TU Delft

Delft University of Technology

Document Version
Final published version

Licence
Dutch Copyright Act (Article 25fa)

Citation (APA)
Mazouz, R., Mathiesen, F. B., Laurenti, L., & Lahijanian, M. (2026). Piecewise Stochastic Barrier Functions. *Automatica*, *189*, Article 113020. https://doi.org/10.1016/j.automatica.2026.113020

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
In case the licence states “Dutch Copyright Act (Article 25fa)”, this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Automatica 189 (2026) 113020

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Piecewise Stochastic Barrier Functions ☆

Rayan Mazouz [a,*,1], Frederik Baymler Mathiesen [b,1], Luca Laurenti [b,c], Morteza Lahijanian [a]

[a] Department of Aerospace Engineering Sciences, University of Colorado Boulder, USA
[b] Delft Center for Systems and Control, Delft University of Technology, The Netherlands
[c] The Italian Institute of Artificial Intelligence for Industry (AI4I), Italy

ARTICLE INFO

Article history:
Received 25 April 2024
Received in revised form 24 October 2025
Accepted 12 March 2026

Keywords:
Barrier certificates
Stochastic systems
Probabilistic safety
Convex optimization
Formal verification

ABSTRACT

Stochastic barrier functions (SBFs) are Lyapunov-like functions that enable safety analysis of stochastic systems. Finding a valid SBF, however, is challenging, as it requires solving a complex functional optimization problem. Existing convex approaches are often limited to low-dimensional systems with simple (polynomial) dynamics, while non-convex approaches lack completeness guarantees. To address these challenges, this paper presents a novel SBF synthesis framework based on piecewise (PW) functions We first outline a general formulation of PW-SBFs. Then, we focus on PW-Constant (PWC) SBFs and show how their simplicity yields computational advantages for general stochastic systems. Specifically, we prove that synthesis of PWC-SBFs reduces to a minimax optimization problem. We then introduce three efficient algorithms to solve this problem, each offering distinct trade-offs, all with completeness guarantees. The first algorithm is based on dual linear programming (LP), which provides an exact solution to the minimax optimization problem. The second is a more scalable algorithm based on counter-example guided inductive synthesis, which involves solving two smaller LPs. The third algorithm solves the minimax problem using gradient descent, which admits even better scalability. We provide an extensive evaluation of these methods on various case studies, including neural network dynamic models, nonlinear switched systems, and high-dimensional linear systems. Our benchmarks demonstrate that PWC-SBFs outperform state-of-the-art methods, namely sum-of-squares and neural barrier functions, and can scale to eight dimensional systems.

© 2026 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

1. Introduction

Autonomous systems deployed in *safety-critical* domains demand thorough verification to ensure reliability. Examples of such systems include autonomous vehicles in human-shared environments (Shalev-Shwartz et al., 2017), aerospace systems (from aerial transportation Lee et al., 2016 to space exploration Maluf et al., 2005), and surgical robotics (Guiochet et al., 2017). However, this verification process faces major challenges arising from the complexities in dynamics as well as uncertainties due to the physics or randomized algorithms of the system. A widely employed method to provide necessary safety guarantees is *stochastic barrier functions* (SBFs) (Kushner, 1967; Prajna et al., 2007; Santoyo et al., 2021). SBFs provide a lower bound on safety probabilities, hence enabling *safety certification*. Despite extensive studies on SBFs, existing methods either lack completeness (in the case of non-convex approaches) or are largely restricted to simplistic settings (in the case of convex methods), namely, polynomial dynamics and low dimensions. This work aims to address these limitations by developing a theoretical and computational framework for generating barrier certificate tailored to complex systems.

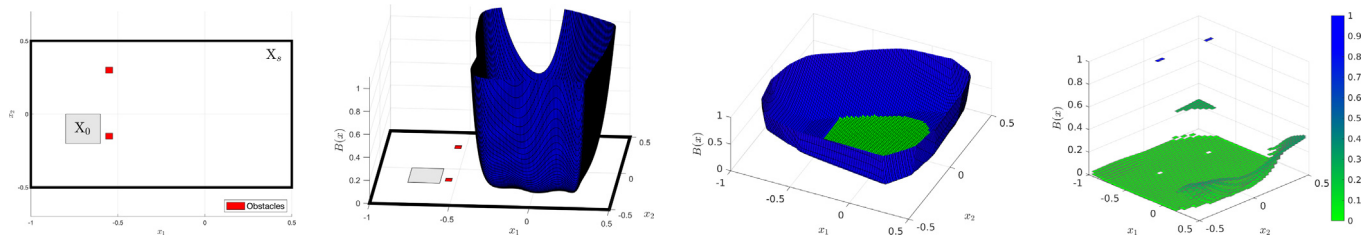
Synthesis of SBFs, similar to Lyapunov functions, reduces to a functional optimization problem: the objective is to minimize the rate of growth of the function along system trajectories (in expectation), as well as its value over the initial states, while ensuring that the function remains above a threshold over the unsafe states. The literature presents two prominent methods for synthesizing SBFs: SOS optimization (Mazouz et al., 2022; Santoyo et al., 2021) and NBF (Dawson et al., 2022; Mathiesen et al., 2022). SOS optimization relies on a predefined SOS polynomial template for the SBF and aims to determine its coefficients using convex optimization. While powerful, this method suffers when

