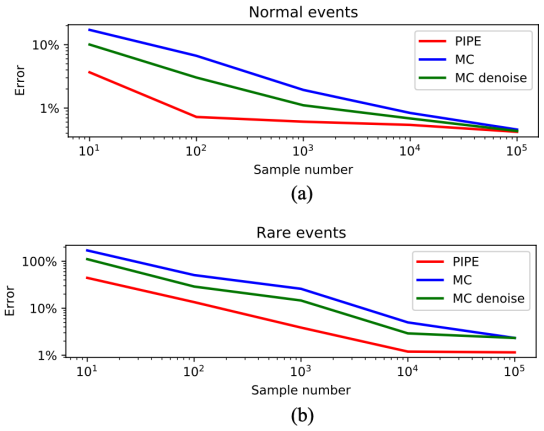


Fig. 4: Percentage error of risk probability estimation for different MC sample numbers for (a) rare events and (b) normal events. PIPE, MC and denoised MC with uniform kernel filtering are compared. Both error and sample number are in log scale.

and $dT = 0.1$. We compare the risk probability estimation error of PIPE and MC, on two regions in the state-time space:

1) Normal event region: $\Omega \times \tau = [-6, -2] \times [4, 6]$, where the average probability is 0.412.
2) Rare event region: $\Omega \times \tau = [-2, 0] \times [8, 10]$, where the average probability is 0.985.

For fairer comparison, we use a uniform filter of kernel size 3 on the MC data to smooth out the noise, as the main cause of inaccuracy of MC estimation is sampling noise. Fig. 4 shows the percentage errors of risk probability inference under different MC sample numbers N. As the sample number goes up, prediction errors for all three approaches decrease. The denoised MC has lower error compared to standard MC as a result of denoising, and their errors tend to converge since the sampling noise contributes less to the error as the sample number increases. On both rare events and normal events, PIPE yields more accurate estimation than MC and denoised MC across all sample numbers. This indicates that PIPE has better sample efficiency than MC and its variants, as it requires less sample data to achieve the same prediction accuracy. This desired feature of PIPE is due to the fact that it incorporates model knowledge into the MC data to further enhance its accuracy by taking the physics-informed neighboring relationships of the data into consideration.

The computation time for different methods are reported in Table I. It can be seen that, although PIPE requires offline training, it offers fast online inference, enabling real-time evaluation for risk probabilities. In addition, the training time is independent of the precision of the training data. MC has no training phase, but incurs a significant online cost when the number of samples is high to achieve the desired accuracy.

C. Adaptation on changing system parameters

In this experiment, we show that PIPE will allow generalization to uncertain parameters of the system. We consider system (31) with varying $\lambda \in [0, 2]$. We use MC data with sample number $N = 10000$ for a fixed set of $\lambda_{\text{train}} = [0.1, 0.5, 0.8, 1]$ for training, and test PIPE after training on $\lambda_{\text{test}} = [0.3, 0.7, 1.2, 1.5, 2]$. Fig. 5 visualizes the results testing instance $\lambda_{\text{test}} = 1.5$. We can see that PIPE is capable of accurately predicting risk probability for systems with unseen and out-of-distribution parameters beyond training, while a neural network without PDE constraints fails to do so. In the prediction error plot, the only region where PIPE has larger prediction error is at $T = 0$ and $x \in \partial\mathcal{C}$ on the boundary of the safe set. This is because the risk probability at this point is not well defined (it can be either 0 or 1), and this point will not be considered in a control scenario as we are always interested in long-term safety where $T \gg 0$. This adaptation feature of the PIPE framework indicates its potential use on stochastic safe control with uncertain system parameters, and it also opens the door for physics-informed learning on a family of PDEs. In general, PDEs with different parameters can have qualitatively different behaviors, so is hard to generalize. The control theory model allows us to have a sense when the PDEs are qualitatively similar with different parameters, and thus allows generalization within the qualitatively similar cases.

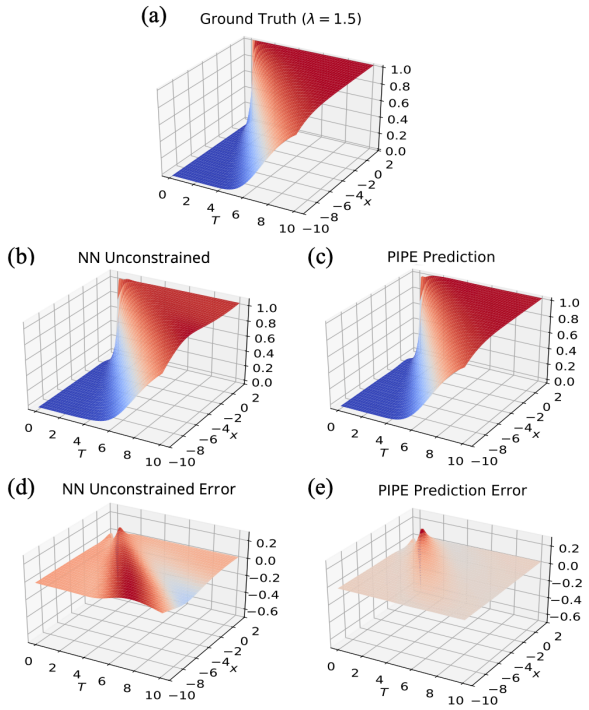


Fig. 5: Risk probability prediction of unconstrained neural network and PIPE on unseen system parameters $\lambda_{\text{test}} = 1.5$. The average absolute prediction error across all $\lambda_{\text{test}} = [0.3, 0.7, 1.2, 1.5, 2]$ is 2.23×10^{-2} for unconstrained neural network, and 0.60×10^{-2} for PIPE.

Method	Offline (Training)	Online (Inference)
PIPE (GPU, L4)	62.26 s	0.0013 s
PIPE (CPU)	262.79 s	0.0012 s
MC (10^1 samples)	N/A	0.0026 s
MC (10^5 samples)	N/A	2.53 s

TABLE I: Comparison of computation time for different methods. The inference time is calculated for evaluating risk probability of one initial state with fixed time horizon $T = 10$.

D. Estimating the gradient of risk probability

In this experiment, we show that PIPE is able to generate accurate gradient predictions of risk probabilities. We consider system (31) with $\lambda = 1$. Similar to the generalization task, we train the PINN with MC data of $N = 1000$ on the sub-region $\Omega \times \tau = [-10, -4] \times [0, 8]$ and test the trained PINN on the full state-time region $\Omega \times \tau = [-10, 2] \times [0, 10]$. We then take the finite difference of the risk probability with regard to the state x to calculate its gradient, for ground truth F, MC estimation $\bar{F}$ and PIPE prediction $\hat{F}_{\boldsymbol{\theta}}$ with and without PDE constraints.[4] Fig. 6 shows the results. It can be seen that the MC gradient estimate is rather noisy, and the unconstrained NN does not provide accurate gradient estimation beyond the region of training samples. In contrast, PIPE gives a much more accurate gradient estimation, because it incorporates the physics model information inside the training process. It is also worth noting that PIPE does not use any governing laws of the risk probability gradient during training, and by considering the risk probability PDE alone, it can provide very accurate estimations of the gradient. The results indicate that PIPE can enable the usage of a lot of first- and higher-order stochastic safe control methods online, by providing accurate and fast estimation of the risk probability gradients.

VIII. Conclusion

In this paper, we propose physics-informed probability estimator (PIPE), a framework to efficiently and accurately estimate long-term risk probabilities in stochastic systems. We first characterize the exact probability distributions of the minimum and maximum barrier function values in a time interval and the first entry and exit times to and from its super level sets. The distributions of these variables are then used to characterize the probability of safety and recovery, the safety margin, and the mean and tail distributions of the failure and recovery times into the solution of certain partial differential equations (PDEs). We then propose a physics-informed learning approach to solve the corresponding PDEs and efficiently estimate the risk probabilities. The method combines data from rare event simulation and the underlying governing PDE we derived, to accurately learn the risk probability as well as its gradient on the entire state-temporal domain of interest. Experiments show better sample efficiencies of the physics-informed learning framework compared to MC, and the ability to generalize to unseen regions in the state-temporal space beyond training. The framework is also robust to uncertain parameters in the system dynamics, and can infer risk probability values of a class of systems with training data only from a fixed number of systems, which provides key foundations for first- and higher-order methods for stochastic safe control. Future work includes extending the current framework to inverse learning [14] and Bayesian learning [21] settings, to account for systems with unknown dynamics where the PDEs for risk characterization are uncertain, and to real-world high-dimensional systems [49], [50].

[4]Note that for PIPE, automatic differentiation [48] can also be used to obtain risk probability gradients, and we did not find qualitative difference with the finite difference methods for this specific experiment.

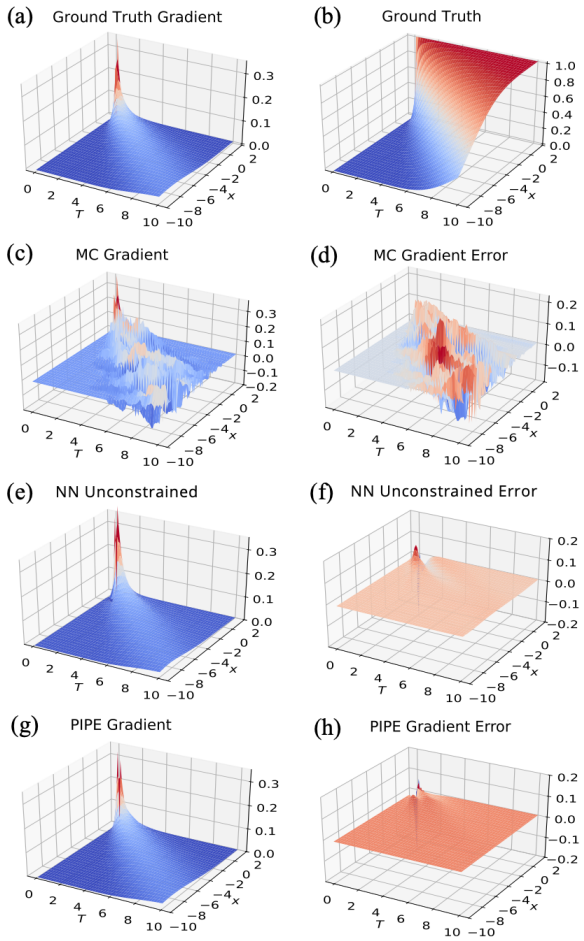


Fig. 6: Gradient of the risk probability prediction of MC, unconstrained NN and PIPE. The average absolute error of gradient prediction is 2.78×10^{-2} for MC, 0.17×10^{-2} for NN without constraints, and 0.06×10^{-2} for PIPE.

References

[1] A. J. McNeil, R. Frey, and P. Embrechts, *Quantitative risk management: concepts, techniques and tools-revised edition*. Princeton university press, 2015.

[2] S. Prajna, A. Jadbabaie, and G. J. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415–1428, 2007.

[3] S. Yaghoubi, K. Majd, G. Fainekos, T. Yamaguchi, D. Prokhorov, and B. Hoxha, "Risk-bounded control using stochastic barrier functions," *IEEE Control Systems Letters*, 2020.

[4] C. Santoyo, M. Dutreix, and S. Coogan, "A barrier function approach to finite-time stochastic system verification and control," *Automatica*, vol. 125, p. 109439, 2021.

[5] R. Cheng, M. J. Khojasteh, A. D. Ames, and J. W. Burdick, "Safe multi-agent interaction through robust control barrier functions with learned uncertainties," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 777–783.

[6] S. Tong, A. Subramanyam, and V. Rao, "Optimization under rare chance constraints," *SIAM Journal on Optimization*, vol. 32, no. 2, pp. 930–958, 2022.

