Scalable Importance Sampling in High Dimensions with Low-Rank Mixture Proposals

Liam A. Kruse, Marc R. Schlichting, and Mykel J. Kochenderfer

***Abstract*—Importance sampling is a Monte Carlo technique for efficiently estimating the likelihood of rare events by biasing the sampling distribution towards the rare event of interest. By drawing weighted samples from a learned *proposal* distribution, importance sampling allows for more sample-efficient estimation of rare events or tails of distributions. A common choice of proposal density is a Gaussian mixture model (GMM). However, estimating full-rank GMM covariance matrices in high dimensions is a challenging task due to numerical instabilities. In this work, we propose using mixtures of probabilistic principal component analyzers (MPPCA) as the parametric proposal density for importance sampling methods. MPPCA models are a type of low-rank mixture model that can be fit quickly using expectation-maximization, even in high-dimensional spaces. We validate our method on three simulated systems, demonstrating consistent gains in sample efficiency and quality of failure distribution characterization.**

I. Introduction

Safety-critical applications such as aircraft controller design or autonomous driving heavily rely on simulations to identify potential failures before real-world deployment. Rigorous safety validation frameworks can characterize failure modes in a controlled simulation environment, reducing the risk of accidents [1]. However, failure events such as collisions or loss of vehicle control might be rarely encountered in simulation due to strict safety thresholds, as shown in Fig. 1. *Importance sampling* (IS) reduces the variance of Monte Carlo failure estimates by biasing the sampling distribution towards the rare event of interest [1], [2]. IS uses a *proposal distribution* to concentrate computational effort on scenarios likely to yield failure events, and assigns *importance weights* to sampled points to correct for the biased distribution. This efficient sampling technique reduces the variance of failure probability estimates compared to direct Monte Carlo sampling.

A common choice of parametric proposal density is a Gaussian mixture model (GMM) [3]. However, learning full-rank covariance matrices in high dimensions, especially when the number of dimensions is greater than the size of the dataset, is a challenging task. The estimate might overfit to the noise in the dataset, and the matrices themselves become ill-conditioned or singular [4]. Furthermore, the memory required to store the matrices grows quadratically with the number of dimensions.

One solution is to constrain the learned covariance matrices to be low-rank, effectively modeling the covariances as full-rank matrices on a learned low-dimensional subspace [5]. In this work, we propose using mixtures of probabilistic principal component analyzers (MPPCA) as the parametric proposal density. MPPCA models offer two key characteristics: 1) they have an analytical likelihood expression, preserving the computational efficiency required for computing importance weights, and 2) they can be fit efficiently even on high-dimensional data using the expectation-maximization (EM) algorithm. Our specific contributions include the following:

- We construct expressive importance sampling proposal distributions using MPPCA models and demonstrate that the MPPCA fitting procedure is tractable even in high-dimensional domains.
- We quantitatively evaluate our approach on simulated importance sampling tasks, including safety validation for an aircraft ground collision avoidance system.

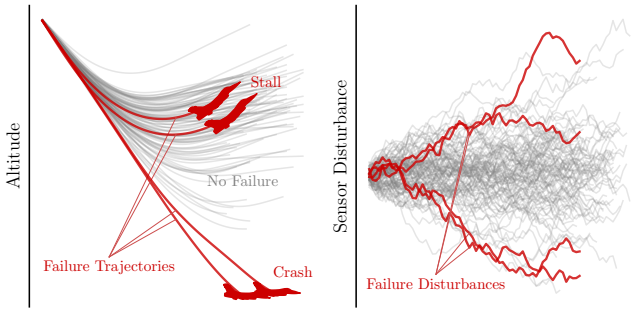

Fig. 1. Safety-critical systems often have extremely low failure rates. Failures can arise from a sequence of disturbances, making the problem high-dimensional. While failures can be multimodal, the number of failure modes is typically much lower than the disturbance dimensionality. This motivates our approach of using low-rank mixture proposals to approximate the distribution over disturbances.

II. Related Work

Importance sampling has received widespread interest in applications such as structural reliability analysis [3], [6], safety validation for autonomous vehicles [7], [8], and estimating the probability of failure for aircraft collision avoidance systems [9]. Many IS methods such as the cross-entropy method or population Monte Carlo iteratively adapt the proposal distribution to move towards the failure distribution [10]. The proposal distribution is often constrained to a parametric set of probability distributions such as the exponential family [7], [10], [11]. Parametric proposals admit analytical updates, which is computationally advantageous when refining the proposal to more closely match the failure distribution [1]. The importance weights can be computed in closed form when

L. A. Kruse, M. R. Schlichting, and M. J. Kochenderfer are with the Stanford Intelligent Systems Laboratory in the Department of Aeronautics and Astronautics at Stanford University, Stanford, CA 94305, USA (email: {lkruse, mschl, mykel}@stanford.edu).