☆ This work was supported in part by National Science Foundation (NSF) grant 2039062 and Air Force Research Laboratory/RVSW under Contract No. FA945324CX026. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Dimitra Panagou under the direction of Editor Christos G. Cassandras.

* Corresponding author.
E-mail addresses: rayan.mazouz@colorado.edu (R. Mazouz), f.b.mathiesen@tudelft.nl, frederik@baymler.com (F.B. Mathiesen), l.laurenti@tudelft.nl (L. Laurenti), morteza.lahijanian@colorado.edu (M. Lahijanian).

[1] Rayan Mazouz and Frederik Baymler Mathiesen contributed equally to this work.

https://doi.org/10.1016/j.automatica.2026.113020
0005-1098/© 2026 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.



(a) Initial and safe sets (b) Degree-30 SOS SBF (c) NBF (d) PWC SBF

Fig. 1. Example of a 2D stochastic system with a non-convex safe set and dynamics $\mathbf{x}_{k+1} = 0.5\mathbf{x}_k + \mathbf{v}$, where noise $\mathbf{v}$ has a Gaussian distribution. Fig. 1(a) shows the initial and safe sets. The SOS barrier of degree 30, the NBF, and the PWC-SBF for this system are shown in Figs. 1(b)–1(d), respectively. The obtained safety probability for 10 time steps and computation time for SOS are $P_{\text{SOS}} = 0.075$ and $\tau_{\text{SOS}} = 197$s, for NBF are $P_{\text{NBF}} = 0.93$ and $\tau_{\text{NBF}} = 3600$s, and for PWC-SBF are $P_{\text{PWC}} = 0.93$ and $\tau_{\text{PWC}} = 69$s.

the system dimensionality is large (Ahmadi & Majumdar, 2014) or the safe set is non-convex. On the other hand, NBF employ neural networks (NNs) to act as barrier functions, addressing the conservatism of SOS in non-convex settings. Yet, NBF face scalability issues due to the need to verify the NN against the SBF conditions, which is challenging for high-dimensional systems.

To illustrate these challenges, consider the 2-dimensional linear system with additive Gaussian noise shown in Fig. 1. The initial and safe sets are depicted in Fig. 1(a). Note that the safe set is non-convex. Figs. 1(b) and 1(c) show the synthesized SOS SBF of degree 30 and NBF, utilizing the state-of-the-art techniques presented in Mazouz et al. (2022) and Mathiesen et al. (2022), respectively. Generally, a higher value of the SBF (especially, over the initial set) indicates a lower safety probability bound for the system. The corresponding lower bound on the probability of remaining in the safe set for 10 time steps is 0.075 for the SOS SBF and 0.93 for the NBF, with computation times of 197 s and 3600 s, respectively. Notice that NBF outperforms SOS by more than 10-fold in safety probability. However, it comes at the cost of an order of magnitude (nearly 20×) increase in the computation time for this 2-dimensional system. In fact, for NBF, the computation time required to guarantee that a trained NN is an SBF quickly explodes as dimensionality increases.

In this manuscript, we introduce an SBF framework based on *piecewise* (PW) functions. We first show a general formulation for PW-SBFs. Then, we focus on PW-Constant (PWC) SBFs, whose simplicity gives rise to an elegant formulation applicable to *general* stochastic systems. Our key insight is that the challenging operations of expectation and function composition required by the martingale condition of SBFs can be significantly mitigated by using constant functions, enabling both simplicity and scalability. Guided by this insight, we prove that the PWC-SBF synthesis problem reduces to a minimax optimization problem, for which we introduce three efficient and complete computational methods. First, we show that, even for nonlinear stochastic systems, the minimax optimization problem can be reduced to a linear program (LP) via a dual formulation. Second, to enable scalability, we introduce an iterative algorithm that requires solving two smaller LPs based on the *counter-example guided inductive synthesis* (LP-CEGIS) principle. However, the LP-CEGIS approach can be memory-intensive. Third, we show that the minimax problem can be solved using a Gradient Descent (GD) algorithm, which admits even better scalability and resolves the memory issue of LP-CEGIS. It however poses a challenge in designing a convergence criterion. The power of PWC-SBFs is illustrated in Fig. 1(d), where the same safety probability bound of 0.93 as NBF is achieved with a computation time of 69 s, an order of magnitude (50×) faster. Our extensive evaluations demonstrate the efficacy of the proposed methods, outperforming state of the art and scaling to eight dimensional systems.