[7] G. Rubino and B. Tuffin, *Rare event simulation using Monte Carlo methods*. John Wiley & Sons, 2009.

[8] H. Janssen, "Monte-carlo based uncertainty analysis: Sampling efficiency and sampling convergence," *Reliability Engineering & System Safety*, vol. 109, pp. 123–132, 2013.

[9] F. Cérou, P. Del Moral, T. Furon, and A. Guyader, "Sequential monte carlo for rare event estimation," *Statistics and computing*, vol. 22, no. 3, pp. 795–808, 2012.

[10] F. Uribe, I. Papaioannou, Y. M. Marzouk, and D. Straub, "Cross-entropy-based importance sampling with failure-informed dimension reduction for rare event simulation," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 9, no. 2, pp. 818–847, 2021.

[11] D. T. Yang, A. M. Goldberg, and L. T. Chong, "Rare-event sampling using a reinforcement learning-based weighted ensemble method," *bioRxiv*, 2024.

[12] M. Ehre, I. Papaioannou, and D. Straub, "Stein variational rare event simulation," *arXiv preprint arXiv:2308.04971*, 2023.

[13] K. Zuev, "Subset simulation method for rare event estimation: an introduction," *arXiv preprint arXiv:1505.03506*, 2015.

[14] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational physics*, vol. 378, pp. 686–707, 2019.

[15] J. Han, A. Jentzen, and W. E, "Solving high-dimensional partial differential equations using deep learning," *Proceedings of the National Academy of Sciences*, vol. 115, no. 34, pp. 8505–8510, 2018.

[16] S. Cuomo, V. S. Di Cola, F. Giampaolo, G. Rozza, M. Raissi, and F. Piccialli, "Scientific machine learning through physics–informed neural networks: Where we are and what's next," *Journal of Scientific Computing*, vol. 92, no. 3, p. 88, 2022.

[17] M. Pereira, Z. Wang, I. Exarchos, and E. Theodorou, "Safe optimal control using stochastic barrier functions and deep forward-backward sdes," in *Conference on Robot Learning*. PMLR, 2021, pp. 1783–1801.

[18] G. S. Misyris, A. Venzke, and S. Chatzivasileiadis, "Physics-informed neural networks for power systems," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 2020, pp. 1–5.

[19] S. Cai, Z. Mao, Z. Wang, M. Yin, and G. E. Karniadakis, "Physics-informed neural networks (pinns) for fluid mechanics: A review," *Acta Mechanica Sinica*, pp. 1–12, 2022.

[20] F. Sahli Costabal, Y. Yang, P. Perdikaris, D. E. Hurtado, and E. Kuhl, "Physics-informed neural networks for cardiac activation mapping," *Frontiers in Physics*, vol. 8, p. 42, 2020.

[21] L. Yang, X. Meng, and G. E. Karniadakis, "B-pinns: Bayesian physics-informed neural networks for forward and inverse pde problems with noisy data," *Journal of Computational Physics*, vol. 425, p. 109913, 2021.

[22] L. Lu, R. Pestourie, W. Yao, Z. Wang, F. Verdugo, and S. G. Johnson, "Physics-informed neural networks with hard constraints for inverse design," *SIAM Journal on Scientific Computing*, vol. 43, no. 6, pp. B1105–B1132, 2021.

[23] K. Shukla, A. D. Jagtap, and G. E. Karniadakis, "Parallel physics-informed neural networks via domain decomposition," *Journal of Computational Physics*, vol. 447, p. 110683, 2021.

[24] J. Sirignano and K. Spiliopoulos, "Dgm: A deep learning algorithm for solving partial differential equations," *Journal of computational physics*, vol. 375, pp. 1339–1364, 2018.

[25] P. Grohs, F. Hornung, A. Jentzen, and P. Von Wurstemberger, *A proof that artificial neural networks overcome the curse of dimensionality in the numerical approximation of Black–Scholes partial differential equations*. American Mathematical Society, 2023, vol. 284, no. 1410.

[26] J. Darbon, G. P. Langlois, and T. Meng, "Overcoming the curse of dimensionality for some hamilton–jacobi partial differential equations via neural network architectures," *Research in the Mathematical Sciences*, vol. 7, no. 3, p. 20, 2020.

[27] J. Darbon and T. Meng, "On some neural network architectures that can represent viscosity solutions of certain high dimensional hamilton–jacobi partial differential equations," *Journal of Computational Physics*, vol. 425, p. 109907, 2021.

[28] T. De Ryck and S. Mishra, "Error analysis for physics-informed neural networks (pinns) approximating kolmogorov pdes," *Advances in Computational Mathematics*, vol. 48, no. 6, p. 79, 2022.

[29] S. Mishra and R. Molinaro, "Estimates on the generalization error of physics-informed neural networks for approximating pdes," *IMA Journal of Numerical Analysis*, vol. 43, no. 1, pp. 1–43, 2023.

[30] ——, "Estimates on the generalization error of physics-informed neural networks for approximating a class of inverse problems for pdes," *IMA Journal of Numerical Analysis*, vol. 42, no. 2, pp. 981–1022, 2022.

[31] Y. Qian, Y. Zhang, Y. Huang, and S. Dong, "Physics-informed neural networks for approximating dynamic (hyperbolic) pdes of second order in time: Error analysis and algorithms," *Journal of Computational Physics*, vol. 495, p. 112527, 2023.

[32] Z. Fang, "A high-efficient hybrid physics-informed neural networks based on convolutional neural network," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 10, pp. 5514–5526, 2021.

[33] G. Pang, L. Lu, and G. E. Karniadakis, "fpinns: Fractional physics-informed neural networks," *SIAM Journal on Scientific Computing*, vol. 41, no. 4, pp. A2603–A2626, 2019.

[34] Y. Jiao, Y. Lai, D. Li, X. Lu, F. Wang, Y. Wang, and J. Z. Yang, "A rate of convergence of physics informed neural networks for the linear second order elliptic pdes," *arXiv preprint arXiv:2109.01780*, 2021.

[35] B. Øksendal, *Stochastic Differential Equations: An Introduction with Applications*, 6th ed., ser. Universitext. Berlin Heidelberg: Springer-Verlag, 2003.

[36] A. N. Borodin, *Stochastic processes*. Springer, 2017.

[37] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH journal*, vol. 1, no. 1, pp. 1–14, 2014.

[38] D. Ferguson, M. Darms, C. Urmson, and S. Kolski, "Detection, prediction, and avoidance of dynamic obstacles in urban environments," in *2008 IEEE Intelligent Vehicles Symposium*, 2008, pp. 1149–1154.

[39] H. Khali, "Adaptive output feedback control of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 41, pp. 177–188, 1996.

[40] M. Ono, M. Pavone, Y. Kuwata, and J. Balaram, "Chance-constrained dynamic programming with application to risk-aware robotic space exploration," *Autonomous Robots*, vol. 39, pp. 555–571, 2015.

[41] Z. Wang, H. Jing, C. Kurniawan, A. Chern, and Y. Nakahira, "Myopically verifiable probabilistic certificates for safe control and learning," *arXiv preprint arXiv:2404.16883*, 2024.

[42] A. Wachi, W. Hashimoto, and K. Hashimoto, "Long-term safe reinforcement learning with binary feedback," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 19, 2024, pp. 21 656–21 663.

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[44] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[45] Y. Shin, J. Darbon, and G. E. Karniadakis, "On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type pdes," *arXiv preprint arXiv:2004.01806*, 2020.

[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[47] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "Deepxde: A deep learning library for solving differential equations," *SIAM Review*, vol. 63, no. 1, pp. 208–228, 2021.

[48] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, "Automatic differentiation in machine learning: a survey," *Journal of machine learning research*, vol. 18, no. 153, pp. 1–43, 2018.

[49] K. Hoshino, Z. Wang, and Y. Nakahira, "Scalable long-term safety certificate for large-scale systems," *IEEE Control Systems Letters*, 2023.

[50] Z. Wang, R. Keller, X. Deng, K. Hoshino, T. Tanaka, and Y. Nakahira, "Physics-informed representation and learning: Control and risk quantification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 19, 2024, pp. 21 699–21 707.

[51] H. Pham, *Continuous-time stochastic control and optimization with financial applications*. Springer Science & Business Media, 2009, vol. 61.

[52] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[53] W. Peng, W. Zhou, J. Zhang, and W. Yao, "Accelerating physics-informed neural network training with prior dictionaries," *arXiv preprint arXiv:2004.08151*, 2020.

[54] D. Gilbarg, N. S. Trudinger, D. Gilbarg, and N. Trudinger, *Elliptic partial differential equations of second order*. Springer, 1977, vol. 224, no. 2.

[55] W. S. Cleveland, "Lowess: A program for smoothing scatterplots by robust locally weighted regression," *American Statistician*, vol. 35, no. 1, p. 54, 1981.

[56] Z. Wang, H. Jing, C. Kurniawan, A. Chern, and Y. Nakahira, "Myopically verifiable probabilistic certificate for long-term safety," in *2022 American Control Conference (ACC)*. IEEE, 2022, pp. 4894–4900.

APPENDIX

A. Proof of Theorem 1 and Theorem 3

The proof of Theorem 1 and Theorem 3 is based on the lemmas presented below. These lemmas relate the distributions of various functionals of a diffusion process $X_t \in \mathbb{R}^n, t \in \mathbb{R}_+$ with deterministic convection-diffusion equations. Let $X_t \in \mathbb{R}^n, t \in \mathbb{R}_+$ be a solution of the SDE

$$dX_t = \mu(X_t)dt + \sigma(X_t)dW_t, \tag{35}$$

We assume sufficient regularity in the coefficients of (35) such that (35) has a unique solution, similar to the assumptions on (1).

We first present a variant of the Feynman–Kac Representation [51, Theorem 1.3.17].

Lemma 1 (Feynman–Kac Representation)**.** *Consider the diffusion process* (35) *with* $X_0 = x \in \mathbb{R}^n$*. Let* $V : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$*,* $\beta : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ *and* $\psi\colon \mathbb{R}^n \to \mathbb{R}$ *be some given functions. The solution of the Cauchy problem*

$$\begin{cases} \frac{\partial U}{\partial T} = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U - VU + \beta, & T > 0, \\ U(x,0) = \psi(x) \end{cases} \tag{36}$$

is given by

$$\begin{aligned} U(x,T) = \mathbb{E}_x \Bigg[& e^{-\int_0^T V(X_t, T-t)\,dt} \psi(X_T) \\ & + \int_0^T e^{-\int_0^\tau V(X_t, T-t)\,dt} \beta(X_\tau, T-\tau)\,d\tau \Bigg]. \end{aligned} \tag{37}$$

While the original statement given in [51, Theorem 1.3.17] presents U as a function of t that moves backward in time, we present U as a function of T that moves forward in time. Even though both approaches express the time derivative with opposite signs, both yield the same probability. The representation in Lemma 1 can be used to derive the following result.

