Adaptive Importance Sampling Based on Fault Tree Analysis for Piecewise Deterministic Markov Process[*]

Guillaume Chennetier[†‡], Hassane Chraibi[†], Anne Dutfoy[†], and Josselin Garnier[‡]

Abstract. Piecewise deterministic Markov processes (PDMPs) can be used to model complex dynamical industrial systems. The counterpart of this modeling capability is their simulation cost, which makes reliability assessment untractable with standard Monte Carlo methods. A significant variance reduction can be obtained with an adaptive importance sampling method based on a cross-entropy procedure. The success of this method relies on the selection of a good family of approximations of the committor function of the PDMP. In this paper original families are proposed. Their forms are based on reliability concepts related to fault tree analysis: minimal path sets and minimal cut sets. They are well adapted to high-dimensional industrial systems. The proposed method is discussed in detail and applied to academic systems and to a realistic system from the nuclear industry.

Key words. rare event simulation, reliability, importance sampling, piecewise deterministic Markov process, fault tree analysis, cross-entropy, PyCATSHOO

MSC codes. 65C05, 62L12, 65C60, 60J25

DOI. 10.1137/22M1522838

1. Introduction. The reliability assessment of industrial systems combines two issues: finding an appropriate framework to model these systems and proposing a method allowing the simulation of rare events since the failure probabilities are generally very low [9, 50]. We focus here on hybrid dynamical industrial systems, i.e., systems whose time-dependent state is described by both continuous and discrete variables. The continuous variables are typically physical variables (such as temperature, pressure, or a level of liquid) that follow deterministic physical laws, and the discrete variables are typically the status of the components of the system that may be affected by random events. Such a system can be modeled by a piecewise deterministic Markov process (PDMP) [23, 26, 2, 55]. The modeling and simulation of hybrid systems is still an active field, and other formalisms similar to PDMPs exist, such as stochastic hybrid systems (SHS) [45] or, more recently, general stochastic hybrid systems which generalize and encompass both PDMPs and SHS [12]. The current work aims to enhance the efficiency of the `PyCATSHOO` toolbox [18], an EDF-developed computer code based on the PDMP formalism.

During the last decade, several attempts to adapt traditional methods of rare event simulation to hybrid systems have been proposed [16, 17, 33, 57, 5, 63, 1]. However, it still appears challenging to significantly reduce the required sample size compared to a standard Monte Carlo method in the case of high-dimensional systems. Preliminary work [16] established the

connection between the optimal instrumental distribution of an importance sampling method for PDMPs and the use of the committor function of the process as an importance function (which we abbreviate IF). An IF (also called reaction coordinate or collective variable in computational physics and chemistry [43]) offers a one-dimensional representation of the dynamics of a high-dimensional system. It associates to a given configuration of the system a real value that can be interpreted as a distance to a specific set of configurations. The committor function is known as the optimal IF to use for modeling phenomena in transition phase theory [44] and for splitting algorithms in rare event simulation [15]. It is the probability of realizing the rare event knowing the current state of the process. When dealing with stochastic processes taking values in standard Euclidean spaces, the committor function is frequently approximated by mixing explicit calculus on stochastic differential equations and machine learning methods [39, 41, 32]. In our case, we can reduce the committor estimation problem to a parametric problem by looking for the best approximation of the committor function among a family of IFs that depend only on the status of the system components, which forms a discrete albeit high-dimensional variable. The failure of a specific component in a specific system configuration has to be encouraged to a greater or lesser extent depending on how it interacts with the other components. These interactions can be described by fault tree analysis through the concepts of minimal path sets and minimal cut sets, which are well known in the reliability community [53]. These concepts are often used to construct quantitative measures of system reliability such as importance indices for the components or approximations of the probability of failure [47, 35, 13]. Here we will use them to design efficient importance sampling strategies. The construction of IFs proposed in [57] to apply a Repetitive Simulation Trials After Reaching Thresholds (RESTART) method to hybrid systems is very close to our philosophy (see also [10] for the same idea with a splitting method for dynamic fault trees).

Given a family of IFs that serve as approximations of the committor function, each IF can be associated with an instrumental distribution for the importance sampling strategy. The search for the best candidate within this family is sequential using an adaptive importance sampling (AIS) method with a cross-entropy (CE) procedure (see [11] for a global perspective on AIS and [24] for a general introduction on CE). Each iteration of the method consists of a simulation phase according to the current distribution and an optimization phase to refine this distribution. Classical parametric families of importance distributions (typically Gaussian mixtures in the literature) require a large number of parameters to sufficiently approximate the optimal distribution when it is quite complex. However, it is well known that in high dimensions, importance sampling becomes tricky [51] because of the degeneracy of the likelihood ratios. This is one of the main concerns in the field, and many recent works try to answer it when the optimal parameters of the instrumental distribution are sought by CE minimization [28, 27, 58, 62]. For the sake of efficiency, the AIS literature is also paying increasing attention to recycling schemes for updating the instrumental distribution and/or estimating the final quantity by reusing samples from past iterations [42, 46, 19]. The paper [46] further proves that the AIS estimator with a standard recycling scheme verifies a central limit theorem under appropriate assumptions. We give easily verifiable and physically interpretable conditions on the process and on the family of IFs that validate these assumptions.

Contribution. The major contribution of this work is to propose families of IFs suitable for approximating the committor functions of high-dimensional hybrid (industrial) systems.

These families of IFs are easy to implement and interpret as they are constructed based on the "minimal paths" and "minimal cuts" of the system, which are classical concepts in reliability analysis. The original parameterization of these IFs also makes them highly flexible, allowing for easy refinement of the committor function approximation even in high dimension. We also provide detailed guidelines for the implementation of the AIS method with recycling schemes and prove the consistency and asymptotic normality of the associated IS estimator.

Structure. The paper is organized as follows. Section 1 presents the notations and the industrial application case. It reminds the reader with the fundamental relation that exists between the committor function U_{opt} of a PDMP and the optimal distribution q_{opt} of an importance sampling method for this process. We propose in section 2 three parametric families of functions $U_{\boldsymbol{\theta}}$ allowing us to approximate the committor function U_{opt} and to construct instrumental densities $q_{\boldsymbol{\theta}}$ producing low variance importance sampling estimators. These $U_{\boldsymbol{\theta}}$ functions are built on reliability properties of the system, and we will explain for this purpose the notions of minimal path sets and minimal cut sets. Our complete algorithm is given in section 3 with its asymptotic properties. We propose an adaptive CE importance sampling algorithm with recycling of past samples (both for a probability estimation and for updating the sampling policy). We also prove the consistency and the asymptotic normality of the estimator produced by the algorithm. Our recommendations for the implementation of the method are described in detail in section 4. The method is applied in section 5 to series/parallels systems and to two different configurations of the spent fuel pool system. Our method gives a dramatic variance reduction when estimating the probability of system failure even in the most complex case. We finally discuss the implications and possible refinements of this work in section 6.

1.1. Modeling hybrid systems with PDMPs.
A system failure is declared when continuous physical variables (e.g., temperature, pressure, or liquid level in a tank) exceed critical threshold values. This only happens when key combinations of components fail and when repairs do not come in time. Component failures and repairs are then seen as random one-time events, while the evolution of continuous variables is dictated by deterministic differential equations derived from physical laws. The high reliability of these systems is explained on the one hand by their high level of redundancy: the system can be reconfigured using several identical components to ensure its operation while waiting for the repair or replacement of broken components. On the other hand, the average waiting time before the failure of a component is generally considerably larger than the average waiting time before its repair. Such behavior lends itself very well to modeling by PDMPs.

PDMPs are a class of stochastic processes introduced and described by Mark Davis in 1984 [23]. Apart from reliability considerations, these processes have since been used in many different areas, in particular, to model chemical and biological phenomena [34, 36, 52]. A considerable work has also been done on the theoretical properties of these processes, in particular on their long-time behaviors [20, 3]. They allow for a sophistication of Markov chain Monte Carlo methods for the simulation of a posteriori distributions [6, 4, 59].

A PDMP describes the evolution of one or more quantities over time. These quantities follow a deterministic trajectory that can change (jump) at random or deterministic times. Moreover, a PDMP is Markovian: the future of its trajectory depends only on its current state and the time already elapsed, but not on its past states. Working with PDMPs therefore

requires managing two difficulties: first, its behavior is neither solely deterministic nor solely stochastic. Second, it is a hybrid process because it is composed of a continuous variable called "position" and a discrete variable called "mode."

The state of a PDMP at time t is denoted $Z_t = (X_t, M_t) \in E$, where $X_t \in \mathbb{X} \subset \mathbb{R}^{d_X}$ for some $d_X \geq 1$ is the position and $M_t \in \mathbb{M}$ is the mode of the PDMP (with $\mathbb{M}$ a finite or countable set and $E = \mathbb{X} \times \mathbb{M}$). We consider in this work PDMPs of fixed duration $t_{\max} > 0$. We denote $\mathcal{Z} = (Z_t)_{t \in [0, t_{\max}]}$ as a complete trajectory, and $\mathcal{E}$ is the set of all possible trajectories of duration $t_{\max}$ on E (the explicit description of the set $\mathcal{E}$ is not necessary for the following but is detailed in [30, section 1.2.3]). We will suppose in the following that one of the coordinates of the "position" variable $X \in \mathbb{X}$ is the total elapsed time, and for any state $z \in E$ we denote by τ_z this elapsed time (in particular for any $t > 0$ we have $\tau_{Z_t} = t$).

Given the state space E, the behavior of a PDMP is characterized by three elements:

1. The flow Φ that gives the deterministic trajectory.
2. The jump intensity λ which determines the distribution of the random jump times.
3. The transition kernel $\mathcal{K}$ which determines the distribution of the postjump locations.

Flow function. If there is no jump between time s and time $s + h$, the mode M_{s+h} of the PDMP remains constant and equal to M_s, and the position X_{s+h} of the PDMP evolves in a deterministic way from X_s as $\phi_{M_s}(X_s, h)$, where ϕ_m is the solution of the differential equation $\dfrac{d\phi_m}{dh} = \boldsymbol{g}(\phi_m, m)$, $\phi_m(x, h = 0) = x$. Here $\boldsymbol{g} : \mathbb{R}^{d_X} \times \mathbb{M} \to \mathbb{R}^{d_X}$ is a Lipschitz function. The trajectory of the process (X_{s+h}, M_{s+h}) is then of the form $\Phi_{(X_s, M_s)}(h)$, where the flow function Φ is defined by the following:

$$(1.1) \qquad \text{For any } z = (x, m) \in E, \ \Phi_z : h \in [0, +\infty) \longmapsto (\phi_m(x, h), m) \in E.$$

Deterministic jumps. By denoting $E_m := \{ z' = (x', m') \in E \mid m' = m \}$, we can write $E = \bigcup_{m \in \mathbb{M}} E_m$. The boundary of the state space E_m for $m \in \mathbb{M}$ is denoted by ∂E_m. When the position reaches the boundary of the state space following the flow Φ, the PDMP jumps. Starting from a state $z = (x, m) \in E_m$ at time s and assuming that no random jump takes place, the boundary ∂E_m is reached at a deterministic time $t_z^\partial \in [0, +\infty]$:

$$(1.2) \qquad t_z^\partial = \inf\{ h > 0 : \Phi_z(h) \in \partial E_m \},$$

with the convention $\inf \emptyset = +\infty$.

Intensity function. For each state $z \in E$, there is a random waiting time T_z at the end of which the process jumps (if it is smaller than the deterministic jump time t_z^∂). The jump intensity λ is a function that associates to each state $z \in E$ a weight $\lambda(z) \in (0, +\infty)$. The larger $\lambda(z)$ is, the more likely it is that the PDMP jumps when it passes through state z. The distribution of T_z depends on λ through the following formula:

$$(1.3) \qquad \mathbb{P}\left(T_z > h \mid Z_{\tau_z} = z \right) = \mathbb{1}_{h < t_z^\partial} \exp\left(-\int_0^h \lambda\left(\Phi_z(h') \right) dh' \right).$$

Transition kernel. Given the jump time and the state z^- from which the process jumps, the arrival state of the process after the jump is randomly chosen according to a Markovian transition kernel $\mathcal{K}(z^-, \cdot)$. It is assumed for any z^- that the transition kernel $\mathcal{K}(z^-, \cdot)$ admits a

probability density function $K(z^-, \cdot)$ with respect to a reference measure ν_{z^-} on $\mathscr{B}(E)$ (where $\mathscr{B}(\cdot)$ indicates the Borelians of a set). So for any $B \in \mathscr{B}(E)$,

$$(1.4) \qquad \mathcal{K}(z^-, B) = \int_B K(z^-, z) \, d\nu_{z^-}(z).$$

Probability density function of a trajectory. A fixed-time trajectory of a PDMP possesses a probability density function with respect to a dominant measure denoted by ζ, which is a mixture of Lebesgue and Dirac measures. The definition of this dominant measure and the density of the PDMP trajectory can be found in Definition 3.2 and Theorem 3.3 of [16], respectively. Using our notations, the expression for this density is as follows. Let $\mathcal{Z} = (Z_t)_{t \in [0, t_{\max}]}$ be a PDMP trajectory, and let $n_{\mathcal{Z}}$ be its number of jumps. We denote by z_0 the initial state of the trajectory and by t_0 the waiting time before the first jump. For $k = 1, \dots, n_{\mathcal{Z}} - 1$, we denote by z_k the state of the process after the kth jump and by t_k the waiting time between the kth and the $(k+1)$th jump. Finally let $t_{n_{\mathcal{Z}}} = t^{\partial}_{z_{n_{\mathcal{Z}}}} = t_{\max} - \sum_{k=0}^{n_{\mathcal{Z}}-1} t_k$. The probability density function of this trajectory can then be expressed as

$$(1.5)$$

$$\pi_{\lambda, K}(\mathcal{Z}) = \prod_{k=0}^{n_{\mathcal{Z}}} \left[\lambda \left(\Phi_{z_k}(t_k) \right) \right]^{\mathbb{1}_{t_k < t^{\partial}_{z_k}}} \times \exp \left[-\int_0^{t_k} \lambda \left(\Phi_{z_k}(u) \right) \, du \right] \times \prod_{k=0}^{n_{\mathcal{Z}}-1} K \left(\Phi_{z_k}(t_k), z_{k+1} \right).$$

We consider the state space E and the flow Φ of the PDMP fixed once and for all in this paper. A distribution $\pi_{\lambda, K}$ of a PDMP trajectory on $\mathcal{E}$ can thus be totally determined by the choice of the jump intensity λ and the jump kernel density K.

System failure. In the case of industrial systems, the position of the PDMP contains the physical variables that determine the failure of the system and if necessary all the variables allowing the process to be Markovian such as the elapsed time. The physical variables evolve in time according to the flow Φ given by physical laws. The mode of the PDMP contains the status of each component (active, inactive, broken, etc.). The failure and repair rates of each component (which can depend on the value of the physical variables) determine the jump intensity λ and the jump kernel density K (see section 5 for a numerical example).