In summary, the key contributions of this work are:

- A general formulation for PW-SBFs with a particular focus on PWC-SBFs, providing a simple formulation for safety verification of general stochastic systems.
- A derivation of the PWC-SBF synthesis problem as a minimax convex optimization problem.
- Three efficient and scalable computational methods for PWC-SBF synthesis:
  - an LP duality-based approach with a proof of zero-duality gap, i.e., obtaining an exact solution to the minimax problem,
  - an LP-CEGIS algorithm with a proof of convergence to the optimal solution in finite time, and
  - a GD algorithm that offers significant reduction in computation time.
- Extensive case studies to illustrate the performance of the proposed methods against state-of-the-art methods of SOS and NBF, and an analysis hereof.

Overall, this work offers a significant advancement in SBF synthesis for safety-critical systems, providing both theoretical insights and computational methods that enhance scalability in complex environments.

1.1. Related work

Barrier functions, also known as barrier certificates (Ames et al., 2014, 2016; Prajna & Jadbabaie, 2004), offer a formal approach to ensure safety of dynamical systems, particularly for nonlinear and hybrid models (Prajna, 2006; Prajna & Jadbabaie, 2004). Barrier functions have been developed both for deterministic (Ames et al., 2019; Prajna & Rantzer, 2005) and stochastic systems (Kushner, 1967; Prajna et al., 2007). Verification of discrete-time systems using SBFs was initially introduced in Kushner (1967) by relying on results from martingale theory, where the conditions for a function to be a SBF have been first established. The use of SBFs for continuous-time hybrid systems with discrete jumps was introduced in Prajna et al. (2007).

Finding a SBF for a given system is a problem that has recently received increased attention, with approaches including SOS optimization (Mazouz et al., 2022; Santoyo et al., 2021), neural networks (Mathiesen et al., 2022), and CEGIS (Ding et al., 2022; Jagtap et al., 2020; Peruffo et al., 2021). Approaches based on SOS optimization (Santoyo et al., 2021) typically encounter significant challenges related to scalability and the representation of non-convex safe sets, and are generally restricted to systems with polynomial dynamics. Work (Mazouz et al., 2022) extends the SOS approach to general dynamics by building upon techniques from Zhang et al. (2018) to construct affine bounds on the dynamics. Nevertheless, scalability continues to pose a substantial challenge.

To address the limitations of SOS approaches, neural barrier functions (NBF) (Dawson et al., 2022; Mathiesen et al., 2022) have recently been proposed. NBF use the representational power of neural networks (NNs) to train a highly amenable SBF candidate, which then needs to be verified to establish whether it is a proper certificate. Work (Žikelić et al., 2023) combines NBF with reinforcement learning to train a deep policy based on a novel formulation of reach-avoid martingales. A major limitation of these methods is their reliance on NN verification, which also suffer from scalability.

In Jagtap et al. (2020), the authors develop an alternative (non-convex) approach for SBF synthesis based on a popular technique in formal methods called CEGIS (Alur et al., 2013). Specifically, Jagtap et al. (2020) parameterizes the SBF using (monomial) basis functions and synthesizes the parameters from finite samples of the state space. The resulting function serves as a candidate SBF, which must then be validated against the SBF conditions over the entire state space. Validation is performed by searching for a counterexample that violates the conditions, typically using either SMT solvers or nonlinear optimization. However, these approaches are incomplete and nonlinear solvers are prone to local minima. In addition, those (non-convex) CEGIS methods suffer from scalability challenges (Barbosa et al., 2019).

This paper aims to address the limitations of the SOS, NBF, and non-convex CEGIS approaches in terms of both conservatism and scalability. Our work mainly builds on the results of Laurenti and Lahijanian (2023), which establishes the connection between SBFs and dynamic programming. With this insight, we propose a novel SBF synthesis method based on PW functions. In contrast to Laurenti and Lahijanian (2023), this paper introduces the new class of PW-SBFs along with tailored synthesis methods, and provides a detailed comparison against state-of-the-art SBF synthesis approaches. Moreover, unlike the non-convex CEGIS approach in Jagtap et al. (2020), all methods proposed in this paper are convex and thus complete and efficient, including our LP-CEGIS method.

