Abstract—This article concerns continuous-time, continuous-space stochastic dynamical systems described by stochastic differential equations (SDE). It presents a new approach to compute probabilistic safety regions, namely sets of initial conditions of the SDE associated to trajectories that are safe with a probability larger than a given threshold. The approach introduces a functional that is minimized at the border of the probabilistic safety region, then solves an optimization problem using techniques from Malliavin Calculus, which computes such region. Unlike existing results in the literature, the new approach allows one to compute probabilistic safety regions without gridding the state space of the SDE.

Index Terms—Formal verification, stochastic processes.

I. BACKGROUND

In control engineering and in formal verification, a fundamental problem is safety analysis: this concerns identifying states of a dynamical model that are safe, namely that are associated to trajectories that do not escape (whether over finite or infinite time) a given set that is deemed to be safe [1], [2], [3]. Dually, one can express a reachability analysis problem by identifying states that are associated with trajectories entering a given target set. In the context of probabilistic models, such as stochastic differential equations (SDEs), we are interested in characterizing and computing the likelihood with which trajectories of the stochastic process either stay within a given set, or dually reach a target set. Alternatively, for stochastic models, we might be interested in computing the set of initial conditions associated with dynamics that are safe with a probability at least equal to, say $p$—this is also known as $p$-safe analysis or computation of the $p$-safe region [4].

In this work, we present a new application of Malliavin Calculus [5] to the computation of the $p$-safe region borrowing ideas from Mathematical Finance: in particular, we leverage techniques for the computation of the Greeks of a derivative [6] for our goal. This allows one to compute probabilistic safety regions without gridding the state space of the SDE: grid-based techniques are by-and-large the standard approach in existing literature.

II. RELATED LITERATURE ON PROBABILISTIC SAFETY

Safety analysis, a standard specification (or requirement) in formal verification, has been studied on dynamical models within the hybrid systems community [3]. Corresponding safety notions for stochastic models have been explicitly introduced only over the past two decades [2], [7]. This work focuses on continuous-time stochastic models: particularly relevant for this setup, Hu et al. [8] has presented a new modeling framework named stochastic hybrid system (SHS), which extends with randomness the deterministic framework of hybrid models by allowing the continuous flow inside each invariant set of the discrete state variables to be governed by stochastic differential equations (SDEs). Hu et al. [8] propose the notion of embedded Markov chain (EMC) and studies the exit probability problem, which is related to reachability analysis: it is shown that this probability over the EMC converges to its counterpart for the original SHS. Bujorianu and Lygeros [9] blend the models from [8] with Markov models with jumps in [10], setting up Markov strings, and thus, obtaining a very general class of models for SHS. Closely related to [9], Bujorianu [11] introduces a general model for SHS and proposes a method based on Dirichlet forms to study the reachability problem over SHS models. Similarly over SHS, Koutsoukos and Riley [12] show that reachability can be characterized as a viscosity solution of a system of coupled Hamilton–Jacobi–Bellman equations. Riley et al. [13] build on [12] by employing Monte Carlo (MC) techniques for estimating probabilities of events, and Riley et al. [13] use multilevel splitting (MLS), a variance-reduction technique. Again over SHS, Esfahani et al. [14] establish a connection between stochastic reach-avoid problems—problems encompassing both reachability and safety, also known as constrained reachability problems—and optimal control problems involving discontinuous payoff functions. Focusing on a particular stochastic optimal control problem, namely the exit-time problem mentioned previously, Esfahani et al. [14] provide its characterization as a solution of a partial differential equation, along with Dirichlet boundary conditions.

Wisniewski et al. [15] present a method to compute protection certificates, which are closely related to the concept of $p$-safe region, elaborated later. As discussed in Remark 1, the authors in [16] and [17] compute the $p$-safe region based on the extended generator of stochastic dynamical systems; these contributions characterize the safety problem as an optimization problem on the space of positive measures, and then, solve it via a moment-based method. [18] characterizes the $p$-safe regions using concepts from the Potential theory.

Alternative techniques leveraging randomized approaches have been presented: Bujorianu and Wisniewski [19] introduce a method for estimating the probability of conflict for two-aircraft encounters at a fixed altitude—a probabilistic safety problem. Along similar lines, the authors in [20] discuss the maximum instantaneous probability of conflict: randomized algorithms are introduced to efficiently estimate this measure of criticality and to provide quantitative bounds on the level of the approximation introduced. Finally, the authors in [21] enable sound verification and correct-by-construction controller synthesis for stochastic models and their hybrid extensions [22]: a stochastic control model satisfying a probabilistic variant of incremental input-to-state

Manuscript received 13 August 2022; accepted 27 December 2022. Date of publication 24 January 2023; date of current version 27 September 2023. The work of Francesco Cosentino was supported in part by The Alan Turing Institute, TUC/000201, under the EPSRC Grant No. EP/N510129/1, in part by the EPSRC grant "Datassig" [EP/S026347/1], in part by The Alan Turing Institute, and in part by Oxford-Man Institute. Recommended by Associate Editor A. A. Malikopoulos. (Corresponding author: Francesco Cosentino.)

Francesco Cosentino and Harald Oberhauser are with the Mathematical Institute, University of Oxford, Oxford OX1 2JD, U.K., and also with The Alan Turing Institute, British Library, London NW1 2DB, U.K. (e-mail: francesco.cosentino211@gmail.com; harald. oberhauser@maths.ox.ac.uk).

Alessandro Abate is with the Computer Science Department, University of Oxford, Oxford OX1 2JD, U.K., and also with The Alan Turing Institute, British Library, London NW1 2DB, U.K. (e-mail: alessandro.abate@cs.ox.ac.uk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TAC.2023.3239460.

Digital Object Identifier 10.1109/TAC.2023.3239460
stability is shown to be abstracted into a finite-state transition system, which is epsilon-approximately bisimilar to the original model.

III. PROBLEM STATEMENT

Let us consider a $d$-dimensional Brownian motion $W_t \in \mathbb{R}^d$ defined on a filtered probability space $(\Omega, \mathcal{F}, (\mathcal{F}_t), \mathbb{P})$, and the following SDE:

$$dX_t = \mu(X_t)dt + \sum_{k=1}^{d} \sigma_k(X_t)dW^k_t, \quad X_0 = x.$$  \hspace{1cm} (1)

This standard setup is adopted by related literature, as surveyed previously. We introduce the following requirements, which are used in [5] and in particular are sufficient for all the results and algorithms proposed in this work. Obtaining stronger requirements, and thus, generalizing our setup, would require modifying the technical results from Malliavin calculus, which is not core to our contributions.