Lemma 2. *Let* $\mathcal{M} \subset \mathbb{R}^n$ *be a domain. Consider the diffusion process* (35) *with the initial condition* $X_0 = x \in \mathbb{R}^n$*. Then the function* $U : \mathbb{R}^n \times [0,\infty) \to \mathbb{R}$ *defined by*

$$U(x,T) := \mathbb{P}_x(X_t \in \mathcal{M}, \forall t \in [0,T]) \tag{38}$$

is the solution to the convection-diffusion equation

$$\begin{cases} \frac{\partial U}{\partial T} = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U, & x \in \mathcal{M}, T > 0, \\ U(x,T) = 0, & x \notin \mathcal{M}, T > 0, \\ U(x,0) = \mathbb{1}_{\mathcal{M}}(x), & x \in \mathbb{R}^n. \end{cases} \tag{39}$$

Proof. Note that Lemma 1 holds for any given functions $V : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$, $\beta : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ and $\psi\colon \mathbb{R}^n \to \mathbb{R}$. We apply Lemma 1 with

$$\beta(x,t) \equiv 0,\ \psi(x) = \mathbb{1}_{\mathcal{M}}(x),\ V(x,t) = \gamma \mathbb{1}_{\mathcal{M}^c}(x),$$

for any $\gamma > 0$, where $\mathcal{M}^c$ is the complement of $\mathcal{M}$. Then we have

$$U(x,T) = \mathbb{E}_x \left[e^{-\int_0^T \gamma \mathbb{1}_{\mathcal{M}^c}(X_t)\,dt} \psi(X_T) \right]. \tag{40}$$

and

$$\begin{cases} \frac{\partial U}{\partial T} = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U, & x \in \mathcal{M}, T > 0, \\ \frac{\partial U}{\partial T} = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U - \gamma U, & x \notin \mathcal{M}, T > 0, \\ U(x,0) = \mathbb{1}_{\mathcal{M}}(x), & x \in \mathbb{R}^n. \end{cases} \tag{41}$$

Now, we take the limit of $\gamma \to \infty$. Since

$$\lim_{\gamma\to\infty} e^{-\int_0^T \gamma \mathbb{1}_{\mathcal{M}^c}(X_t)\,dt} = \begin{cases} 0, & \text{if } X_t \notin \mathcal{M}, \exists t \in [0,T], \\ 1, & \text{otherwise.} \end{cases},$$

(40) becomes

$$\begin{aligned} U(x,T) &= \mathbb{E}_x[\mathbb{1}_{\{X_t \in \mathcal{M}, \forall t \in [0,T]\}}] && (42) \\ &= \mathbb{P}_x(X_t \in \mathcal{M}, \forall t \in [0,T]). && (43) \end{aligned}$$

Meanwhile, under the limit of $\gamma \to \infty$, the part of (41) with $x \notin \mathcal{M}$ and $T > 0$ reduces to the algebraic condition

$$U(x,T) = 0, \quad x \notin \mathcal{M}, T > 0. \tag{44}$$

Combining (41), (42), and (44) and gives Lemma 2. ∎

Using Lemma 2, we can prove Theorem 1 and Theorem 3 as follows.

Proof. (Theorem 1) Consider the augmented space space $Z_t = [\phi(X_t), X_t^\intercal]^\intercal \in \mathbb{R}^{n+1}$. The stochastic process Z_t is the solution to (18) with parameters ρ and ζ defined in (19) with the initial state

$$Z_0 = z = [\phi(x), x^\intercal]^\intercal. \tag{45}$$

Let us define

$$\mathcal{M} = \{z \in \mathbb{R}^{n+1} : z[1] \geq \ell\}. \tag{46}$$

From Lemma 2,

$$F(x,T;\ell) = \mathbb{P}_x(z_t \in \mathcal{M}, \forall t \in [0,T])$$

is the solution to the convection-diffusion equation

$$\begin{cases} \frac{\partial F}{\partial T} = \frac{1}{2}\operatorname{tr}(\zeta\zeta^\intercal \operatorname{Hess} F) + \mathcal{L}_\rho F, & z \in \mathcal{M}, T > 0, \\ F(z,T) = 0, & z \notin \mathcal{M}, T > 0, \\ F(z,0) = \mathbb{1}_{\mathcal{M}}(z), & z \in \mathbb{R}^{n+1}. \end{cases} \tag{47}$$

To obtain (20.B), define $D = \zeta\zeta^\intercal$ and apply the vector identity[5]

$$\nabla \cdot (D\nabla F) = \mathcal{L}_{\nabla \cdot D} F + \operatorname{tr}(D \operatorname{Hess} F). \tag{48}$$

∎

Proof. (Theorem 3) Consider the augmented space of $Z_t = [\phi(X_t), X_t^\intercal]^\intercal \in \mathbb{R}^{n+1}$ in (16). The stochastic process Z_t is a solution of (18) with parameters ρ and ζ defined in (19) with the initial state

$$Z_0 = z = [\phi(x), x^\intercal]^\intercal. \tag{49}$$

Let us define

$$\mathcal{M} = \{z \in \mathbb{R}^{n+1} : z[1] < \ell\}. \tag{50}$$

[5]The identity is a direct consequence of the Leibniz rule $\partial_i(D_{ij}\partial_j F) = (\partial_i D_{ij})\partial_j F + D_{ij}\partial_i\partial_j F$.

The CDF of $\Theta_x(T)$ is given by

$$\mathbb{P}(\Theta_x(T) < \ell) = \mathbb{P}_x(\forall t \in [0,T], \phi(X_t) < \ell) \tag{51}$$
$$= \mathbb{P}_x(\forall t \in [0,T], Z_t \in \mathcal{M}), \tag{52}$$

which, by Lemma 2, is the solution to the convection-diffusion equation (39), yielding (22.B) after the application of identity (48). ■

B. Proof of Theorem 2 and Theorem 4

We first present Lemma 3 and Lemma 4, which link the escape time of a diffusion process to PDE solutions via a Feynman–Kac formulation, and then use these lemmas to prove Theorem 2 and Theorem 4.

Lemma 3. *Consider the diffusion process* (35) *with the initial point* $X_0 = x$. *Define the escape time as* (68). *Let* $\psi(x), V(x)$ *be continuous functions and* V *be non-negative. If* $U : \mathbb{R}^n \to \mathbb{R}$ *is the bounded solution to the boundary value problem*

$$\begin{cases} \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U - VU = 0, & x \in \mathcal{M}, \\ U(x) = \psi(x), & x \notin \mathcal{M}, \end{cases} \tag{53}$$

then

$$U(x) = \mathbb{E}_x\left[\psi(X_{H_\mathcal{M}}) e^{-\int_0^{H_\mathcal{M}} V(X_s)\,ds}\right]. \tag{54}$$

Proof. We first define a mapping $\eta : \mathbb{R} \to \mathbb{R}$ by

$$\eta(s) := U(X_s) e^{-\int_0^s V(X_v)\,dv}. \tag{55}$$

It satisfies

$$\eta(q) = \int_0^q d\eta(s) + \eta(0) \tag{56}$$
$$= \int_0^q e^{-\int_0^s V(X_v)\,dv}[-V(X_s)U(X_s) + \mathcal{L}_\mu U(X_s) + \tfrac{1}{2}\operatorname{tr}(\sigma(X_s)\sigma^\intercal(X_s)\operatorname{Hess} U(X_s))]ds \tag{57}$$
$$+ \int_0^q e^{-\int_0^s V(X_v)\,dv}\mathcal{L}_\sigma U(X_s)\,dW_s + \eta(0)$$
$$= \int_0^q e^{-\int_0^s V(X_v)\,dv}\mathcal{L}_\sigma U(X_s)\,dW_s + \eta(0), \tag{58}$$

where (57) is from Itô's Lemma; (58) is from (53) with $x \in \mathcal{M}$. Thus, its expectation satisfies

$$\mathbb{E}_x[\eta(q)] = \mathbb{E}_x[\eta(0)] \tag{59}$$
$$= \mathbb{E}_x[U(X_0)] \tag{60}$$
$$= U(x). \tag{61}$$

where (59) holds because the right hand side of (58) has zero mean; (60) is due to (55); and (61) is from the assumption that $X_0 = x$.

Next, we set

$$q = H_\mathcal{M}. \tag{62}$$

As (62) implies $X_q \notin \mathcal{M}$, condition (53) with $x \notin \mathcal{M}$ yields

$$U(X_q) = \psi(X_q). \tag{63}$$

Finally, we have

$$U(x) = \mathbb{E}_x[\eta(q)] \tag{64}$$
$$= \mathbb{E}_x\left[U(X_q) e^{-\int_0^q V(X_v)\,dv}\right] \tag{65}$$
$$= \mathbb{E}_x\left[\psi(X_q) e^{-\int_0^q V(X_v)\,dv}\right] \tag{66}$$
$$= \mathbb{E}_x\left[\psi(X_{H_\mathcal{M}}) e^{-\int_0^{H_\mathcal{M}} V(X_v)\,dv}\right], \tag{67}$$

where (64) is due to (61); (65) is due to (55); (66) is due to (63), and (67) is due to (62).

■

Lemma 4. *Consider the diffusion process* (35) *with the initial condition* $X_0 = x \in \mathbb{R}^n$. *Define the escape time,*

$$H_\mathcal{M} := \inf\{t \in \mathbb{R}_+ : X_t \notin \mathcal{M}\}. \tag{68}$$

The CDF of the escape time

$$U(x,t) = \mathbb{P}_x(H_\mathcal{M} \le t), \quad t > 0, \tag{69}$$

is the solution to

$$\begin{cases} \frac{\partial U}{\partial t}(x,t) = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U, & x \in \mathcal{M}, t > 0, \\ U(x,t) = 1, & x \notin \mathcal{M}, t > 0, \\ U(x,0) = \mathbb{1}_{\mathcal{M}^c}(x), & x \in \mathbb{R}^n. \end{cases} \tag{70}$$

Proof. Let $\gamma \in \mathbb{C}$ be a spectral parameter with $\operatorname{Re}(\gamma) > 0$. For each fixed γ, let $\hat{U}(\cdot,\gamma) : \mathbb{R}^n \to \mathbb{C}$ be the solution of

$$\begin{cases} \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} \hat{U}) + \mathcal{L}_\mu \hat{U} - \gamma \hat{V}\hat{U} = 0, & x \in \mathcal{M}, \\ \hat{U}(x,\gamma) = \hat{\psi}(x,\gamma), & x \notin \mathcal{M}. \end{cases} \tag{71}$$

According to Lemma 3, $\hat{U}(x,\gamma)$ is given by

$$\hat{U}(x,\gamma) = \mathbb{E}_x\left[\hat{\psi}(X_{H_\mathcal{M}}) e^{-\gamma\int_0^{H_\mathcal{M}} \hat{V}(X_s)\,ds}\right], \tag{72}$$

where X_s is the diffusion process in (35). Now, take

$$\hat{\psi}(x,\gamma) = 1/\gamma, \quad \hat{V}(x) = 1, \tag{73}$$

in (71) and (72). Then, (71) becomes

$$\begin{cases} \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} \hat{U}) + \mathcal{L}_\mu \hat{U} - \gamma \hat{U} = 0, & x \in \mathcal{M}, \\ \hat{U}(x,\gamma) = 1/\gamma, & x \notin \mathcal{M}, \end{cases} \tag{74}$$

and (72) becomes

$$\hat{U}(x,\gamma) = \mathbb{E}_x\left[\frac{1}{\gamma} e^{-\gamma H_\mathcal{M}}\right]$$
$$= \int_0^\infty \frac{1}{\gamma} e^{-\gamma t} p_{H_\mathcal{M}|X_0}(t \mid x)\,dt$$
$$= \int_0^\infty e^{-\gamma t}\mathbb{P}_x(H_\mathcal{M} \le t)\,dt \tag{75}$$

where integration by parts is used in (75). Here, $p_{H_\mathcal{M}|X_0}(t|x) = \frac{d}{dt}\mathbb{P}_x(H_\mathcal{M} \le t)$ denotes the probability density function of $H_\mathcal{M}$ conditioned on $X_0 = x$.