Other related works include (Anand et al., 2022; Edwards et al., 2024; Nejati & Zamani, 2023; Prajna & Jadbabaie, 2004). Specifically, Nejati and Zamani (2023) proposes a data-driven framework for synthesis of safe controllers using multiple control barrier certificates. The dynamics of the system are, however, assumed to be deterministic. In Prajna and Jadbabaie (2004), an SBF approach is presented for the safety verification of (non)-deterministic hybrid systems with uncertain disturbance inputs. The certificate is constructed from a collection of barriers, where each function corresponds to a discrete mode of the system. In Edwards et al. (2024), a tool was presented for the formal verification and control synthesis of deterministic discrete- and continuous-time systems. In Anand et al. (2022), a method is presented for verification of interconnected stochastic systems, based on small-gain theory. However, none of these works address the synthesis of PW-SBFs for general discrete-time stochastic systems, as we propose in this paper.

1.2. Basic notations

The basic notations that we use in the paper are as follows. We denote the set of natural, real, and non-negative real numbers as $\mathbb{N}$, $\mathbb{R}$, and $\mathbb{R}_{\geq 0}$, respectively. Random variables are denoted using bold symbols, e.g., $\mathbf{x} \in \mathbb{R}^n$ denotes a random variable taking values in n-dimensional Euclidean space, whereas $x \in \mathbb{R}^n$ represents a point in the same space.

We denote Pr as a probability measure, and P_s as the probability of safety. A probabilistic transition from a compact set $X_i \subset \mathbb{R}^n$ to a compact (measurable) set $X_j \subset \mathbb{R}^n$ is denoted by $p_{ij} \in [0, 1]$, and its lower and upper bounds by $\underline{p}_{ij}, \overline{p}_{ij} \in [0, 1]$, respectively, i.e., $0 \leq \underline{p}_{ij} \leq p_{ij} \leq \overline{p}_{ij} \leq 1$. For $K \in \mathbb{N}$ compact sets $X_1, X_2, \ldots, X_K \subset \mathbb{R}^n$, a feasible transition probability distribution from X_i is denoted by $p_i = (p_{i1}, \ldots, p_{iK})$ such that $p_{ij} \in [\underline{p}_{ij}, \overline{p}_{ij}]$ for all $j \in \{1, \ldots, K\}$ and $\sum_{j=1}^{K} p_{ij} = 1$. The set of all feasible transition probabilities from X_i is denoted by $\mathcal{P}_i$.

2. Problem formulation

We consider a stochastic process described by the following stochastic difference equation

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{v}_k), \tag{1}$$

where state $\mathbf{x}_k$ takes values in $\mathbb{R}^n$, and noise $\mathbf{v}_k$ is an independent and identically distributed random variable taking values in $\mathbb{R}^v$ with associated probability distribution $p_{\mathbf{v}}$. Further, $f : \mathbb{R}^n \times \mathbb{R}^v \to \mathbb{R}^n$ is the vector field representing the one step dynamics of system (1). We assume that f is almost everywhere continuous. Intuitively, $\mathbf{x}_k$ is a general model of a stochastic process with possibly nonlinear dynamics and non-additive noise.

To define a probability measure for system (1), for a measurable set X, we define the one-step stochastic kernel

$$T(X \mid x) := \int_{\mathbb{R}^v} \mathbf{1}_X(f(x, v)) p_{\mathbf{v}}(dv), \tag{2}$$

where $\mathbf{1}_X$ is the indicator function for set X, defined as

$$\mathbf{1}_X(x) = \begin{cases} 1 & \text{if } x \in X \\ 0 & \text{otherwise.} \end{cases}$$

It follows that for an initial condition $x_0 \in \mathbb{R}^n$, T induces a unique probability measure $\Pr^{x_0}$ (Bertsekas & Shreve, 2004), such that for measurable sets $X_0, X_k \subseteq X$ where $k \in \mathbb{N}$, it holds that

$$\Pr^{x_0}[\mathbf{x}_0 \in X_0] = \mathbf{1}_{X_0}(x_0),$$
$$\Pr^{x_0}[\mathbf{x}_k \in X_k \mid \mathbf{x}_{k-1} = x] = T(X_k \mid x).$$

The definition of $\Pr^{x_0}$ allows one to make probabilistic statements over the trajectories of system (1). This work seeks to quantify *probabilistic safety*, which together with its dual, *probabilistic reachability* (Laurenti & Lahijanian, 2023), are commonly used to quantify safety for stochastic dynamical systems. Work (Abate et al., 2008) presents a generalization of invariance.