Assumption 1: We suppose that the vector fields $\mu, \sigma_k \in C^\infty_\lambda(\mathbb{R}^d, \mathbb{R}^d)$, $k = 1, 2, \ldots, d$, where $C^\infty_\lambda(\mathbb{R}^d, \mathbb{R}^d)$ indicates the space of infinitely differentiable functions with bounded derivatives and bounded linear growth from $\mathbb{R}^d$ to $\mathbb{R}^d$. Moreover, if we call $\sigma \in C^\infty_\lambda(\mathbb{R}^d, \mathbb{R}^{d \times d})$ the matrix whose columns are the vectors $\sigma_k, k \geq 1$, we assume that $\sigma$ satisfies the uniform ellipticity condition, i.e., $\sigma \sigma^\top$ is uniformly positive definite.

If Assumption 1 holds, then it is well-known that the SDE (1) has a unique strong solution $X^*_t$ [23], and whenever clear from the context, we shall omit the index $x$.

Let us consider a bounded and smooth region $A$ and let $\partial A$ denote the border of $A$. We call $\tau^*_A$ the exit time of $X^*_t$ from the region $A$, i.e.,

$$\tau^*_A = \inf\{t \geq 0 : X^*_t \notin A\}, \quad x \in A.$$  \hspace{1cm} (2)

Whenever clear from context we shall omit the indexes $x, A$.

We define $A^p_T$ to be the $p$-safe portion of a region $A$, or equivalently $p$-safe region of $A$, as the initial points $x \in A$ such that if $X_t$ starts from $x$, then it stays in $A$ longer than $T$ with probability greater than $p$, i.e.,

$$A^p_T = \{x \in A : \mathbb{P}(\tau^*_x \geq T) \geq p\}.$$  \hspace{1cm} (3)

Again, whenever clear from the context, we shall omit the indexes $p, T$.

Remark 1: In [16], the authors study a more general problem, namely the probabilistic reach-avoid problem, defined next. Given a safe set $S$ and an unsafe set $U$, they compute the probability to leave $S$ before entering in $U$, before a prespecified time $T$, i.e., $\{x \in S : \mathbb{P}(\tau^*_U \leq \tau^*_S, \tau^*_U \leq T) \leq q\}$. If we consider a set $U$ s.t. $\overline{U} = S$, then the wanted quantity becomes $\{x \in S : \mathbb{P}(\tau^*_S < T) \leq q\}$, which is exactly the dual of $A^p_T$ for $p = 1 - q$. An analysis of the approximation error is not presented, and since the approach is radically different from the one presented in this article (cf. discussion in Section II and in the next section), a quantitative comparison between the two approaches is not pursued.

The standard way to compute $A^p_T$ is to discretize the region $A$ and to compute the value $\mathbb{P}(\tau^*_x \geq T)$ at any point in the introduced grid (cf. Related Literature). In the following instead, using ideas from mathematical finance and results from the Malliavin calculus, we show how to compute $A^p_T$ with a grid-free technique. The approach hinges on the observation that the border of $A^p_T$ can be expressed as

$$\partial A^p_T = \arg\min \frac{1}{2} \left( \mathbb{P}(\tau^*_x \geq T) - p \right)^2.$$  \hspace{1cm} (4)

The main idea of this approach is thus to solve such optimization problem: indeed, assuming differentiability and excluding convexity issues, we know that, setting up the recursion

$$x_{j+1} = x_j - \lambda (\mathbb{P}(\tau^*_x \geq T) - p) D_x \mathbb{P}(\tau^*_x \geq T)$$  \hspace{1cm} (5)

then $x_j \to x^*$, where $x^* \in \partial A^p_T$, for $\lambda > 0$ small enough. Equation (3) represents a standard gradient descent (GD) step. We remark that in principle other optimization algorithms can be used to solve the problem in (2); in this work, we focus on first-order gradient-based optimization procedures, of which GD is an exemplar. As an alternative instance to standard GD, in the case study, we employ ADAM [24], a state-of-the-art optimization procedure. Using the GD in (3) not only allows identifying the set $A^p_T$: in Section VI, we also provide a procedure to explore its border. Furthermore, in Theorem 2, we prove that the “interior” of the obtained region delimited by $\partial A^p_T$ is in $A^p_T$, which implies that there is no need to check these internal points. Moreover, in Proposition 1 and Corollary 1, we show how to check if a point $x$ is “inside” $A^p_T$ without computing $\mathbb{P}(\tau^*_x \geq T)$, but only using the gradient at a specific point on $\partial A^p_T$, which is generated by the optimization procedure.

A. Grid-Based Versus Grid-Free Approaches

Evidently, the GD step in (3) depends on the two quantities $\mathbb{P}(\tau^*_x \geq T)$ and $D_x \mathbb{P}(\tau^*_x \geq T)$: it should be clear that if we can compute them (or approximations thereof) with a grid-free method, then the overall procedure will result in a grid-free computation of the $p$-safe portion of the region. An important advantage of such grid-free approach is that if the $p$-safe region $A^p_T$ is not empty—however small, even if it was a zero-measure set—then it will be found. Instead, grid-based approaches (broadly all those presented in the previous section on related work) will find the set $A^p_T$ only if it intersects with the introduced grid. As an extreme instance, if the $p$-safe region consists of only one point, the procedure introduced here shall find it, up to a numerical precision related to the approximation of $\mathbb{P}(\tau^*_x \geq T)$ and $D_x \mathbb{P}(\tau^*_x \geq T)$; on the contrary, this might not be possible for grid-based approaches, unless the grid is selected to intersect such point, which is usually not known beforehand. Another extreme case can be identified when the $p$-safe region of interest is not bounded: in such case, the approach underpinning grid-based methods can be quite inefficient, while the grid-free-based approach presented here shall converge to its border $\partial A^p_T$, and explore as much of it as computationally feasible.