the proposal density has an analytical likelihood expression. However, iteratively refining proposals in high dimensions is challenging because the importance weights degenerate as the problem dimensionality increases [12]. Furthermore, full-rank covariance estimates can become numerically ill-conditioned in high dimensions, especially if the number of samples is small [4]. Neural network-based IS methods often scale more effectively to higher dimensions. Demange-Chryst *et al.* [13] approximate the failure distribution using a variational autoencoder, which requires iterative re-training throughout the IS procedure. Similarly, Delecki *et al.* [14] employ a denoising diffusion model as the proposal distribution, training it through a cross-entropy-like approach. While these methods produce high-fidelity results and perform well in high-dimensional settings, they are computationally expensive. In this work, we use mixtures of probabilistic principal component analyzers as the parametric proposal density. MPPCA models perform local linear dimensionality reduction, promoting scalable performance even in high dimensions.

Non-parametric approaches to IS offer flexible failure distribution representations that are not constrained to a specific distribution family. Sequential Monte Carlo methods can represent complex, multimodal failure distributions with a collection of samples [10], [15]. Multilevel-splitting relies on Markov chain Monte Carlo (MCMC) estimation to guide a series of conditional distributions towards the failure distribution [16], [17]. However, MCMC can require many iterations to find all failure modes and accurately reflect the failure density [10]. In general, non-parametric strategies do not admit closed-form updates. We use MPPCA proposal densities because they can be analytically updated using the EM algorithm, resulting in a computationally efficient search over the space of sampling distributions. Furthermore, we demonstrate that mixtures with even a small number of components are sufficiently expressive to model a range of failure modes.

Of particular interest to this work is the scalability of Gaussian mixture models. Two common frameworks for modeling low-rank GMMs include mixtures of probabilistic principal component analyzers [5] and mixtures of factor analyzers (MFAs) [18]. In this work, we focus on MPPCA models because they can be fit with closed-form expectation-maximization updates. Richardson and Weiss [19] use MFA models for image generation, demonstrating that low-rank GMMs can be trained on full-sized images despite the high dimensionality. Covariance matrix adaptation (CMA) is an evolutionary optimization algorithm that iteratively adapts a covariance matrix to improve sample efficiency [20]. Low-rank updates make the optimization process robust and sample efficient.

III. Importance Sampling and MPPCA

We next outline the theory behind importance sampling and mixtures of probabilistic principal component analyzers.

A. Importance Sampling

Simulations allow engineers to evaluate the performance of algorithms and models in diverse scenarios, including scenarios that are rare, dangerous, or costly to replicate in real-world testing. Assessing the probability of failure events can require a prohibitively large number of Monte Carlo simulations, especially if the event of interest is rare.

Consider an outcome space $\mathbf{x} \in \mathbb{R}^d$ with probability density function $p(\mathbf{x})$ and a *cost function* $f(\mathbf{x})$ such that a *failure event* occurs if and only if $f(\mathbf{x}) \leq 0$. The probability of failure P_F is given by the integral

$$P_F = \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{1}\{f(\mathbf{x}) \leq 0\}\right] = \int \mathbb{1}\{f(\mathbf{x}) \leq 0\} \cdot p(\mathbf{x}) d\mathbf{x} \quad (1)$$

We can estimate P_F via Monte Carlo simulations by drawing N_s samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_s}\}$ from $p(\mathbf{x})$ and taking the mean:

$$\hat{P}_F = \frac{1}{N_s} \sum_{n=1}^{N_s} \mathbb{1}\{f(\mathbf{x}_n) \leq 0\}. \quad (2)$$

This estimate is unbiased and has a coefficient of variation

$$\delta_{\hat{P}_F} = \sqrt{\frac{1 - P_F}{N_s P_F}}. \quad (3)$$

Since the coefficient of variation is inversely proportional to the failure probability, many samples might be required to precisely estimate P_F, especially if P_F is small [6].

Importance sampling is a technique that aims to reduce the variance of $\hat{P}_F$ by sampling from an alternative sampling distribution—or *proposal distribution*—denoted by $q(\mathbf{x})$. So long as the support of $q(\mathbf{x})$ contains the failure domain, the probability of failure integral in Eq. (1) can be rewritten as

$$P_F = \int \frac{\mathbb{1}\{f(\mathbf{x}) \leq 0\} \cdot p(\mathbf{x})}{q(\mathbf{x})} \cdot q(\mathbf{x}) d\mathbf{x}. \quad (4)$$

The importance sampling estimate of P_F is given by

$$\hat{P}_F = \frac{1}{N_s} \sum_{n=1}^{N_s} \mathbb{1}\{f(\mathbf{x}_n) \leq 0\} \cdot \frac{p(\mathbf{x}_n)}{q(\mathbf{x}_n)} \quad (5)$$

where the samples are distributed according to the proposal distribution $q(\mathbf{x})$. An appropriate choice of proposal distribution can therefore reduce the variance of the estimate of P_F.

