A Generalizable Physics-informed Learning Framework for Risk Probability Estimation

Zhuoyuan Wang ZHUOYUAW@ANDREW.CMU.EDU
Yorie Nakahira YORIE@CMU.EDU
Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA, USA

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

Accurate estimates of long-term risk probabilities and their gradients are critical for many stochastic safe control methods. However, computing such risk probabilities in real-time and in unseen or changing environments is challenging. Monte Carlo (MC) methods cannot accurately evaluate the probabilities and their gradients as an infinitesimal devisor can amplify the sampling noise. In this paper, we develop an efficient method to evaluate the probabilities of long-term risk and their gradients. The proposed method exploits the fact that long-term risk probability satisfies certain partial differential equations (PDEs), which characterize the neighboring relations between the probabilities, to integrate MC methods and physics-informed neural networks. We provide theoretical guarantees of the estimation error given certain choices of training configurations. Numerical results show the proposed method has better sample efficiency, generalizes well to unseen regions, and can adapt to systems with changing parameters. The proposed method can also accurately estimate the gradients of risk probabilities, which enables first- and second-order techniques on risk probabilities to be used for learning and control.

Keywords: Stochastic safe control; physics-informed learning; risk probability estimation.

1. Introduction

Safe control for stochastic systems is important yet a key challenge for deploying autonomous systems in the real world. In the past decades, many stochastic control methods have been proposed to ensure safety of systems with noises and uncertainties, including stochastic reachabilities (Prandini and Hu, 2008), chance-constrained predictive control (Nakka et al., 2020) and *etc*. Despite the huge amount of stochastic safe control methods, many of them rely on accurate estimates of long-term risk probabilities or their gradients to guarantee long-term safety (Abate et al., 2008; Chapman et al., 2019; Santoyo et al., 2021; Wang et al., 2021). To get such accurate estimates is non-trivial, and here we list the challenges.

- *High computational complexity*. Estimating long-term risk is computationally expensive, because the possible state trajectories scale exponentially with regard to the time horizon. Besides, the values of risk probability are often small in safety-critical systems, and thus huge amounts of sample trajectories are needed to capture the rare unsafe event (Janssen, 2013).
- *Sample inefficiency.* Generalization of risk estimation to the full state space is hard to achieve for sample-based methods, since each point of interest requires one separate simulation. The sample complexity increases linearly with respect to the number of points for evaluation. Besides, most

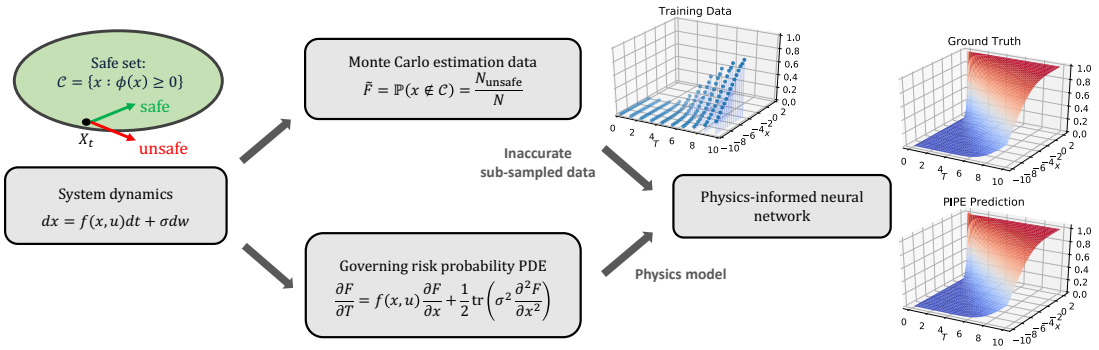

Figure 1: The overview diagram of the proposed PIPE framework. The system takes form (1) with safe set defined as (2). For training data, one can acquire the empirical risk probabilities by simulating the system dynamics and calculating the ratio of unsafe trajectories over all trajectories. For the physics model, we know that the mapping between state-time pair and the risk probability satisfies a governing convection diffusion equation (Theorem 4). 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 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.

of the existing methods require re-evaluation of the risk probability for any changes of system parameters, which further degrades sample efficiency (Zuev, 2015).