A general, formal comparison between the two approaches is questionable: while grid-free strategies search for solutions within an uncountable infinite set, grid-based procedures search for solutions within a predefined, possibly finite set.

IV. MALLIavin CALCULUS FOR STOPPING TIMES

In the pricing theory, a branch of mathematical finance, a classical problem is to evaluate the variation of the price of a derivative, in response to a change of the underlying asset price or volatility [6]. These quantities are known as Greeks. Given an underlying asset, whose price $P_t$ is the solution of an SDE starting in $p_0$, the price $D$ of a derivative is given as the expectation of a functional of $P_t$.

In [16], the authors compute $\mathbb{P}(\tau^*_U \geq \tau^*_S, \tau^*_U \leq T) \leq q$, whereas, here, we use $\leq$, as it does not change the outcome, while greatly simplifying the comparison between [16] and this article.

1We employ here for simplicity a quadratic function $f(x) = 0.5(x - p)^2$, however, any other differentiable function $f$ minimized in $x = p$ is also appropriate for the task.
i.e., \( D_{p_0} = \mathbb{E}[f(P_{p_0}^T)] \), \( T > 0 \). If we call \( p_0 \) the initial price of the underlying asset, the Greek representing the sensitivity with respect to the initial price is called \( \Delta \), and is formally defined as \( \Delta = \partial D_{p_0}/\partial p_0 \). Through Malliavin calculus, it is possible to provide explicit formulae for the Greeks [5], [25], [26], [27], [28], [29]. We refer to [30] for a computational perspective on these methods.

In our problem setup, we are interested to compute the quantity \( D_x \mathbb{P}(\tau^x > T) \) used in (3), where \( \tau^x \) is a specific exit time related to the probabilistic safety property: this is challenging because it involves the derivative of a nonsmooth indicator functional of the exit time. Under Assumption 1, it is possible to show that \( X_\tau^x \) is a.s. differentiable with respect to the starting point \( x \) [31]. Following the notations in [5], let us introduce \( J_t = D_x X_t \), and \( \mu = D_x \mu(x), \sigma_k = D_x \sigma(x) \) \in \( C_\infty^\infty(\mathbb{R}^d; \mathbb{R}^{d \times d}) \); then \( J_t \) solves

\[
dJ_t = \mu(X_t) dt + \sum_{k=1}^d \sigma_k(X_t) dW_t^k, \quad J_0 = I_d.
\]

The main results we leverage is the following.

**Theorem 1**: [5, Th. 2.18] If Assumption 1 holds true, calling \( \tau \) the time when \( \int_0^\tau \operatorname{dist}(X_t, \partial A)^2 dt = 1 \), then

\[
\frac{\partial}{\partial \varepsilon} \mathbb{P}(\tau^{x + \varepsilon} > T) \bigg|_{\varepsilon = 0} = \mathbb{E} [I_{\{\tau > T\}} H_{\varepsilon} \tau]
\]

where \( H_{\varepsilon} \tau = \sum_{k=1}^d \beta_k \sigma_k(X_t) \cdot J_t \cdot \varepsilon \). Setting the directions \( \varepsilon = e_i, i = 1, \ldots, d \), we can obtain the gradient \( D_x \mathbb{P}(\tau^x > T) \), which lies at the core of our procedure in (3): without this result, this derivative should be estimated alternatively, for instance numerically. Therefore, we have that \( D_x \mathbb{P}(\tau^x > T) = \mathbb{E}[I_{\{\tau^x > T\}} H_T] \), where

\[
H_T = \int_0^\tau \frac{I_{\{t < \tau\}}}{\operatorname{dist}(X_t, \partial A)^2} \beta_t \cdot dW_t, \quad \beta_t = \sigma^{-1}(X_t) \cdot J_t \tag{4}
\]

and \( \sigma \) is the matrix whose columns are the vectors \( \sigma_k \). Note that the dimension of \( \beta_t, H_t \) in (4) and Theorem 1 are different.

**V. PROPERTIES OF THE REGION \( A_T^p \)**

While Theorem 1 can be useful for the problem at hand, from an algorithmic point of view, there are still a few subtle points to be handled.

First, we do not know whether the quantity \( \mathbb{P}(\tau^x > T) \) is convex or not. Nevertheless, we know that \( x \) is in \( \partial A_T^p \) if \( \mathbb{P}(\tau^x > T) = p \), which implies that the quantity in (3) \( \mathbb{P}(\tau^x > T) - p \) is zero, regardless of the value of the gradient \( D_x \mathbb{P}(\tau^x > T) \). Therefore, if we end at a point \( x \) where

\[
\mathbb{P}(\tau^x > T) \neq p \quad \text{and} \quad (\mathbb{P}(\tau^x > T) - p) D_x \mathbb{P}(\tau^x > T) = 0
\]

then we know that \( x \notin \partial A_T^p \), thus we are in a local saddle or local maximum point.

Second, we observe that the GD scheme in (3) converges to a point, however, in general, it does not “discover” the entire border \( \partial A_T^p \). Besides, if \( A_T^p \) is the union of two (or more) disconnected regions, then the GD scheme will converge solely to one of them. The former issue can be mitigated algorithmically, by finding a way to “explore” the border defined by the condition \( \{ \mathbb{P}(\tau^x > T) = p \} \); this is discussed in the next Section. However, we cannot in general solve the latter problem, which is related to the issue of convergence to local-versus-global optima, and which is intrinsic to GD schemes.

Still, we shall shed some further light on the shape of \( A_T^p \). Let us start noticing that if \( x \in A \), then \( \mathbb{P}(\tau^x > T) \geq 0 \), therefore, for any \( p > 0, T > 0 \) a.s. \( A_T^p \neq \emptyset, A_T^p \subseteq A = A_T^0 \). However, we cannot be sure that the p-safe region \( A_T^p \) is a connected set, as we can in general argue that \( A_T^p = \bigcup_i A_{x_i} \), namely \( A_T^p \) consists of a countably infinite union of sets, wherein any \( A_{x_i} \) is a bounded connected set. Each component \( A_{x_i} \) is endowed with interesting properties.

**Definition 1**: We say that a surface (see [32] for a formal definition) is closed if it partitions the space, e.g., \( \mathbb{R}^d \), into one bounded connected region and one unbounded region. We denote this bounded region as the **interior of the surface**.