B. Low-Rank Mixture Models

Principal component analysis (PCA) is a classic statistical technique for dimensionality reduction. Tipping and Bishop [5] reformulate PCA within a maximum likelihood framework, resulting in an associated probability density. Consider the following latent variable model:

$$\mathbf{x} = \mathbf{W}\mathbf{z} + \mu + \epsilon \quad (6)$$

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (7)$$

$$\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}) \quad (8)$$

where $\mathbf{W}$ is a rectangular *factor loading* matrix of size $d \times \ell$ and ℓ is a latent dimension such that $\ell \ll d$. The latent vector $\mathbf{z}$ is of length ℓ, the mean vector μ is of length d, and ϵ is added

noise with diagonal covariance $\sigma^2\mathbf{I}$. For MPPCA, the noise is assumed to be isotropic; in the more general MFA framework, the noise assumption is relaxed to merely be diagonal. In the case of isotropic noise, the implied conditional distribution is

$$p(\mathbf{x} \mid \mathbf{z}) = \left(2\pi\sigma^2\right)^{-d/2} \exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{x} - \mathbf{W}\mathbf{z} - \mu\|^2\right\} \quad (9)$$

The Gaussian prior over the latent variables is given by

$$p(\mathbf{z}) = (2\pi)^{-\ell/2} \exp\left\{-\frac{1}{2}\mathbf{z}^\top\mathbf{z}\right\} \quad (10)$$

Thus, the marginal density over $\mathbf{x}$ can be expressed as

$$p(\mathbf{x}) = \eta|\mathbf{C}|^{-1/2} \exp\left\{-\frac{1}{2}(\mathbf{x} - \mu)^\top \mathbf{C}^{-1}(\mathbf{x} - \mu)\right\} \quad (11)$$

with normalizing constant $\eta = (2\pi)^{-d/2}$ and model covariance $\mathbf{C} = \sigma^2\mathbf{I} + \mathbf{W}\mathbf{W}^\top$. Tipping and Bishop [5] show that the likelihood of a dataset $\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is maximized when the columns of the factor loading matrix $\mathbf{W}$ span the principal subspace of the data. Equation (11) defines the marginal likelihood for a single probabilistic principal component analyzer (PPCA); multiple PPCA models can be combined in a mixture to model more complex distributions.

IV. Methodology

In this section, we present the two adaptive importance sampling methods used in our study. We also describe the analytical EM procedure for optimizing proposal distributions.

A. Importance Sampling Methods

The cross-entropy (CE) method [3], [21] attempts to learn the parameters of a parametric proposal distribution that minimizes the KL divergence between the optimal IS density and the proposal. CE importance sampling introduces a series of intermediate failure domains that gradually approach the true failure domain. At each step, the intermediate failure region is defined such that $\rho \cdot N_s$ samples fall in the region, where the ρ-quantile is chosen by the user. The proposal distribution parameters are then fit via maximum likelihood estimation over these samples. The expectation-maximization algorithm is often used to fit the search distribution, though it must be adjusted to account for importance-weighted samples [3].

Like the CE method, sequential importance sampling (SIS) introduces a series of intermediate failure distributions that gradually approach the optimal IS density [6], [15]. Samples for each intermediate distribution are obtained by resampling weighted particles from the previous distribution and then moved to regions of high likelihood under the next failure distribution through Markov chain Monte Carlo. In this work, we use a conditional sampling Metropolis–Hastings algorithm to move the samples [6]. The first samples in the Markov chains are discarded during a specified *burn-in* period, ensuring that the retained samples are drawn from the stationary distribution.

B. Fitting MPPCA Models

Consider a mixture of K probabilistic principal component analyzers as presented in Eq. (6). The log-likelihood of observing a dataset $\{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_N\}$ is

$$\mathcal{L} = \sum_{i=1}^{N}\left\{\sum_{k=1}^{K}\pi_k p(\mathbf{x}_n \mid k)\right\} \quad (12)$$

where $p(\mathbf{x} \mid k)$ represents the density of the kth PPCA model and π_k is the corresponding mixing proportion, with $\pi_k \geq 0$ and $\sum_{k=1}^{K}\pi_k = 1$. Tipping and Bishop [5] derive an iterative expectation-maximization procedure with closed-form updates for MPPCA parameters $\mathbf{W}_k$, μ_k, π_k, and σ_k^2 for components $k = 1, \ldots, K$ that is guaranteed to find a local maximum of Eq. (12). In classical EM parameter updates for a mixture model, the *responsibility* of component k for generating sample $\mathbf{x}_n$ is given by

$$r_{nk} = \frac{\pi_k p(\mathbf{x}_n \mid k)}{\sum_j \pi_k p(\mathbf{x}_n \mid j)} \quad (13)$$