The system fails when the position reaches a critical region $\mathbb{X}_D$ determined by threshold values for the physical variables. The critical region $\mathbb{X}_D$ can only be reached in certain modes (the critical thresholds cannot be reached by the physical variables when all components are functioning, for example). Let $\mathbb{M}_D$ denote the set of modes m such that the position can reach $\mathbb{X}_D$ by following the flow ϕ_m corresponding to the mode m. The set of states corresponding to a failed system is therefore $D = \mathbb{X}_D \times \mathbb{M}_D \subset E$. The set of faulty trajectories is written $\mathcal{D} \subset \mathcal{E}$ and corresponds to

$$(1.6) \qquad \mathcal{D} := \left\{ (Z_t)_{t \in [0, t_{\max}]} \in \mathcal{E} \ : \ \exists t \in [0, t_{\max}] \text{ such that } Z_t \in D \right\}.$$

Objective. We denote by λ_0, K_0, and K_0 the jump intensity, jump kernel, and jump kernel density of the PDMP modeling the system whose failure probability we wish to estimate, and we denote by $\pi_0 \equiv \pi_{\lambda_0, K_0}$ the corresponding PDMP distribution. Our objective is to estimate the following probability:

$$(1.7) \qquad P := \mathbb{P}_{\pi_0} (\mathcal{Z} \in \mathcal{D}) = \mathbb{E}_{\pi_0} \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \right].$$

Simulation cost. Most of the computational cost for complex industrial systems comes from solving the differential equations that define the flow of the PDMP. We consider in the following that simulating several tens of uniform random variables and evaluating the density of a PDMP trajectory have a negligible cost compared to the computation of the flow between two jumps. This will guide the choice of our jump time simulation method and our optimization strategy for AIS (both are described in section 4). Note that, even if the flow appears in the formula of the density of a trajectory (1.5), it does not have to be computed again since it was already evaluated to generate the trajectory.

Test case: The spent fuel pool (SFP). We propose a redesigned version of the system presented in [18]. It is a simplification of the real operation of the storage pools of water for spent fuel from nuclear reactors. The water in the pool cools the fuel and provides protection from radiation. Conversely, the fuel heats the water in the pool, which will eventually boil, evaporate, and allow the fuel to damage the structure and contaminate the outside environment. System failure is declared when the water level in the pool has reached a critical level. All the components of the system (shown in Figure 1) are designed to keep the water in the pool cold enough to prevent it from evaporating.

On this system, the position of the PDMP corresponds to the water temperature and the water level in the pool; its mode is the combination of the status (active, inactive, broken) of the components. The domain D is defined by all the states of E in which the water temperature (first coordinate of the position) is $100°C$ and the water level (second coordinate
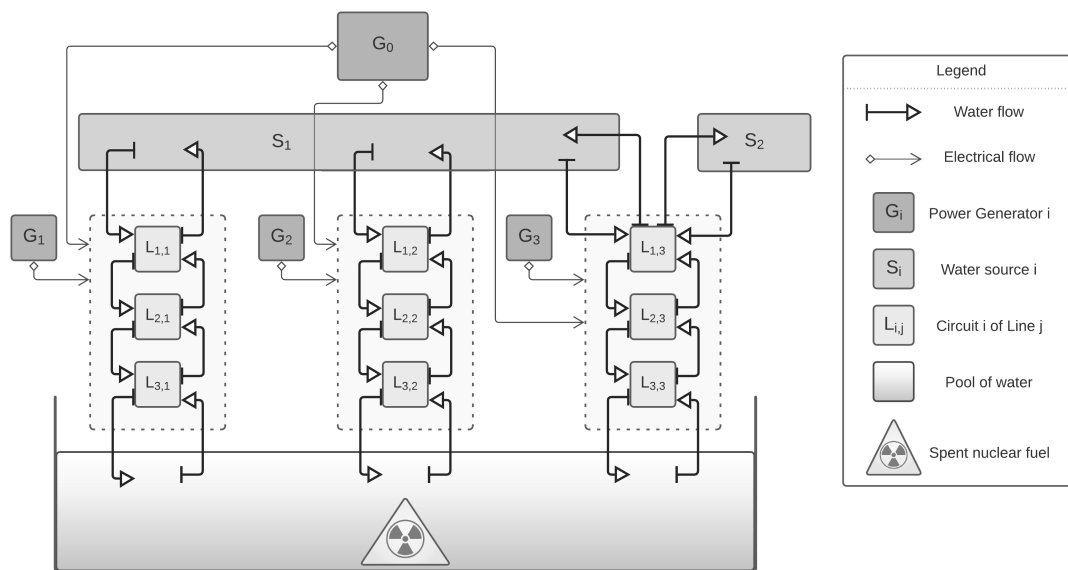
Figure 1. *Representation of the spent fuel pool.* *The temperature of an external cold water source* S_1 *is transferred to the pool by means of three sealed circuits connected by heat exchangers* $L_{1,1}$, $L_{2,1}$, *and* $L_{3,1}$ *forming a line* L_1. *The system has a general power supply* G_0. *In the event of a problem with one of these components, the system is equipped with two other lines* L_2 *and* L_3 *identical to* L_1, *an emergency diesel generator for each line* G_1, G_2, *and* G_3, *and a second outside water source* S_2 *accessible only to the third line* L_3.

of the position) is lower or equal to the critical threshold. We notice that these positions are not accessible for the flow for all modes, for example if no component is broken. If on the other hand, all the generators G_0, G_1, G_2, and G_3 are and remain broken, for example, then the flow reaches one of these positions in finite time. A more formal description of the modes allowing the flow to bring the process into D is given in section 2. This description will turn out to be important to build a good estimator of the failure probability. The system data allowing us to compute the flow and the different failure and repair rates of each component allowing us to build the jump intensities and kernels are given in section 5.

1.2. Importance sampling with PDMPs. We are interested in estimating P the probability that the system fails before $t_{\max}$. In the nuclear and hydraulic energy sector, the reliability requirements for the systems are very high. The system failure is therefore a rare event with a probability of 10^{-5} or smaller.

Crude Monte Carlo estimator. It is well known that crude Monte Carlo (CMC) methods are quite inefficient in this case because an overwhelming majority of realizations will not produce any failure. A CMC estimator of P from an independent and identically distributed (i.i.d.) sample $\mathcal{Z}_1, \ldots, \mathcal{Z}_N \sim \pi_0$ is given by

$$(1.8) \qquad \widehat{P}_N^{\mathrm{CMC}} := \frac{1}{N} \sum_{k=1}^{N} \mathbb{1}_{\mathcal{Z}_k \in \mathcal{D}} \xrightarrow[N \to \infty]{\text{a.s.}} \mathbb{E}_{\pi_0} \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \right] = P.$$

An accurate CMC estimator of P (say with a coefficient of variation of 0.1) requires about $100/P$ realizations. Each realization implies to simulate a PDMP trajectory which is very expensive, especially for large industrial systems (as previously mentioned, the deterministic parts of the trajectory correspond to complex physical phenomena and result from the resolution of expensive computer codes). It is therefore unthinkable to simulate several tens of millions of trajectories to estimate the probability of system failure.

Importance sampling (IS) estimator. Introduced in 1951 by Kahn and Harris [31], IS is used to estimate the expectation of a random quantity under an arbitrary distribution. In the context of rare event simulation, this method allows us to estimate the probability P of the rare event using an instrumental distribution q whose support is included in the support of the original distribution π_0 and that realizes the event more frequently than it. See [9] for a rare event perspective of IS, [56] for a recent review of IS methods in reliability assessment, and [29] for recent advances in IS for a more general purpose. From the formula

$$P = \mathbb{E}_{\pi_0} \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \right] = \int_{\mathcal{E}} \mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \pi_0(\mathcal{Z}) \, d\zeta(\mathcal{Z})$$

$$(1.9) \qquad = \int_{\mathcal{E}} \mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \frac{\pi_0(\mathcal{Z})}{q(\mathcal{Z})} q(\mathcal{Z}) \, d\zeta(\mathcal{Z}) = \mathbb{E}_q \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \frac{\pi_0(\mathcal{Z})}{q(\mathcal{Z})} \right],$$

we can deduce the form of the IS estimator of P with an i.i.d. sample $\mathcal{Z}_1, \ldots, \mathcal{Z}_N \sim q$:

$$(1.10) \qquad \widehat{P}_N^{\mathrm{IS}} := \frac{1}{N} \sum_{k=1}^{N} \mathbb{1}_{\mathcal{Z}_k \in \mathcal{D}} \frac{\pi_0(\mathcal{Z}_k)}{q(\mathcal{Z}_k)} \xrightarrow[N \to \infty]{\text{a.s.}} \mathbb{E}_q \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \frac{\pi_0(\mathcal{Z})}{q(\mathcal{Z})} \right] = P.$$

Optimal importance process for PDMPs. The variance of $\widehat{P}_N^{\mathrm{IS}}$ strongly depends on the choice of q. The optimal distribution $q_{\mathrm{opt}} : \mathcal{Z} \mapsto \frac{1}{P}\mathbb{1}_{\mathcal{Z} \in \mathcal{D}}\,\pi_0(\mathcal{Z})$ produces a zero-variance IS estimator. Although inaccessible in practice, this form guides us on the choice of the instrumental density to use. We also know from [16] that the process of distribution q_{opt} is a PDMP with the same state space and with the same deterministic flow as the original PDMP of distribution π_0.

The optimal jump intensity λ_{opt} and optimal jump kernel density K_{opt} given in (1.12) can be expressed in terms of the committor function of the process U_{opt}. It is defined here as follows for any states $z^-, z \in E$:

$$(1.11) \qquad U_{\mathrm{opt}}(z) := \mathbb{E}_{\pi_0}\left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \mid Z_{\tau_z} = z\right] \quad \text{and} \quad U_{\mathrm{opt}}^-(z^-) := \int_E U_{\mathrm{opt}}(z)\, K_0\left(z^-, z\right) d\nu_{z^-}(z).$$

U_{opt} represents the probability of reaching $\mathcal{D}$ knowing the current state of the trajectory, and U_{opt}^- represents the same quantity with the additional assumption that the process jumps immediately. Theorems 4.3 and 4.4 of [16] give us the following result.

Theorem 1.1 (optimal jump intensity and jump kernel). *For states $z^-, z \in E$, the optimal jump intensity and optimal jump kernel density are given by*

$$(1.12) \qquad \lambda_{opt}(z) = \lambda_0(z) \times \frac{U_{opt}^-(z)}{U_{opt}(z)} \quad and \quad K_{opt}\left(z^-, z\right) = K_0\left(z^-, z\right) \times \frac{U_{opt}(z)}{U_{opt}^-(z^-)}.$$

We have $q_{\lambda_{opt}, K_{opt}}(\mathcal{Z}) = q_{opt}(\mathcal{Z}) = \frac{1}{P}\mathbb{1}_{\mathcal{Z} \in \mathcal{D}}\,\pi_0(\mathcal{Z})$, and this distribution produces a zero-variance IS estimator of P.

Remark: Understanding these equations gives a good intuition of the behavior of the optimal process. If, from a given state $z^- \in E$,

1. the process is k times more likely to reach D before the end of the simulation by jumping now than by not jumping, then the optimal intensity λ_{opt} must be k times larger than the nominal intensity λ_0;
2. the process is k times more likely to reach D before the end of the simulation by jumping to state z than by jumping randomly according to $\mathcal{K}$, then the value of the optimal jump kernel density K_{opt} between z^- and z must be k times larger than that of the nominal jump kernel density K_0.

The optimal distribution q_{opt} is thus completely characterized by the original distribution π_0 and by the committor function U_{opt}.

2. Parametric approximation of the committor function $U_{\mathbf{opt}}$. In practice the true committor function is not accessible. The best we can do is to replace it by a function that we call IF and whose behavior is as close as possible to U_{opt}. Rather than trying to learn the committor function among a nonparametric class of functions, we look for the best approximation of U_{opt} among a parametric family of IFs $(U_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta}$. Here $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^{d_{\boldsymbol{\theta}}}$ is a parameter of dimension $d_{\boldsymbol{\theta}}$. Each IF $U_{\boldsymbol{\theta}}$ is associated with a PDMP importance distribution $q_{\boldsymbol{\theta}} \equiv \pi_{\lambda_{\boldsymbol{\theta}}, K_{\boldsymbol{\theta}}}$ whose jump intensity $\lambda_{\boldsymbol{\theta}}$ and jump kernel $K_{\boldsymbol{\theta}}$ are defined by

$$(2.1) \qquad \lambda_{\boldsymbol{\theta}}(z) = \lambda_0(z) \times \frac{U_{\boldsymbol{\theta}}^-(z)}{U_{\boldsymbol{\theta}}(z)} \quad \text{and} \quad K_{\boldsymbol{\theta}}\left(z^-, z\right) = K_0\left(z^-, z\right) \times \frac{U_{\boldsymbol{\theta}}(z)}{U_{\boldsymbol{\theta}}^-(z^-)}.$$

The more faithful is the approximation $U_{\boldsymbol{\theta}}$ to the committor function U_{opt}, the closer the instrumental distribution $q_{\boldsymbol{\theta}}$ should be to the optimal distribution q_{opt}, and the larger the variance reduction of the IS method should be. The committor function can be interpreted as a proximity measure between a state of the process and the set D.

IFs without position dependency. Since it is sufficient to stay long enough in $\mathbb{M}_D$ to end up in D, the main obstacle to overcome in order to realize the rare event is to reach $\mathbb{M}_D$. Moreover, the committor function appears only as ratios evaluated at identical positions but distinct modes (see (2.1)), which removes some of the position dependence. It is therefore reasonable to restrict ourselves to IFs which depend only on the mode and not on the position of the process. It remains to determine how to quantify the proximity of a mode (which represents the status of the system components) to the set $\mathbb{M}_D$. For this purpose we will exploit a static and Boolean representation of the system, and make use of concepts from the reliability theory: the minimal path sets (MPS) and minimal cut sets (MCS).

2.1. MPS and MCS. In the static point of view, we consider the final mode of the trajectory, and we declare that the trajectory has failed if that mode belongs to $\mathbb{M}_D$ (without taking account the time during which the position evolves to D when its mode belongs to $\mathbb{M}_D$ as well as the possibility that the system is repaired during this time). The path sets and cut sets can be defined as follows.

- The path sets are the sets of components whose operation prevents the system failure.
- The cut sets are the sets of components whose malfunction causes the system failure.

A path/cut set is minimal if it contains no other path/cut set. These concepts can be understood very well with examples.

Series and parallel systems. Here are two extreme examples of industrial systems to keep in mind for the following. A series system is a configuration of components in which the failure of any one component is sufficient to cause system failure (see Figure 2). A parallel system is a configuration of components in which the failure of all components is necessary to cause system failure (see Figure 3). A series system has only one MPS, and a parallel system has $d_{\boldsymbol{c}}$ MPS. In contrast, a series system has $d_{\boldsymbol{c}}$ MCS, and a parallel system has only one MCS.