**Theorem 2 (No holes)**: Let the \( \partial A \) be a closed surface such that \( \partial A_0 \subseteq \partial A_T^p \). Then, the interior of \( \partial A_0 \) is in \( A_T^p \).

**Proof**: Let us indicate with \( A_{x_i} \) the interior of \( \partial A_{x_i} \). We prove the thesis if for any \( x \in A_{x_i}, \mathbb{P}(\tau^x > T) \geq p \)—we omit the index \( A \) in the next steps. If we define \( \theta = \) the exit time from \( A_{x_i} \), then

\[
\mathbb{P}(\tau^x > T) = \mathbb{P}(\tau^x \geq T - \theta^x \mid \theta^x \leq T) \mathbb{P}(\theta^x \leq T)
\]

\[
+ \mathbb{P}(\tau^x \geq T \mid \theta^x \geq T) \mathbb{P}(\theta^x \geq T)
\]

\[
= \mathbb{P}(\tau^x \geq T - \theta^x \mid \theta^x \leq T) \mathbb{P}(\theta^x \leq T) + \mathbb{P}(\theta^x \geq T)
\]

where, thanks to the definition of \( \theta^x, \tau^x \) we have that \( \mathbb{P}(\tau^x > T \mid \theta^x \geq T) = 1 \), indeed \( \theta^x \leq \tau^x \) a.s. since \( A_0 \subseteq A \) and by definition of exit time.

Since \( \theta^x \geq 0 \) a.s., \( \mathbb{P}(\tau^x > T - \theta^x \mid \theta^x \leq T) \geq \mathbb{P}(\tau^x > T \mid \theta^x \geq T) \) then

\[
\mathbb{P}(\tau^x \geq T \mid \theta^x \leq T) \mathbb{P}(\theta^x \leq T) + \mathbb{P}(\theta^x \geq T)
\]

\[
\geq p \mathbb{P}(\theta^x \leq T) + p \mathbb{P}(\theta^x \geq T) = p
\]

because \( \tau^x \in \partial A_T^p \).

From an algorithmic point of view, Theorem 2 is remarkable: once the algorithm has obtained a closed surface for \( \{ x \in \mathbb{P}(\tau^x \geq T) = p \} \), we know that all the points inside are in \( A_T^p \) without the need to check any further. Nevertheless, let us recall that we cannot know if this is the only part of \( A_T^p \) as there could be other bounded sets that are not connected with the one just found.

Once we have identified (part of) \( A_T^p \), an important question is how to check if a point lies inside \( A_T^p \). There are different ways to check if a point is inside a region, such as the winding number, or the Point-in-Polygon algorithm [33], [34], [35], [36], but computationally these methods are quite expensive and generalizations to large dimensions do not seem to be treated in the literature, at least from an algorithmic point of view.

Remember that to compute \( \partial A_T^p \), we use a gradient-based optimization algorithm, requiring the computation of the quantity \( D_x \mathbb{P}(\tau^x > T) \) for any point in the sequence (3). Hence, it would be useful to understand if one point is inside the safety region using the information given by \( D_x \mathbb{P}(\tau^x > T) \): this is handled by the next result.

**Proposition 1**: Let us suppose that a region \( A \in \mathbb{R}^d \) is defined by a differentiable function \( \alpha \), i.e., \( A := \{ x : \alpha(x) \leq 0 \} \) and \( \partial A := \{ x : \alpha(x) = 0 \} \). Moreover, let us suppose that \( A \) is connected. Then, a point \( x \) is inside \( A \) if

\[
x = x^* \iff \frac{D_\theta \alpha(x)}{\|D_\theta \alpha(x)\|_{x=x^*}}
\]
where \( x^* := \arg\min_{y \in \partial A} ||x - y|| \). If instead \( x = x^* + ||x - x^*|| \frac{D_\alpha(x)}{||D_\alpha(x)||}_{x=x^*} \) then \( x \) is outside.

**Proof:** Let us consider \( S = S(x, ||x - x^*||) \) the open sphere with center \( x \) and radius \( ||x - x^*|| \); we know that if \( x \) is in \( A \), then \( S \subseteq A \), vice versa \( S \subseteq \mathbb{R}^d \) if \( x \) is outside \( A \).

Note that \( x - x^* \) is perpendicular to the tangential plane to \( \alpha \) in \( x^* \), as it is also the gradient \( D_\alpha(x) \big|_{x=x^*} \), therefore,

\[
x = x^* + ||x - x^*|| \frac{D_\alpha(x)}{||D_\alpha(x)||}_{x=x^*}
\]

Since \( A \) is connected, sign \( \{\alpha(x)\} \) is the same for any \( x \in S \) and given that \( \alpha(x^*) = 0 \), the sign can be deduced by the direction of the gradient, which means that if the \( D_\alpha(x) \big|_{x=x^*} \), points to \( x \) than \( \alpha(x) > 0 \) and \( x \not\in A \), if the \( -D_\alpha(x) \big|_{x=x^*} \) points to \( x \), then \( \alpha(x) > 0 \) and \( x \in A \).

Once we have found a closed surface bounding \( A \), then thanks to Proposition 1, we know how to check if a point \( x \) is inside \( A \), by estimating the gradient in \( \partial A \), which we compute during the optimization procedure. This means that we do not have to compute \( \mathbb{P}(\tau^s \geq T) \). Unfortunately we cannot know a priori if it is outside because we do not know beforehand whether \( A^p \) is connected or not.

**Corollary 1:** Let the \( \partial A \) be a closed surface such that \( \partial A \subseteq \partial A^p \) and \( A \) its interior. Denoting by \( x^* := \arg\min_{y \in \partial A^p} ||x - y|| \), then a point \( x \) is inside \( A \), if

\[
x = x^* - ||x - x^*|| \frac{D_\alpha(x)}{||D_\alpha(x)||}_{x=x^*}
\]

**Proof:** The proof follows closely Proposition 1 considering \( A \) in place of \( A \). The difference is that the sign of the points in \( S \) is the same for the points inside, while we cannot say the same if \( x \) is outside the region \( A \). It could be that, if \( x \) is outside \( A \), but \( S \cap A_j \neq \emptyset \) for some \( j \neq i \).