- *Lack of direct solutions*. For control affine systems, recent study (Chern et al., 2021) suggests that the risk probability can be characterized by the solution of partial differential equations (PDEs). It provides an analytical approach to calculate the risk probability, but to solve the time-varying PDE and get the actual probability value is non-trivial.
- *Noisy estimation of probability gradients.* Estimating gradients of risk probabilities is difficult, as sampling noise is amplified by the infinitesimal divisor while taking derivatives over the estimated risk probabilities.

To resolve the abovementioned issues, we propose a physics-informed learning framework PIPE (Physics-Informed Probability Estimator) to estimate risk probabilities in an efficient and generalizable manner. Fig. 1 shows the overview diagram of the proposed PIPE framework. The framework takes both data and physics models into consideration, and by combining the two, we achieve better sample efficiency and the ability to generalize to unseen regions in the state space and unknown parameters in the system dynamics. The use of deep neural networks enables the efficient learning of complex PDEs, and the consideration of physics models further enhances efficiency as it allows imperfect noisy data for training. The resulting framework takes only inaccurate sample data in a sparse sub-region of the state space for training, and is able to accurately predict the risk probability over the whole state space for systems with different parameters.

2. Related Works

2.1. Stochastic safe control

Stochastic safe control is a heated topic in recent years, as safety under uncertainties and noises becomes the key challenge of many real-world autonomous systems. Stochastic reachability analysis

takes the stochastic dynamics model and forward rollouts the possible trajectories to get the safety probability of any given state, and use this information to design suitable safe controllers (Abate et al., 2008; Prandini and Hu, 2008; Patil and Tanaka, 2022). Conditional Value-at-Risk considers the risk measure of the system and guarantees the expected risk value to always decrease conditioned on the previous states of the system, and thus guarantees safety (Samuelson and Yang, 2018; Singletary et al., 2022). Chance-constrained predictive control takes probabilistic safety requirements as the constraints in an optimization-based controller, and solves a minimal distance control to a nominal controller to find its safe counterpart (Nakka et al., 2020; Zhu and Alonso-Mora, 2019; Pfrommer et al., 2022). While these methods provide theoretical guarantees on safety, all of them require accurate estimation of risk probabilities or their gradients to yield desirable performance, and to get accurate estimates itself challenging. We tackle this problem by combining physics models and data to provide accurate estimates of risk probability and its gradient.

2.2. Rare event simulation

Rare event simulation considers the problem of estimating the probability of rare events in the system, and is highly related to risk probability estimation because the risk probability is often small in safety-critical systems. Here, we list a few widely adopted approaches for rare event simulation. Standard MC forward runs the system dynamics multiple times to empirically estimate the risk probability by calculating the unsafe trajectory numbers over the total trajectory number (Rubino and Tuffin, 2009). Standard MC is easy to implement, but needs huge amounts of sample trajectories to get accurate estimation, which becomes impractical when the required accuracy is high. Importance sampling methods calculate the risk probability on a shifted new distribution to improve the sample efficiency, but needs good prior information on the re-sampled distribution for reasonable performance enhancement, which is hard to achieve for complex systems (Cérou et al., 2012; Botev et al., 2013). Subset simulation calculates the risk probability conditioned on intermediate failure events that are easier to estimate, to further improve sample efficiency (Au and Beck, 2001). However, computational efficiency remains an issue and generalization to the entire state space is hard to achieve, as the estimation can only be conducted at a single point once (Zuev, 2015). There are no known methods that can compute the risk probability in an integrated way for the entire state space, and to do so with high sample efficiency. We address this problem by proposing a learning framework that considers both data and model to give generalizable prediction results with high sample efficiency.