Definition 1 (*Probabilistic Safety*)**.** Let $X_s \subset \mathbb{R}^n$ be a compact set representing the safe set, $X_0 \subseteq X_s$ be the initial set, and $N \in \mathbb{N}$ be the time horizon. Then, *probabilistic safety* is defined as

$$P_s(X_s, X_0, N) = \inf_{x_0 \in X_0} \Pr^{x_0}[\forall k \in \{0, \ldots, N\}, \mathbf{x}_k \in X_s].$$

The problem we consider in this work is as follows.

Problem 1 (*Safety Certificate*)**.** Consider a compact safe set $X_s \subset \mathbb{R}^n$ and an initial set $X_0 \subseteq X_s$. Then, for a given threshold $\delta_s \in [0, 1]$, certify whether, starting from X_0, system (1) remains in X_s for N time steps with at least probability δ_s, i.e.,

$$P_s(X_s, X_0, N) \geq \delta_s.$$

Problem 1 seeks to compute the probability that $\mathbf{x}_k$ remains within a given safe set, e.g., it avoids obstacles and undesirable states. Computation of this probability is particularly challenging because system (1) is stochastic and f can be a nonlinear (and non-polynomial) function. Further, since set X_s can be non-convex, the problem is generally non-convex. This implies that convex optimization tools cannot be used out-of-the-box for this problem in an efficient way.

Approach overview. Our approach to Problem 1 is based on SBFs, which are value functions to guarantee a lower bound on the probability of safety in the sense of Problem 1. We review the theory of SBFs in Section 3. To address the inefficiencies of existing synthesis techniques for continuous barrier functions due to, e.g., non-convexity of X_s, in Section 4, we focus on *piecewise* functions. We show that, based on such functions, barrier certificates can be provided for the safety of system (1) by establishing a lower bound on P_s. Further, we show in Section 5 that by choosing a piecewise constant function, the synthesis problem reduces to an LP, for which we introduce three efficient computational frameworks.

3. Stochastic barrier functions

In this section, we provide an overview of stochastic barrier functions.

Consider system (1) with safe set $X_s \subset \mathbb{R}^n$, initial set $X_0 \subseteq X_s$, and unsafe set $X_u = \mathbb{R}^n \setminus X_s$. Then, a function $B : \mathbb{R}^n \to \mathbb{R}$ is a *stochastic barrier function* (SBF) if there exist scalars $\eta, \beta \geq 0$ such that

$$B(x) \geq 0 \qquad \forall x \in \mathbb{R}^n, \tag{3a}$$

$$B(x) \geq 1 \qquad \forall x \in X_u, \tag{3b}$$

$$B(x) \leq \eta \qquad \forall x \in X_0, \tag{3c}$$

$$\mathbb{E}[B(f(x, \mathbf{v})) \mid x] \leq B(x) + \beta \qquad \forall x \in X_s, \tag{3d}$$

where $\mathbf{v} \in \mathbb{R}^n$ is a random variable with density $p_{\mathbf{v}}$. Given an SBF B, then a lower bound on the probability of safety for system (1) can be obtained according to the following proposition.

Theorem 1 (*Laurenti and Lahijanian (2023)*)**.** *If there exists function B that satisfies conditions (3a)–(3d), then for a horizon $N \in \mathbb{N}$, it follows that*

$$P_s(X_s, X_0, N) \geq 1 - \min\{(\eta + \beta N), 1\}. \tag{4}$$

Note that Eq. (4) provides a lower bound on the probability of remaining in the safe set. Hence, B is called a barrier certificate if the right hand side of Eq. (4) is greater than or equal to threshold δ_s, i.e., $1 - \min\{(\eta + \beta N), 1\} \geq \delta_s$. A major benefit of using barrier certificates for safety analysis is that static conditions (3a)–(3d) enable probabilistic reasoning on the time evolution of the stochastic system without the need to evolve the system. A popular approach to finding an SBF B is to limit the search to the set of SOS polynomials of a given degree. Then, an SOS optimization problem can be formulated to synthesize the parameters of the barrier polynomial, where the objective function is $\min \eta + N\beta$, subject to conditions (3a)–(3d) (Mazouz et al., 2022; Prajna et al., 2007; Santoyo et al., 2021). The SOS approach however becomes very conservative when X_s is non-convex. Recent techniques address this issue by using the power of neural networks as universal approximators to learn a NBF (Dawson et al., 2022; Mathiesen et al., 2022). Nevertheless, while training NBFs can be done efficiently, checking conditions (3a)–(3d) against the network is challenging, which limits scalability of NBFs. To tackle these challenges, we propose piecewise barrier functions below.