### VI. Exploration of the Border \( \partial A_i^p \)

In this section, we provide an algorithmic solution to the problem discussed previously, namely how to explore points on the border characterized by

\[
\arg\min_{\tau} \frac{1}{2} (\mathbb{P}(\tau^s \geq T) - p)^2.
\]

In the following, we treat in detail the case of models with dimension \( d = 2, 3 \); if the model dimension is greater than 3, we show that the procedure can be iterated and scaled up.

![Fig. 1.]{fig1.png} Representation of how the algorithm explores the border: given the points \( \{x_m, x_{m-1}, \ldots\} \) already found by the algorithm, it proposes the red point \( x_m + \gamma \times \text{dir}\|\text{dir}\| \) as the new point of the region, and from there, it runs the GD to find the new point \( x_{m+1} \). It is possible that on the half plane where the algorithm looks for the new point does not exist a point of \( \partial A_i^p \), therefore, it is necessary to update the constraint; see Algorithm 2.

#### A. Dimension 2

Algorithm 1 explains how to proceed if \( d = 2 \). Let us discuss the main steps of Algorithm 1.

1) Line 5: If we move along the border of the region \( A_i \) just found, in a, say, clockwise manner, thanks to Theorem 2, we know that we can stop when we have found a closed point (\( x \approx x_i \)) and all the points inside are in \( A_i \). Given that the set could be the union of different disjoint sets, we should still explore the rest of the region \( A \), i.e., \( A \setminus A_i \).

2) Line 8: Defining \( x_0 = x + \gamma \times \text{dir} \), we know that

\[
\mathbb{P}(\tau^s \geq t) \approx \mathbb{P}(\tau^s \geq t) + D_\phi \mathbb{P}(\tau^s \geq t)(\gamma \times \text{dir})
\]

\[+ ||\gamma \times \text{dir}||^2 \|\text{ERR} - \text{err} + ||\gamma \times \text{dir}||^2 \|\text{ERR} \]

which means that for small \( \gamma \), we do not go far from the border \( \partial A_i^p \).

It is important to remark that \( \gamma \) and \( \lambda \) are two different parameters, which can be chosen independently, however, for more insights, see Section VII.

#### B. Dimension 3

In dimension 3, we can explore the desired border along its “sections,” as introduced next. Without loss of generality, let us suppose that the region \( A \) is the sphere of center 0 and radius 1. Let us fix the discretization parameter \( \delta > 0 \), which is related to the error we can tolerate. We can discretize the first directions \( x_1 \) to create the planes \( x_1 = \pm i \delta, i \in \mathbb{N}, 0 \leq i \leq \delta^{-1} \). The sections of the border are, therefore, the curves resulting from the intersections between the border and the considered planes.

We thus run Algorithm 1 constrained on any given plane \( x_1 = \pm i \delta \) that we are considering; see Fig. 2. Then, the same must be done for the other directions \( x_2 \) and \( x_3 \).

Note that once we select a plane (say \( \phi \), or a region), it could happen that \( \min_{x_\phi} \mathbb{P}(\tau^s \geq T) < p \), which means that there is no intersection between \( \phi \) and \( A_i^p \) and we must pass on to examining another plane (or region).

#### C. Higher Dimensions

We can apply the same reasoning on models with any dimension: namely, if we are in \( \mathbb{R}^d \), then we can partition the considered region...
A in sets of dimension $d - 1$. Continuing this procedure, we can go back recursively to the case $d = 2$. An alternative grid-free approach is to “explore the border” without constraints that are relative to some sections, i.e., to generalize directly from the case $d = 2$. Let us suppose that $x_\star$ is a point on the border; then we can compute $d - 1$ orthonormal vectors $\{g_1, \ldots, g_{d-1}\}$ to $D_x \mathbb{P}(\tau \geq T)|_{x=x_\star}$, thus running the procedure recursively from any new point $x_\star \pm \gamma g_j, j = 1, \ldots, d - 1$, until we obtain a closed surface.

VII. EXPERIMENTS

In this section, we present a case study: the code can be found at.

For the experiments, we use a simulation-based approach, i.e., we use MC techniques, and to reduce the variance, we use antithetic Brownian paths \cite{antithetic1, antithetic2}.

Remark 2: We remark that the way $\mathbb{P}(\tau \geq T)$ and $D_x \mathbb{P}(\tau \geq T)$ are computed, it is not relevant for the idea presented in this work. Indeed, it is enough to be able to compute the quantities $\mathbb{E}[\mathbb{1}_{\{\tau \geq T\}}]$ and $\mathbb{E}[\mathbb{1}_{\{\tau \geq T\}} H_T]$ and plug them into the GD procedure.

A. Complexity

Let us recall the definition of $H_T$ and $\beta_t$ from (4). Computing $H_T$ can be expensive. To estimate the expectation via MC methods, we use $N$ simulations and a time discretization step of $n^{-1}$, i.e., we split the time interval $[0, T]$ in $n$ steps. The stochastic processes to be simulated are $X_t, J_t, \beta_t, H_T$, and $\text{dist}(X_t, \partial A)$. The realization of the stochastic process $\beta$ has a total cost of $N nd^3$, where $d^3$ is the cost related to the matrix inversion $\sigma^{-1}$, plus matrix multiplications. Moreover, an optimization problem to compute $\text{dist}(X_t, \partial A)$ has to be solved $N n$ times. Nevertheless, we have to simulate $H_T, J_t$ only if $t < \tau_1$. It is important to remark that we have analyzed the computational cost of computing only one step of the GD procedure, but many are necessary to converge and explore the space.

If we are interested in a relatively low-dimensional problem, the matrix inversion can be solved analytically. A second improvement is to consider particular forms for the region $A$ that can be advantageous for

\footnote{[Online]. Available: https://github.com/FraCose/Grid-free_prob_safety.}

Fig. 2. Surface $\partial A_{1,5}^{0.5}$ (in blue) found when the region $A$ is a (black) sphere, computed sectioning the region $A$ across 2-D planes. We have discussed at the end of Section VI, the use of alternative grid-free approaches for exploration in 3-D (and higher-dimensional) cases.