Spent fuel pool example. The test case described in subsection 1.1 can be represented as a series/parallel diagram (see Figure 4) facilitating its decomposition into MPS and MCS. The MPS correspond to all vertical combinations and the MCS to all horizontal combinations.
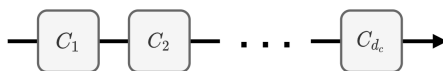
Figure 2. *Series system* with $d_{\boldsymbol{c}}$ components. It has one MPS: $\{\mathtt{C}_1,\ldots,\mathtt{C}_{d_{\boldsymbol{c}}}\}$ and $d_{\boldsymbol{c}}$ MCS: $\{\mathtt{C}_1\},\ldots,\{\mathtt{C}_{d_{\boldsymbol{c}}}\}$.
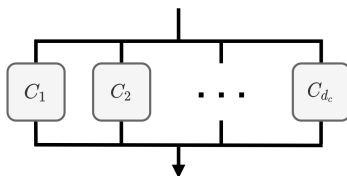
Figure 3. *Parallel system* with $d_{\boldsymbol{c}}$ components. It has $d_{\boldsymbol{c}}$ MPS: $\{\mathtt{C}_1\},\ldots,\{\mathtt{C}_{d_{\boldsymbol{c}}}\}$ and one MCS: $\{\mathtt{C}_1,\ldots,\mathtt{C}_{d_{\boldsymbol{c}}}\}$.
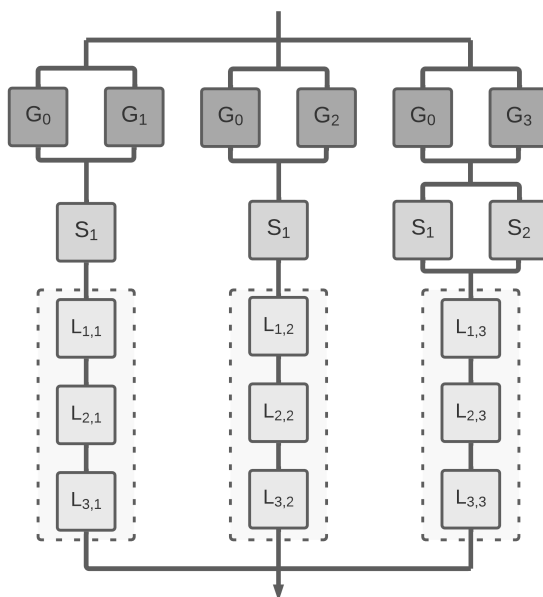
Figure 4. *Series/parallel diagram of the spent fuel pool of Figure* 1. *There are* 8 *MPS and* 69 *MCS in this system. MPS examples:* $\{G_0, S_1, L_{1,1}, L_{2,1}, L_{3,1}\}$ *and* $\{G_0, S_2, L_{1,3}, L_{2,3}, L_{3,3}\}$. *MCS examples:* $\{L_{1,1}, L_{3,2}, L_{1,3}\}$ *and* $\{G_0, S_1, G_3\}$.

Formal definition with Boolean algebra. Letting $d_{\boldsymbol{c}} \in \mathbb{N}^*$ and $\boldsymbol{c} = (c_1, \ldots, c_{d_{\boldsymbol{c}}}) \in \{0,1\}^{d_{\boldsymbol{c}}}$, we donote $[\![1, d_{\boldsymbol{c}}]\!] = \{1, \ldots, d_{\boldsymbol{c}}\}$, and for any $I \subset [\![1, d_{\boldsymbol{c}}]\!]$,

$$\bigvee_{j \in I} c_j = \max\{c_j, j \in I\} \quad \text{and} \quad \bigwedge_{j \in I} c_j = \min\{c_j, j \in I\}.$$

Letting $\boldsymbol{c}' = (c_1', \ldots, c_{d_{\boldsymbol{c}}}') \in \{0,1\}^{d_{\boldsymbol{c}}}$, we write $\boldsymbol{c} \leq \boldsymbol{c}'$ if and only if $c_j \leq c_j'$ for all $j \in [\![1, d_{\boldsymbol{c}}]\!]$. Let φ be a Boolean function, i.e., from $\{0,1\}^{d_{\boldsymbol{c}}}$ to $\{0,1\}$. The function φ is nondecreasing if and only if $\varphi(\boldsymbol{c}) \leq \varphi(\boldsymbol{c}') \iff \boldsymbol{c} \leq \boldsymbol{c}'$ for all $\boldsymbol{c}, \boldsymbol{c}' \in \{0,1\}^{d_{\boldsymbol{c}}}$. Combining Theorem 1.14 and Theorem 1.21 from [21], we obtain the following decomposition.

Theorem 2.1 (complete disjunctive normal form of an nondecreasing Boolean function). *If φ is an nondecreasing Boolean function, then it admits a unique decomposition (except for the numbering of the terms) of the form*

$$(2.2) \qquad \varphi(\boldsymbol{c}) = \bigvee_{i=1}^{d_{\varphi}} \bigwedge_{j \in I_i} c_j \quad \text{for all } \boldsymbol{c} \in \{0,1\}^{d_{\boldsymbol{c}}},$$

where d_{φ} is an integer and $I_1, \ldots, I_{d_{\varphi}} \subset [\![1, n]\!]$ are such that $I_i \not\subset I_{i'}$ for all $i, i' \in [\![1, d_{\varphi}]\!]$ with $i \neq i'$.

If a system has $d_{\boldsymbol{c}}$ components, we now consider that the PDMP mode $m \in \mathbb{M}$ can be converted into a multidimensional binary variable $b_m = (c_1, \ldots, c_{d_{\boldsymbol{c}}}) \in \{0,1\}^{d_{\boldsymbol{c}}}$, where, for $j \in [\![1, d_{\boldsymbol{c}}]\!]$, c_j is 0 if the jth component is broken and 1 otherwise.

Definition 2.2 (structure function φ). *We call the structure function of the system the Boolean function φ which associates* 1 *to the modes for which the system works and* 0 *to the other ones:*

$$(2.3) \qquad \varphi : b_m \longmapsto 1 - \mathbb{1}_{m \in \mathbb{M}_D}.$$

Definition 2.3 (coherent system). *A system is said to be coherent if:*
- *it works when none of its components is down, i.e., $\varphi((1,\ldots,1)) = 1$;*
- *it does not work when all its components are broken, i.e., $\varphi((0,\ldots,0)) = 0$;*
- *if it does not work in a given mode, then it does not work with the additional failure of a component, and conversely, if it works in a given mode, then it still works if broken components are repaired, i.e., φ is nondecreasing.*

MPS decomposition. We assume from now that the system is coherent, so we can apply Theorem 2.1 to its structure function φ. There are then d_{MPS} sets of components $I_1, \ldots, I_{d_{\mathrm{MPS}}} \subset [\![1, d_{\boldsymbol{c}}]\!]$ such that none of them is contained in another and such that the operation of all the components of a set causes the operation of the system independently of the status of the components in the other sets. These sets are the unique minimal path sets of the system. By switching to the complementary with $\overline{c} = 1 - c$ for any $c \in \{0, 1\}$, we can notice that

$$(2.4) \qquad \varphi(b_m) = \bigvee_{i=1}^{d_{\mathrm{MPS}}} \bigwedge_{j \in I_i} c_j \iff \overline{\varphi(b_m)} = \bigwedge_{i=1}^{d_{\mathrm{MPS}}} \bigvee_{j \in I_i} \overline{c_j} \ .$$

Therefore the minimal path sets can be alternatively defined such that the failure of at least one component in each set causes the system failure.

MCS decomposition. Let us first note that since φ is a nondecreasing Boolean function, the function $b_m \mapsto \varphi\left(\overline{b_m}\right)$ is also nondecreasing (it is called the dual function of φ). Indeed, letting $b_m \le b'_m \in \{0, 1\}^{d_{\boldsymbol{c}}}$, we have $\overline{b_m} \ge \overline{b'_m}$ so $\varphi\left(\overline{b_m}\right) \ge \varphi\left(\overline{b'_m}\right)$ and finally $\overline{\varphi\left(\overline{b_m}\right)} \le \overline{\varphi\left(\overline{b'_m}\right)}$. We can therefore apply Theorem 2.1 to it and get a set of d_{MCS} lists of indices $J_1, \ldots, J_{d_{\mathrm{MCS}}} \subset [\![1, d_{\boldsymbol{c}}]\!]$ with $J_i \not\subset J_{i'}$ for any $i, i' \in [\![1, d_{\mathrm{MCS}}]\!]$ with $i \ne i'$ such that

$$(2.5) \qquad \overline{\varphi\left(\overline{b_m}\right)} = \bigvee_{i=1}^{d_{\mathrm{MCS}}} \bigwedge_{j \in J_i} c_j, \quad \text{i.e.,} \quad 1 - \varphi(b_m) = \overline{\varphi(b_m)} = \bigvee_{i=1}^{d_{\mathrm{MCS}}} \bigwedge_{j \in J_i} \overline{c_j} \ .$$

These d_{MCS} sets of components $J_1, \ldots, J_{d_{\mathrm{MCS}}}$ are the unique minimal cuts sets of the system. If all the components of a MCS are broken, then this causes (from the static point of view) the system failure. And again by switching to the complementary, if at least one component per MCS works, the system failure is prevented:

$$(2.6) \qquad \overline{\varphi(b_m)} = \bigvee_{i=1}^{d_{\mathrm{MCS}}} \bigwedge_{j \in J_i} \overline{c_j} \iff \varphi(b_m) = \bigwedge_{i=1}^{d_{\mathrm{MCS}}} \bigvee_{j \in J_i} c_j \ .$$

2.2. Parametric families of IFs. We below propose 3 families of IFs $(U_{\boldsymbol{\theta}}^{(\mathrm{BC})})_{\boldsymbol{\theta} \in \Theta}$, $(U_{\boldsymbol{\theta}}^{(\mathrm{MPS})})_{\boldsymbol{\theta} \in \Theta}$, and $(U_{\boldsymbol{\theta}}^{(\mathrm{MCS})})_{\boldsymbol{\theta} \in \Theta}$ to approximate the committor function U_{opt}, each based on a

different reliability idea and parameterized by a vector $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^{d_{\boldsymbol{\theta}}}$ for some $d_{\boldsymbol{\theta}} \geq 1$. To each of these families of approximations of the committor function corresponds a family $(q_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta}$ of instrumental distributions obtained by replacing U_{opt} by $U_{\boldsymbol{\theta}}$ in (1.12).

One comment: since our IF only appears in the intensities and jump kernels as the ratio $U_{\boldsymbol{\theta}}^-/U_{\boldsymbol{\theta}}$, we only need to approximate a function proportional to U_{opt} and not U_{opt} itself. This means that unlike U_{opt}, $U_{\boldsymbol{\theta}}$ does not have to be interpreted as a probability and can therefore be larger than 1. On the other hand, we impose $U_{\boldsymbol{\theta}=\mathbf{0}} \equiv 1$ to ensure that $q_{\mathbf{0}} = \pi_0$ the unbiased distribution and $U_{\boldsymbol{\theta}}$ must be increasing in each coordinate of $\boldsymbol{\theta}$. The intuition for the practitioner should be clear: the "larger" $\boldsymbol{\theta}$ is, the more we bias and the more we push the trajectory to $\mathcal{D}$, but the more unstable the likelihood ratio $\pi_0/q_{\boldsymbol{\theta}}$ could become. One just has to find a suitable trade-off.

Broken components–based IF (BC-IF). A natural idea is to design $U_{\boldsymbol{\theta}}$ as an increasing function in the number of broken components. We borrow and extend an idea from [16]. Let $d_{\boldsymbol{c}}$ be the total number of components in the system, let $\beta^{(\mathrm{BC})}(z)$ be the number of broken components when the process is in the state $z \in E$, and let $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{d_{\boldsymbol{c}}}) \in \mathbb{R}^{d_{\boldsymbol{c}}}$. We propose the following IF:

$$(2.7) \qquad U_{\boldsymbol{\theta}}^{(\mathrm{BC})}(z) := \exp\left[\left(\sum_{j=1}^{d_{\boldsymbol{c}}} \theta_j \mathbb{1}_{\beta^{(\mathrm{BC})}(z) \geq j}\right)^2\right] = \exp\left[\left(\sum_{j=1}^{\beta^{(\mathrm{BC})}(z)} \theta_j\right)^2\right].$$

Theoretical arguments in favor of this particular function are given in [16] for a one-dimensional form: $U_{\boldsymbol{\theta}}^{(\mathrm{BC})}(z) = e^{(\theta_1 b_z)^2}$ (equivalent to (2.7) by imposing $\theta_1 = \cdots = \theta_{d_c}$). The intuition is that a good parametric form should favor each new component failure more and more strongly, because it is less and less likely that additional components will break without the previous ones being repaired. The form $x \mapsto \exp(x^2)$ then guarantees that the ratios $U_{\boldsymbol{\theta}}^-/U_{\boldsymbol{\theta}}$ are strictly increasing in $\beta^{(\mathrm{BC})}$.

The authors of [16] have successfully tested this one-dimensional approach on a parallel system with three components and a single failing mode (when all components are broken). It is reasonable to think that the number of components is not the core of the problem for parallel systems and that our $d_{\boldsymbol{c}}$-dimensional extension adds enough flexibility if needed for very large parallel systems. On the other hand, there is concern that this approach is too naive to approximate U_{opt} in the case of systems with both many components and multiple failing modes. Typical failures do not involve a large number of components but a small number of key components. Using $U_{\boldsymbol{\theta}}^{(\mathrm{BC})}$ produces trajectories where blindly selected components will break. With small $\boldsymbol{\theta}$ it may not break the right components, and with large $\boldsymbol{\theta}$ it may break too many and produce implausible trajectories (with a low likelihood ratio in the IS estimator (1.10)). We must therefore take into account the role played by each component within the system.

MPS-based IF (MPS-IF). An MPS is said to be damaged when at least one of its components is broken. It is necessary that all minimal path sets are damaged to reach system failure. For $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{d_{\mathrm{MPS}}}) \in \mathbb{R}^{d_{\mathrm{MPS}}}$, we recycle the idea expressed in (2.7) but this time with an increasing function in $\beta^{(\mathrm{MPS})}(z)$ as the number of damaged MPS in state z:

$$(2.8) \qquad U_{\boldsymbol{\theta}}^{(\mathrm{MPS})}(z) := \exp\left[\left(\sum_{i=1}^{d_{\mathrm{MPS}}} \theta_i \mathbb{1}_{\beta^{(\mathrm{MPS})}(z)\geq i}\right)^2\right] = \exp\left[\left(\sum_{j=1}^{\beta^{(\mathrm{MPS})}(z)} \theta_j\right)^2\right].$$

The process simulated from $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})}$ therefore prioritizes the failure of components involved in a large number of still undamaged MPS (and conversely will prioritize preventing the repair of components involved in a large number of already damaged MPS).

The MPS-IF does not consider the number of broken components per MPS but only the presence of at least one broken component. This means that the form $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})}$ does not look for breaking two components in a single MPS if they are not present in other MPS. Yet, even if this second broken component is not necessary to achieve system failure, it prevents the system from being safe in the case when the first component is repaired. Describing the functioning of the system with MCS rather than MPS leads us to a more flexible parametric family since this time we have to look at the proportion of broken components in each MCS.

MCS-based IF (MCS-IF). For any state $z \in E$ and any $i \in [\![1, d_{\mathrm{MCS}}]\!]$, we define $\beta_i^{(\mathrm{MCS})}(z)$ as the proportion of broken components in the ith MCS when the process is in the state z. We rank these proportions in descending order, and we also define $\beta_{(i)}^{(\mathrm{MCS})}(z)$ as the ith largest proportion of broken components among all MCS in the state z:

$$(2.9) \qquad U_{\boldsymbol{\theta}}^{(\mathrm{MCS})}(z) := \exp\left[\left(\sum_{i=1}^{d_{\mathrm{MCS}}} \theta_i \beta_{(i)}^{(\mathrm{MCS})}(z)\right)^2\right].$$

Here again, the process simulated from $U_{\boldsymbol{\theta}}^{(\mathrm{MCS})}$ prioritizes the failure of components involved in a large number of MCS or in MCS of small size (to bring the proportions of broken components closer to 1 more quickly).