However, we must adjust the responsibility calculation since we are sampling from the proposal distribution [3]. The importance sampling responsibilities are

$$r_{nk} = \frac{w_n\pi_k q(\mathbf{x}_n \mid k)}{\sum_j \pi_k q(\mathbf{x}_n \mid j)} \quad (14)$$

where $p(\mathbf{x}_n)$ is the prior density, $q(\mathbf{x}_n)$ is the proposal density, and $w_n = p(\mathbf{x}_n)/q(\mathbf{x}_n)$ is the *likelihood ratio*.

V. Experiments

This section presents our problem domains and evaluation metrics before discussing experimental results.

A. Data Simulators

We validate our proposed approach using three simulated environments. Each system is defined by a continuous cost function $f(\mathbf{x})$ that maps a d-dimensional sample $\mathbf{x}$ to a scalar value. Recall from Section III-B that $f(\mathbf{x}) \leq 0$ defines a failure event, while positive values indicate that the sample falls outside of the failure region. Larger values correspond to samples that are farther away from the failure region boundary.

1) Branches: Chiron *et al.* [22] develop the following analytical expression given for even dimensions with $d \geq 2$:

$$f(\mathbf{x}) = \min\left\{\begin{array}{c} \beta + \frac{1}{\sqrt{d}}\sum_{i=1}^{d} x_i \\ \beta - \frac{1}{\sqrt{d}}\sum_{i=1}^{d} x_i \\ \beta + \frac{1}{\sqrt{d}}\left(\sum_{i=1}^{d/2} x_i - \sum_{i=d/2+1}^{d} x_i\right) \\ \beta + \frac{1}{\sqrt{d}}\left(-\sum_{i=1}^{d/2} x_i + \sum_{i=d/2+1}^{d} x_i\right) \end{array}\right\}$$

This system of four cost functions has a probability of failure which is independent of the number of random variables. We set the parameter $\beta = 3.5$ to ensure that the probability of failure is small.

TABLE I
HYPERPARAMETERS

System	K	L	samples / iter.	trials	ρ
Branches ($d = 40$)	8	8	10,000	50	0.2
Branches ($d = 60$)	8	8	10,000	50	0.2
Oscillator ($d = 100$)	8	8	10,000	50	0.2
Oscillator ($d = 200$)	8	8	10,000	50	0.2
F-16 GCAS	8	8	10,000	50	0.2

2) Duffing Oscillator: The second example is the Duffing oscillator introduced by Zuev [23]. We consider its discretized form in the frequency domain as presented by Papaioannou, Geyer, and Straub [24]. The cost function is the maximal displacement $u(t)$ of the oscillator at $t_{\max} = 2$ seconds:

$$f(\mathbf{x}) = \min\{u_1 - u(t_{\max}), u(t_{\max}) - u_2\}$$

We define $u_1 = 0.1$ and $u_2 = -0.06$, corresponding to two failure modes. The displacement of the oscillator satisfies

$$\begin{aligned} &m\ddot{u}(t) + c\dot{u}(t) + k\left(u(t) + \gamma u(t)^3\right) \\ &= -m\sigma \sum_{i=1}^{d/2} \left(x_i \cos(\omega_i t) + x_{d/2+i} \sin(\omega_i t)\right), \end{aligned}$$

for all $t \geq 0$. We set $m = 1{,}000\,\mathrm{kg}$, $c = 200\pi\,\mathrm{Ns/m}$, $k = 1000(2\pi)^2\,\mathrm{N/m}$, $\gamma = 1\,\mathrm{m}^{-2}$, $\omega_i = i\Delta\omega$, $\Delta\omega = 30\pi/d$, and $\sigma = \sqrt{0.01\Delta\omega}$. The initial conditions are set to $u(0) = 0\,\mathrm{m}$ and $\dot{u}(0) = 1.5\,\mathrm{m/s}$ [13].