Fig. 3. Plots show the points found while exploring the border of the 3-D region $A_{1,5}^{0.5}$, with respect to one plane (or section), when $A$ is a sphere. (Top) Full-view of the problem. (Middle) Same plot but zoomed, with color scale showing the safety probabilities of the points explored during exploration. It can be seen how starting from the red-cross point, with a probability approximately 0.8, we arrive at the points with the desired probability 0.5. The presence of points with different colors to the one corresponding to the desired probability 0.5 means that in those regions, the GD has explored adjacent points. (Bottom) Planar section of the space considered, fixing direction $x_3$. The black points represent those we consider being part of the border (up to an approximation error), while the circle points are those explored via the GD method. It can be seen that we explore points associated with probabilities between 0.2 and 0.8.
computing $\text{dist}(X_t, \partial A)$, e.g., a sphere, a parallelepiped or a simplex—although nonsmooth regions are not covered by the assumptions of this work.

B. Bias

It is important to remark that the steps done by the GD algorithm are stochastic and biased. Indeed, we do not compute the exact probability $P(\tau^* \geq T)$, but we discretize the time, therefore, computing $P(\tau^*_n \geq T)$; recall that in [39], it is shown that $|P(\tau^* \geq T) - P(\tau^*_n \geq T)| \leq O(n^{-1/2})$, where $\tau^*_n$ represents the discrete stopping time of the Euler scheme associated with (1). Moreover, the algorithm is stochastic since we approximate $P(\tau^*_n \geq T)$ using MC techniques. Therefore, we have to consider that [39]

$$\left| P(\tau^* \geq T) - P(\tau^*_n \geq T) \right| \leq O(n^{-1/2}) + O\left(\frac{1}{\sqrt{n}}\right) Z$$

where the hat denotes an MC estimator of the quantity of interest and $Z$ represents a standard normal random variable. A similar error bound might be derived for the other term $|D_x P(\tau^* \geq T) - D_x P(\tau^*_n \geq T)|$ [39], [40], [41], though an adaptation is needed due to the presence of $\tau^*_n$ in the definition of $H$ in Theorem 1. Due to these biases, we have noticed that reducing the variance helps the GD to converge better: for instance, when the (norm of the) gradient $D_x P(\tau^* \geq T)$ becomes small, the error could dominate and the GD step could not work properly.

C. Hyperparameters

The hyperparameters to be chosen for the procedure are the following:

1) $n$, time discretization step—in principle, the higher the better, but $n$ has a big impact on the computational time, since it cannot be parallelized.

2) $N$, Monte Carlo simulations—increasing $N$ reduces the variance of the MC methods. $N$ has a relatively low impact since the number of samples can be parallelized.

3) $\lambda$, the “learning rate” of the GD procedure in (3)---$\lambda$ must be chosen carefully. While we are doing the first minimization, i.e., while we are searching for a first point on $\partial A^p_T$ (exploration phase), $\lambda$ can be quite high (more than $1 \times 10^{-3}$) [24]. Instead, if we are considering the minimization procedure in Algorithm 1, since we should be almost close to the border we should select a small $\lambda$.

4) $\gamma$, the “border exploration” parameter in Algorithm 1—$\gamma$ indicates how fine-grained we wish the approximation of $\partial A^p_T$ to be.

D. Acceleration of the exploration

Algorithm 1 is a good starting point to explore the border $\partial A^p_T$, but care must be taken. For the following discussions, we consider the problem to be in a 2-D space.

First, we would like to explore with an orientation, e.g., clockwise, such that we do not go back to a region already explored. To solve this problem, we check if there are already points computed in the direction we are going to explore. However, selecting an “optimal” number of points is an open question that depends on the curvature of $\partial A^p_T$, which a priori is unknown. Another heuristic is to constrain the algorithm to search the new point on $\partial A^p_T$ in a given region; see Fig. 1. A more sophisticated alternative is to split the region into subspaces and search the border $\partial A^p_T$ locally. This technique would also increase the level of parallelization [42].

Second, going toward the direction that is perpendicular to the gradient, as per Section VI, is only an approximation and sometimes, depending on the local curvature, can be quite imprecise. To improve this approximation, we have considered the following procedure. Let us imagine that we have computed a certain number of points on $\partial A^p_T$, in order $\{x_1, x_2, \ldots, x_m\}$. We can compute the parabola equation (since the plane is fixed) that approximates the points $\{x_1, x_2, \ldots, x_m\}$, and use this equation as a second possible approximation. This can be thought as an approximation of the second-order information of the curve $P(\tau^*_n \geq T) - p = 0$ in $x$. Later, we can choose the new direction as a weighted average of the perpendicular to $D_x P(\tau^*_n \geq T)$ and the value of the approximated parabola $p(x)$ in $x = 2x_m - x_{m-1}$; see Fig. 1 and the code for more insight. Experimentally, this procedure accelerates the exploration, since it reduces the approximation error relative to the gradient.

Finally, it is better to constraint the space where the algorithm searches for the next point of the border. In Fig. 1, it is shown how we proceed. Once one point $x_m$ on the border is found, i.e., $P(\tau^*_n \geq T) \approx p$, we compute the gradient ($D$) and the direction to follow (dir). Given dir and $\gamma$, it is possible to search the new point only in the part of the plane where there are not “recent points” considering the line perpendicular to the direction passing through the point guess $x_m + \gamma \times \text{dir}[(\text{dir})]$. It is possible that the constraint does not allow the optimization procedure to find a point $P(\tau^*_n \geq T) \approx p$, therefore, if the solution of the GD returns, after a certain number of iterations, a point $x$, s.t. $P(\tau^*_n \geq T) \neq p$, then it is important to update the direction dir and the corresponding constraint. The candidate we have chosen for the updated direction is $2x_m - x_{m-1}$, up to some rescaling, but other choices are available. For example, we have experimented that selecting $2x_m - x_{m}$ accelerates the procedure over the choice $x_m - x_{m-1}$. Moreover, it is necessary to reduce the step exploration $\gamma$, such that we get closer to the point $x_m$, and by continuity of $\partial A^p_T$, we will find the point sooner or later. In Algorithm 2, we present a pseudocode of the procedure.