2.3. Physics-informed neural networks

Physics-informed neural networks (PINNs) are neural networks that are trained to solve supervised learning tasks while respecting any given laws of physics described by general nonlinear partial differential equations (Raissi et al., 2019). PINNs take both data and the physics model of the system into account, and are able to solve the forward problem of getting PDE solutions, and the inverse problem of discovering underlying governing PDEs from data. PINNs have been widely used in power systems (Misyris et al., 2020), fluid mechanics (Cai et al., 2022) and medical care (Sahli Costabal et al., 2020). For stochastic safe control, previous works use PINNs to solve the initial value problem of deep backward stochastic differential equations to derive an end-to-end myopic safe controller (Han et al., 2018; Pereira et al., 2021). However, to the best of our knowledge there is no work that considers PINNs for risk probability estimation, especially on the full

state-time space scale. In this work, we take the first step towards leveraging PINNs on the problem of risk probability estimation.

3. Problem Formulation

We consider a nonlinear stochastic control system dynamics characterized by the following stochastic differential equation (SDE):

$$dx = f(x, u)dt + \sigma dw, \tag{1}$$

where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the control input, w_t is a n-dimensional Wiener process starting from $w_0 = 0$ and σ is the magnitude of the noise. We assume that function f is parameterized by some parameter λ. Safety of the system is defined as the state staying within a safe set $\mathcal{C}$, which is the super-level set of a function $\phi(x) : \mathbb{R}^n \to \mathbb{R}$, *i.e.,*

$$\mathcal{C} = \{x \mid \phi(x) \geq 0\}. \tag{2}$$

This definition of safety can characterize a large variety of practical safety requirements (Prajna et al., 2007; Ames et al., 2019). For the stochastic system (1), since safety can only be guaranteed in the sense of probability, we consider the long-term safety probability F_s and recovery probability F_r of the system defined as below.

Definition 1 (Safety probability) *Starting from initial state $x_0 = x \in \mathcal{C}$, the safety probability F_s of system (1) for outlook time horizon T is defined as the probability of state x_t staying in the safe set $\mathcal{C}$ over the time interval $[0, T]$, i.e.,*

$$F^s(x, T) = \mathbb{P}(x_t \in \mathcal{C}, \forall t \in [0, T] \mid x_0 = x). \tag{3}$$

Definition 2 (Recovery probability) *Starting from initial state $x_0 = x \notin \mathcal{C}$, the recovery probability F_r of system (1) for outlook time horizon T is defined as the probability of state x_t to get back to the safe set during the time interval $[0, T]$, i.e.,*

$$F^r(x, T) = \mathbb{P}(\exists t \in [0, T], x_t \in \mathcal{C} \mid x_0 = x). \tag{4}$$

Both safety probability and recovery probability are specific realizations of risk probability, depending on whether the initial state is safe or not and whether the safe or unsafe events are of interest in the system (they are complementary). In the rest of the paper, we will denote risk probability as F, and it refers to safety probability or recovery probability depending on the initial state of the system is within the safe set or not.

Remark 3 *The risk probability defined above characterizes the long-term behaviour of the stochastic system. In many literature, people may consider Gaussian approximation to measure the risk at each time step (Liu and Tomizuka, 2015; Akametalu et al., 2014). However, if we know the probability of risk at each time step being ϵ, then the probability bound of risk for k time steps will be $1 - (1 - \epsilon)^k \to 1$. This value is very conservative because the derivation does not capture the dynamic relationships between time steps. Previous studies also consider approximations of long-term safety, e.g., supermartingale (Prajna et al., 2007), Chebyshev's inequality (Boucheron et al., 2013; Farokhi et al., 2021) and Boole's inequality (Li et al., 2021). Those approximations allows efficient calculations but can be conservative as well.*