Now, for fixed x, let $U(x,t)$ be the inverse Laplace transformation of $\hat{U}(x,\gamma)$. In other words,

$$\hat{U}(x,\gamma) = \int_0^\infty U(x,t) e^{-\gamma t}\,dt. \tag{76}$$

On one hand, comparing (75) and (76) gives

$$U(x,t) = \mathbb{P}_x\left(H_{\mathcal{M}} \le t\right). \tag{77}$$

On the other hand, taking the inverse Laplace transformation of the PDE (74) for $\hat{U}$ yields the PDE satisfied by U:

$$\begin{cases} \frac{\partial U}{\partial t}(x,t) = \frac{1}{2}\operatorname{tr}(\sigma\sigma^\intercal \operatorname{Hess} U) + \mathcal{L}_\mu U, & x \in \mathcal{M}, t > 0, \\ U(x,t) = 1, & x \notin \mathcal{M}, t > 0, \\ U(x,0) = \mathbb{1}_{\mathcal{M}^c}(x), & x \in \mathbb{R}^n. \end{cases} \tag{78}$$

■

Now, we are ready to prove Theorem 2 and Theorem 4.

Proof. (Theorem 2) Let the augmented state Z_t be the solution of (18) with the initial state $Z_0 = z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. Let

$$\mathcal{M} = \{z \in \mathbb{R}^{n+1} : z[1] \ge \ell\}. \tag{79}$$

Thus, we have $\Gamma_x(\ell) = H_{\mathcal{M}}$, for G be the solution of the convection-diffusion equation with $D = \zeta\zeta^\intercal$

$$\begin{cases} \frac{\partial G}{\partial t} = \frac{1}{2}\nabla\cdot(D\nabla G) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D} G, & z \in \mathcal{M}, t > 0, \\ G(z,t) = 1, & z \notin \mathcal{M}, t > 0. \\ G(z,0) = \mathbb{1}_{\mathcal{M}^c}(z), & z \in \mathbb{R}^{n+1}, \end{cases}$$

which is equivalent to (21.B). From Lemma 4, we have $G(z,t;l) = \mathbb{P}(\Gamma_x(l) \le t)$. ■

Proof. (Theorem 4) Let the augmented state space Z_t be the solution of (18) with the initial state $Z_0 = z = [\phi(x), x^\intercal]^\intercal \in \mathbb{R}^{n+1}$. Let

$$\mathcal{M} = \{z \in \mathbb{R}^{n+1} : z[1] < \ell\}. \tag{80}$$

We have $\Psi_x(\ell) = H_{\mathcal{M}}$ for N be the solution of the convection diffusion equation with $D = \zeta\zeta^\intercal$

$$\begin{cases} \frac{\partial N}{\partial t}(z,t) = \frac{1}{2}\nabla\cdot(D\nabla N) + \mathcal{L}_{\rho - \frac{1}{2}\nabla\cdot D} N, & z \in \mathcal{M}, t > 0, \\ N(x,t) = 1, & z \notin \mathcal{M}, t > 0, \\ N(x,0) = \mathbb{1}_{\mathcal{M}^c}(z), & z \in \mathbb{R}^{n+1}, \end{cases}$$

which is equivalent to (23.B). From Lemma 4, we have $N(z,T;l) = \mathbb{P}(\Psi_x(l) \le T)$. ■

C. Proof of Theorems for Physics-informed Learning

Proof. (Corollary 1) We know that u is the solution to the PDE of interest. We can construct $\bar{u}$ such that

$$\bar{u} = \begin{cases} u, & (x,T) \in D_s \\ u + \frac{d}{d_{\max}}(M+\delta), & (x,T) \in \Omega\times\tau \backslash D_s \end{cases} \tag{81}$$

where d characterizes the distance between $(x,t) \in \Omega\backslash D_s$ and set D_s, $d_{\max} = \max_{(x,t)\in\Omega\backslash D_s} d$ is the maximum distance, and $0 < \delta < \min\{\delta_1, \delta_2\}$ is a constant. Then we have

$$\sup_{(x,T)\in\Omega\times\tau} |\bar{u}(x,t) - u(x,t)| = M + \delta \tag{82}$$

Since we assume the neural network has sufficient representation power, by universal approximation theorem [52], for δ given above, there exist $\bar{\boldsymbol{\theta}}$ such that

$$\sup_{(x,T)\in\Omega\times\tau} |F_{\bar{\boldsymbol{\theta}}}(x,t) - \bar{u}(x,t)| \le \delta. \tag{83}$$

Then we have $\bar{\boldsymbol{\theta}}$ satisfies

$$\begin{aligned} \sup_{(x,T)\in\Sigma_s} |F_{\bar{\boldsymbol{\theta}}}(x,T) - \tilde{u}(x,T)| &< \delta_1, \\ \sup_{(x,T)\in D_s} |F_{\bar{\boldsymbol{\theta}}}(x,T) - u(x,T)| &< \delta_2, \end{aligned} \tag{84}$$

and

$$\sup_{(x,T)\in\Omega\times\tau} |F_{\bar{\boldsymbol{\theta}}}(x,t) - u(x,t)| \ge M. \tag{85}$$

■

The proof of Theorem 5 is based on [53], [54] by adapting the results on elliptic PDEs to parabolic PDEs. We first give some supporting definitions and lemmas. We define the second order parabolic operator L w.r.t. u as follows.

$$\begin{aligned} L[u] := &-\frac{\partial u}{\partial T} + \sum_{i,j} a_{i,j}(x,T)\frac{\partial^2 u}{\partial x_i x_j} \\ &+ \sum_i b_i(x,T)\frac{\partial u}{\partial x_i} + c(x,T)u. \end{aligned} \tag{86}$$

Let $A(x) := [a_{i,j}(x,T)]_{i,j} \in \mathbb{R}^{d\times d}$ and $b(x,T) := [b_i(x)]_i \in \mathbb{R}^d$. We consider uniform parabolic operators, where A is positive definite, with the smallest eigenvalue $\lambda > 0$.

Let $\Omega \times \tau \in \mathbb{R}^{d+1}$ be a bounded domain of interest, where Σ is the boundary of $\Omega\times\tau$. We consider the following partial differential equation for the function $u(\cdot)$.

$$\begin{aligned} L[u](x,T) &= q(x,T), \quad (x,T) \in \Omega\times\tau \\ u(x,T) &= \tilde{u}(x,T), \quad (x,T) \in \Sigma \end{aligned} \tag{87}$$

We know that the risk probability PDEs (18)-(21) are parabolic PDEs and can be written in the form of (87). Specifically, in our case we have $c(x,T) \equiv 0$ and $A = \frac{1}{2}\sigma^2 I$ which gives $\lambda = \frac{1}{2}\sigma^2$.

For any space $\Omega \in \mathbb{R}^d$, for any function $f : \mathbb{R}^d \to \mathbb{R}$, we define the L_1 norm of the function f on Ω to be

$$\|f\|_{L_1(\Omega)} := \int_\Omega f(\mathbf{X}) d\mathbf{X}. \tag{88}$$

With this definition, we know that

$$\mathbb{E}_{\mathbf{X}}[f] = \|f\|_{L_1(\Omega)}/|\Omega|, \tag{89}$$

where $\mathbf{X}$ is uniformly sampled from Ω, and $|\Omega|$ denote the size of the space Ω.

Corollary 2. ***Weak maximum principle***. *Suppose that $D = \Omega\times\tau$ is bounded, L is uniformly parabolic with $c \le 0$ and that $u \in C^0(\bar{D}) \cap C^2(D)$ satisfies $Lu \ge 0$ in D, and $M = \max_{\bar{D}} u \ge 0$. Then*

$$\max_{\bar{D}} u = \max_{\Sigma} u. \tag{90}$$

Corollary 3. ***Comparison principle.*** *Suppose that D is bounded and L is uniformly parabolic. If $u, v \in C^0(\bar{D}) \cap C^2(D)$ satisfies $Lu \le Lv$ in D and $u \ge v$ on Σ, then $u \ge v$ on D.*

Theorem 7. *Let $Lu = q$ in a bounded domain D, where L is parabolic, $c \le 0$ and $u \in C^0(\bar{D}) \cap C^2(D)$. Then*

$$\sup_D |u| \le \sup_\Sigma |u| + C \sup_D \frac{|q|}{\lambda}, \tag{91}$$

where C is a constant depending only on diam D and $\beta = \sup|b|/\lambda$.

Proof. (Theorem 7) Let D lie in the slab $0 < x_1 < d$. Without loss of generality, we assume $\lambda_1 \geq \lambda > 0$. Set $L_0 = -\frac{\partial}{\partial t} + a^{ij}\frac{\partial^2}{\partial x_i x_j} + b^i \frac{\partial}{\partial x_i}$. For $\alpha \geq \beta + 1$, we have

$$L_0 e^{\alpha x_1} = (\alpha^2 a^{11} + \alpha b^1) e^{\alpha x_1} \geq \lambda(\alpha^2 - \alpha\beta) e^{\alpha x_1} \geq \lambda \quad (92)$$

Consider the case when $Lu \geq q$. Let

$$v = \sup_{\Sigma} u^+ + (e^{\alpha d} - e^{\alpha x_1}) \sup_{D} \frac{|q^-|}{\lambda}, \quad (93)$$

where $u^+ = \max(u, 0)$ and $q^- = \min(q, 0)$. Then, since $Lv = L_0 v + cv \leq -\lambda \sup_{\Sigma}(|q^-|/\lambda)$ by maximum principle (Corollary 2), we have

$$L(v - u) \leq -\lambda(\sup_{D} \frac{|q^-|}{\lambda} + \frac{q}{\lambda}) \leq 0 \text{ in } D, \quad (94)$$

and $v - u \geq 0$ on Σ. Hence, from comparison principle (Corollary 3) we have

$$\sup_{D} u \leq \sup_{D} v \leq \sup_{\Sigma}|u| + C \sup_{D} \frac{|q|}{\lambda} \quad (95)$$

with $C = e^{\alpha d} - 1$. Consider $Lu \leq q$, we can get similar results with flipped signs. Combine both cases we have for $Lu = q$

$$\sup_{D}|u| \leq \sup_{\Sigma}|u| + C \sup_{D} \frac{|q|}{\lambda}. \quad (96)$$

■

Theorem 8. *Suppose that $D \in \mathbb{R}^{d+1}$ is a bounded domain, L is the parabolic operator defined in* (86) *and $u \in C^0(\bar{D}) \cap C^2(D)$ is a solution to the risk probability PDE. If the PINN $F_{\boldsymbol{\theta}}$ satisfies the following conditions:*

1) $\sup_{(x,T)\in\Sigma}|F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)| < \delta_1$*;*
2) $\sup_{(x,T)\in D}|W_{F_{\boldsymbol{\theta}}}(x,T)| < \delta_2$*;*
3) $F_{\boldsymbol{\theta}} \in C^0(\bar{D}) \cap C^2(D)$*,*

Then the error of $F_{\boldsymbol{\theta}}$ over Ω is bounded by

$$\sup_{(x,T)\in D} |F_{\boldsymbol{\theta}}(x,T) - u(x,t)| \leq \delta_1 + C\frac{\delta_2}{\lambda} \quad (97)$$

where C is a constant depending on D and L.