Dimension reduction. A key point for the success of the method is to get a flexible but reasonably sized family. One way to reduce the dimension of $\boldsymbol{\theta}$ is to impose the equality of some groups of coordinates (by ordered packets). For example, to get a vector $\boldsymbol{\theta} \in \mathbb{R}^{d_{\boldsymbol{\theta}}}$ from $\widetilde{\boldsymbol{\theta}} \in \mathbb{R}^{d_{\widetilde{\boldsymbol{\theta}}}}$ with $d_{\boldsymbol{\theta}} = k \times d_{\widetilde{\boldsymbol{\theta}}}$, we impose $\theta_i = \widetilde{\theta}_{\lfloor \frac{i-1}{k} \rfloor + 1}$ for $i \in [\![1, d_{\boldsymbol{\theta}}]\!]$. This is why, by choosing $\mathbb{1}_{\beta \geq i}$ in (2.7) and (2.8) instead of simply $\mathbb{1}_{\beta = i}$, and $\beta_{(i)}^{(\mathrm{MCS})}$ instead of $\beta_i^{(\mathrm{MCS})}$ in (2.9), we keep a consistent (and increasing in each coordinate) $U_{\boldsymbol{\theta}}$ even with $d_{\widetilde{\boldsymbol{\theta}}} = 1$.

3. Algorithm and asymptotic properties. To each parametric family $(U_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta}$ of IFs designed to approximate U_{opt} corresponds a family of instrumental distributions $(q_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta} \equiv (\pi_{\lambda_{\boldsymbol{\theta}}, K_{\boldsymbol{\theta}}})_{\boldsymbol{\theta} \in \Theta}$ designed to approximate q_{opt}. Each of these distributions is defined by the jump intensity $\lambda_{\boldsymbol{\theta}}$ and density jump kernel $K_{\boldsymbol{\theta}}$ given in (2.1).

The AIS method by CE that we present allows us to jointly determine a good candidate within the family $(q_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta}$ and to estimate the probability P of the rare event that interests us.

3.1. Estimation procedure. We follow the CE minimization principle [24]. We are looking for a candidate within the family $(q_{\boldsymbol{\theta}})_{\boldsymbol{\theta} \in \Theta}$ as close as possible to the target distribution q_{opt} in the sense of the Kullback–Leibler divergence:

$$\underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\,\mathcal{D}_{\mathrm{KL}}\left(q_{\mathrm{opt}}\|q_{\boldsymbol{\theta}}\right) = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\,\mathbb{E}_{q_{\mathrm{opt}}}\left[\log\left(\frac{q_{\mathrm{opt}}(\mathcal{Z})}{q_{\boldsymbol{\theta}}(\mathcal{Z})}\right)\right],$$

$$= \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\,\int -\log\left(q_{\boldsymbol{\theta}}(\mathcal{Z})\right)\frac{\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\,\pi_0(\mathcal{Z})}{P}d\zeta(\mathcal{Z}),$$

$$(3.1) \qquad\qquad = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\,\mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\log\left(q_{\boldsymbol{\theta}}\left(\mathcal{Z}\right)\right)\right].$$

Sequential algorithm. The function $\boldsymbol{\theta}\mapsto\mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\log\left(q_{\boldsymbol{\theta}}\left(\mathcal{Z}\right)\right)\right]$ is estimated by IS under an initial instrumental distribution $q_{\boldsymbol{\theta}^{(1)}}$; we determine $\boldsymbol{\theta}^{(2)}$ which minimizes this estimate and we repeat the scheme. To save the simulation budget, we reuse at each iteration all the trajectories already drawn to perform the minimization step. Similarly, all the trajectories generated during the algorithm are recycled to produce the final estimator of P. To summarize, at iteration $\ell\in\mathbb{N}^*$, we have the following:

1. **Simulation phase.** Generate n_ℓ trajectories $\mathcal{Z}_1^{(\ell)},\dots,\mathcal{Z}_{n_\ell}^{(\ell)}\overset{\text{i.i.d.}}{\sim}q_{\boldsymbol{\theta}^{(\ell)}}$.
2. **Optimization phase.** Update the parameter of the instrumental distribution with the ℓ last samples drawn $(\mathcal{Z}_k^{(1)})_{k=1}^{n_1},\dots,\left(\mathcal{Z}_k^{(\ell)}\right)_{k=1}^{n_\ell}$:

$$(3.2) \qquad \boldsymbol{\theta}^{(\ell+1)} = \underset{\boldsymbol{\theta}\in\Theta}{\arg\min}\left\{-\sum_{r=1}^{\ell}\sum_{k=1}^{n_r}\mathbb{1}_{\mathcal{Z}_k^{(r)}\in\mathcal{D}}\frac{\pi_0(\mathcal{Z}_k^{(r)})}{q_{\boldsymbol{\theta}^{(r)}}(\mathcal{Z}_k^{(r)})}\log\left[q_{\boldsymbol{\theta}}(\mathcal{Z}_k^{(r)})\right]\right\}.$$

Estimation phase. At the final iteration L (with $N_L=\sum_{\ell=1}^{L}n_\ell$ the total budget), we reuse all past samples to get the final estimator of P:

$$(3.3) \qquad \widehat{P}_{N_L} = \frac{1}{N_L}\sum_{\ell=1}^{L}\sum_{k=1}^{n_\ell}\mathbb{1}_{\mathcal{Z}_k^{(\ell)}\in\mathcal{D}}\frac{\pi_0(\mathcal{Z}_k^{(\ell)})}{q_{\boldsymbol{\theta}^{(\ell)}}(\mathcal{Z}_k^{(\ell)})}.$$

3.2. Asymptotic optimality and confidence interval. Using theorems 2 and 3 from [46], we can determine sufficient criteria to ensure the consistency and asymptotic normality of the estimator (3.3).

Hypothesis 3.1 (assumptions on the PDMP). The PDMP of distribution π_0 with states in E, jump intensity λ_0, and jump kernel K_0 verifies the following conditions:

1. There exist $\lambda_{\min},\lambda_{\max}>0$ such that for any $z\in\mathrm{supp}(\lambda)$, $\lambda_{\min}\leq\lambda_0(z)\leq\lambda_{\max}$.
2. There exist $K_{\min},K_{\max}>0$ such that for any $z^-\in E$ and $z\in\mathrm{supp}\left(K_0\left(z^-,\cdot\right)\right)$, $K_{\min}\leq K_0(z^-,z)\leq K_{\max}$.
3. There exists $t_\varepsilon>0$ such that for any $z^-\in\partial E$ and $z\in\mathrm{supp}\left(K_0\left(z^-,\cdot\right)\right)$, $t_z^\partial\geq t_\varepsilon$.

Hypothesis 3.2 (assumptions on the parametric family $(U_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\Theta}$ and on the set Θ). The set of parameters Θ and the parametric family $(U_{\boldsymbol{\theta}})_{\boldsymbol{\theta}\in\Theta}$ of IFs verify the following conditions:

1. Θ is a compact subset of $\mathbb{R}^{d_{\boldsymbol{\theta}}}$ for some $d_{\boldsymbol{\theta}}>0$.
2. $\boldsymbol{\theta}_{\mathrm{opt}}\in\Theta$ is the unique minimizer of $\boldsymbol{\theta}\mapsto\mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\log\left(q_{\boldsymbol{\theta}}\left(\mathcal{Z}\right)\right)\right]$.
3. There exist $U_{\min},U_{\max}>0$ such that for any $\boldsymbol{\theta}\in\Theta$ and $z\in E$, $U_{\min}\leq U_{\boldsymbol{\theta}}(z)\leq U_{\max}$.

Theorem 3.3. *Under Hypothesis* 3.1 *and Hypothesis* 3.2, *with* $V:\boldsymbol{\theta}\mapsto\mathbb{E}_{\pi_0}\left[\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\frac{\pi_0(\mathcal{Z})}{q_{\boldsymbol{\theta}}(\mathcal{Z})}\right]-P^2$, *we have*

$$(3.4) \qquad \boldsymbol{\theta}^{(L)}\xrightarrow{a.s.}\boldsymbol{\theta}_{opt} \quad and \quad \sqrt{N_L}\left(\widehat{P}_{N_L}-P\right)\xrightarrow{\mathcal{L}}\mathcal{N}\left(0,V\left(\boldsymbol{\theta}_{opt}\right)\right)$$

if one of the two following conditions are satisfied:
  1. $n_\ell > 0$ *for any* $\ell > 0$ *and* $L \to +\infty$,
  2. $L < \infty$, $n_{L-1} \to +\infty$ *and* $n_L/n_{L-1} \to +\infty$.

The asymptotics can therefore be taken either in the number of iterations L or in the size of the last two samples n_{L-1} and n_L. These are two different yet specific ways to make the total number of simulated trajectories tend towards infinity. The first case, more standard, is proven in Appendix A.1. In the second case, with a fixed number of iterations L, if the number of simulated trajectories at the second to last iteration n_{L-1} tends to infinity, then we minimize $\boldsymbol{\theta} \mapsto \mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \log\left(q_{\boldsymbol{\theta}}\left(\mathcal{Z}\right)\right)\right]$, which gives $\boldsymbol{\theta}_{\text{opt}}$. Thus at the last iteration L, the trajectories are generated according to $q_{\boldsymbol{\theta}_{\text{opt}}}$. It is then sufficient that n_L/n_{L-1} tends to infinity for the proportion of trajectories drawn according to $q_{\boldsymbol{\theta}_{\text{opt}}}$ to converge to one.

We can also propose a consistent estimator of the asymptotic variance $V(\boldsymbol{\theta}_{\text{opt}})$:

$$(3.5) \qquad \widehat{\sigma}_{N_L}^2 = \frac{1}{N_L} \sum_{\ell=1}^{L} \sum_{k=1}^{n_\ell} \mathbb{1}_{\mathcal{Z}_k^{(\ell)} \in \mathcal{D}} \frac{\pi_0(\mathcal{Z}_k^{(\ell)})^2}{q_{\boldsymbol{\theta}^{(\ell)}}(\mathcal{Z}_k^{(\ell)})^2} - \widehat{P}_{N_L}^2.$$

It follows that an asymptotic confidence interval for P with the conditions of Theorem 3.3 is given by

$$(3.6) \qquad \mathbb{P}\left(P \in \left[\widehat{P}_{N_L} - v_{1-\alpha/2}\,\widehat{\sigma}_{N_L}\,N_L^{-1/2}\,;\,\widehat{P}_{N_L} + v_{1-\alpha/2}\,\widehat{\sigma}_{N_L}\,N_L^{-1/2}\right]\right) \longrightarrow 1 - a,$$

where $v_{1-\alpha/2}$ is the $(1-\alpha/2)$-quantile of the $\mathcal{N}(0,1)$ distribution.

4. Implementation guidelines. We discuss here the implementation of our method.

Initialization of the algorithm. It is known that the choice of the initial distribution of an AIS method is crucial. A common option that has the advantage of being without a priori is to take $q_{\boldsymbol{\theta}^{(0)}} = \pi_0$, but it is not suitable to deal with rare events. Let us recall that we would like to minimize $\mathbb{E}_{\pi_0}\left[\mathbb{1}_{Z \in \mathcal{D}} \log q_{\boldsymbol{\theta}}(Z)\right]$ but we minimize in practice an empirical approximation (3.2). With a poorly chosen initial auxiliary distribution, the minimizer of the approximation could be too far from the true minimizer. It is then difficult to find the right track over the iterations, and the final result of the procedure depends therefore a lot on this initial choice.

In the starting configuration of our system, the first spontaneous jump can only be the failure of a component since none are broken at the initial time. We are therefore able to set a time limit $\widetilde{t}$ and a threshold probability $\widetilde{p}$ and to determine the smallest $\widetilde{\theta}$ (of dimension 1) such that the probability under $\pi_{\widetilde{\theta}}$ that the first spontaneous jump takes place before the time $\widetilde{t}$ is larger than $\widetilde{p}$. We can then start the CE with $\boldsymbol{\theta}^{(0)} = \left(\widetilde{\theta}, \ldots, \widetilde{\theta}\right)$ and

$$(4.1) \qquad \widetilde{\theta} := \inf\left\{\theta \in \mathbb{R}_+ : \mathbb{P}_{q_\theta}\left(T_{z_0} \le \widetilde{t} \,|\, Z_0 = z_0\right) \ge \widetilde{p}\right\},$$

with T_{z_0} the time of the first random jump and hence the time of the first component failure.

Sampling size policy and stopping criterion. Given a fixed simulation budget, we would like to choose n_ℓ as the sample size at iteration q and L as the total number of iterations. Let us recall that $N_L = \sum_{\ell=1}^{L} n_\ell$ is the total number of trajectories to be drawn.

If we do not opt for a recycling scheme, it is imperative to set n_ℓ large enough (at the very least 100) even if it means doing few iterations, because we cannot rely on past samples to

approximate the true objective function in (3.2). On the other hand, with a recycling scheme, we would ideally like to choose n_ℓ as small as possible in order to perform as many iterations as possible, thus as many minimizations as possible, and to give ourselves the best chances to get close to q_{opt}.

In practice, it all depends on the optimization method used to solve (3.2). If it is expensive, we cannot afford too many iterations. Moreover, the cost of calling the function to be minimized depends linearly on the number of terms in the sum, so the minimization will be more and more expensive with each iteration (using a recycling scheme). To get an idea, if we do not want the cost dedicated to the optimization to exceed the cost dedicated to the simulation, the number of iterations L should be smaller than $2c_S/c_O - 1$, with c_S the computational cost of simulating one trajectory and c_O the computational cost of minimizing (3.2) for a single trajectory.

For a fixed simulation budget N_L, we propose to determine n_ℓ at iteration q as the minimum between the remaining budget $N_L - N_{q-1}$ and the (random) smallest number of trajectories to be drawn such that n_{CE} of them belong to $\mathcal{D}$ (with for example $n_{\mathrm{CE}} = 10$ in the case of a total budget $N_L = 10^4$).

Numerical optimization. It is necessary at each iteration to call an optimization program to solve (3.2). Let us point out that with a large number of iterations and few new trajectories at each sample, it is not useful to determine the true minimizer (which would imply using sophisticated methods adapted to nonconvex problems). We only need to improve our instrumental distribution a little bit at each iteration.

We employed the BFGS method [22] that can be found in many toolboxes. We used the function `minimize(·,method=BFGS)` from the `scipy.optimize` toolbox in Python [61]. This function performs better when given the explicit gradient of the function to be minimized rather than letting it approximate it by finite differences. We give in Appendix A.2 the explicit gradient of the function $\boldsymbol{\theta} \mapsto \log q_{\boldsymbol{\theta}}(\cdot)$ involved in (3.2).

One last piece of advice: a classical stopping criterion of a BFGS method is to obtain a sufficiently small gradient (i.e., close to zero). The default threshold in `scipy.optimize` is 10^{-5}. It is better to drastically lower this threshold to 10^{-20}, for example, because since the set of possible trajectories is a very-high-dimensional space, the densities of the trajectories are very small and the gradient of the objective function is small.

PDMP simulation. It is assumed that a suitable numerical code can be called up to compute the flow. It represents the main computational cost of the simulation. There are several methods, exact or approximate, to simulate the jump times of a PDMP [36, 7, 60, 37, 49]. Since our main constraint comes from the computation of the flow, we will opt for a method that is sparing in the number of calls to the flow. We adapt for this purpose the algorithm 3.4 of [55] which is intended for the case where the flow is explicitly known. This algorithm is based on a thinning principle which is usual for simulating time inhomogeneous Poisson processes [38]. The `PyCATSHOO` toolbox [18] is an EDF-developed computer code that enables such simulations.