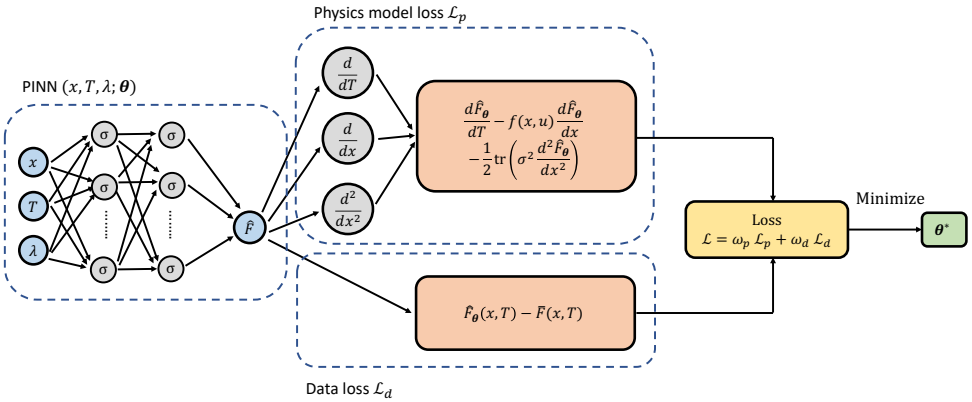

Figure 2: The training scheme of the physics-informed neural network (PINN) in PIPE.

The value of the risk probability F over the state space is crucial for safe control of these systems, but in practice it is hard to obtain all risk probability information accurately, *e.g.,* the failure probability of manufacturing robot arm is very small and thus hard to estimate (Lasota et al., 2014). In this paper, the goal is to accurately estimate the risk probability and its gradient over the entire state space, and to achieve adaptation in changing system dynamics. Specifically, we want to find the mapping between the state-time pair (x, T) and the risk probability F over the entire state-time space $\mathbb{R}^n \times \mathbb{R}$ for any system parameter λ, and to estimate the gradient of risk probability F with regard to the state x. We list four specific goals for the problem, generalization to unseen regions in the state-time space, efficient estimation with fixed number of sample data, adaptation on changing system parameters, and accurate estimation of probability gradients.

4. Proposed Method

In this section, we propose a model-based data-driven approach, PIPE, to efficiently estimate risk probabilities. We first introduce a PDE whose solution will characterize the risk probability. Then we combine MC data and the PDE model to form a physics-informed neural network to learn the risk probability. The PIPE formulation enjoys advantages from both PDE and MC, and achieves better performance and efficiency than any of the method alone.

Theorem 4 *(Chern et al., 2021) For a stochastic control affine system in the form of (1), the risk probability F (for both safety probability F^s defined in Definition 1 and recovery probability F^r defined in Definition 2) is characterized by the following convection diffusion equation,*

$$W_F(x,T) := \frac{\partial F}{\partial T}(x,T) - f(x,u)\frac{\partial F}{\partial x}(x,T) - \frac{1}{2}\,\mathrm{tr}\left(\sigma^2 \frac{\partial^2 F}{\partial x^2}(x,T)\right) = 0, \tag{5}$$

with initial condition $F(x,0) = \mathbb{1}(x \in \mathcal{C})$. For safety probability, the boundary condition is $F^s(x,T) = 0,\ x \notin \mathcal{C}$. For recovery probability, the boundary condition is $F^r(x,T) = 1,\ x \in \partial\mathcal{C}$.

Theorem 4 states that the risk probability of a control system can be analytically expressed as a PDE. The PDE consists of a convection term and a diffusion term. The convection term characterizes how the risk probability changes as the system evolves under its deterministic part of the dynamics. The diffusion term characterizes the effect of stochastic noises on the risk probability

value as the noise term in the dynamics diffuses with time. The initial condition says when the outlook time T is 0, the risk probability value is the indicator function of whether the state is within the safe set. The boundary condition says that on the boundary of safe set $\mathcal{C}$, the risk probability F is 0 if we consider safety probability, and is 1 if we consider recovery probability. We use $W_F(x,T)$ to denote the function value that the PDE risk probability F should satisfy, to better define the loss function in the learning framework later.

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 (5). 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 (Paszke et al., 2019) and TensorFlow (Abadi et al., 2016). 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

$$\mathcal{L}(\boldsymbol{\theta}) = \omega_p \mathcal{L}_p(\boldsymbol{\theta}) + \omega_d \mathcal{L}_d(\boldsymbol{\theta}), \tag{6}$$

where

$$\begin{aligned} \mathcal{L}_p(\boldsymbol{\theta}) &= \frac{1}{|\mathcal{P}|} \sum_{(x,T)\in\mathcal{P}} \|W_{\hat{F}_{\boldsymbol{\theta}}}(x,T)\|_2^2, \\ \mathcal{L}_d(\boldsymbol{\theta}) &= \frac{1}{|\mathcal{D}|} \sum_{(x,T)\in\mathcal{D}} \|\hat{F}_{\boldsymbol{\theta}}(x,T) - \bar{F}(x,T)\|_2^2. \end{aligned} \tag{7}$$

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.