3) F-16 Ground Collision Avoidance: Finally, we simulate a diving F-16 fighter jet controlled by a ground collision avoidance system (GCAS). We use the dynamics model introduced by Heidlauf *et al.* [25] and the `JAX` code implementation[1] by So and Fan [26]. In this experiment, we model the effect of a short-term sensor drift in the roll and pitch angle sensors. The F-16 GCAS system is activated at 1,000 ft and consists of two sequential phases: 1) leveling the wings and 2) increasing the pitch angle until the aircraft is recovered from the dive. The initial roll and pitch angles are sampled from $\phi_0 \sim \mathcal{N}(0, 0.15^2)$ and $\theta_0 \sim \mathcal{N}(-0.52, 0.05^2)$, respectively, while the initial altitude is 950 ft. Roll and pitch sensor disturbances δ are modeled as a discrete-time approximation of the Wiener process $\delta_{t+1} \sim \delta_t + \epsilon$ with $\epsilon \sim \mathcal{N}(0, 0.01^2)$. There are two possible failure modes: a collision with the ground (i.e., altitude $h \leq 0$) or an aerodynamic stall (i.e., angle of attack $\alpha \geq \alpha_c$). The critical angle of attack α_c is 25 degrees for the F-16 model. The cost function for this experiment is

$$f(\mathbf{h}_{1:T}, \alpha_{1:T}) = \text{minimum}\left(\frac{1}{950}\min_t \mathbf{h}, \frac{1}{\alpha_c}\left(\alpha_c - \min_t \alpha\right)\right)$$

The total dimensionality of the disturbance space is $d = 202$.

B. Metrics

We obtain a reference failure probability for each dataset using the Monte Carlo estimate given in Eq. (2) and then calculate the importance sampling estimate $\hat{P}_F$ using Eq. (5). Next, we compute a series of metrics to evaluate the quality of the learned proposal density:

- *Relative error of $\hat{P}_F$*: This metric is defined as the signed difference between the IS estimate of the probability of failure and reference failure probability, normalized by the reference value. A positive value indicates that $\hat{P}_F$ is an overestimate of the true value. A *lower* absolute relative error indicates better performance.
- *Average negative log likelihood (NLL)*: An evaluation batch of samples $\hat{\mathbf{x}}$ is drawn from the learned proposal distribution, and for each sample the negative log-likelihood is computed under the prior distribution. A *lower* average NLL indicates that the samples are more likely under the prior distribution.
- *Coverage*: The coverage metric proposed by Naeem *et al.* [27] measures the proportion of real samples whose neighborhood (as defined by its k-nearest neighbors) contains at least one generated sample. A *higher* value indicates that more modes of the real samples are represented in the samples drawn from the learned proposal.
- *Number of statistically different bins (NDB)*: Richardson and Weiss [19] propose the NDB metric as a simple method to evaluate generative models based on relative proportions of samples that fall into C predetermined bins. A *lower* value indicates that learned distribution more closely represents the data distribution. We report NDB/C to normalize the values between 0 and 1.
- *Total number of samples* (N_{total}): The total number of samples is a proxy for the sample efficiency of each method. *Fewer* total samples indicate higher efficiency in finding an effective proposal distribution.

C. Experimental Setup

We evaluate MPPCA proposal densities against GMM proposals with full-rank covariance matrices. The proposals are iteratively refined using the CE and SIS methods until the data log-likelihood converges. We perform IS on the Branches problem with 40 and 60 dimensions, and on the Duffing Oscillator problem with 100 and 200 dimensions. Problem hyperparameters are provided in Table I. All experiments run on a CPU, but parallelizing trials can improve efficiency. Code to reproduce the experimental results is available at `https://github.com/sisl/MPPCAImportanceSampling`.

D. Results

Table II presents the experimental results across the five problem configurations. MPPCA proposals fit via the cross-entropy method (CE-MPPCA) obtain the smallest relative error of $\hat{P}_F$ on three of the five experimental configurations. MPPCA proposals fit via SIS obtain the smallest relative error on the remaining two problems. This indicates that IS with MPPCA proposals provide more reliable probability estimates compared to GMM proposals. Both CE-MPPCA and SIS-MPPCA score lower on the NDB/C metric than CE-GMM and SIS-GMM, which indicates that samples from the learned