Proof. (Theorem 8) Denote $h_1 = L[F_{\boldsymbol{\theta}}] - q$, and $h_2 = F_{\boldsymbol{\theta}} - u$. Since $F_{\boldsymbol{\theta}}$ and u both fall in $C^0(\bar{D}) \cap C^2(D)$, from Theorem 7 we have

$$\begin{aligned} \sup_{D}|h_2(x,t)| &\leq \sup_{\Sigma}|h_2(x,t)| + C \sup_{D} \frac{|h_1(x,t)|}{\lambda} \\ &\leq \delta_1 + C\frac{\delta_2}{\lambda} \end{aligned} \quad (98)$$

which gives (97). ■

Lemma 5. *Let $D \subset \mathbb{R}^{d+1}$ be a domain. Define the regularity of D as*

$$R_D := \inf_{(x,T)\in D, r>0} \frac{|B(x,T,r) \cap D|}{\min\left\{|D|, \frac{\pi^{(d+1)/2} r^{d+1}}{\Gamma((d+1)/2+1)}\right\}}, \quad (99)$$

where $B(x,T,r) := \left\{y \in \mathbb{R}^{d+1} \mid \|y - (x,T)\| \leq r\right\}$ and $|S|$ is the Lebesgue measure of a set S. Suppose that D is bounded and $R_D > 0$. Let $q \in C^0(\bar{D})$ be an l-Lipschitz continuous function on $\bar{D}$. Then

$$\begin{aligned} \sup_{D}|q| \leq \max\Bigg\{ &\frac{2\|q\|_{L_1(\bar{D})}}{R_D|D|}, \\ &2l \cdot \left(\frac{\|q\|_{L_1(\bar{D})} \cdot \Gamma((d+1)/2+1)}{l R_D \cdot \pi^{(d+1)/2}}\right)^{\frac{1}{d+2}} \Bigg\}. \end{aligned} \quad (100)$$

Proof. (Lemma 5) According to the definition of l-Lipschitz continuity, we have

$$l\|(x,T) - (\bar{x},\bar{T})\|_2 \geq |q(x,T) - q(\bar{x},\bar{T})|, \quad \forall (x,T), (\bar{x},\bar{T}) \in \bar{D}, \quad (101)$$

which follows

$$\begin{aligned} \|q\|_{L_1(\bar{D})} &\geq \int_{D^+} |q(x,T)| dx dT \\ &\geq \int_{D^+} |f(\bar{x},\bar{T})| - l\|(x,T) - (\bar{x},\bar{T})\| dx dT, \end{aligned} \quad (102)$$

where $D^+ := \{(x,T) \in \bar{D} \mid |q(\bar{x},\bar{T})| - l\|(x,T) - (\bar{x},\bar{T})\| \geq 0\}$. Without loss of generality, we assume that $(\bar{x},\bar{T}) \in \arg\max_{\bar{D}}|q|$ and $q(\bar{x},\bar{T}) > 0$. Denote that

$$B_1 := B\left(\bar{x}, \bar{T}, \frac{q(\bar{x},\bar{T})}{2l}\right) \cap D. \quad (103)$$

It obvious that $B_1 \subset D^+$. Note that the Lebesgue measure of a hypersphere in $\mathbb{R}^{d+1}$ with radius r is $\pi^{(d+1)/2} r^{d+1}/\Gamma((d+1)/2+1)$. Then (102) becomes

$$\begin{aligned} &\|q\|_{L_1(\bar{D})} \geq \int_{B_1} q(\bar{x}) - l\|(x,T) - (\bar{x},\bar{T})\| dx dT \\ &\geq |B_1| \cdot \frac{q(\bar{x},\bar{T})}{2} \\ &\geq \frac{q(\bar{x},\bar{T})}{2} \cdot R_D \cdot \min\left\{|D|, \frac{\pi^{(d+1)/2} q(\bar{x},\bar{T})^{d+1}}{2^{d+1} l^{d+1} \Gamma((d+1)/2+1)}\right\} \\ &= \sup_{D}|q| \cdot \frac{R_D}{2} \cdot \min\left\{|D|, \frac{\pi^{(d+1)/2} q(\bar{x},\bar{T})^{d+1}}{2^{d+1} l^{d+1} \Gamma((d+1)/2+1)}\right\}, \end{aligned} \quad (104)$$

which leads to (100). ■

Now we are ready to prove Theorem 5.

Proof. (Theorem 5) From condition 1, $\mathbf{Y}$ is uniformly sampled from Σ. From condition 2, $\mathbf{X}$ is uniformly sampled from D. Then we have

$$\mathbb{E}_{\mathbf{Y}}\left[|F_{\boldsymbol{\theta}}(\mathbf{Y}) - \tilde{u}(\mathbf{Y})|\right] = \|F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)\|_{L_1(\Sigma)}/|\Sigma|, \quad (105)$$

$$\mathbb{E}_{\mathbf{X}}\left[|W_{F_{\boldsymbol{\theta}}}(\mathbf{X})|\right] = \|W_{F_{\boldsymbol{\theta}}}(x,T)\|_{L_1(D)}/|D|. \quad (106)$$

From condition 1 and 2 we know that

$$\|F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)\|_{L_1(\Sigma)} < \delta_1|\Sigma| \quad (107)$$

$$\|W_{F_{\boldsymbol{\theta}}}(x,T)\|_{L_1(D)} < \delta_2|D| \quad (108)$$

Also from condition 3 we have that $F_{\boldsymbol{\theta}} - \tilde{u}$ and $W_{F_{\boldsymbol{\theta}}}$ are both l-Lipschitz continuous on $\bar{D}$. From Lemma 5 we know that

$$\begin{aligned}
&\sup_{(x,T)\in\Sigma} |F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)| \\
&\leq \max\Bigg\{ \frac{2\|F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)\|_{L_1(\Sigma)}}{R_\Sigma|\Sigma|}, \\
&\quad 2l \cdot \left(\frac{\|F_{\boldsymbol{\theta}}(x,T) - \tilde{u}(x,T)\|_{L_1(\Sigma)} \cdot \Gamma(d/2+1)}{lR_\Sigma \cdot \pi^{d/2}} \right)^{\frac{1}{d+1}} \Bigg\} \\
&< \max\left\{ \frac{2\delta_1|\Sigma|}{R_\Sigma|\Sigma|}, 2l \cdot \left(\frac{\delta_1|\Sigma| \cdot \Gamma(d/2+1)}{lR_\Sigma \cdot \pi^{d/2}} \right)^{\frac{1}{d+1}} \right\},
\end{aligned} \tag{109}$$

$$\begin{aligned}
&\sup_{(x,T)\in D} |W_{F_{\boldsymbol{\theta}}}(x,T)| \\
&\leq \max\Bigg\{ \frac{2\|W_{F_{\boldsymbol{\theta}}}(x,T)\|_{L_1(D)}}{R_D|D|}, \\
&\quad 2l \cdot \left(\frac{\|W_{F_{\boldsymbol{\theta}}}(x,T)\|_{L_1(D)} \cdot \Gamma((d+1)/2+1)}{lR_D \cdot \pi^{(d+1)/2}} \right)^{\frac{1}{d+2}} \Bigg\} \\
&< \max\left\{ \frac{2\delta_2|D|}{R_D|D|}, 2l \cdot \left(\frac{\delta_2|D| \cdot \Gamma((d+1)/2+1)}{lR_D \cdot \pi^{(d+1)/2}} \right)^{\frac{1}{d+2}} \right\}.
\end{aligned} \tag{110}$$

Let

$$\begin{aligned}
\tilde{\delta}_1 &= \max\left\{ \frac{2\delta_1|\Sigma|}{R_\Sigma|\Sigma|}, 2l \cdot \left(\frac{\delta_1|\Sigma| \cdot \Gamma(d/2+1)}{lR_\Sigma \cdot \pi^{d/2}} \right)^{\frac{1}{d+1}} \right\}, \\
\tilde{\delta}_2 &= \max\left\{ \frac{2\delta_2|D|}{R_D|D|}, 2l \cdot \left(\frac{\delta_2|D| \cdot \Gamma((d+1)/2+1)}{lR_D \cdot \pi^{(d+1)/2}} \right)^{\frac{1}{d+2}} \right\}.
\end{aligned} \tag{111}$$

Then from Theorem 8 we know that

$$\sup_{(x,T)\in D} |F_{\boldsymbol{\theta}}(x,T) - u(x,T)| \leq \tilde{\delta}_1 + C\frac{\tilde{\delta}_2}{\lambda}. \tag{112}$$

Given that $\lambda = \frac{1}{2}\sigma^2$, replace C with $2C$ we get

$$\sup_{(x,T)\in D} |F_{\boldsymbol{\theta}}(x,T) - u(x,T)| \leq \tilde{\delta}_1 + C\frac{\tilde{\delta}_2}{\sigma^2}, \tag{113}$$

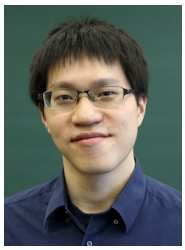
which completes the proof.

■



Zhuoyuan Wang received his B.E. degree in Automation from Tsinghua University, Beijing, China, in 2020 and is currently pursuing a Ph.D. degree in Electrical and Computer Engineering at Carnegie Mellon University, Pittsburgh, PA, USA.

His research interests include safety-critical control for stochastic systems, physics-informed learning, safe reinforcement learning and application to robotic systems. He is a recipient of the Michel and Kathy Doreau Graduate Fellowship at Carnegie Mellon University.



Albert Chern is an Assistant Professor in Computer Science and Engineering at University of California San Diego, La Jolla, CA, USA, since 2020. He received his Ph.D. in Applied and Computational Mathematics at California Institute of Technology, Pasadena, CA, USA, in 2017, and was a Postdoctoral Researcher in Mathematics at Technische Universität Berlin, Berlin, Germany, from 2017 to 2020. Chern's research interest lies in applications of differential geometry to computational math, fluid dynamics, computer graphics, and the interplay between stochastic processes and differential equations. Chern was a recipient of the NSF CAREER Award in 2023.



Yorie Nakahira is an Assistant Professor in the Department of Electrical and Computer Engineering at Carnegie Mellon University. She received B.E. in Control and Systems Engineering from Tokyo Institute of Technology in 2012 and Ph.D. in Control and Dynamical Systems from California Institute of Technology in 2019. Her research interests include the fundamental theory of optimization, control, and learning and its application to neuroscience, cell biology, smart grid, cloud computing, finance, autonomous robots.

D. Proof of Theorems for Physics-informed Learning (Continued)

Proof. (Theorem 6)

The proof is based on [45, Theorem 3.6]. We first introduce some definitions. A generic partial differential equation (PDE) has the form

$$\mathcal{L}[u](x) = p(x),\ \forall x \in D, \quad \mathcal{B}[u](x) = q(x),\ \forall x \in \Gamma \subseteq \partial D, \tag{114}$$

where $\mathcal{L}[\cdot]$ is a differential operator and $\mathcal{B}[\cdot]$ could be Dirichlet, Neumann, Robin, or periodic boundary conditions. We define the physics-model loss for the PINN training with data m for a neural network instance h as

$$\begin{aligned}\text{Loss}_m^{\text{PINN}}(h;\boldsymbol{\lambda}) =& \frac{\lambda_r}{m_r}\sum_{i=1}^{m_r}\left\|\mathcal{L}[h]\left(x_r^i\right) - a\left(x_r^i\right)\right\|^2 + \\ & \frac{\lambda_b}{m_b}\sum_{i=1}^{m_b}\left\|\mathcal{B}[h]\left(x_b^i\right) - b\left(x_b^i\right)\right\|^2,\end{aligned} \tag{115}$$

where $\boldsymbol{\lambda} = [\lambda_r, \lambda_b]$ are hyperparameters.

We now restate the assumptions used in [45].

Assumption 3. *[45, Assumption 3.1]*

Let $D \in \mathbb{R}^{d+1}$ and $\Gamma \in \mathbb{R}^d$ be the state-time space of interest and a subset of its boundary, respectively. Let μ_r and μ_b be probability distributions defined on D and Γ. Let ρ_r be the probability density of μ_r with respect to $(d+1)$-dimensional Lebesgue measure on D. Let ρ_b be the probability density of μ_b with respect to d-dimensional Hausdorff measure on Γ. We assume the following conditions.