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.

5. Performance Analysis

In this section, 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 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 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

$$\sup_{(x,T)\in D} |F_{\bar{\boldsymbol{\theta}}}(x,T) - u(x,T)| \geq M. \tag{8}$$

Theorem 6 *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. $\mathbb{E}_{\mathbf{Y}}[|F_{\boldsymbol{\theta}}(\mathbf{Y}) - \tilde{u}(\mathbf{Y})|] < \delta_1$, *where $\mathbf{Y}$ is uniformly sampled from Σ*
2. $\mathbb{E}_{\mathbf{X}}[|W_{F_{\boldsymbol{\theta}}}(\mathbf{X})|] < \delta_2$, *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

$$\sup_{(x,T)\in D} |F_{\boldsymbol{\theta}}(x,T) - u(x,T)| \leq C_1\delta_1 + C_2\frac{\delta_2}{\lambda} \tag{9}$$

where C_1, C_2 are constants depending on D, Σ, and W.

See extended version of this paper for proofs (Wang and Nakahira, 2023). Corollary 5 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 6 says that when the PDE constraint is imposed on the full space-time domain, the prediction of the PINN has bounded error, and the error scales linearly with the training loss.

6. Experiments

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

$$dx = \lambda\, dt + dw. \tag{10}$$

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, the risk probability F is characterized by the solution of the following convection diffusion equation

$$\frac{\partial F}{\partial T}(x,T) = \lambda\frac{\partial F}{\partial x}(x,T) + \frac{1}{2}\operatorname{tr}\left(\frac{\partial^2 F}{\partial x^2}(x,T)\right), \tag{11}$$

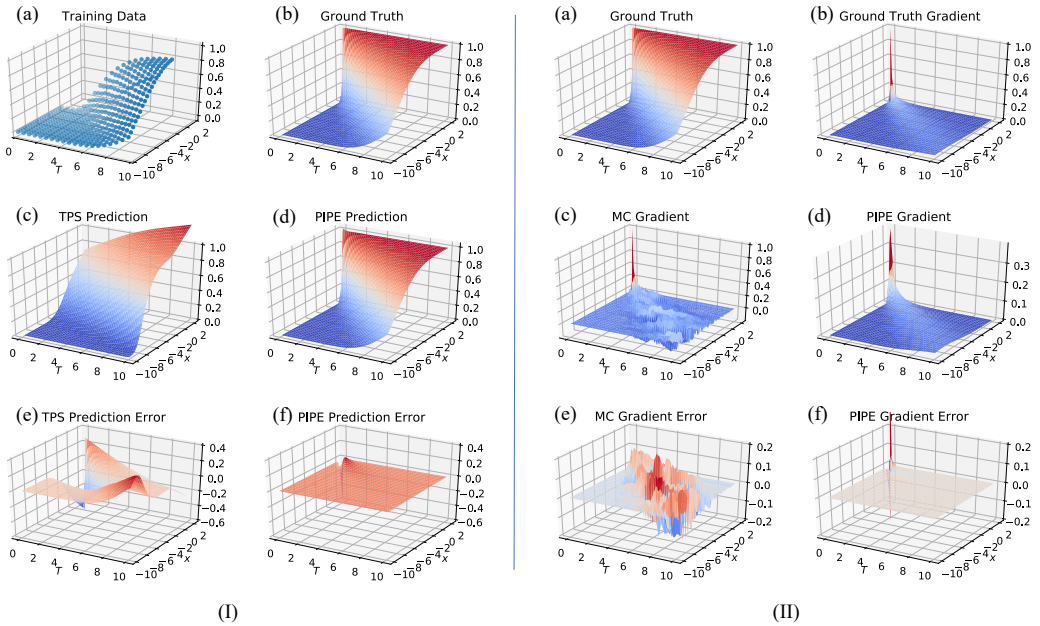

Figure 3: (I) Settings and results of the risk probability generalization task. PIPE and TPS fitting are compared. The average absolute error of prediction is 9.2×10^{-2} for TPS, and 0.3×10^{-2} for PIPE. (II) Gradient of the risk probability prediction of PIPE and MC. The average absolute error of gradient prediction is 2.78×10^{-2} for MC, and 0.06×10^{-2} for PIPE.