E. Case study

The model considered for the experiment is

\[
\begin{align*}
\frac{dX_t^{(1)}}{dt} &= \left(\frac{1}{2} X_t^{(1)} + \frac{1}{3} X_t^{(2)}\right) dt \\
\frac{dX_t^{(2)}}{dt} &= \left(\frac{1}{2} X_t^{(1)} + \frac{1}{3} X_t^{(2)}\right) dt \\
\frac{dX_t^{(3)}}{dt} &= \left(\frac{1}{3} X_t^{(1)} + \frac{1}{2} X_t^{(2)}\right) dt + \frac{1}{3} \omega_1, \omega_2, \omega_1, \omega_2, \omega_1 \right) dW_t \\
\omega_1 &= 2\sqrt{1 - \rho + \sqrt{1 + 2\rho}} \\
\omega_2 &= -\sqrt{1 - \rho + \sqrt{1 + 2\rho}}.
\end{align*}
\]

Possibly as a weighted average of the gradient and some local approximation of the curvature as explained before.
If we define $dW_t = \sigma dW_t$, where $\sigma$ is the diffusion matrix in (5), then we have that $\text{Corr}(dW_t^{(i)}, dW_t^{(j)}) = \rho_{i,j} \neq 0$ and $i, j \in \{1, 2, 3\}$. In the experiment, we have used $\rho = 0.5$. For the region $A$, we have considered a sphere with center at the origin and radius equal to 100.

We consider the problem of computing $A_{\rho}^{\lambda}(x) = \{ x \in A : \mathbb{P}(\tau^x \geq 1) \geq 0.5 \}$. We start at a point $x_0$ where $\mathbb{P}(\tau_{x_0} \geq 1) \neq 0.5$, then we minimize $\frac{1}{2}(\mathbb{P}(\tau^x \geq 1) - 0.5)^2$ until we obtain a point $x_\ast$, s.t. $\mathbb{P}(\tau^{x_\ast} \geq 1) - 0.5 < \varepsilon$—in this case, err represents the approximation errors due to the computation of $\mathbb{P}(\tau^x \geq 1)$. From $x_\ast$, we fix $x^{(3)}$ and start Algorithm 1, i.e., we fix the plane $x^{(3)} = x^{(3)}$; see Fig. 3 for the results of the experiments. See Fig. 2 for a 3-D representation of $\partial A_{\rho}^{\lambda}$. Instead of using plain GD, we use ADAM [24], a version of GD with momentum and adaptive learning rate that has shown impressive performance in machine learning and it is considered the state-of-the-art optimization tool. In particular, we prefer to include momentum, because we do not know whether $\frac{1}{2}(\mathbb{P}(\tau^x \geq 1) - 0.5)^2$ is convex as a function of $x$.

The hyperparameters chosen are $N = 10000$, $n = 200$, maximum iteration of the GD (any time we use it) = 50, $\lambda = 5 \times 10^{-2}$, $\gamma = 1.5$. With reference to the previous discussion on the approximation of the second-order information of $\partial A_{\rho}^{\lambda}$, in order to compute the new direction, i.e., dir in Fig. 1, in addition to the gradient information, we use also the parabola approximating the previous four points found on the border.

VIII. CONCLUSION

We have presented a new approach to find and compute probabilistic safety regions for SDEs without resorting to the discretization of their state space, which is by and large the standard approach in literature. This is done by formulating an optimization problem: to solve this, we have borrowed techniques and ideas from Malliavin calculus and mathematical finance.

We have elaborated two formal results that allow one to explore relevant parts of the regions of interest, thus focusing computational load related to probabilistic safety computation for continuous-space models, such as SDEs. We have discussed possible algorithmic issues related to the procedure, and offered strategies to cope with them.

We conclude suggesting that more work on the generalisation of the approach on high-dimensional models in a completely automatic fashion is a goal worth pursuing.

ACKNOWLEDGMENT

The authors want to thank The Alan Turing Institute and the University of Oxford for the financial support given.