[1] `https://github.com/MIT-REALM/jax-f16`

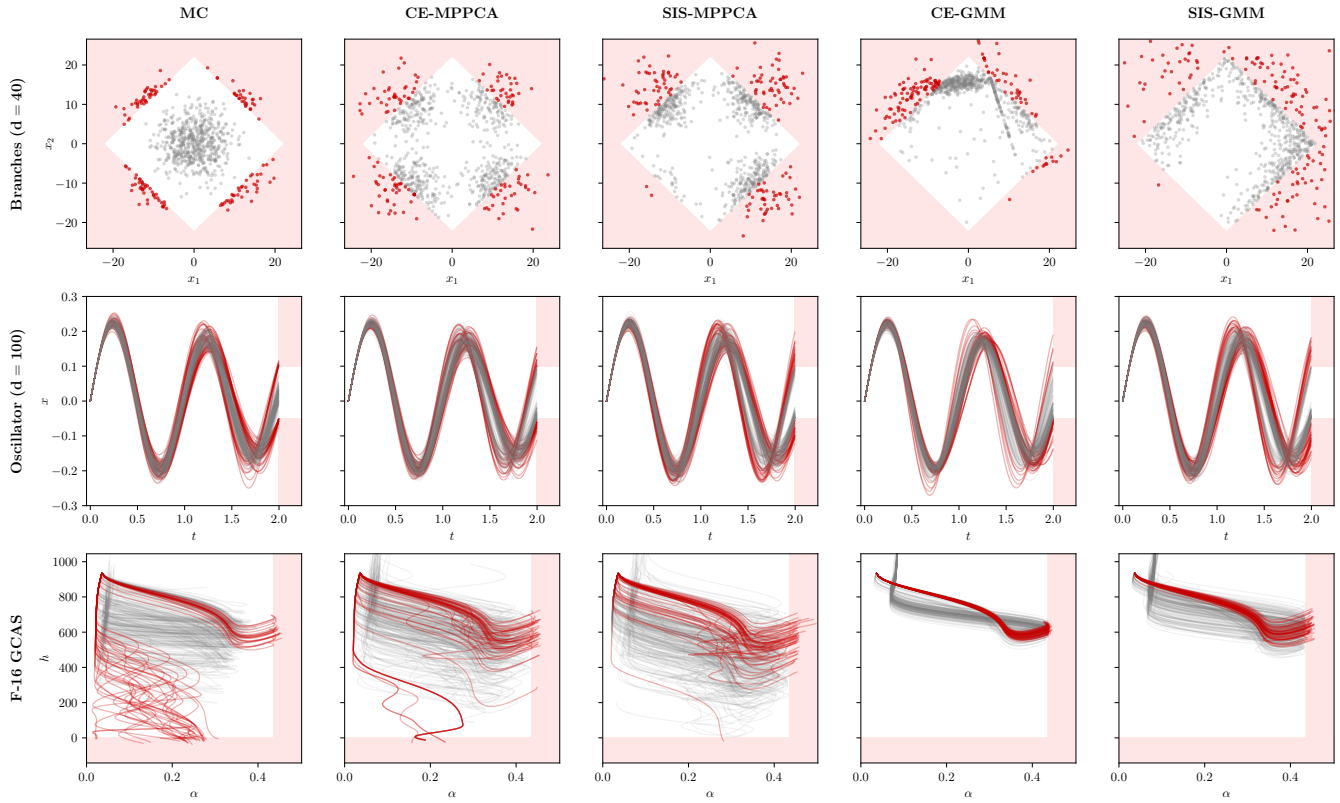

Fig. 2. Results for the Branches problem with $d = 40$ (top row), the Duffing oscillator with $d = 100$ (middle row), and the F-16 GCAS system (bottom row). Red samples represent failure events where $f(\mathbf{x}) \leq 0$, while gray samples represent outcomes where $f(\mathbf{x}) > 0$. Across all experiments, the ratio of failure samples to non-failure samples is fixed at $1/4$, regardless of the system's failure rate or the effectiveness of the method. The shaded red areas denote the failure regions where the cost function is ≤ 0.

MPPCA proposals are a closer match to samples from the true failure distributions. The performance of different methods varies more for the average NLL metric, with no single approach consistently outperforming the others in all cases. Note that both average NLL and coverage are computed in *disturbance space* for the oscillator and F-16 problems, e.g., the proposals are over system inputs rather than output trajectories. Even though the GMM proposals obtain higher average coverage scores on the F-16 problem, a visual inspection of Fig. 2 reveals that the GMMs experience a mode collapse. The learned GMM disturbance models only induce failures due to aerodynamic stalls, while the MPPCA proposals learn disturbance distributions that induce both stalls and ground collisions. The CE method converges with fewer samples than SIS, regardless of proposal density. This is likely because SIS relies on MCMC to gradually move samples towards the optimal failure distribution.

VI. Conclusion and Future Work

Importance sampling is a powerful technique for validating safety-critical systems [1]. In this work we present a technique to scale parametric IS methods using low-rank mixture proposal distributions. Mixtures of probabilistic principal component analyzers are parameter-efficient models that can be updated analytically via expectation-maximization, even in high-dimensional spaces. Closed-form parameter updates are critical for fast and efficient adaptive IS methods. We empirically demonstrated that MPPCA proposal distributions obtain more reliable estimates of the probability of failure compared to GMM proposals on a range of challenging, high-dimensional systems.