MPS/MCS decomposition. Listing all the MPS or MCS of a system is an NP-hard problem. This task can be done by hand on the SFP system that we present in subsection 1.1, but it becomes impractical in the case of very large, highly redundant systems. The literature presents more methods to determine the MCS than the MPS of a system, but the two problems

are strictly equivalent. Fault trees are the most common representation of systems in the static Boolean approach, and the search for the MCS of the system belongs to the field called fault tree analysis (see [53] for a recent survey). This is an old but still active field in the industrial and academic communities. New approaches based on the differential logic calculus offer other perspectives on the decomposition of the structure function [54].

5. Numerical experiments. In this section we present the results obtained with our method, first on series and parallel systems and then on the SFP system represented in Figure 1. We compare the performances of the AIS method with each of the three families of IFs (BC-IF, MPS-IF, and MCS-IF) to a CMC method.

5.1. Series and parallel systems. We study series and parallel systems with $d_{\boldsymbol{c}}$ components. We set $\mathbb{M} = \{0,1\}^{d_{\boldsymbol{c}}}$. The mode of the system is $m = \left(m^{(1)}, \ldots, m^{(d_{\boldsymbol{c}})}\right)$, where, for $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$, the status of the jth component $m^{(j)} = 1$ if the component is active and 0 if it is broken. The mode at time $t \geq 0$ thus corresponds to the current status of each component: $M_t = (M_t^{(1)}, \ldots, M_t^{(d_{\boldsymbol{c}})})$. Therefore in the case of series systems, $M_t \in \mathbb{M}_D$ if there is $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$ such that $M_t^{(j)} = 0$, and in the case of parallel systems, $M_t \in \mathbb{M}_D$ if $M_t^{(j)} = 0$ for any $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$.

Under distribution π_0, for $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$, the jth component has a jump rate $\lambda_0^{(j)}$ that depends on its status (in other words it has a constant failure rate and a constant repair rate). From the state z^-, the next jump occurs at a random time of jump intensity $\lambda_0(z^-) = \sum_{j=1}^{d_{\boldsymbol{c}}} \lambda_0^{(j)}(z^-)$. At each jump from state z^-, only one component is randomly selected with probability $\lambda_0^{(j)}(z^-)/\lambda_0(z^-)$ for $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$, and it then changes status.

The system failure is reached either as soon as the process has spent a total time larger than $x_{\max}^{(1)}$ in $\mathbb{M}_D$ (global grace period) or when it remains in $\mathbb{M}_D$ a time larger than $x_{\max}^{(2)}$ without leaving it (local grace period). We write $X_t = (X_t^{(1)}, X_t^{(2)}, X_t^{(3)})$ as the position of the process at time $t \geq 0$ with $X_t^{(1)}$ the total time spent in $\mathbb{M}_D$ during the entire trajectory, $X_t^{(2)}$ the elapsed time since the entry in $\mathbb{M}_D$ if the process is there and 0 otherwise, and finally $X_t^{(3)} = t$ the total elapsed time. For an initial time $t_0 > 0$ and a departure state $Z_{t_0} = (X_{t_0}, M_{t_0})$, the flow of the PDMP is given by $\Phi_{Z_{t_0}} : h \mapsto (X_{t_0+h}, M_{t_0+h})$ with

$$X_{t_0+h}^{(1)} = X_{t_0}^{(1)} \mathbb{1}_{M_{t_0} \notin \mathbb{M}_D} + \left(X_{t_0}^{(1)} + h\right) \mathbb{1}_{M_{t_0} \in \mathbb{M}_D}, \tag{5.1}$$

$$X_{t_0+h}^{(2)} = \left(X_{t_0}^{(2)} + h\right) \mathbb{1}_{M_{t_0} \in \mathbb{M}_D}, \tag{5.2}$$

$$X_{t_0+h}^{(3)} = t_0 + h. \tag{5.3}$$

Importance distribution for series and parallel systems. In this subsection we will denote by $\beta_z \equiv \beta^{(\mathrm{BC})}(z)$ the number of broken components in state $z \in E$. As seen in subsection 2.1,

1. A series system with $d_{\boldsymbol{c}}$ components has 1 MPS containing all the components and $d_{\boldsymbol{c}}$ MCS each containing 1 component. Thus in series systems, $U_{\boldsymbol{\theta}}^{(\mathrm{MCS})} = U_{\boldsymbol{\theta}}^{(\mathrm{BC})}$ and $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})} = \exp\left[\theta_1^2 \mathbb{1}_{\beta_z \geq 1}\right]$.

2. A parallel system with $d_{\boldsymbol{c}}$ components has $d_{\boldsymbol{c}}$ MPS each containing 1 component and 1 MCS containing all the components. Thus in parallel systems, $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})} = U_{\boldsymbol{\theta}}^{(\mathrm{BC})}$ and $U_{\boldsymbol{\theta}}^{(\mathrm{MCS})} = \exp[(\theta_1 \beta_z / d_{\boldsymbol{c}})^2]$.

Table 1

Marginal importance jump intensity of the jth component for the importance process on series and parallel systems. The jump intensity $\lambda_{\boldsymbol{\theta}}$ and the jump kernel density $K_{\boldsymbol{\theta}}$ of the importance distribution $q_{\boldsymbol{\theta}}$ are simply written: $\lambda_{\boldsymbol{\theta}}(z) = \sum_{j=1}^{d_c} \lambda_{\boldsymbol{\theta}}^{(j)}(z)$ and $K_{\boldsymbol{\theta}}(z, z^{(j)}) = \lambda_{\boldsymbol{\theta}}^{(j)}/\lambda_{\boldsymbol{\theta}}(z)$, with $z^{(j)}$ the same state as z except for $m^{(j)}$, which is the status of the jth component.

	$\lambda_{\boldsymbol{\theta}}^{(j)}$ in a **series system**	$\lambda_{\boldsymbol{\theta}}^{(j)}$ in a **parallel system**
With $U_{\boldsymbol{\theta}}^{(\text{BC})}$	$\lambda_0^{(j)}(z) \exp\left[\theta_{\beta_z + m^{(j)}}^2 + 2(2m^{(j)} - 1)\theta_{\beta_z + m^{(j)}} \sum_{i=1}^{\beta_z} \theta_i\right]$	
With $U_{\boldsymbol{\theta}}^{(\text{MPS})}$	$\lambda_0^{(j)}(z) \exp\left[\theta_1^2 \left(\mathbb{1}_{\beta_z \geq 2(1-m^{(j)})} - \mathbb{1}_{\beta_z \geq 1}\right)\right]$	Same as broken components case
With $U_{\boldsymbol{\theta}}^{(\text{MCS})}$	Same as broken components case	$\lambda_0^{(j)}(z) \exp\left[(\theta_1/d_c)^2 \left(1 + 2(2m^{(j)} - 1)\beta_z\right)\right]$

Table 2

Results on the series system case (with jump rates of the five components from Table 7 in the appendix). The 3 AIS versions were initialized according to the method described in section 4: with the smallest one-dimensional $\boldsymbol{\theta}$ such that the probability that at least one component failure occurs before the end of the simulation is larger than $1/3$. At each iteration, we generate trajectories until we have $n_{CE} = 10$ failures before updating $\boldsymbol{\theta}$ for $N_L = 10^3$ and $n_{CE} = 50$ for $N_L = 10^4$. We stop when the total budget N_L is reached. The effective dimension of the vector $\boldsymbol{\theta}$ is not modified. For the CMC method, we simply generate N_L trajectories, and we count the proportion of faulty trajectories. The estimated probability $\widehat{P}_{N_L}$ is given by (3.3), and the estimated asymptotic standard deviation $\widehat{\sigma}_{N_L}$ is given by (3.5).

Method	N_L	$\widehat{P}_{N_L}$	$\widehat{\sigma}_{N_L}$	95% confidence interval
	10^5	4×10^{-5}	6.32×10^{-3}	$\left[8.02 \times 10^{-7} ; 7.92 \times 10^{-5}\right]$
CMC	10^6	2.9×10^{-5}	5.38×10^{-3}	$\left[1.84 \times 10^{-5} ; 3.96 \times 10^{-5}\right]$
	10^7	2.7×10^{-5}	5.19×10^{-3}	$\left[2.38 \times 10^{-5} ; 3.02 \times 10^{-5}\right]$
IS with $U_{\boldsymbol{\theta}}^{(\text{MPS})}$	10^3	2.82×10^{-5}	2.26×10^{-5}	$\left[2.68 \times 10^{-5} ; 2.96 \times 10^{-5}\right]$
	10^4	2.91×10^{-5}	2.24×10^{-5}	$\left[2.87 \times 10^{-5} ; 2.95 \times 10^{-5}\right]$
IS with $U_{\boldsymbol{\theta}}^{(\text{MCS})} = U_{\boldsymbol{\theta}}^{(\text{BC})}$	10^3	2.96×10^{-5}	2.25×10^{-5}	$\left[2.82 \times 10^{-5} ; 3.09 \times 10^{-5}\right]$
	10^4	2.89×10^{-5}	2.25×10^{-5}	$\left[2.84 \times 10^{-5} ; 2.93 \times 10^{-5}\right]$

For these cases, we give in Table 1 explicit expressions of the jump intensity $\lambda_{\boldsymbol{\theta}}$ and jump kernel $K_{\boldsymbol{\theta}}$ of the importance density $q_{\boldsymbol{\theta}}$ from the marginal jump intensities $(\lambda_{\boldsymbol{\theta}}^{(j)})_{j=1}^{d_c}$.

Each of the two systems presents a different challenge for IS. The series system requires multimodal importance distribution since the failure can come from any component, and the importance distribution for a parallel system must produce sequences where all components fail that are plausible from the perspective of jump times.

Results. We compare on a series and on a parallel system the performance of a CMC method with a sample size ranging from 10^5 to 10^7 to our three versions of the AIS method corresponding to the three families of approximations of the committor function with a sample size ranging from 10^3 to 10^4. For both the series and parallel systems, we generated trajectories of duration $t_{\max} = 1500$ with global grace period $x_{\max}^{(1)} = 75$ and local grace period $x_{\min}^{(2)} = 50$. Each system has five components. The jump parameters of the two systems are described in Table 7. The results obtained on the series system and parallel system are described in Table 2, and Table 3, respectfully.

The AIS method performs better than the CMC method in all configurations. The estimated probabilities are of the same order and the confidence intervals produced by the AIS

Table 3
Results on the parallel system case *(with jump rates of the five components from Table 7 in appendix). Same notation as in Table 2, except that the 3 IS forms were initialized with the smallest one-dimensional $\boldsymbol{\theta}$ such that the probability that at least one component failure occurs before $t_{\max}/d_{\boldsymbol{c}}$ is larger than $1/3$.*

Method	N_L	$\widehat{P}_{N_L}$	$\widehat{\sigma}_{N_L}$	95% confidence interval
	10^5	8×10^{-5}	8.94×10^{-3}	$\left[2.46 \times 10^{-5} \, ; 1.35 \times 10^{-4}\right]$
CMC	10^6	6.7×10^{-5}	7.54×10^{-3}	$\left[5.10 \times 10^{-5} \, ; 8.30 \times 10^{-5}\right]$
	10^7	6.73×10^{-5}	8.2×10^{-3}	$\left[6.22 \times 10^{-5} \, ; 7.24 \times 10^{-5}\right]$
IS with $U_{\boldsymbol{\theta}}^{(\text{MPS})} = U_{\boldsymbol{\theta}}^{(\text{BC})}$	10^3	4.85×10^{-5}	1.77×10^{-4}	$\left[3.76 \times 10^{-5} \, ; 5.94 \times 10^{-5}\right]$
	10^4	5.80×10^{-5}	2.53×10^{-4}	$\left[5.29 \times 10^{-5} \, ; 6.31 \times 10^{-5}\right]$
IS with $U_{\boldsymbol{\theta}}^{(\text{MCS})}$	10^3	5.94×10^{-5}	2.95×10^{-4}	$\left[4.12 \times 10^{-5} \, ; 7.77 \times 10^{-5}\right]$
	10^4	7.01×10^{-5}	7.14×10^{-4}	$\left[5.61 \times 10^{-5} \, ; 8.41 \times 10^{-5}\right]$

method for a given sample size are of comparable length to the confidence intervals produced by the CMC method for a sample size 10^4 times larger. It can be seen that the best performance is obtained on the series system despite a slightly lower failure probability. This result is not surprising since the failed trajectories of a series system generally contain few jumps and thus produce likelihood ratios that are easier to stabilize. Since only the failure (and nonrepair) of a single component is necessary for the system to fail, the MPS and broken components/MCS methods have the same effectiveness here. For the parallel system on the other hand, the broken components/MPS method benefits from additional degrees of freedom compared to the MCS method, which seems to make a small difference at the end. In particular, the broken components/MPS form allows the speed at which component failures must follow each other until the failure mode is reached to be dosed precisely.

5.2. The spent fuel pool. The roles of the components $(c_j)_{j=1}^{d_{\boldsymbol{c}}=15}$ are described in Figure 1. We set $\mathbb{M} = \{-1, 0, 1\}^{d_{\boldsymbol{c}}}$. The mode of the system is $m = \left(m^{(1)}, \ldots, m^{(d_{\boldsymbol{c}})}\right)$, where, for $j \in \{1, \ldots, d_{\boldsymbol{c}}\}$, the status of the jth component $m^{(j)} = 1$ if the component is active, 0 if it is inactive, and -1 if it is broken. The mode $M_t = (M_t^{(1)}, \ldots, M_t^{(d_{\boldsymbol{c}})})$ of the process at time $t \geq 0$ corresponds to the current status of each component. Recall that we have $M_t \in \mathbb{M}_D$ if, at time $t \geq 0$, all MPS are damaged or equivalently if at least one MCS has all its components broken.

We write $X_t = (X_t^{(1)}, X_t^{(2)}, X_t^{(3)})$ as the position of the process at time $t \geq 0$ with $X_t^{(1)}$ the temperature of the water in the pool in °C, $X_t^{(2)}$ the water level in the pool in meters (m), and $X_t^{(3)} = t$ the total elapsed time. The evolution of these variables is described by the system of ordinary differential equations:

$$\text{(5.4)} \qquad \frac{dX_t^{(1)}}{dt} = \mathbb{1}_{X_t^{(1)} < 100} \times \frac{r + \rho C Q (x_S^{(1)} - X_t^{(1)}) \mathbb{1}_{M_t \notin \mathbb{M}_D}}{\rho C A X_t^{(2)}},$$

$$\text{(5.5)} \qquad \frac{dX_t^{(2)}}{dt} = -\mathbb{1}_{X_t^{(1)} = 100} \times \frac{r}{\rho C A \ell},$$

$$\text{(5.6)} \qquad \frac{dX_t^{(3)}}{dt} = 1,$$

where the physical parameters are given in the Table 4 (values taken from [18]).

Table 4
Physical parameters of the SFP. Values taken from [18].