1) *D is at least of class $C^{0,1}$.*
2) *ρ_r and ρ_b are supported on D and Γ, respectively. Also, $\inf_D \rho_r > 0$ and $\inf_\Gamma \rho_b > 0$.*
3) *For $\epsilon > 0$, there exists partitions of D and Γ, $\left\{D_j^\epsilon\right\}_{j=1}^{K_r}$ and $\left\{\Gamma_j^\epsilon\right\}_{j=1}^{K_b}$ that depend on ϵ such that for each j, there are cubes $H_\epsilon\left(\mathbf{z}_{j,r}\right)$ and $H_\epsilon\left(\mathbf{z}_{j,b}\right)$ of side length ϵ centered at $\mathbf{z}_{j,r} \in D_j^\epsilon$ and $\mathbf{z}_{j,b} \in \Gamma_j^\epsilon$, respectively, satisfying $D_j^\epsilon \subset H_\epsilon\left(\mathbf{z}_{j,r}\right)$ and $\Gamma_j^\epsilon \subset H_\epsilon\left(\mathbf{z}_{j,b}\right)$.*
4) *There exists positive constants c_r, c_b such that $\forall \epsilon > 0$, the partitions from the above satisfy $c_r\epsilon^d \le \mu_r\left(D_j^\epsilon\right)$ and $c_b\epsilon^{d-1} \le \mu_b\left(\Gamma_j^\epsilon\right)$ for all j. There exists positive constants C_r, C_b such that $\forall x_r \in D$ and $\forall x_b \in \Gamma, \mu_r\left(B_\epsilon\left(x_r\right) \cap D\right) \le C_r\epsilon^d$ and $\mu_b\left(B_\epsilon\left(x_b\right) \cap \Gamma\right) \le C_b\epsilon^{d-1}$ where $B_\epsilon(x)$ is a closed ball of radius ϵ centered at x. Here C_r, c_r depend only on (D, μ_r) and C_b, c_b depend only on (Γ, μ_b).*
5) *When $d = 1$, we assume that all boundary points are available. Thus, no random sample is needed on the boundary.*

Assumption 4. *[45, Assumption 3.2]*

Let k be the highest order of the derivative for the PDE of interest. For some $0 < \alpha \le 1$, let $p \in C^{0,\alpha}(D)$ and $q \in C^{0,\alpha}(\Gamma)$.

1) *For each training dataset m, let $\mathcal{H}_m$ be a class of neural networks in $C^{k,\alpha}(D) \cap C^{0,\alpha}(\bar{D})$ such that for any $h \in \mathcal{H}_m, \mathcal{L}[h] \in C^{0,\alpha}(D)$ and $\mathcal{B}[h] \in C^{0,\alpha}(\Gamma)$.*
2) *For each $m, \mathcal{H}_m$ contains a network u_m^* satisfying $\text{Loss}_m^{\text{PINN}}\left(u_m^*;\lambda\right) = 0$.*
3) *And,*

$$\sup_m \left[\mathcal{L}\left[u_m^*\right]\right]_{\alpha;D} < \infty, \quad \sup_m \left[\mathcal{B}\left[u_m^*\right]\right]_{\alpha;\Gamma} < \infty.$$

We also know that safety-related PDEs are second-order linear parabolic equations. Let U be a bounded domain in $\mathbb{R}^d$ and let $U_T = U \times (0, T]$ for some fixed time $T > 0$. Let (x, t) be a point in $\mathbb{R}^{d+1}$. For a parabolic PDE, we define

$$\begin{cases} -u_t + L[u] = s, & \text{in } U_T \\ u = \varphi, & \text{in } \partial U \times [0, T] \\ u = v, & \text{in } \bar{U} \times \{t = 0\} \end{cases} \tag{116}$$

where $s : U_T \mapsto \mathbb{R}, v : \bar{U} \mapsto \mathbb{R}, \varphi : \partial U \times [0, T] \mapsto \mathbb{R}$, and

$$\begin{aligned} L[u] = & \sum_{i,j=1}^d D_i\left(a^{ij}(x,t)D_j u + b^i(x,t)u\right) \\ & + \sum_{i=1}^d c^i(x,t)D_i u + d(x,t)u. \end{aligned} \tag{117}$$

Assumption 5. *[45, Assumption 3.4]*

Let $\lambda(x,t)$ be the minimum eigenvalues of $\left[a^{ij}(x,t)\right]$ and $\alpha \in (0,1)$. Suppose a^{ij}, b^i are differentiable and let $\tilde{b}^i(x,t) = \sum_{j=1}^d D_j a^{ij}(x,t) + b^i(x,t) + c^i(x,t)$ and $\tilde{c}(x,t) = \sum_{i=1}^d D_i b^i(x,t) + d(x,t)$. Let $\Omega = U \times (0,T)$.

1) *For some constant* $\lambda_0, \lambda(x,t) \geq \lambda_0 > 0$ *for all* $(x,t) \in \Omega$.
2) $a^{ij}, \tilde{b}^i, \tilde{c}^i$ *are Hölder continuous (exponent* α*) in* Ω*, and* $\left|a^{ij}\right|_{\alpha,}$ $\left|\mathrm{d}\tilde{b}^i\right|_{\alpha,}$ $\left|\mathrm{d}^2\tilde{c}\right|_{\alpha}$ *are uniformly bounded.*
3) s *is Hölder continuous (exponent* α*) in* U_T *and* $\left|\mathrm{d}^2 f\right|_{\alpha} < \infty$. v *and* φ *are Hölder continuous (exponent* α*) on* $\bar{U} \times \{t = 0\}$*, and* $\partial U \times [0,T]$*, respectively, and* $v(x) = \varphi(x,0)$ *on* ∂U.
4) *There exists* $\theta > 0$ *such that* $\theta^2 a^{11}(x,t) + \theta \tilde{b}^1(x,t) \geq 1$ *in* Ω.
5) *For* $x' \in \partial U$*, there exists a closed ball* $\bar{B}$ *in* $\mathbb{R}^d$ *such that* $\bar{B} \cap \bar{U} = \{x'\}$.
6) *There are constants* $\Lambda, v \geq 0$ *such that for all* $(x,t) \in \Omega$

$$\begin{aligned} \sum_{i,j} \left|a^{ij}(x,t)\right|^2 &\leq \Lambda, \\ \lambda_0^{-2}\left(\|\boldsymbol{b}(x,t)\|^2 + \|\boldsymbol{c}(x,t)\|^2\right) + \lambda_0^{-1}|d(x,t)| &\leq v^2, \end{aligned} \tag{118}$$

where $\boldsymbol{b}(x,t) = \left[b^1(x,t), \cdots, b^d(x,t)\right]^T$ *and* $\boldsymbol{c}(x,t) = \left[c^1(x,t), \cdots, c^d(x,t)\right]^T$.

Now that we have listed all assumptions needed in [45, Theorem 3.6], we are ready to prove Theorem 6.

Note that conditions 1-5 in Assumption 2 match with Assumption 3 ([45, Assumption 3.1]), thus it is satisfied.

Since we assume for any training data set m_r, architecture-wise the physics-informed neural network has enough representation power to characterize the solution of the sasfety-related PDE u, Assumption 4 ([45, Assumption 3.2]) is satisfied.

For Assumption 5 ([45, Assumption 3.4]), condition 1 is satisfied because we assume the noise magnitude σ has non-zero eigenvalues in Assumption 1, and $a(x,t) = \sigma^\intercal \sigma$ which indicates all eigenvalues of a are greater than 0, thus λ_0 can be found and this condition is satisfied. For condition 2, we have f, g and σ in the system dynamics are bounded and continuous from Assumption 1, and we know $a = \sigma^\intercal \sigma$, $b \equiv 0$, $c = f + g\, u_{\text{nominal}}$ and $d \equiv 0$. Thus, $a^{ij}, \tilde{b}^i, \tilde{c}^i$ are Hölder continuous (exponent α) in Ω, and $\left|a^{ij}\right|_{\alpha,}$ $\left|\mathrm{d}\tilde{b}^i\right|_{\alpha,}$ $\left|\mathrm{d}^2\tilde{c}\right|_{\alpha}$ are uniformly bounded, indicating the condition is satisfied. For condition 3, we know that $s \equiv 0$ and $\varphi \equiv 0$ or $\varphi \equiv 1$, and $v \equiv 0$ or $v \equiv 1$ depending on the specific choice of the safety-related probability, thus they are Hölder continuous and we know that condition 3 is satisfied. For condition 4, since we have $a^{11} > 0$ in Assumption 1 and we know $b \equiv 0$, we can always find $\theta > 0$ so that the condition is satisfied. For condition 5, it is assumed in the regularity condition in Assumption 2 thus is satisfied. For condition 6, we have f, g, σ bounded in Assumption 1 and we know that $a = \sigma^\intercal \sigma$, $b \equiv 0$, $c = f + g\, u_{\text{nominal}}$ and $d \equiv 0$. Thus, we can find constants $\Lambda, v \geq 0$ to bound $\sum_{i,j} \left|a^{ij}(x,t)\right|^2$ and $\lambda_0^{-2}\left(\|\boldsymbol{b}(x,t)\|^2 + \|\boldsymbol{c}(x,t)\|^2\right) + \lambda_0^{-1}|d(x,t)|$ for all $(x,t) \in \Omega$, indicating this condition is satisfied.

Till here, we have shown that all assumptions needed in [45, Theorem 3.6] are satisfied. As the number of training data $m_r \to \infty$, the loss function (25) is equivalent to the Hölder regularized empirical loss [45, Equation (3.3)]. Thus, all conditions in [45, Theorem 3.6] hold and we have

$$\lim_{m_r \to \infty} F_{m_r} = u.$$

■

In the following sections, we provide details for the experiments presented in the paper and some additional experiments.

E. Generalization to unseen regions

In the generalization task in section VII-A, we use a down-sampled sub-region of the system to train the proposed PIPE framework, and test the prediction results on the full state-time space. We showed that PIPE is able to give accurate inference on the entire state space, while standard fitting strategies cannot make accurate predictions. In the paper, we only show the fitting results of thin plate spline interpolation. Here, we show the results of all fitting strategies we tested for this generalization tasks. The fitting strategies are

1) Polynomial fitting of 5 degrees for both state x and time T axes. The fitting sum of squares error (SSE) on the training data is 0.1803.
2) Lowess: locally weighted scatterplot smoothing [55]. The training SSE is 0.0205.
3) Cubic spline interpolation. The training SSE is 0.
4) Biharmonic spline interpolation. The training SSE is 2.52×10^{-27}.
5) TPS: thin plate spline interpolation. The training SSE is 1.64×10^{-26}.

All fittings are conducted via the MATLAB Curve Fitting Toolbox. Fig. 7 visualizes the fitting results on the full state space. Polynomial fitting performs undesirably because the polynomial functions cannot capture the risk probability geometry well. Lowess fitting also fails at inference since it does not have any model information of the data. Given the risk probability data, cubic spline cannot extrapolate outside the training region, and we use 0 value to fill in the unseen region where it yields NAN for prediction. Biharmonic and TPS give similar results as they are both spline interpolation methods. None of these fitting methods can accurately predict the risk probability in unseen regions, because they purely rely on training data and do not incorporate any model information of the risk probability for prediction.