4. Piecewise stochastic barrier functions

In this paper, we introduce the notion of *piecewise* SBF (PW-SBF). Consider a partition $X_1, \ldots, X_K$ of the compact safe set X_s in K compact sets, i.e., $\bigcup_{i=1}^{K} X_i = X_s$ and $X_i \cap X_j = \emptyset$, for all $i \neq j \in \{1, \ldots, K\}$, such that vector field f is continuous in each region X_i, and we further assume that the boundary of each X_i has measure zero w.r.t. $T(\cdot \mid x)$ for any $x \in X_s$. Furthermore, let $B_i : X_i \to \mathbb{R}$ be a real-valued continuous function for every $i \in \{1, \ldots, K\}$. We define PW function

$$B(x) = \begin{cases} B_i(x) & \text{if } x \in X_i \\ 1 & \text{otherwise.} \end{cases} \tag{5}$$

The following corollary characterizes the general result of Theorem 1 for the PW function B.

Corollary 1 (*Piecewise SBF*)**.** *Piecewise function $B(x)$ in Eq. (5) is a stochastic barrier function for system (1) if $\forall i \in \{1, \ldots, K\}$ there exist scalars $\beta_i, \eta \geq 0$ such that*

$$B_i(x) \geq 0 \qquad \forall x \in X_i, \tag{6a}$$

$$B_i(x) \leq \eta \qquad \forall x \in X_i \cap X_0, \tag{6b}$$

$$\sum_{j=1}^{K} \mathbb{E}[B_j(\mathbf{x}') \mid \mathbf{x}' \in X_j] \cdot T(X_j \mid x) + T(X_u \mid x) \leq B_i(x) + \beta_i \qquad \forall x \in X_i, \tag{6c}$$

where $\mathbf{x}' = f(x, \mathbf{v})$ and T is the transition kernel in Eq. (2). Then, it holds that

$$P_s(X_s, X_0, N) \geq 1 - (\eta + N \cdot \max_{i \in \{1,\ldots,K\}} \beta_i). \tag{7}$$

Proof. The proof builds on Theorem 1. It is clear that PW function B in (5) under conditions (6a)–(6b) satisfies conditions (3a)–(3c). Then, it is enough to show that $\max_{i\in\{1,\ldots,K\}} \beta_i$ from Condition (6c) upper bounds $\mathbb{E}[B(\mathbf{x}')] - B(x)$ in Condition (3d). For $x \in X_i$ and PW $B(x)$, we have

$$\mathbb{E}[B(f(x, \mathbf{v}))] - B(x) \tag{8}$$

$$= \sum_{j=1}^{K} \mathbb{E}[B_j(f(x, \mathbf{v})) \mid f(x, \mathbf{v}) \in X_j] \cdot T(X_j \mid x) + 1 \cdot T(X_u \mid x) - B_i(x) \tag{9}$$

$$\leq \beta_i, \tag{10}$$

where the equality in (9) holds by the law of total expectation, $B(x' \in X_u) = 1$, and the inequality in (10) holds by Condition (6c). Hence, for every $x \in X_s = \bigcup_{i=1}^{K} X_i$, the expression in (8) is $\leq \max_{i\in\{1,\ldots,K\}} \beta_i$. □

Corollary 1 enables one to formulate an optimization problem to synthesize a PW-SBF. Specifically, the objective is to find B_is that minimize $(\eta + N \cdot \max\{\beta_i\}_{i=1}^{K})$ subject to conditions (6a)–(6c). The benefit of this formulation is that both the size of the partition and type of functions B_is (e.g., linear, polynomial, exponential, etc.) are design parameters. This provides flexibility for the SBF to fit different shapes of X_s. This flexibility, however, may introduce challenges as it can lead to a non-convex optimization problem, even for simple choices for B_i such as polynomial or linear functions. We introduce a set of simple but effective choices that lend themselves to efficient computational tools that outperform the state-of-the-art SBF synthesis techniques.

A major difficulty in the optimization problem results from the product of the expectation term and the transition kernel function in condition (6c), namely, $\mathbb{E}[B_j(\mathbf{x}') \mid \mathbf{x}' \in X_j] \cdot T(X_j \mid x)$. The first term not only requires to perform an expectation operation, but also a composition of B_j with the nonlinear function f. The second term T is a nonlinear function of x that involves an integral of probability density function $p_{\mathbf{v}}$. To reduce complexity, one can choose to use constant values for B_is, which simply avoids the need to perform the expectation and composition operations. This

leads to a PW Constant (PWC) SBF. Furthermore, while the analytical form of T may be hard to obtain, its bounds can be efficiently computed using, e.g., the procedure in Skovbekk et al. (2023), for general f and non-standard $p_{\mathbf{v}}$ (e.g., non-symmetric, non-unimodal, etc.). In the next section, we detail the optimization problem for these choices.