Physical parameters	Value	Description
r	$2.106 \times 10^{10}\,\mathrm{J \cdot h^{-1}}$	Residual power of the fuel
C	$4180\,\mathrm{J \cdot Kg^{-1} \cdot {}^\circ K^{-1}}$	Mass heat capacity
ρ	$990\,\mathrm{Kg \cdot m^{-3}}$	Density of the water
A	$77\mathrm{m}^2$	Area of the pool
$x_S^{(1)}$	$15^\circ\mathrm{C}$	Temperature of the water sources
Q	$550\,\mathrm{m^3\,h^{-1}}$	The debit water
ℓ	$2.257 \times 10^6\,\mathrm{J \cdot Kg^{-1}}$	Latent heat of vaporization
$t_{\max}$	$3600\,\mathrm{h}$	Duration of the mission
$x_0^{(2)}$	$19\,\mathrm{m}$	Initial level of water in the pool
$x_{\min}^{(2)}$	$16\,\mathrm{m}$	Critical threshold of the level of water in the pool

Under distribution π_0, each component $m^{(j)}$ has a jump rate $\lambda_0^{(j)}$ which depends on its status and on the values of the physical variables of the system. The jump intensity of the PDMP in a state $z \in E$ is the sum of the jump rates of the components in state z: $\lambda_0(z) = \sum_{j=1}^{d_c} \lambda_0^{(j)}(z)$. At each jump from state z^-, a component c_j is randomly selected with probability $\lambda_0^{(j)}(z^-)/\lambda_0(z^-)$ and changes status (it is repaired if it was down and fails otherwise). The system automatically reconfigures itself by enabling or disabling components so that exactly 1 MPS has all its components active if possible (be careful not to confuse inactive component $m^{(j)} = 0$ and broken component $m^{(j)} = -1$).

It is assumed that no water can be reinjected into the pool in case of evaporation for the duration of the mission $t_{\max}$. Once in $\mathbb{M}_D$ there is a first grace period before the temperature of the water reaches 100°C, but this temperature can go back down once the system is repaired. Then we have a second grace period before the water level in the pool reaches a critical threshold $x_{\min}^{(2)}$. In our model, the evaporated water is lost, and the water level cannot rise again if the system is repaired.

Results. We carry out three series of numerical simulations on the spent fuel pool system.
1. We first compare the performance of each version of our AIS method to a CMC method on a standard case with jump rates described in Table 8 in the appendix, results described in Table 5, and a probability of system failure of about 10^{-5}.
2. We then check the stability of the best version of our method which seems to be based on $U_{\boldsymbol{\theta}}^{(\text{MPS})}$. In Figure 5 we represent 50 confidence intervals at the 95% level obtained with the AIS MPS-IF method with a sample size of 10^3 trajectories still on the standard case (Table 8), and we compare them to the confidence interval obtained with the CMC method and a sample size of 10^7.
3. Since the method is stable, we can trust the confidence intervals produced and confront it with even rarer events for which it cannot be compared to a CMC method. Therefore we test the AIS MPS-IF method on an extreme case with jump rates described in Table 9 in appendix, results described in Table 6, and a probability of system failure of about 10^{-7}.

The first observation on Table 5 is that even the BC-IF method, which does not distinguish the role of each component, manages to drastically reduce the variance of the estimator compared to the CMC method (almost by a factor of 1000). Surprisingly, the performance of

Table 5

Results on the standard SFP case (with jump rates from Table 8). *The 3 AIS methods were initialized according to the same way described in section* 4: *with the smallest one-dimensional $\boldsymbol{\theta}$ such that the probability that at least one component failure occurs before the end of the simulation is larger than* 1/3. *At each iteration, we generate trajectories until we have $n_{CE} = 10$ failures before updating $\boldsymbol{\theta}$ for $N = 10^3$ and $n_{CE} = 50$ for $N = 10^4$. We stop when the total budget N is reached. The effective dimension of the vector $\boldsymbol{\theta}$ is reduced to 8. For the CMC method, we simply generate N trajectories and we count the proportion of faulty trajectories. The estimated probability $\widehat{P}_{N_L}$ is given by* (3.3), *and the estimated asymptotic standard deviation $\widehat{\sigma}_{N_L}$ is given by* (3.5).

Method	N_L	$\widehat{P}_{N_L}$	$\widehat{\sigma}_{N_L}$	95% confidence interval
CMC	10^5	2×10^{-5}	4.47×10^{-3}	$\left[0 \, ; 4.77 \times 10^{-5}\right]$
	10^6	1.3×10^{-5}	3.61×10^{-3}	$\left[5.93 \times 10^{-6} \, ; 2.01 \times 10^{-5}\right]$
	10^7	1.77×10^{-5}	4.21×10^{-3}	$\left[1.51 \times 10^{-5} \, ; 2.03 \times 10^{-5}\right]$
AIS with $U_{\boldsymbol{\theta}}^{(\mathrm{BC})}$	10^3	2.16×10^{-5}	2.35×10^{-4}	$\left[7.05 \times 10^{-6} \, ; 3.63 \times 10^{-5}\right]$
	10^4	1.79×10^{-5}	3.01×10^{-4}	$\left[1.37 \times 10^{-5} \, ; 2.22 \times 10^{-5}\right]$
AIS with $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})}$	10^3	2.19×10^{-5}	6.59×10^{-5}	$\left[1.78 \times 10^{-5} \, ; 2.60 \times 10^{-5}\right]$
	10^4	1.99×10^{-5}	2.01×10^{-5}	$\left[1.96 \times 10^{-5} \, ; 2.03 \times 10^{-5}\right]$
AIS with $U_{\boldsymbol{\theta}}^{(\mathrm{MCS})}$	10^3	1.05×10^{-5}	1.27×10^{-4}	$\left[2.65 \times 10^{-6} \, ; 1.83 \times 10^{-5}\right]$
	10^4	1.50×10^{-5}	2.04×10^{-4}	$\left[1.10 \times 10^{-5} \, ; 1.90 \times 10^{-5}\right]$

Table 6

Results with the MPS-IF approximation on the extreme SFP case (with jump rates from Table 9). *Same method as in Table* 5 *except that the initialization follows the method described in section* 4 *with the smallest one-dimensional $\boldsymbol{\theta}$ such that the probability that at least one component failure occurs before the end of the simulation is larger than* 0.9.

Method	N_L	$\widehat{P}_{N_L}$	$\widehat{\sigma}_{N_L}$	95% confidence interval
AIS with $U_{\boldsymbol{\theta}}^{(\mathrm{MPS})}$	10^3	3.31×10^{-7}	1.11×10^{-6}	$\left[2.63 \times 10^{-7} \, ; 4.00 \times 10^{-7}\right]$
	10^4	3.83×10^{-7}	1.26×10^{-6}	$\left[3.58 \times 10^{-7} \, ; 4.08 \times 10^{-7}\right]$

the MCS-IF method is closer to the BC-IF method than to the MPS-IF method. The latter is extremely efficient, with a variance reduction of 10^4. We may explain this by two reasons. The first one is that, as we have seen, the MPS-IF method is more adapted to parallel systems than the MCS-IF method, yet the structure of reliable industrial systems relies on the redundancy of components and thus on parallelism. The second one is that since we are dealing with a dynamic system in continuous time and not in discrete time, it seems more appropriate to decide how fast to go through the stages until $\mathbb{M}_D$ is reached, as the MPS-IF method allows, rather than to decide which stages are to be gone through in priority, as the MCS-IF method allows.

Note that the coefficient of variation in Table 5, which can be used as an indicator of the performance of an estimator, does not necessarily decrease with the sample size. This is due to the fact that in the first iterations of the method, the parameter $\boldsymbol{\theta}$ is not yet well chosen, and a poor importance distribution in high dimension tends to produce too small likelihood ratios. The coefficient of variation is underestimated at this time.

Figure 5 confirms the performance of the AIS MPS-IF method. The majority of the confidence intervals produced by the AIS method with a sample size of 10^3 are shorter than the confidence interval produced by the CMC method with a sample size of 10^7. Only 1
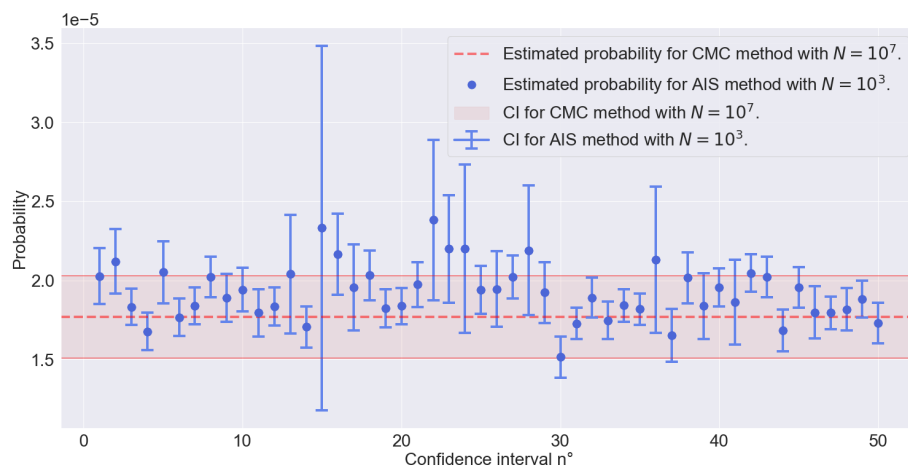
Figure 5. *Comparison of* **50** *confidence intervals at* **95%** *level obtained with MPS-IF approxima-tion.* *Each confidence interval corresponds to a run of the AIS MPS-IF method on the standard case of the SFP (Table* 8*) with* 10^3 *trajectories (same conditions as for Table* 5*). They are compared to the confidence intervals obtained with the CMC method on* 10^7 *trajectories.*

interval out of 50 is significantly larger than the interval produced by CMC, but it is relevant since it gives a probability of failure between 1×10^{-5} and 3.5×10^{-5}. We deduce that the AIS MPS-IF method is robust and that we can therefore have confidence in its estimates.

Finally, we see on the Table 6 that the AIS MPS-IF method still offers excellent perfor-mances for a 100 times rarer event. A reliable estimate of the probability that is of order 10^{-7} can be obtained with a sample size smaller than 10^4.

6. Conclusion. This work contains a comprehensive methodology for assessing the relia-bility of hybrid dynamic industrial systems, as well as a demonstration of its efficiency.

1. We have presented the mathematical modeling of the system under the form of a PDMP.
2. We have emphasized the role played by the committor function of the system in the optimality conditions of an IS method for estimating its failure probability.
3. We have proposed three different families of parametric approximations of the com-mittor function. The forms of these families are based on the decomposition of the system structure function into MPS and MCS.
4. We have proposed an AIS algorithm based on a CE procedure and a recycling scheme of past samples. The convergence and asymptotic normality of the estimator have been demonstrated. They make it possible to construct asymptotic confidence intervals of the failure probability.
5. Finally, the different versions of our method have been tested and compared on differ-ent test cases.

It is found that each version of our AIS method is considerably more efficient than a CMC method in all cases. If we compare the different AIS versions between them, it appears that the best performances are obtained when we approximate the committor function with an increasing function in the number of MPS with a broken component. The variance of the estimator produced is more than 10,000 times smaller than that of a CMC method on the

examples. It allows to estimate with accuracy a probability of failure of order 10^{-7} with a sample size smaller than 10^4.

Multilevel CE and improved CE. Sometimes, it is challenging to determine an initial instrumental distribution beforehand that enables the realization of the event $\{\mathcal{Z} \in \mathcal{D}\}$. When the event takes the form $\{\varphi(\mathcal{Z}) > \overline{\gamma}\}$, a common technique is to adaptively set intermediate thresholds $\gamma_1 < \gamma_2 < \ldots < \overline{\gamma}$ and replace the indicator $\mathbb{1}_{\varphi(\mathcal{Z}) > \overline{\gamma}}$ with $\mathbb{1}_{\varphi(\mathcal{Z}) > \gamma_\ell}$ at each step ℓ, as explained in Algorithm 1.1 of [24]. A further refinement of this technique is the improved CE method presented in [58], where the indicator function is replaced by a continuous approximation of the form $g(\mathcal{Z}; s) = F_{\mathcal{N}(0,1)}(\frac{\varphi(\mathcal{Z}) - \overline{\gamma}}{s})$, with $F_{\mathcal{N}(0,1)}$ the cumulative distribution function of the $\mathcal{N}(0,1)$ distribution (such that we have $g(\mathcal{Z}; s) \xrightarrow[s \to 0]{} \mathbb{1}_{\varphi(\mathcal{Z}) > \overline{\gamma}}$).

In our case, as we have seen, even though the event $\{X_t \in \mathbb{X}_D\}$ can generally be expressed as a threshold exceedance, the intermediate steps to be crossed are primarily determined by the modes of $\mathbb{M}$ that are not ordered. The IF can serve both to parameterize the importance distribution and to define the intermediate thresholds by ordering the modes. The major drawback of MCS-IF here is that it classifies the modes in a different order depending on the value of the vector $\boldsymbol{\theta}$ (unlike BC-IF and MPS-IF).

Extension to other applications with reverse IS trick. Our method can also serve other purposes. Recall that with the reverse importance trick, one can always estimate what the probability of failure would have been under another distribution $\widetilde{\pi}$:

$$(6.1) \qquad \mathbb{E}_{\widetilde{\pi}} \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \right] = \mathbb{E}_{q_{\boldsymbol{\theta}}} \left[\mathbb{1}_{\mathcal{Z} \in \mathcal{D}} \frac{\widetilde{\pi}(\mathcal{Z})}{q_{\boldsymbol{\theta}}(\mathcal{Z})} \right].$$

For example, a reliability sensitivity analysis can be carried out to measure the influence of variations of the jump intensity and the jump kernel (or hyperparameters of the jump intensity and the jump kernel such as failure rates of components of an industrial system) on its failure probability. Any classical sensitivity index [48] can be constructed from an input/output data set $((\lambda, K)^{(i)}, \widehat{P}_{\pi_{(\lambda, K)^{(i)}}})_{i=1,\ldots,n}$ with

$$(6.2) \qquad \widehat{P}_{\pi_{(\lambda, K)^{(i)}}} = \frac{1}{N_L} \sum_{\ell=1}^{L} \sum_{k=1}^{n_\ell} \mathbb{1}_{\mathcal{Z}_k^{(\ell)} \in \mathcal{D}} \frac{\pi_{(\lambda, K)^{(i)}}(\mathcal{Z}_k^{(\ell)})}{q_{\boldsymbol{\theta}^{(\ell)}}(\mathcal{Z}_k^{(\ell)})}$$

for $i = 1, \ldots, n$. Thus the trajectories already simulated can be recycled to estimate new quantities.

Application to other rare event methods. As mentioned earlier, approximating the committor function of the process enables the efficient implementation of variance reduction methods other than IS. Importance splitting is a family of methods used to estimate the probability of a rare event by decomposing it into a nested intersection of less rare events. The principle is to generate a set of trajectories of the process, this time following its original distribution π_0, but duplicating the most promising trajectories along the way and discarding the others. It is up to the user to choose an IF that determines whether a trajectory is promising or not, and this choice primarily determines the method's performance. Such methods have already been applied to PDMPs in the literature. For example, adaptive multilevel splitting (AMS) [14, 8] was applied to particle transport in [40], and the interacting particle systems (IPS) method

[25] was applied to industrial systems similar to ours in [17]. The optimal IF to use in AMS is the committor function U_{opt}. In the case of the IPS algorithm, the optimal IF (more exactly, the potential function used to select the promising particles) can also be expressed in terms of the committor function, although the relationship is more complex. The families of IFs we have proposed in this paper could therefore be used to efficiently implement splitting algorithms. The latter do not generally compete with a well-implemented IS method; however, their performance degrades little when the IF is not well chosen. They generally require less a priori knowledge about the system.