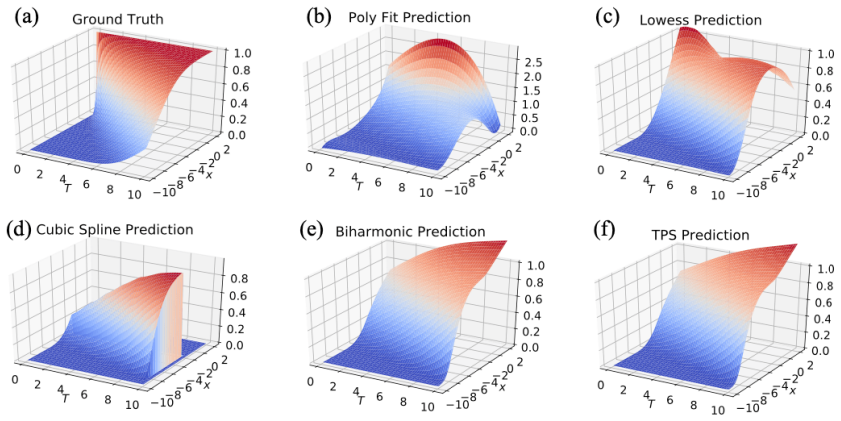


Fig. 7: Results of different fitting strategies on the risk probability generalization task.

We also compare the prediction results for different network architectures in the proposed PIPE framework, to examine the effect of network architectures on the risk probability prediction performance. The network settings we consider are different hidden layer numbers (1-4) and different numbers of neurons in one hidden layer (16, 32, 64). We use 3 hidden layers, 32 neurons per layer as baseline (the one used in the paper). Table II and Table III report the averaged absolute error of the predictions for different layer numbers and neuron numbers per layer, respectively. We trained the neural networks 10 times with random initialization to report the standard deviation. We can see that as the number of layers increases, the prediction error of the risk probability drops, but in a relatively graceful manner. The prediction error for a single layer PIPE is already very small, which indicates that the proposed PIPE framework is very efficient in terms of computation and storage. The prediction accuracy tends to saturate when the hidden layer number reaches 3, as there is no obvious improvement when we increase the layer number from 3 to 4. This means for the specific task we consider, a 3-layer neural net has enough representation. Under the same layer number, as the neuron number per layer increases, the risk probability prediction error decreases. This indicates that with larger number of neuron in each layer (*i.e.,* wider neural networks), the neural network can better capture the mapping between state-time pair and the risk probability. However, the training time increases significantly for PIPEs with more neurons per layer (152s for 16 neurons and 971s for 64 neurons), and the gain in prediction accuracy becomes marginal compared to the amount of additional computation involved. We suggest to use a moderate number of neurons per layer to achieve desirable trade-offs between computation and accuracy.

# Hidden Layer	1	2	3	4
Prediction Error ($\times 10^{-3}$)	4.773 ± 0.564	$\mathbf{2.717 \pm 0.241}$	2.819 ± 0.619	2.778 ± 0.523

TABLE II: Risk probability prediction error of PIPE for different numbers of hidden layers.

# Neurons	16	32	64
Prediction Error ($\times 10^{-3}$)	2.743 ± 0.313	2.931 ± 0.865	$\mathbf{2.599 \pm 0.351}$

TABLE III: Risk probability prediction error of PIPE for different neuron numbers per layer.

F. Efficient estimation of risk probability

In the efficient estimation task in section VII-B, we showed that PIPE will give better sample efficiency in risk probability prediction, in the sense that it achieves the same prediction accuracy with less sample numbers. Here, we visualize the prediction errors of Monte Carlo (MC) and the proposed PIPE framework to better show the results. Fig. 8 shows the prediction error comparison plots for MC and PIPE with different sample numbers N. As the sample number increases, the errors for both MC and PIPE decrease because of better resolution of the data. PIPE gives more accurate predictions than MC across all sample numbers, since it combines data and physical model of the system together. From Fig. 8 we can see that, PIPE indeed provides smoother and more accurate estimation of the risk probability. The visualization results further validate the efficacy of the proposed PIPE framework.

# Hidden Layer	1	2	3	4
Prediction Error ($\times 10^{-4}$)	14.594 ± 2.109	7.302 ± 0.819	6.890 ± 0.613	$\mathbf{6.625 \pm 0.574}$

TABLE IV: Risk probability gradient prediction error of PIPE for different numbers of hidden layers.

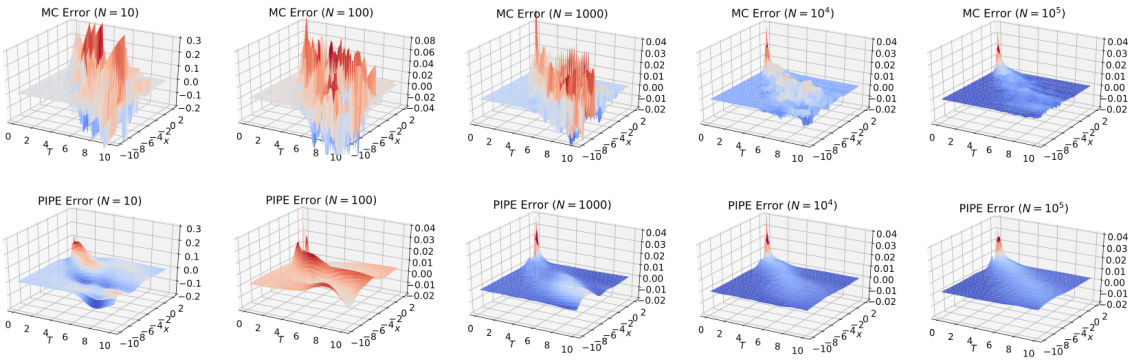


Fig. 8: Prediction errors for Monte Carlo (left) and PIPE (right) with different sample numbers.

# Neurons	16	32	64
Prediction Error ($\times 10^{-4}$)	7.049 ± 0.767	6.890 ± 0.613	$\mathbf{6.458 \pm 0.794}$

TABLE V: Risk probability gradient prediction error of PIPE for different neuron numbers per layer.

G. Adaptation on changing system parameters

For the adaptation task described in section VII-C, we trained PIPE with system data of parameters $\lambda_{\text{train}} = [0.1, 0.5, 0.8, 1]$ and tested over a range of unseen parameters over the interval $\lambda = [0, 2]$. Here, we show additional results on parameters $\lambda_{\text{test}} = [0.3, 0.7, 1.2, 2]$ to further illustrate the adaptation ability of PIPE. Fig. 9 shows the results. It can be seen that PIPE is able to predict the risk probability accurately on both system parameters with very low error over the entire state-time space. This result indicates that PIPE has solid adaptation ability on uncertain parameters, and can be used for stochastic safe control with adaptation requirements.

H. Estimating the gradient of risk probability

For the gradient estimation task in section VII-D, we presented that PIPE is able to predict the risk probability gradients accurately by taking finite difference on the predicted risk probabilities. This result shows that PIPE can be used for first- and higher-order methods for safe control, by providing accurate gradient estimations in a real-time fashion. Similar to the generalization task, here we report the gradient prediction errors with different network architectures in PIPE, to examine the effect of network architectures on the gradient estimation performance. Table IV and Table V show the averaged absolute error of gradient predictions for different layer numbers and neuron numbers per layer. We trained the neural networks for 10 times with random initialization to report the standard deviation. We can see that as the number of hidden layer increases, the gradient prediction error keeps dropping, and tends to saturate after 3 layers. With the increasing neuron numbers per layer, the gradient prediction error decreases in a graceful manner. Similar to the generalization task, larger networks with more hidden

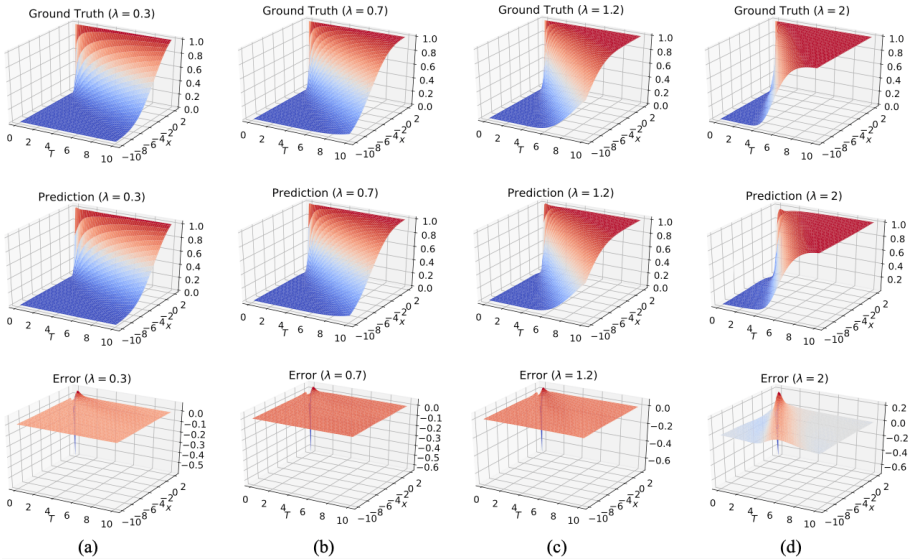


Fig. 9: Risk probability prediction results on systems with unseen parameters $\lambda_{\text{test}} = [0.3, 0.7, 1.2, 2]$.

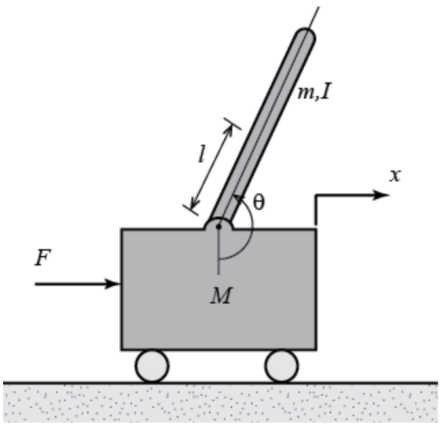


Fig. 10: Inverted pendulum on a cart.

layers and more neurons per layer can give more accurate estimation of the gradient, but the computation time scales poorly compared to the accuracy gain. Based on these results, we suggest to use moderate numbers of layers and neurons per layer to acquire desirable gradient prediction with less computation time.

In this section, we provide additional experiment results on risk estimation of the inverted pendulum on a cart system, as well as safe control with risk estimation through the proposed PIPE framework.

I. Risk estimation of inverted pendulum on a cart system

We consider the inverted pendulum on a cart system, with dynamics given by

$$\frac{d\mathbf{x}}{dt} = f(\mathbf{x}) + g(\mathbf{x})u + \sigma\ \tilde{I}\ dW_t, \tag{119}$$

where $\mathbf{x} = [x, \dot{x}, \theta, \dot{\theta}]^\top$ is the state of the system and $u \in \mathbb{R}$ is the control, with x and $\dot{x}$ being the position and velocity of the cart, and θ and $\dot{\theta}$ being the angle and angular velocity of the pendulum. We use $\mathbf{x}$ to denote the state of the system to distinguish from cart's position x. Then, we have

$$f(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & m+M & 0 & ml\cos\theta \\ 0 & 0 & 1 & 0 \\ 0 & ml\cos\theta & 0 & ml^2 \end{bmatrix}^{-1} \begin{bmatrix} \dot{x} \\ ml\dot{\theta}^2\sin\theta - b_x\dot{x} \\ \dot{\theta} \\ mgl\sin\theta - b_\theta l\dot{\theta} \end{bmatrix}, \tag{120}$$

$$g(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & m+M & 0 & ml\cos\theta \\ 0 & 0 & 1 & 0 \\ 0 & ml\cos\theta & 0 & ml^2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \tag{121}$$

where m and M are the mass of the pendulum and the cart, g is the acceleration of gravity, l is the length of the pendulum, and b_x and b_θ are constants. The last term in (119) is the additive noise, where W_t is 4-dimensional Wiener process with $W_0 = \mathbf{0}$, σ is the magnitude of the noise, and

$$\tilde{I} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{122}$$

Fig. 10 visualizes the system.