5. Piecewise constant SBF synthesis

In this section, we first formally set up an optimization problem for synthesis of PWC-SBFs. Then, we introduce three efficient computational methods to solve the optimization problem, namely, a Linear Programming (LP) duality-based approach, a LP-CEGIS procedure, and a GD-based method.

5.1. PWC-SBF synthesis

For $i, j \in \{1, \dots, K\}$, let $\underline{p}_{ij}, \overline{p}_{ij} \in [0, 1]$ denote the lower and upper bounds of the transition kernel $T(X_j \mid x)$ for every $x \in X_i$, respectively, i.e.,

$$\underline{p}_{ij} \leq T(X_j \mid x) \leq \overline{p}_{ij} \qquad \forall x \in X_i. \tag{11}$$

Similarly, we use $\underline{p}_{iu}, \overline{p}_{iu} \in [0, 1]$ for the bounds of T to the unsafe set X_u, i.e.,

$$\underline{p}_{iu} \leq T(X_u \mid x) \leq \overline{p}_{iu} \qquad \forall x \in X_i. \tag{12}$$

Note that these bounds can be computed efficiently for general f and $p_{\mathbf{v}}$ by using, e.g., techniques in Adams et al. (2022), Cauchi et al. (2019), Laurenti et al. (2020), Skovbekk et al. (2023). We define the set of all feasible values for the transition kernel for all $x \in X_i$ as

$$\mathcal{P}_i = \Big\{ p_i = (p_{i1}, \dots, p_{iK}, p_{iu}) \in [0, 1]^{K+1} \quad s.t. \\ \sum_{j=1}^{K} p_{ij} + p_{iu} = 1, \\ \underline{p}_{ij} \leq p_{ij} \leq \overline{p}_{ij} \quad \forall j \in \{1, \dots, K, u\} \Big\}. \tag{13}$$

Note that $\mathcal{P}_i$ is a convex polytopic set, specifically a simplex. The following theorem sets up an optimization problem for synthesis of a PWC-SBF.

Theorem 2 (*PWC-SBF Synthesis*). *Given a K-partition of X_s, let $\mathcal{B}_K$ be the set of PWC functions in the form of Eq. (5) with $B_i(x) = b_i \in \mathbb{R}$ for every $i \in \{1, \dots, K\}$, and let $\mathcal{P} = \mathcal{P}_1 \times \cdots \times \mathcal{P}_K$, where each $\mathcal{P}_i$ is the set of probability distributions defined in Eq. (13). Then, $B^* \in \mathcal{B}_K$ is a PWC-SBF that maximizes RHS of Eq. (7) if B^* is a solution to*

$$B^* = \arg\min_{B \in \mathcal{B}_K} \max_{(p_i)_{i=1}^K \in \mathcal{P}} \eta + N\beta \tag{14}$$

subject to

$$b_i \geq 0 \qquad \forall i \in \{1, \dots, K\}, \tag{15a}$$

$$b_i \leq \eta \qquad \forall i : X_i \cap X_0 \neq \emptyset, \tag{15b}$$

$$\sum_{j=1}^{K} b_j \cdot p_{ij} + p_{iu} \leq b_i + \beta_i \qquad \forall i \in \{1, \dots, K\}, \tag{15c}$$

$$0 \leq \beta_i \leq \beta \qquad \forall i \in \{1, \dots, K\}. \tag{15d}$$

Proof. It suffices to show that if for all i, $B_i(x)$ satisfy conditions (15a)–(15c), then PW function $B(x)$ as defined in Eq. (5) satisfies conditions (6a)–(6c). The optimization problem in equation (14) aims to maximize the safety probability over feasible values for the transition kernel $\mathcal{P}_i$, which is expressed as a minimax problem. Each β_i is bounded according to constraint (15c), and by Corollary 1, $\max_{i \in \{1, \dots, K\}} \beta_i$ upper bounds $\mathbb{E}[B(\mathbf{x}')] - B(x)$. Hence, optimizing over objective B^* also optimizes the right hand side of (7), thereby maximizing the lower bound on safety probability $P_s(X_s, X_0, N)$. □

Given a partition of X_s, Theorem 2 provides a method to synthesize a PWC-SBF that optimizes a lower bound on $P_s(X_s, X_0, N)$. This theorem immediately gives rise to questions on optimality (w.r.t. general SBFs) and computability. We address the computability question in Section 5.2. For optimality, the following proposition establishes that, in the limit of a partition of size large enough, a PWC-SBF converges to a value of safety smaller or equal to the optimal safety probability $\underline{P}_s^*(X_s, X_0, N)$ obtainable with continuous SBFs.