Appendix A.

A.1. Proof of Theorem 3.3. The results come directly from showing that we verify the hypotheses of Theorems 2 and 3 from [46]. We know by Hypotheses 3.1 and 3.2 that Θ is compact and that we have $\mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\log q_{\boldsymbol{\theta}}(\mathcal{Z})\right] > \mathbb{E}_{\pi_0}\left[-\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\log q_{\boldsymbol{\theta}_{\text{opt}}}(\mathcal{Z})\right]$ if $\boldsymbol{\theta} \neq \boldsymbol{\theta}_{\text{opt}}$. Moreover, for any $z \in E$, the continuity of the application $\boldsymbol{\theta} \mapsto U_{\boldsymbol{\theta}}(z)$ implies the continuity of $\boldsymbol{\theta} \mapsto \mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\,\pi_0(\mathcal{Z})\log q_{\boldsymbol{\theta}}(\mathcal{Z})$ for any $\mathcal{Z} \in \mathcal{E}$. To obtain the convergence of the sequence $(\boldsymbol{\theta}^{(L)})_{L>0}$, it remains to be shown that

$$(A.1) \qquad \mathbb{E}_{\pi_0}\left[\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\sup_{\boldsymbol{\theta}\in\Theta}\left\{-\log q_{\boldsymbol{\theta}}(\mathcal{Z})\right\}\right] < +\infty,$$

$$(A.2) \qquad \sup_{\boldsymbol{\theta}\in\Theta}\left\{\mathbb{E}_{\pi_0}\left[\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\frac{\pi_0(\mathcal{Z})}{q_{\boldsymbol{\theta}}(\mathcal{Z})}\sup_{\widetilde{\boldsymbol{\theta}}\in\Theta}\left\{-\log\pi_{\widetilde{\boldsymbol{\theta}}}(\mathcal{Z})\right\}^2\right]\right\} < +\infty,$$

and to get the asymptotic normality of the estimator $\widehat{P}_{N_L}$, we have to prove that there exists $\eta > 0$ such that

$$(A.3) \qquad \sup_{\boldsymbol{\theta}\in\Theta}\left\{\mathbb{E}_{\pi_0}\left[\mathbb{1}_{\mathcal{Z}\in\mathcal{D}}\left(\frac{\pi_0(\mathcal{Z})}{q_{\boldsymbol{\theta}}(\mathcal{Z})}\right)^{1+\eta}\right]\right\} < +\infty.$$

From the definitions (2.1) of $\lambda_{\boldsymbol{\theta}}$ and $K_{\boldsymbol{\theta}}$, and from Hypotheses 3.1 and 3.2, we obtain that, for any $z \in \text{supp}(\lambda_0)$, $\lambda_{\min}U_{\min}/U_{\max} \leq \lambda_{\boldsymbol{\theta}}(z) \leq \lambda_{\max}U_{\max}/U_{\min}$, and for any $z^- \in E$ and any $z \in \text{supp}(K_0(z^-,\cdot))$, $K_{\min}U_{\min}/U_{\max} \leq K(z^-,z) \leq K_{\max}U_{\max}/U_{\min}$. Then from the definition (1.5) of the density of a PDMP trajectory $\mathcal{Z} \in \mathcal{E}$ with $n_{\mathcal{Z}}$ jumps, there exist $c_{\min}, c_{\max} > 0$ such that

$$(A.4) \quad \sup_{\boldsymbol{\theta}\in\Theta} q_{\boldsymbol{\theta}}(\mathcal{Z}) \leq \prod_{k=0}^{n_{\mathcal{Z}}}\left(\lambda_{\max}\frac{U_{\max}}{U_{\min}}\right)^{\mathbb{1}_{t_k < t_{z_k}^{\partial}}}\prod_{k=0}^{n_{\mathcal{Z}}-1}K_{\max}\frac{U_{\max}}{U_{\min}} \leq (c_{\max})^{n_{\mathcal{Z}}},$$

$$(A.5) \quad \inf_{\boldsymbol{\theta}\in\Theta} q_{\boldsymbol{\theta}}(\mathcal{Z}) \geq \prod_{k=0}^{n_{\mathcal{Z}}}\left(\lambda_{\min}\frac{U_{\min}}{U_{\max}}\right)^{\mathbb{1}_{t_k < t_{z_k}^{\partial}}}\exp\left[-\lambda_{\max}\frac{U_{\max}}{U_{\min}}t_k\right]\prod_{k=0}^{n_{\mathcal{Z}}-1}K_{\min}\frac{U_{\min}}{U_{\max}} \geq (c_{\min})^{n_{\mathcal{Z}}}.$$

Using (A.4) and (A.5), we see that conditions (A.1)–(A.3) are dominated by the following: for any constant $c > 0$, $\mathbb{E}_{\pi_0}\left[c^{n_{\mathcal{Z}}}\right] < +\infty$. We have $n_{\mathcal{Z}} = n_{\mathcal{Z}}^{(\lambda)} + n_{\mathcal{Z}}^{(\partial)}$ with $n_{\mathcal{Z}}^{(\lambda)}$ the number of spontaneous jumps with jump rate λ and $n_{\mathcal{Z}}^{(\partial)}$ the number of jumps at boundaries. At most, the process reaches the state space boundary "almost immediately" after each spontaneous jump and a time t_ε after reaching another boundary. So $n_{\mathcal{Z}}^{(\partial)} \leq n_{\mathcal{Z}}^{(\lambda)} + t_{\max}/t_\varepsilon$, and thus $n_{\mathcal{Z}} \leq 2n_{\mathcal{Z}}^{(\lambda)} + t_{\max}/t_\varepsilon$. We just need to prove that $\mathbb{E}_{\pi_0}\left[c^{n_{\mathcal{Z}}^{(\lambda)}}\right] < +\infty$ for any constant c.

We define $\widetilde{\mathcal{Z}}$ as a jump process analogous to a PDMP but with some jumps rejected and not taking place. It is characterized by its flow Φ, its constant jump intensity $\lambda_{\max}$, and its jump kernel $\widetilde{\mathcal{K}}$ defined as follows:

$$(A.6) \qquad \widetilde{\mathcal{K}}\left(z^-, dz\right) = \begin{cases} \mathcal{K}_0\left(z^-, dz\right)\dfrac{\lambda_0(z^-)}{\lambda_{\max}} + \left(1 - \dfrac{\lambda_0(z^-)}{\lambda_{\max}}\right)\delta_{z^-}(dz) & \text{if } z^- \notin \partial E, \\ \mathcal{K}_0\left(z^-, dz\right) & \text{otherwise.} \end{cases}$$

Thus a part of the spontaneous jumps are "rejected" because the process remains on the same state at each jump with probability $(1 - \frac{\lambda_0(z^-)}{\lambda_{\max}})$. Following theorem 5.5 from [23], we notice that the generator of this process is the same as that of the PDMP $\mathcal{Z}$. Indeed, by denoting $\mathcal{Q}_0$ as the generator of the PDMP $\mathcal{Z}$ and $\widetilde{\mathcal{Q}}$ as the generator of the process $\widetilde{\mathcal{Z}}$, for any state $z^- \in E$ and f a function of the domain of the generator $\widetilde{\mathcal{Q}}$ (see detail in [23]),

$$\widetilde{\mathcal{Q}}f(z^-) = \langle \nabla f(z^-), \boldsymbol{g}(z^-)\rangle + \lambda_{\max}(z^-)\int_E \left[f(z) - f(z^-)\right]\widetilde{\mathcal{K}}(z^-, dz)$$

$$= \langle \nabla f(z^-), \boldsymbol{g}(z^-)\rangle + \lambda_0(z^-)\int_E \left[f(z) - f(z^-)\right]\mathcal{K}_0(z^-, dz) + \left(1 - \frac{\lambda_0(z^-)}{\lambda_{\max}}\right)\left[f(z^-) - f(z^-)\right]$$

$$= \mathcal{Q}_0 f(z^-).$$

Since the generator characterizes the distribution of the process, the trajectories of the PDMP $\mathcal{Z}$ and of the jump process $\widetilde{\mathcal{Z}}$ are identically distributed. In particular, their number of jumps $n_{\mathcal{Z}}$ and $n_{\widetilde{\mathcal{Z}}}$ also follow the same law (as well as $n_{\mathcal{Z}}^{(\lambda)}$ and $n_{\widetilde{\mathcal{Z}}}^{(\lambda)}$). If we denote $\widetilde{n}_{\widetilde{\mathcal{Z}}}^{(\lambda)}$ as the number of proposed jumps with jump intensity $\lambda_{\max}$ including the rejected ones, it is straightforward to see that $\widetilde{n}_{\widetilde{\mathcal{Z}}}^{(\lambda)}$ follows a Poisson distribution with intensity $\lambda_{\max}$ and that $\widetilde{n}_{\widetilde{\mathcal{Z}}}^{(\lambda)} \geq n_{\widetilde{\mathcal{Z}}}^{(\lambda)}$. Finally for any $c \geq 1$,

$$(A.7) \qquad \mathbb{E}_{\pi_0}\left[c^{n_{\mathcal{Z}}^{(\lambda)}}\right] = \mathbb{E}\left[c^{n_{\widetilde{\mathcal{Z}}}^{(\lambda)}}\right] \leq \mathbb{E}\left[c^{\widetilde{n}_{\widetilde{\mathcal{Z}}}^{(\lambda)}}\right] = \mathbb{E}\left[c^{\mathcal{P}(\lambda_{\max})}\right] = e^{\lambda_{\max}(c-1)} < +\infty.$$

This completes the proof of the theorem.

A.2. Gradient of the log-likelihood for instrumental distributions. At each iteration of the cross-entropy procedure, the minimization program (3.2) must be solved. The only quantity depending on $\boldsymbol{\theta}$ in the objective function is $\mathcal{Z} \mapsto \log q_{\boldsymbol{\theta}}(\mathcal{Z})$. Let us recall that the probability density function of any trajectory $\mathcal{Z} \in \mathcal{E}$ is given by (1.5). For all states $z^-, z \in E$, we note $r_{\boldsymbol{\theta}}(z^-, z) = U_{\boldsymbol{\theta}}^-(z^-)/U_{\boldsymbol{\theta}}(z)$. For $i \in \{1, \ldots, d_{\boldsymbol{\theta}}\}$, the derivative of $\log q_{\boldsymbol{\theta}}(\mathcal{Z})$ in θ_i is given by

$$\partial_{\theta_i}\log q_{\boldsymbol{\theta}}(\mathcal{Z}) = \sum_{k=0}^{n_{\mathcal{Z}}}\left[\mathbb{1}_{t_{z_k} < t_{z_k}^\partial}\frac{\partial_{\theta_i}r_{\boldsymbol{\theta}}\left(\Phi_{z_k}(t_k), \Phi_{z_k}(t_k)\right)}{r_{\boldsymbol{\theta}}\left(\Phi_{z_k}(t_k), \Phi_{z_k}(t_k)\right)}\right.$$

$$\left. - \int_0^{t_k}\lambda_0\left(\Phi_{z_k}(u)\right)\partial_{\theta_i}r_{\boldsymbol{\theta}}\left(\Phi_{z_k}(u), \Phi_{z_k}(u)\right)du\right] - \sum_{k=0}^{n_{\mathcal{Z}}-1}\frac{\partial_{\theta_i}r_{\boldsymbol{\theta}}\left(\Phi_{z_k}(t_k), z_{k+1}\right)}{r_{\boldsymbol{\theta}}\left(\Phi_{z_k}(t_k), z_{k+1}\right)},$$

with $\partial_{\theta_i}r_{\boldsymbol{\theta}}(z^-, z) = \dfrac{1}{U_{\boldsymbol{\theta}}(z)^2}\int_E \left[U_{\boldsymbol{\theta}}(z)\partial_{\theta_i}U_{\boldsymbol{\theta}}(z^+) - U_{\boldsymbol{\theta}}(z^+)\partial_{\theta_i}U_{\boldsymbol{\theta}}(z)\right]K_0(z^-, z^+)\,d\nu_{z^-}(z^+).$

A.3. Jump parameters for the series/parallel systems and the spent fuel pool system.
The marginal jump rates of each system component are presented according to its status and
according to the value of the position. The jump intensity of the process in a given state is
the sum of the marginal jump rates in that state.

Table 7

*Marginal jump intensity of each component for the **series** and **parallel** systems.*

Component	Series system	Parallel system
	Marginal jump intensity $\lambda_0^{(j)}$ for $j = 1, \ldots, d_c$	
c_1	$1 \cdot 10^{-9} \, \mathbb{1}_{m^{(1)}=0} + 1 \cdot 10^{-6} \, \mathbb{1}_{m^{(1)}=1}$	$6 \cdot 10^{-5} \, \mathbb{1}_{m^{(1)}=0} + 1 \cdot 10^{-4} \, \mathbb{1}_{m^{(1)}=1}$
c_2	$5 \cdot 10^{-9} \, \mathbb{1}_{m^{(2)}=0} + 5 \cdot 10^{-6} \, \mathbb{1}_{m^{(2)}=1}$	$2 \cdot 10^{-4} \, \mathbb{1}_{m^{(2)}=0} + 5 \cdot 10^{-4} \, \mathbb{1}_{m^{(2)}=1}$
c_3	$5 \cdot 10^{-9} \, \mathbb{1}_{m^{(3)}=0} + 1 \cdot 10^{-6} \, \mathbb{1}_{m^{(3)}=1}$	$2 \cdot 10^{-4} \, \mathbb{1}_{m^{(3)}=0} + 1 \cdot 10^{-3} \, \mathbb{1}_{m^{(3)}=1}$
c_4	$1 \cdot 10^{-9} \, \mathbb{1}_{m^{(4)}=0} + 5 \cdot 10^{-6} \, \mathbb{1}_{m^{(4)}=1}$	$6 \cdot 10^{-5} \, \mathbb{1}_{m^{(4)}=0} + 5 \cdot 10^{-4} \, \mathbb{1}_{m^{(4)}=1}$
c_5	$8 \cdot 10^{-9} \, \mathbb{1}_{m^{(5)}=0} + 6 \cdot 10^{-6} \, \mathbb{1}_{m^{(5)}=1}$	$5 \cdot 10^{-4} \, \mathbb{1}_{m^{(5)}=0} + 8 \cdot 10^{-4} \, \mathbb{1}_{m^{(5)}=1}$

Table 8

*Marginal jump intensity of each component for the **standard SFP case**.*

Component	Marginal jump intensity $\lambda_0^{(j)}$ for $j = 1, \ldots, d_c$		
c_i	when $m^{(i)} = -1$	when $m^{(i)} = 0$	when $m^{(i)} = 1$
$c_1 = \mathtt{G}_0$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-6}$	$6 \cdot 10^{-6}$
$c_{i+1} = \mathtt{G}_i, \ i = 1, 2, 3$	$8 \cdot 10^{-2}$	$2 \cdot 10^{-6}$	$30 \cdot 10^{-6}$
$c_5 = \mathtt{S}_1$	$1 \cdot 10^{-2}$	$4 \cdot 10^{-6}$	$20 \cdot 10^{-6}$
$c_6 = \mathtt{S}_2$	$3 \cdot 10^{-2}$	$1 \cdot 10^{-6}$	$5 \cdot 10^{-6}$
$c_{6+i} = \mathtt{L}_{i,1}, \ i = 1, 2, 3$	$(6 - 0.03 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.05 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.1 X_t^{(1)}) \cdot 10^{-6}$
$c_{9+i} = \mathtt{L}_{i,2}, \ i = 1, 2, 3$	$(6 - 0.03 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.05 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.1 X_t^{(1)}) \cdot 10^{-6}$
$c_{12+i} = \mathtt{L}_{i,3}, \ i = 1, 2, 3$	$(6 - 0.03 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.05 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.1 X_t^{(1)}) \cdot 10^{-6}$