Future work will explore connections between the number of system failure modes and the choice of latent MPPCA factors. Characterizing the principal subspace of the failure distribution will allow us to identify a meaningful number of latent factors for each probabilistic principal component analyzer model, refining the trade-off between model complexity and failure distribution fidelity. We will also develop generative performance metrics that consider sample density and coverage in both disturbance space and trajectory space.

Acknowledgments

Toyota Research Institute (TRI) provided funds to assist the authors with their research, but this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity.

TABLE II
RESULTS

Dataset	IS Method	$(\hat{P}_F - P_F)/P_F$	Avg NLL($\hat{x}$)	Coverage	NDB/C	N_{total}
BRANCHES $d = 40$ $P_F = 9.55 \times 10^{-4}$	CE-MPPCA	**–0.024 ± 0.018**	65.660 ± 0.191	**0.943 ± 0.004**	**0.128 ± 0.049**	**30,000 ± 0**
	SIS-MPPCA	–0.043 ± 0.096	66.976 ± 0.365	0.760 ± 0.015	0.707 ± 0.059	83,200 ± 8,352
	CE-GMM	–0.997 ± 0.006	**64.270 ± 4.676**	0.686 ± 0.199	0.938 ± 0.062	39,200 ± 2,712
	SIS-GMM	–0.053 ± 0.593	73.533 ± 2.223	0.667 ± 0.037	0.731 ± 0.060	84,400 ± 9,200
BRANCHES $d = 60$ $P_F = 9.33 \times 10^{-4}$	CE-MPPCA	**0.001 ± 0.027**	**94.034 ± 0.185**	**0.944 ± 0.005**	**0.167 ± 0.050**	**30,000 ± 0**
	SIS-MPPCA	–0.003 ± 0.045	95.435 ± 0.429	0.855 ± 0.025	0.575 ± 0.073	82,800 ± 6,939
	CE-GMM	–0.999 ± 0.002	95.074 ± 8.768	0.592 ± 0.232	0.961 ± 0.044	37,600 ± 4,270
	SIS-GMM	0.204 ± 2.100	103.779 ± 1.513	0.709 ± 0.033	0.716 ± 0.080	82,800 ± 6,939
OSCILLATOR $d = 100$ $P_F = 9.55 \times 10^{-4}$	CE-MPPCA	–0.012 ± 0.024	**149.890 ± 0.321**	**0.945 ± 0.007**	**0.126 ± 0.057**	**30,000 ± 0**
	SIS-MPPCA	**–0.008 ± 0.050**	152.536 ± 0.784	0.937 ± 0.011	0.297 ± 0.120	127,200 ± 23,919
	CE-GMM	–0.999 ± 0.000	165.675 ± 17.590	0.918 ± 0.122	0.894 ± 0.069	38,600 ± 6,003
	SIS-GMM	–0.823 ± 0.259	156.262 ± 2.302	0.876 ± 0.062	0.676 ± 0.128	118,000 ± 20,099
OSCILLATOR $d = 200$ $P_F = 9.39 \times 10^{-4}$	CE-MPPCA	0.004 ± 0.031	**291.721 ± 0.664**	**0.953 ± 0.008**	**0.201 ± 0.062**	**30,000 ± 0**
	SIS-MPPCA	**–0.003 ± 0.079**	294.698 ± 0.773	0.950 ± 0.009	0.298 ± 0.095	138,800 ± 30,570
	CE-GMM	–0.999 ± 0.000	313.461 ± 33.544	0.945 ± 0.137	0.935 ± 0.109	38,000 ± 9,591
	SIS-GMM	–0.999 ± 0.001	303.757 ± 3.464	0.745 ± 0.120	0.785 ± 0.104	123,600 ± 17,292
F-16 GCAS $d = 202$ $P_F = 6.11 \times 10^{-4}$	CE-MPPCA	**–0.492 ± 0.624**	292.656 ± 4.816	0.517 ± 0.177	**0.936 ± 0.044**	**69,000 ± 11,357**
	SIS-MPPCA	–0.642 ± 0.207	297.620 ± 2.484	0.601 ± 0.042	0.961 ± 0.029	158,800 ± 26,354
	CE-GMM	–0.999 ± 0.000	329.508 ± 58.463	0.654 ± 0.341	0.981 ± 0.023	108,600 ± 76,026
	SIS-GMM	–0.998 ± 0.000	**288.815 ± 1.089**	**0.893 ± 0.021**	0.975 ± 0.021	171,600 ± 23,009

REFERENCES

[1] A. Corso, R. Moss, M. Koren, R. Lee, and M. Kochenderfer, "A survey of algorithms for black-box safety validation of cyber-physical systems," *Journal of Artificial Intelligence Research*, vol. 72, pp. 377–428, 2021.

[2] A. B. Owen, *Monte Carlo Theory, Methods and Examples*. Stanford University, 2013, https://artowen.su.domains/mc/.