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 (11) as ground truth for comparison, as given by $F(x,T) = (2-x)\exp\left\{\frac{-((2-x)-\lambda T)^2}{2T}\right\}/\sqrt{2\pi T^3}$. 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 the number of trajectories where the state recovers to the safe set during the time horizon $[0,T]$ over the full trajectory number, *i.e.*, $\bar{F}(x,T) = \mathbb{P}(\exists t \in [0,T], x_t \in \mathcal{C} \mid x_0 = x) = \frac{N_{\text{recovery}}}{N}$, where N is the 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 (Kingma and Ba, 2014) 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 (6) 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 (Lu et al., 2021). Experiment details, simulation results on other systems, and applications to stochastic safe control can be found in the extended version of this paper (Wang and Nakahira, 2023). Codes are available at: https://github.com/jacobwang925/PIPE-L4DC.

6.1. 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 (10) with $\lambda = 1$. We train the PINN with data only on the sub-region of the state-time space $\Omega \times \tau = [-10,-2] \times [0,10]$, 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

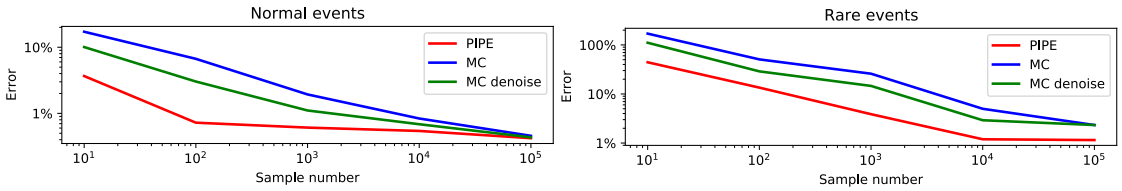

Figure 4: Percentage error of risk probability estimation for different MC sample numbers for rare events and normal events. PIPE, MC and denoised MC with uniform kernel filtering are compared. Both error and sample number are in log scale.

number $N = 1000$, and is down-sampled to $dx = 0.4$ and $dT = 0.5$. For comparison, we use thin plate spline (TPS) fitting on the training data to infer the risk probability on the whole state space. We also examined other fitting methods such as cubic spline and polynomial fitting, but TPS performs the best over all fitting strategies. Fig. 3 (I) visualizes the training data samples and shows the results. The spline fitting does not include any physical model constraint, thus fails to generalize to unseen regions in the state space. On the contrary, PIPE can infer the risk probability value very accurately on the whole state space due to its combination of data and the physics model.

6.2. 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 (10) with $\lambda = 1$. The training data is sampled on the state-time space $\Omega \times \tau = [-10, 2] \times [0, 10]$ with $dx = 0.2$ 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.

6.3. Adaptation on changing system parameters

In this experiment, we show that PIPE will allow generalization to uncertain parameters of the system. We consider system (10) 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 $\lambda_{\text{test}} = [0.3, 0.7, 1.2, 1.5, 2]$. We only present $\lambda_{\text{test}} = 1.5$ due to space limitation, but similar results on different λ_{test} can be found at the project webpage. Fig. 5 shows the results. We can see that PIPE

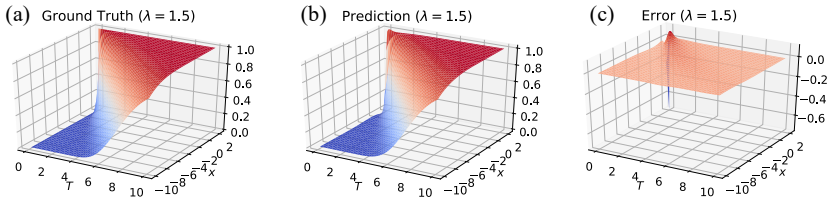

Figure 5: Risk probability prediction of PIPE on unseen system parameters. The average absolute error of the prediction is 0.70×10^{-2}.

is able to predict risk probability for systems with unseen and even out of distribution parameters during training. In the prediction error plot, the only region that 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.