Table 9

*Marginal jump intensity of each component for the **extreme SFP case**.*

Component	Marginal jump intensity $\lambda_0^{(j)}$ for $j = 1, \ldots, d_c$		
c_i	when $m^{(i)} = -1$	when $m^{(i)} = 0$	when $m^{(i)} = 1$
$c_1 = \mathtt{G}_0$	$4 \cdot 10^{-2}$	$4 \cdot 10^{-6}$	$6 \cdot 10^{-6}$
$c_{i+1} = \mathtt{G}_i, \ i = 1, 2, 3$	$10 \cdot 10^{-2}$	$15 \cdot 10^{-6}$	$30 \cdot 10^{-6}$
$c_5 = \mathtt{S}_1$	$1 \cdot 10^{-2}$	$4 \cdot 10^{-6}$	$20 \cdot 10^{-6}$
$c_6 = \mathtt{S}_2$	$3 \cdot 10^{-2}$	$1 \cdot 10^{-6}$	$5 \cdot 10^{-6}$
$c_{6+i} = \mathtt{L}_{i,1}, \ i = 1, 2, 3$	$(12 - 0.04 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.1 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.1 X_t^{(1)}) \cdot 10^{-6}$
$c_{9+i} = \mathtt{L}_{i,2}, \ i = 1, 2, 3$	$(12 - 0.04 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.1 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.1 X_t^{(1)}) \cdot 10^{-6}$
$c_{12+i} = \mathtt{L}_{i,3}, \ i = 1, 2, 3$	$(15 - 0.05 X_t^{(1)}) \cdot 10^{-2}$	$(1 + 0.08 X_t^{(1)}) \cdot 10^{-6}$	$(3 + 0.08 X_t^{(1)}) \cdot 10^{-6}$

REFERENCES

[1] A. Abate, H. Blom, M. Bouissou, N. Cauchi, H. Chraibi, J. Delicaris, S. Haesaert, A. Hartmanns, M. Khaled, A. Lavaei, et al., *Arch-comp21 category report: Stochastic models*, in Proceedings of the 8th International Workshop on Applied Verification of Continuous and Hybrid Systems 2021, EasyChair, Brussels, 2021, pp. 55–89.

[2] R. Arismendi, A. Barros, and A. Grall, *Piecewise deterministic markov process for condition-based maintenance models — Application to critical infrastructures with discrete-state deterioration*, Reliab. Eng. Syst. Safe., 212 (2021), 107540.

[3] R. Azaïs, J.-B. Bardet, A. Génadot, N. Krell, and P.-A. Zitt, *Piecewise deterministic Markov process — Recent results*, ESAIM Proc. Surveys, 44 (2014), pp. 276–290.

[4] J. Bierkens, P. Fearnhead, and G. Roberts, *The zig-zag process and super-efficient sampling for Bayesian analysis of big data*, Ann. Statist., 47 (2019), pp. 1288–1320.

[5] H. A. Blom, H. Ma, and G. B. Bakker, *Interacting particle system-based estimation of reach probability for a generalized stochastic hybrid system*, IFAC-PapersOnLine, 51 (2018), pp. 79–84.

[6] A. Bouchard-Côté, S. J. Vollmer, and A. Doucet, *The bouncy particle sampler: A nonreversible rejection-free Markov chain Monte Carlo method*, J. Amer. Statist. Assoc., 113 (2018), pp. 855–867.

[7] M. Bouissou, H. Elmqvist, M. Otter, and A. Benveniste, *Efficient Monte Carlo simulation of stochastic hybrid systems*, in Proceedings of The 10th International Modelica Conference, 2014.

[8] C.-E. Bréhier, M. Gazeau, L. Goudenege, T. Lelièvre, and M. Rousset, *Unbiasedness of some generalized adaptive multilevel splitting algorithms*, Ann. Appl. Probab., 26 (2016), pp. 3559–3601.

[9] J. A. Bucklew, *Introduction to Rare Event Simulation*, Springer, New York, 2004.

[10] C. E. Budde and M. Stoelinga, *Automated rare event simulation for fault tree analysis via minimal cut sets*, in Proceedings of the International Conference on Measurement, Modelling and Evaluation of Computing Systems, Springer, New York, 2020, pp. 259–277.

[11] M. F. Bugallo, V. Elvira, L. Martino, D. Luengo, J. Miguez, and P. M. Djuric, *Adaptive importance sampling: The past, the present, and the future*, IEEE Signal Process. Mag., 34 (2017), pp. 60–79.

[12] M.-L. Bujorianu and J. Lygeros, *General stochastic hybrid systems: Modelling and optimal control*, in Proceedings of the 2004 43rd IEEE Conference on Decision and Control, Vol. 2, IEEE, 2004, pp. 1872–1877.

[13] M. Čepin, *Assessment of Power System Reliability: Methods and Applications*, Springer, New York, 2011.

[14] F. Cérou and A. Guyader, *Adaptive multilevel splitting for rare event analysis*, Stoch. Anal. Appl., 25 (2007), pp. 417–443.

[15] F. Cérou, A. Guyader, and M. Rousset, *Adaptive multilevel splitting: Historical perspective and recent results*, Chaos, 29 (2019), 043108.

[16] H. Chraibi, A. Dutfoy, T. Galtier, and J. Garnier, *On the optimal importance process for piecewise deterministic markov process*, ESAIM Probab. Stat., 23 (2019), pp. 893–921.

[17] H. Chraibi, A. Dutfoy, T. Galtier, and J. Garnier, *Optimal potential functions for the interacting particle system method*, Monte Carlo Methods Appl., 27 (2021), pp. 137–152.

[18] H. Chraibi, J. C. Houdebine, and A. Sibler, *PyCATSHOO: Toward a new platform dedicated to dynamic reliability assessments of hybrid systems*, in Proceedings of the 13th International Conference on Probabilistic Safety Assessment and Management (PSAM 13), 2016.

[19] J.-M. Cornuet, J.-M. Marin, A. Mira, and C. P. Robert, *Adaptive multiple importance sampling*, Scand. J. Stat., 39 (2012), pp. 798–812.

[20] O. L. Costa and F. Dufour, *Stability and ergodicity of piecewise deterministic Markov processes*, SIAM J. Control Optim., 47 (2008), pp. 1053–1077.

[21] Y. Crama and P. L. Hammer, *Boolean Functions: Theory, Algorithms, and Applications*, Cambridge University Press, Cambridge, 2011.

[22] Y.-H. Dai, *Convergence properties of the BFGS algoritm*, SIAM J. Optim., 13 (2002), pp. 693–701.

[23] M. H. A. Davis, *Piecewise-deterministic Markov processes: A general class of non-diffusion Stochastic models*, J. R. Stat. Soc. Ser. B. Stat. Methodol., 46 (1984), pp. 353–388.

[24] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, *A tutorial on the cross-entropy method*, Ann. Oper. Res., 134 (2005), pp. 19–67.

[25] P. Del Moral and J. Garnier, *Genealogical particle analysis of rare events*, Ann. Appl. Probab., 15 (2005), pp. 2496–2534.

[26] L. Desgeorges, P.-Y. Piriou, T. Lemattre, and H. Chraibi, *Formalism and semantics of pycatshoo: A simulator of distributed stochastic hybrid automata*, Reliab. Eng. Syst. Safe., 208 (2021), 107384.

[27] M. EL MASRI, J. MORIO, AND F. SIMATOS, *Improvement of the cross-entropy method in high dimension for failure probability estimation through a one-dimensional projection without gradient estimation*, Reliab. Eng. Syst. Safe., 216 (2021), 107991.

[28] M. ELMASRI, J. MORIO, AND F. SIMATOS, *Optimal Projection to Improve Parametric Importance Sampling in High Dimension*, arXiv:2107.06091, 2021.

[29] V. ELVIRA AND L. MARTINO, *Advances in importance sampling*, Wiley StatsRef, 2021, pp. 1–14.

[30] T. GALTIER, *Accelerated Monte-Carlo Methods for Piecewise Deterministic Markov Processes for a Faster Reliability Assessment of Power Generation Systems Within the PyCATSHOO Toolbox*, Ph.D. thesis, Université de Paris, Paris, 2019.

[31] H. KAHN AND T. E. HARRIS, *Estimation of particle transmission by random sampling*, National Bureau of Standards Applied Mathematics Series, 12 (1951), pp. 27–30.

[32] Y. KHOO, J. LU, AND L. YING, *Solving for high-dimensional committor functions using artificial neural networks*, Res. Math. Sci., 6 (2019), pp. 1–13.

[33] J. KRYSTUL AND H. A. BLOM, *Sequential Monte Carlo simulation of rare event probability in stochastic hybrid systems*, IFAC Proc. Volumes, 38 (2005), pp. 176–181.

[34] A. LASOTA, M. C. MACKEY, AND J. TYRCHA, *The statistical dynamics of recurrent biological events*, J. Math. Biol., 30 (1992), pp. 775–800.

[35] W.-S. LEE, D. L. GROSH, F. A. TILLMAN, AND C. H. LIE, *Fault tree analysis, methods, and applications – A review*, IEEE Trans. Reliab., 34 (1985), pp. 194–203.

[36] V. LEMAIRE, M. THIEULLEN, AND N. THOMAS, *Exact simulation of the jump times of a class of piecewise deterministic Markov processes*, J. Sci. Comput., 75 (2018), pp. 1776–1807.

[37] V. LEMAIRE, M. THIEULLEN, AND N. THOMAS, *Thinning and multilevel Monte Carlo methods for piecewise deterministic (Markov) processes with an application to a stochastic Morris–Lecar model*, Adv. Appl. Probab., 52 (2020), pp. 138–172.

[38] P. W. LEWIS AND G. S. SHEDLER, *Simulation of nonhomogeneous Poisson processes by thinning*, Nav. Res. Logist. Q., 26 (1979), pp. 403–413.

[39] Q. LI, B. LIN, AND W. REN, *Computing committor functions for the study of rare events using deep learning*, J. Chem. Phys., 151 (2019), 054112.

[40] H. LOUVIN, E. DUMONTEIL, T. LELIÈVRE, M. ROUSSET, AND C. M. DIOP, *Adaptive multilevel splitting for monte carlo particle transport*, in EPJ Web of Conferences 153, EDP Sciences, Les Ulis, France, 2017, 06006.

[41] D. LUCENTE, S. DUFFNER, C. HERBERT, J. ROLLAND, AND F. BOUCHET, *Machine Learning of Committor Functions for Predicting High Impact Climate Events*, arXiv:1910.11736, 2019.

[42] J.-M. MARIN, P. PUDLO, AND M. SEDKI, *Consistency of adaptive importance sampling and recycling schemes*, Bernoulli, 25 (2019), pp. 1977–1998.

[43] R. T. MCGIBBON, B. E. HUSIC, AND V. S. PANDE, *Identification of simple reaction coordinates from complex dynamics*, J. Chem. Phys., 146 (2017), 044109.

[44] P. METZNER, C. SCHÜTTE, AND E. VANDEN-EIJNDEN, *Transition path theory for Markov jump processes*, Multiscale Model. Simul., 7 (2009), pp. 1192–1219.

[45] G. POLA, M.-L. BUJORIANU, J. LYGEROS, AND M. D. DI BENEDETTO, *Stochastic hybrid models: An overview*, IFAC Proc. Volumes, 36 (2003), pp. 45–50.

[46] F. PORTIER AND B. DELYON, *Asymptotic optimality of adaptive importance sampling*, Adv. Neural Inform. Process. Syst. 31, 2018.

[47] A. RAUZY, *New algorithms for fault trees analysis*, Reliab. Eng. Syst. Safe., 40 (1993), pp. 203–211.

[48] S. RAZAVI, A. JAKEMAN, A. SALTELLI, C. PRIEUR, B. IOOSS, E. BORGONOVO, E. PLISCHKE, S. L. PIANO, T. IWANAGA, W. BECKER, ET AL., *The future of sensitivity analysis: An essential discipline for systems modeling and policy support*, Environ. Model. Softw., 137 (2021), 104954.

[49] M. G. RIEDLER, *Almost sure convergence of numerical approximations for piecewise deterministic Markov processes*, J. Comput. Appl. Math., 239 (2013), pp. 50–71.

[50] G. RUBINO AND B. TUFFIN, *Rare Event Simulation Using Monte Carlo Methods*, Wiley, New York, 2009.

[51] R. Y. RUBINSTEIN AND P. W. GLYNN, *How to deal with the curse of dimensionality of likelihood ratios in Monte Carlo simulation*, Stoch. Models, 25 (2009), pp. 547–568.

[52] R. RUDNICKI AND M. TYRAN-KAMIŃSKA, *Piecewise deterministic markov processes in biological models*, in Semigroups of Operators-Theory and Applications, Springer, 2015, pp. 235–255.

[53] E. RUIJTERS AND M. STOELINGA, *Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools*, Comput. Sci. Rev., 15 (2015), pp. 29–62.

[54] P. RUSNAK, E. ZAITSEVA, F. COOLEN, M. KVASSAY, AND V. LEVASHENKO, *Logic differential calculus for reliability analysis based on survival signature*, IEEE Trans. Depend. Secure Comput., (2022).

[55] B. D. SAPORTA, F. DUFOUR, AND H. ZHANG, *Numerical Methods for Simulation and Optimization of Piecewise Deterministic Markov Processes: Application to Reliability*, Mathematics and Statistics Series, ISTE, London, 2016.

[56] A. TABANDEH, G. JIA, AND P. GARDONI, *A review and assessment of importance sampling methods for reliability analysis*, Struct. Saf., 97 (2022), 102216.

[57] P. TURATI, N. PEDRONI, AND E. ZIO, *Advanced restart method for the estimation of the probability of failure of highly reliable hybrid dynamic systems*, Reliab. Eng. Syst. Safe., 154 (2016), pp. 117–126.

[58] 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 J. Uncertain. Quantif., 9 (2021), pp. 818–847.

[59] P. VANETTI, A. BOUCHARD-CÔTÉ, G. DELIGIANNIDIS, AND A. DOUCET, *Piecewise-deterministic Markov Chain Monte Carlo*, arXiv:1707.05296, 2017.

[60] R. VELTZ, *A New Twist for the Simulation of Hybrid Systems Using the True Jump Method*, arXiv:1504.06873, 2015.

[61] P. VIRTANEN, R. GOMMERS, T. E. OLIPHANT, M. HABERLAND, T. REDDY, D. COURNAPEAU, E. BUROVSKI, P. PETERSON, W. WECKESSER, J. BRIGHT, ET AL., *Scipy 1.0: Fundamental algorithms for scientific computing in Python*, Nat. Methods, 17 (2020), pp. 261–272.

[62] Z. WANG AND J. SONG, *Cross-entropy-based adaptive importance sampling using von mises-fisher mixture for high dimensional reliability analysis*, Struct. Saf., 59 (2016), pp. 42–52.

[63] P. ZULIANI, C. BAIER, AND E. M. CLARKE, *Rare-event verification for stochastic hybrid systems*, in Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control, 2012, pp. 217–226.