[3] S. Geyer, I. Papaioannou, and D. Straub, "Cross entropy-based importance sampling using Gaussian densities revisited," *Structural Safety*, vol. 76, pp. 15–27, 2019.

[4] O. Ledoit and M. Wolf, "A well-conditioned estimator for large-dimensional covariance matrices," *Journal of Multivariate Analysis*, vol. 88, no. 2, pp. 365–411, 2004.

[5] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.

[6] I. Papaioannou, C. Papadimitriou, and D. Straub, "Sequential importance sampling for structural reliability analysis," *Structural Safety*, vol. 62, pp. 66–75, 2016.

[7] M. O'Kelly, A. Sinha, H. Namkoong, R. Tedrake, and J. C. Duchi, "Scalable end-to-end autonomous vehicle testing via rare-event simulation," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[8] Z. Huang, M. Arief, H. Lam, and D. Zhao, "Evaluation uncertainty in data-driven self-driving testing," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 2019, pp. 1902–1907.

[9] Y. Kim and M. J. Kochenderfer, "Improving aircraft collision risk estimation using the cross-entropy method," *Journal of Air Transportation*, vol. 24, no. 2, pp. 55–62, 2016.

[10] M. J. Kochenderfer, S. M. Katz, A. L. Corso, and R. J. Moss, *Algorithms for Validation*. MIT Press, 2025.

[11] D. Zhao, H. Lam, H. Peng, *et al.*, "Accelerated evaluation of automated vehicles safety in lane-change scenarios based on importance sampling techniques," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 595–607, 2016.

[12] Z. I. Botev and D. P. Kroese, "An efficient algorithm for rare-event probability estimation, combinatorial optimization, and counting," *Methodology and Computing in Applied Probability*, vol. 10, pp. 471–505, 2008.

[13] J. Demange-Chryst, F. Bachoc, J. Morio, and T. Krauth, "Variational autoencoder with weighted samples for high-dimensional non-parametric adaptive importance sampling," *Transactions on Machine Learning Research*, 2024.

[14] H. Delecki, M. R. Schlichting, M. Arief, A. Corso, M. Vazquez-Chanlatte, and M. J. Kochenderfer, "Diffusion-based failure sampling for cyber-physical systems," *arXiv preprint arXiv:2406.14761*, 2024.

[15] P. Del Moral, A. Doucet, and A. Jasra, "Sequential Monte Carlo samplers," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 68, no. 3, pp. 411–436, 2006.

[16] H. Kahn and T. E. Harris, "Estimation of particle transmission by random sampling," *National Bureau of Standards Applied Mathematics Series*, vol. 12, pp. 27–30, 1951.

[17] J. Norden, M. O'Kelly, and A. Sinha, "Efficient black-box assessment of autonomous vehicle safety," *Machine Learning for Autonomous Driving Workshop at the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2019.

[18] Z. Ghahramani, G. E. Hinton, *et al.*, "The EM algorithm for mixtures of factor analyzers," University of Toronto, Tech. Rep. CRG-TR-96-1, 1996.

[19] E. Richardson and Y. Weiss, "On GANs and GMMs," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31, 2018.

[20] N. Hansen, "The CMA evolution strategy: A tutorial," *arXiv preprint arXiv:1604.00772*, 2016.

[21] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, pp. 19–67, 2005.

[22] M. Chiron, C. Genest, J. Morio, and S. Dubreuil, "Failure probability estimation through high-dimensional elliptical distribution modeling with multiple importance sampling," *Reliability Engineering & System Safety*, vol. 235, p. 109 238, 2023.

[23] K. Zuev, *Advanced stochastic simulation methods for solving high-dimensional reliability problems*. Hong Kong University of Science and Technology (Hong Kong), 2009.

[24] I. Papaioannou, S. Geyer, and D. Straub, "Improved cross entropy-based importance sampling with a flexible mixture model," *Reliability Engineering & System Safety*, vol. 191, p. 106 564, 2019.

[25] P. Heidlauf, A. Collins, M. Bolender, and S. Bak, "Verification challenges in F-16 ground collision avoidance and other automated maneuvers," in *International Workshop on Applied Verification for Continuous and Hybrid Systems (ARCH)*, 2018, pp. 208–217.

[26] O. So and C. Fan, "Solving Stabilize-Avoid Optimal Control via Epigraph Form and Deep Reinforcement Learning," in *Proceedings of Robotics: Science and Systems*, 2023.

[27] M. F. Naeem, S. J. Oh, Y. Uh, Y. Choi, and J. Yoo, "Reliable fidelity and diversity metrics for generative models," in *International Conference on Machine Learning (ICML)*, PMLR, 2020, pp. 7176–7185.