6.4. 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 (10) with $\lambda = 1$. Similar to the generalization task, we train the PINN with MC data of $N = 1000$ on the sub-region $\Omega \times \tau = [-10, -2] \times [0, 10]$ 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 $\hat{F}_{\boldsymbol{\theta}}$. Fig. 3 (II) shows the results. It can be seen that PIPE gives much more accurate estimation of the risk probability gradient than MC, and this is due to the fact that PIPE incorporates physics model information inside the training process. It is also interesting 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 estimation of the gradient. The results indicates that PIPE can enable the usage a lot of first- and higher-order stochastic safe control methods online, by providing accurate and fast estimation of the risk probability gradients.

7. Conclusion

In this paper, we proposed a generalizable physics-informed learning framework, PIPE, to estimate the risk probability in stochastic safe control systems. The PIPE framework combines data from Monte Carlo and the underlying governing PDE of the risk probability, to accurately learn the risk probability as well as its gradient. PIPE has better sample efficiencies compared to MC, and is able to generalize to unseen regions in the state space beyond training. The resulting PIPE 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. The proposed PIPE framework provides key foundations for first- and higher-order methods for stochastic safe control, and opens the door for robust physics-informed learning for generic PDEs. Future work includes applications to high-dimensional and real-world systems.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

Alessandro Abate, Maria Prandini, John Lygeros, and Shankar Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.

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

Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. IEEE, 2019.

Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic engineering mechanics*, 16(4):263–277, 2001.

Zdravko I Botev, Pierre L'Ecuyer, and Bruno Tuffin. Markov chain importance sampling with applications to rare event probability estimation. *Statistics and Computing*, 23(2):271–285, 2013.

Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (pinns) for fluid mechanics: A review. *Acta Mechanica Sinica*, pages 1–12, 2022.

Frédéric Cérou, Pierre Del Moral, Teddy Furon, and Arnaud Guyader. Sequential monte carlo for rare event estimation. *Statistics and computing*, 22(3):795–808, 2012.

Margaret P Chapman, Jonathan Lacotte, Aviv Tamar, Donggun Lee, Kevin M Smith, Victoria Cheng, Jaime F Fisac, Susmit Jha, Marco Pavone, and Claire J Tomlin. A risk-sensitive finite-time reachability approach for safety of stochastic dynamic systems. In *2019 American Control Conference (ACC)*, pages 2958–2963. IEEE, 2019.

Albert Chern, Xiang Wang, Abhiram Iyer, and Yorie Nakahira. Safe control in the presence of stochastic uncertainties. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 6640–6645. IEEE, 2021.

William S Cleveland. Lowess: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*, 35(1):54, 1981.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

Farhad Farokhi, Alex Leong, Iman Shames, and Mohammad Zamani. Safe learning of uncertain environments. *arXiv preprint arXiv:2103.01413*, 2021.

David Gilbarg, Neil S Trudinger, David Gilbarg, and NS Trudinger. *Elliptic partial differential equations of second order*, volume 224. Springer, 1977.

Jiequn Han, Arnulf Jentzen, and Weinan E. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences*, 115(34):8505–8510, 2018.

Hans Janssen. Monte-carlo based uncertainty analysis: Sampling efficiency and sampling convergence. *Reliability Engineering & System Safety*, 109:123–132, 2013.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Przemyslaw A Lasota, Gregory F Rossano, and Julie A Shah. Toward safe close-proximity human-robot interaction with standard industrial robots. In *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 339–344. IEEE, 2014.

Nan Li, Ilya Kolmanovsky, and Anouck Girard. An analytical safe approximation to joint chance-constrained programming with additive gaussian noises. *IEEE Transactions on Automatic Control*, 66(11):5490–5497, 2021.