The safe set is defined in (2) with barrier function $\phi(\mathbf{x}) = 1 - (\frac{\mathbf{x}_3}{\pi/3})^2 = 1 - (\frac{\theta}{\pi/3})^2$. Essentially the system is safe when the angle of the pendulum is within $[-\pi/3, \pi/3]$. We consider spatial-temporal space $\Omega \times \tau = [[-10, 10] \times [-\pi/3, \pi/3] \times [-\pi, \pi]] \times [0, 1]$. We collect training and testing data on $\Omega \times \tau$ with grid size $N_{\Omega\text{-train}} = 13$ and $N_{\tau\text{-train}} = 10$ for training and $N_{\Omega\text{-test}} = 25$ and $N_{\tau\text{-test}} = 10$ for testing. The nominal controller we choose is a proportional controller $N(\mathbf{x}) = -K\mathbf{x}$ with $K = [0, -0.9148, -22.1636, -14.3992]^\top$. The sample number for MC simulation is set to be $N = 1000$ for both training and testing. Table VI lists the parameters used in the simulation.

Parameters	Values
M	1
m	0.1
g	9.8
l	0.5
b_x	0.05
b_θ	0.1

TABLE VI: Parameters used in the inverted pendulum simulation.

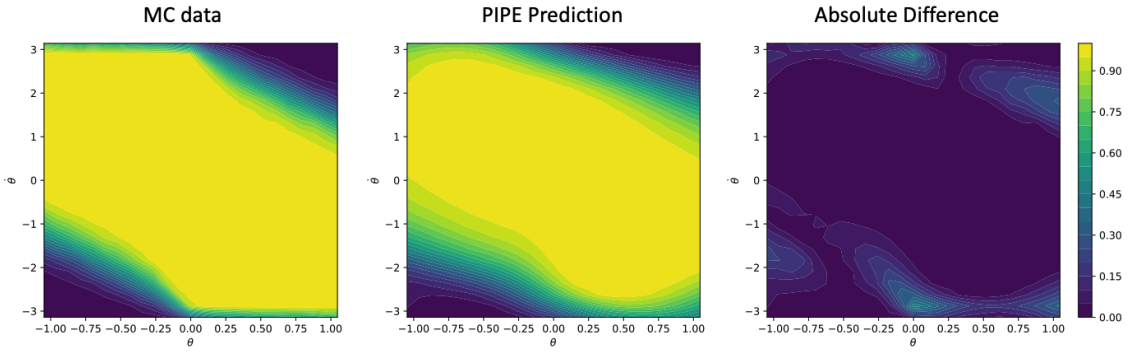


Fig. 11: Safety probability from MC simulation and PIPE prediction, and their absolute difference. Results on outlook time horizon $T = 0.6$, initial velocity $v = 0$. The x-axis shows the initial angle, and y-axis shows the initial angular velocity.

We train PIPE with the same configuration listed in section VII. According to Theorem 1, the PDE that characterizes the safety probability of the pendulum system is

$$\frac{\partial F}{\partial T}(\mathbf{x}, T) = (f(\mathbf{x}) - g(\mathbf{x})K\mathbf{x}) \frac{\partial F}{\partial \mathbf{x}}(\mathbf{x}, T) + \frac{1}{2}\sigma^2 \tilde{I} \operatorname{tr}\left(\frac{\partial^2 F}{\partial \mathbf{x}^2}(\mathbf{x}, T)\right), \tag{123}$$

which is a high dimensional and highly nonlinear PDE that cannot be solved effectively using standard solvers. Here we can see the advantage of combining data with model and using a learning-based framework to estimate the safety probability. Fig. 11, Fig. 12 and Fig. 13 show the results of the PIPE predictions. We see that despite the rather high dimension and nonlinear dynamics of the pendulum system, PIPE is able to predict the safety probability of the system with high accuracy. Besides, since PIPE takes the model knowledge into training loss, the resulting safety probability prediction is smoother thus more reliable than pure MC estimations.

J. Safe control with PIPE

We consider the control affine system (1) with $f(x) \equiv Ax = 2x$, $g(x) \equiv 1$, $\sigma(x) \equiv 2$. The safe set is defined as in (2) and the barrier function is chosen to be $\phi(x) := x - 1$. The safety specification is given as the forward invariance condition. The nominal controller is a proportional controller $N(x) = -Kx$ with $K = 2.5$. The closed-loop system with this controller has

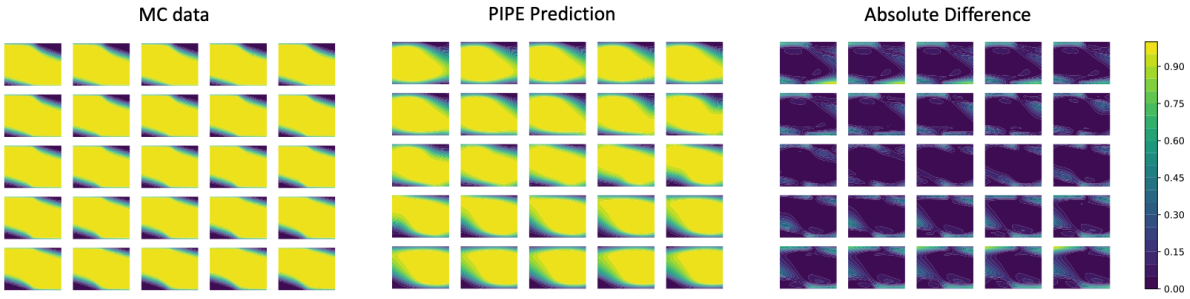


Fig. 12: Safety probability from MC simulation and PIPE prediction, and their absolute difference. Results on outlook time horizon $T = 0.3$. The 5x5 plots show results on 25 different initial velocities uniformly sampled in $[-10, 10]$. The x-axis and y-axis (omitted) are the initial angle and the initial angular velocity as in Fig. 11. One can see that the safety probability shift as the velocity changes, and the safety probability is symmetric with regard to the origin $v = 0$ due to the symmetry of the dynamics.

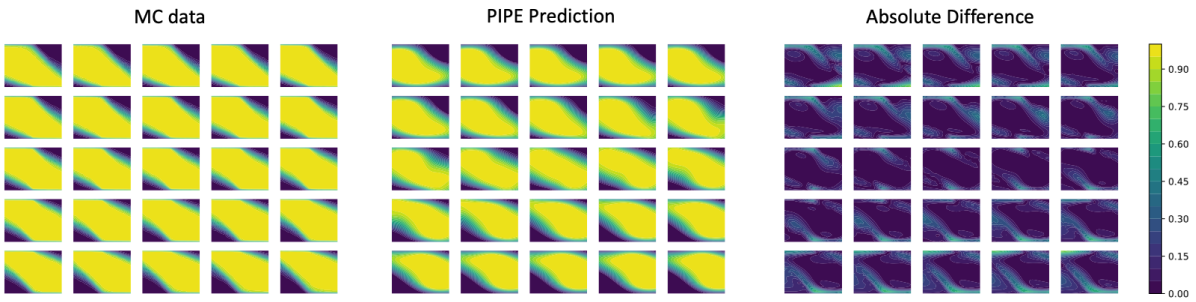


Fig. 13: Safety probability from MC simulation and PIPE prediction, and their absolute difference. Results on outlook time horizon $T = 1$. The 5x5 plots show results on 25 different initial velocities uniformly sampled in $[-10, 10]$.

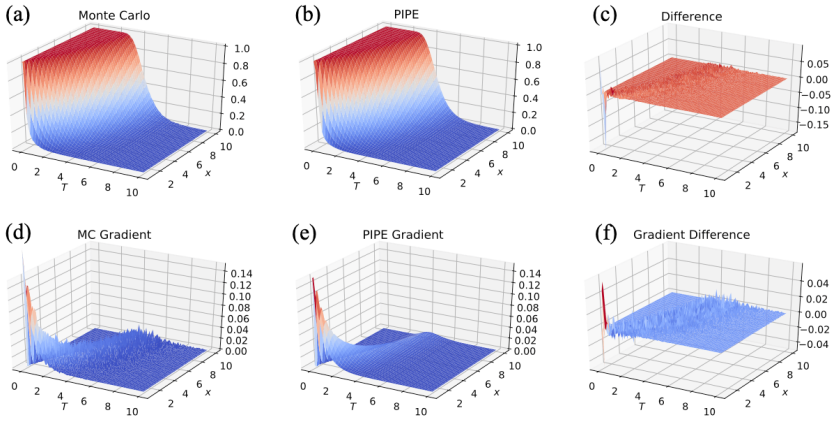


Fig. 14: Safety probability and its gradient of Monte Carlo samples and PIPE estimation.

an equilibrium at $x = 0$ and tends to move into the unsafe set in the state space. We run simulations with $dt = 0.1$ for all controllers. The initial state is set to be $x_0 = 3$. For this system, the safety probability satisfies the following PDE

$$\frac{\partial F}{\partial T}(x,T) = -0.5x\frac{\partial F}{\partial x}(x,T) + 2\,\mathrm{tr}\left(\frac{\partial^2 F}{\partial x^2}(x,T)\right), \tag{124}$$

with initial and boundary conditions

$$\begin{aligned} F(x,t) &= 0,\ x \le 1, \\ F(x,0) &= \mathbb{1}(x \ge 1). \end{aligned} \tag{125}$$

We first estimate the safety probability $F(x,T)$ of the system via PIPE. The training data $\bar{F}(x,T)$ is acquired by running MC on the system dynamics for given initial state x_0 and nominal control N. Specifically,

$$\bar{F}(x,T) = \mathbb{P}(\forall t \in [0,T], x_t \in \mathcal{C} \mid x_0 = x) = \frac{N_{\text{safe}}}{N}, \tag{126}$$

with $N = 100$ being the number of sample trajectories. The training data is sampled on the state-time region $\Omega \times \tau = [1,10] \times [0,10]$ with grid size $dx = 0.5$ and $dt = 0.5$. We train PIPE with the same configuration as listed in section VII. We test the estimated safety probability and its gradient on the full state space $\Omega \times \tau$ with $dx = 0.1$ and $dt = 0.1$. Fig. 14 shows the results. It can be seen that the PIPE estimate is very close to the Monte Carlo samples, which validates the efficacy of the framework. Furthermore, the PIPE estimation has smoother gradients, due to the fact that it leverages model information along with the data.

We then show the results of using such estimated safety probability for control. Fig. 15 shows the results. For the baseline stochastic safe controllers for comparison, refer to [56] for details. We see that the PIPE framework can be applied to long-term safe control methods discussed in [56]. Together with the PIPE estimation, long-term safety can be ensured in stochastic control systems.

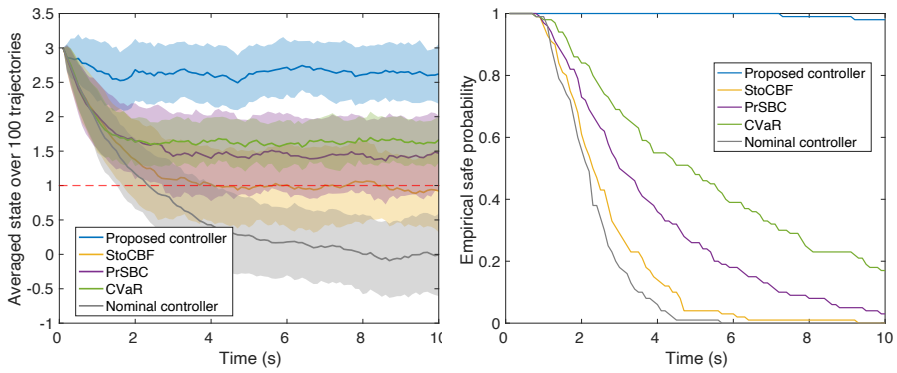


Fig. 15: Safe control with probability estimation from PIPE compared with other baselines.