REFERENCES

[1] A. Abate, S. Amin, M. Prandini, J. Lygeros, and S. Sastry, “Computational approaches to reachability analysis of stochastic hybrid systems,” in Proc. Hybrid Syst. Comput. Control, 2007, pp. 4–17.
[2] A. Abate, M. Prandini, J. Lygeros, and S. Sastry, “Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems,” Automatica, vol. 44, no. 11, pp. 2724–2734, Nov. 2008.
[3] F. Blanchini and S. Miani, Set-Theoretic Methods in Control, 1st ed. Cambridge, MA, USA: Birkhäuser, 2007.
[4] R. Wisniewski and M. L. Bujorianu, “Stochastic safety analysis of stochastic hybrid systems,” in Proc. IEEE 56th Ann. Conf. Decis. Control, Melbourne, Australia, Dec. 2017, pp. 2390–2395.
[5] P. Malliavin and A. Thalmaier, Stochastic Calculus of Variations in Mathematical Finance. Berlin, Germany: Springer-Verlag, 2006.
[6] J. C. Hull, Options, Futures, and Other Derivatives, Global Edition. London, U.K.: Pearson, 2017.
[7] A. Lavaei, S. Soudjani, A. Abate, and M. Zamani, “Automated verification and synthesis of stochastic hybrid systems: A survey,” Automatica, vol. 146, 2022, Art. no. 110617. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0001895222004794.
[8] J. Hu, J. Lygeros, and S. Sastry, “Towards a theory of stochastic hybrid systems,” in Proc. 3rd Int. Workshop Hybrid Syst., Comput. Control, Pittsburgh, PA, USA, 2000, pp. 160–173.
[9] M. Bujorianu and J. Lygeros, “Toward a general theory of stochastic hybrid systems,” in Stochastic Hybrid Syst., H. Blom and J. Lygeros, Eds., vol. 337. Berlin, Germany: Springer, 2006.
[10] M. H. A. Davis, Markov Models and Optimization (Monographs on Statistics and Applied Probability), vol. 49. London, U.K.: Chapman & Hall, 1993.
[11] M. L. Bujorianu, “Extended stochastic hybrid systems and their reachability problem,” in Proc. 7th Int. Workshop Hybrid Syst., Comput. Control, Philadelphia, PA, USA, 2004, pp. 234–249.
[12] X. D. Koutsoukos and D. Riley, “Computational methods for reachability analysis of stochastic hybrid systems,” in Proc. Int. Workshop Hybrid Syst., Comput. Control, Santa Barbara, CA, USA, 2006, pp. 377–391.
[13] D. Riley, X. D. Koutsoukos, and K. Riley, “Reachability analysis for stochastic hybrid systems using multilevel splitting,” in Proc. 12th Int. Conf. Hybrid Syst. Comput. Control, San Francisco, CA, USA, 2009, pp. 436–446.
[14] P. Mohajerin Esfahani, D. Chatterjee, and J. Lygeros, “The stochastic reach-avoid problem and set characterization for diffusions,” Automatica, vol. 70, pp. 43–56, 2016.
[15] R. Wisniewski, M. Svenstrup, A. Pedersen, and C. Steiniche, “Certificate for safe emergency shutdown of wind turbines,” in Proc. Automat. Control Conf., 2013, pp. 3667–3672.
[16] R. Wisniewski, M. L. Bujorianu, and C. Sloth, “Safe and slow-sensitive analysis of stochastic hybrid processes,” IEEE Trans. Autom. Control, vol. 65, no. 12, pp. 5220–5235, Dec. 2020.
[17] R. Wisniewski and C. Sloth, “Safety analysis of stochastic dynamical systems,” in Proc. IFAC Conf. Anal. Des. Hybrid Syst., 2015, pp. 62–67.
[18] M. L. Bujorianu and R. Wisniewski, “New insights on p-safety of stochastic hybrid systems,” in Proc. IEEE 58th Conf. Decis. Control, Nice, France, 2019, pp. 4433–4438.
[19] J. Hu and M. Prandini, “Aircraft conflict detection: A method for computing the probability of collision based on Markov chain approximation,” in Proc. 7th Eur. Control Conf., Cambridge, U.K., 2003, pp. 2003–2225.
[20] M. Prandini, J. Hu, J. Lygeros, and S. Sastry, “A probabilistic approach to aircraft conflict detection,” IEEE Trans. Intell. Transp. Syst., vol. 1, no. 4, pp. 199–220, Dec. 2000.
[21] M. Zamani, P. M. Esfahani, R. Majumdar, A. Abate, and J. Lygeros, “Symbolic control of stochastic systems via approximately bisimilar finite abstractions,” IEEE Trans. Automat. Control, vol. 59, no. 12, pp. 3135–3150, Dec. 2014.
[22] M. Zamani, P. M. Esfahani, A. Abate, and J. Lygeros, “Symbolic models for stochastic control systems without stability assumptions,” in Proc. 12th Eur. Control Conf., Zurich, Switzerland, 2013, pp. 4213–4227.
[23] N. Ikeda and S. Watanabe, Stochastic Differential Equations and Diffusion Processes. Amsterdam, Netherlands: Elsevier, 2014.
[24] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.
[25] E. Fournier, J.-M. Lasry, J. Lebouchou, P.-L. Lions, and N. Touzi, “Applications of malliavin calculus to monte carlo methods in finance,” Finance Stochastics, vol. 3, no. 4, pp. 391–412, 1999.
[26] E. Fournier, J.-M. Lasry, J. Lebouchou, and P.-L. Lions, “Applications of Malliavin calculus to monte-carlo methods in finance,” II. “Finance Stochastics,” vol. 5, no. 2, pp. 201–236, 2001.
[27] N. Privault and X. Wei, “A Malliavin calculus approach to sensitivity analysis in insurance,” Insuranc, Math. Econ., vol. 35, no. 3, pp. 679–690, 2004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167668704000964.
[28] B. Vafadary, G. Pages, and J. Primet, “A quantization tree method for pricing and hedging multidimensional american options,” Math. Finance, vol. 15, no. 1, pp. 119–168, Jan. 2005.
[29] E. Gobet and R. Munos, “Sensitivity analysis using FT\text{-}Malliavin calculus and martingales, and application to stochastic optimal control,” SIAM J. Control. Optim., vol. 43, pp. 1676–1713, 2005.
[30] E. Gobet and A. Kohatsu-Higa, “Computation of Greeks for barrier and look-back options using Malliavin calculus,” Electron. Commun. Probab., vol. 8, pp. 51–62, 2003.
[31] H. Kunita, “Stochastic differential equations and stochastic flows of diffeomorphisms,” in Proc. Ecole d’Ete de Probabilites de Saint-Flour, 1984, pp. 143–303.

[32] L. C. Kinsey, Topology of Surfaces. New York, NY, USA: Springer, 1993.

[33] E. Haines, “Point in polygon strategies,” in Graphics Gems. Amsterdam, The Netherlands: Elsevier, 1994, pp. 24–46.

[34] K. Hormann and A. Agathos, “The point in polygon problem for arbitrary polygons,” Comput. Geometry, vol. 20, no. 3, pp. 131–144, Nov. 2001.

[35] G. N. Kumar and M. Bangi, “An extension to winding number and point-in-polygon algorithm,” IFAC-PapersOnLine, vol. 51, no. 1, pp. 548–553, 2018.

[36] C.-W. Huang and T.-Y. Shih, “On the complexity of point-in-polygon algorithms,” Comput. Geosci., vol. 23, no. 1, pp. 109–118, Feb. 1997.

[37] P. Glasserman, Monte Carlo Methods in Financial Engineering. New York, NY, USA: Springer, 2003.

[38] H. T. Huynh, V. S. Lai, and I. Soumaré, “Solution of stochastic differential equations,” in Stochastic Simulation and Applications in Finance With MATLAB Programs. Hoboken, NJ, USA: Wiley, 2012, pp. 123–148.

[39] E. Gobet, “Weak approximation of killed diffusion using euler schemes,” Stochastic Processes Appl., vol. 87, no. 2, pp. 167–197, 2000.

[40] E. Gobet and S. Menozzi, “Discrete sampling of functionals of ITO processes,” in Lecture Notes in Mathematics. Berlin, Germany: Springer, 2007, pp. 355–374.

[41] E. Gobet and S. Menozzi, “Stopped diffusion processes: Boundary corrections and overshoot,” Stochastic Processes Appl., vol. 120, no. 2, pp. 130–162, Feb. 2010.

[42] K. G. Suffern, “Quadtree algorithms for contouring functions of two variables,” Comput. J., vol. 33, no. 5, pp. 402–407, May 1990.