Changliu Liu and Masayoshi Tomizuka. Safe exploration: Addressing various uncertainty levels in human robot interactions. In *2015 American Control Conference (ACC)*, pages 465–470. IEEE, 2015.

Lu Lu, Xuhui Meng, Zhiping Mao, and George Em Karniadakis. Deepxde: A deep learning library for solving differential equations. *SIAM Review*, 63(1):208–228, 2021.

George S Misyris, Andreas Venzke, and Spyros Chatzivasileiadis. Physics-informed neural networks for power systems. In *2020 IEEE Power & Energy Society General Meeting (PESGM)*, pages 1–5. IEEE, 2020.

Yashwanth Kumar Nakka, Anqi Liu, Guanya Shi, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung. Chance-constrained trajectory optimization for safe exploration and learning of nonlinear systems. *IEEE Robotics and Automation Letters*, 6(2):389–396, 2020.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Apurva Patil and Takashi Tanaka. Upper bounds for continuous-time end-to-end risks in stochastic robot navigation. In *2022 European Control Conference (ECC)*, pages 2049–2055. IEEE, 2022.

Wei Peng, Weien Zhou, Jun Zhang, and Wen Yao. Accelerating physics-informed neural network training with prior dictionaries. *arXiv preprint arXiv:2004.08151*, 2020.

Marcus Pereira, Ziyi Wang, Ioannis Exarchos, and Evangelos Theodorou. Safe optimal control using stochastic barrier functions and deep forward-backward sdes. In *Conference on Robot Learning*, pages 1783–1801. PMLR, 2021.

Samuel Pfrommer, Tanmay Gautam, Alec Zhou, and Somayeh Sojoudi. Safe reinforcement learning with chance-constrained model predictive control. In *Learning for Dynamics and Control Conference*, pages 291–303. PMLR, 2022.

Stephen Prajna, Ali Jadbabaie, and George J Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8): 1415–1428, 2007.

Maria Prandini and Jianghai Hu. Application of reachability analysis for stochastic hybrid systems to aircraft conflict prediction. In *2008 47th IEEE Conference on Decision and Control*, pages 4036–4041. IEEE, 2008.

Maziar Raissi, Paris Perdikaris, and George 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*, 378:686–707, 2019.

Gerardo Rubino and Bruno Tuffin. *Rare event simulation using Monte Carlo methods*. John Wiley & Sons, 2009.

Francisco Sahli Costabal, Yibo Yang, Paris Perdikaris, Daniel E Hurtado, and Ellen Kuhl. Physics-informed neural networks for cardiac activation mapping. *Frontiers in Physics*, 8:42, 2020.

Samantha Samuelson and Insoon Yang. Safety-aware optimal control of stochastic systems using conditional value-at-risk. In *2018 Annual American Control Conference (ACC)*, pages 6285–6290. IEEE, 2018.

Cesar Santoyo, Maxence Dutreix, and Samuel Coogan. A barrier function approach to finite-time stochastic system verification and control. *Automatica*, 125:109439, 2021.

Andrew Singletary, Mohamadreza Ahmadi, and Aaron D Ames. Safe control for nonlinear systems with stochastic uncertainty via risk control barrier functions. *arXiv preprint arXiv:2203.15892*, 2022.

Zhuoyuan Wang and Yorie Nakahira. A generalizable physics-informed learning framework for risk probability estimation. *arXiv preprint arXiv:2305.06432*, 2023.

Zhuoyuan Wang, Haoming Jing, Christian Kurniawan, Albert Chern, and Yorie Nakahira. Myopically verifiable probabilistic certificate for long-term safety. *arXiv preprint arXiv:2110.13380*, 2021.

Hai Zhu and Javier Alonso-Mora. Chance-constrained collision avoidance for mavs in dynamic environments. *IEEE Robotics and Automation Letters*, 4(2):776–783, 2019.

Konstantin Zuev. Subset simulation method for rare event estimation: an introduction. *arXiv preprint arXiv:1505.03506*, 2015.