Proposition 1. *Let $\underline{P}_s^*(X_s, X_0, N)$ denote the optimal safety probability obtained from Theorem 1 from the class of continuous SBFs, i.e., the set of continuous functions satisfying conditions (3a)–(3d). Consider a uniform partition of X_s in K compact sets and call $(\eta + N\beta)_K^*$ the resulting optimal safety probability bound obtained from solving Theorem 2. Then, it holds that*

$$\underline{P}_s^*(X_s, X_0, N) \leq \lim_{K \to \infty} (1 - (\eta + N\beta)_K^*).$$

Proof. First, we need to show that for all $x \in X_u$, the optimal choice of the barrier is always $B(x) = 1$. Note

$$\mathbb{E}[B(f(x, \mathbf{v}))] = \int_{v : f(x,v) \in X_s} B(f(x, v)) p_{\mathbf{v}}(dv) + \\ \int_{v : f(x,v) \in X_u} B(f(x, v)) p_{\mathbf{v}}(dv).$$

Using this fact, we can rewrite Condition (3d) as the fact that $\forall x \in X_s$ it must hold that

$$\int_{v : f(x,v) \in X_s} B(f(x, v)) p_{\mathbf{v}}(dv) + \\ \int_{v : f(x,v) \in X_u} B(f(x, v)) p_{\mathbf{v}}(dv) \leq B(x) + \beta N.$$

As for $x \in X_u$, $B(x)$ only appear on the left hand side of the inequality; hence, the value of β required to satisfy the inequality is minimized by taking the smallest possible value of $B(x)$ for $x \in X_u$. Because of Condition (3b), the smallest possible value is 1. The rest of the proof follows by the fact that, within X_s, PWC-SBFs are dense w.r.t. to the set of continuous SBFs. □

This proposition shows that, despite their simplicity, PWC-SBFs are as expressive as complex forms of SBFs such as polynomials and NBFs. That is, PWC-SBFs can compute the optimal safety probability as well as or even better than continuous SBFs. In fact, our evaluations in Section 6 clearly demonstrate this point. Further, we also note that, from the proof of Proposition 1, it becomes evident that the choice of $B(x) = 1$ for all $x \in X_u$ in Eq. (5) is the optimal choice for PW-SBFs. Lastly, it is worth mentioning that a rigorous sub-optimality analysis of the PWC-SBF for a given partition is non-trivial and remains an open question.

5.2. Computation of PWC-SBFs

The optimization problem in Theorem 2 is a minimax problem with the decision variables

$$b = (b_1, \dots, b_K) \in \mathbb{R}^K_{\geq 0}, \tag{16a}$$

$$p_i = (p_{i1}, \dots, p_{iK}, p_{iu}) \in \mathcal{P}_i \qquad \forall i \in \{1, \dots, K\}, \tag{16b}$$

$$\beta_i \in \mathbb{R}_{\geq 0} \qquad \forall i \in \{1, \ldots, K\}, \tag{16c}$$

$$\eta, \beta \in \mathbb{R}_{\geq 0}. \tag{16d}$$

A major difficulty in computability of this optimization problem is in constraint (15c), which includes products of decision variables b_j and p_{ij}. As a consequence, the minimax optimization problem in Theorem 2 is bilinear, meaning that it is linear if either b_j or p_{ij} are fixed. This class of problems is generally non-convex; hence, convex solvers, which provide efficiency and guaranteed convergence, cannot be utilized. We propose three approaches to (losslessly) convexify the problem, namely dual Linear Programming (LP), LP Counter-Example Guided Inductive Synthesis (LP-CEGIS), and Gradient Decent (GD) such that the optimal solution can be efficiently computed. Each approach has its own advantages and disadvantages. The dual LP approach allows use of standard LP solvers to exactly compute the optimal solution at the cost of lesser scalability. LP-CEGIS mitigates the scalability problem by decoupling the inner maximization and outer minimization, but may still require prohibitive amounts of memory. To address this problem, GD is presented, which utilizes sparsity and provides efficient gradient computation. However, GD requires tweaking of hyperparameters to achieve good convergence in practice, due to the non-smoothness of the objective function. We first present the dual LP.

5.2.1. Dual linear program

Using duality, we first introduce an exact approach that encodes the optimization problem in Theorem 2 as a linear program.

Observe that the set of feasible transition kernels $\mathcal{P}_i$ in Eq. (13) is a simplex

$$\mathcal{P}_i = \{p_i : H_i\, p_i \leq h_i\}, \tag{17}$$

where matrix $H_i \in \mathbb{R}^{2(K+1)\times(K+1)}$ and vector $h_i \in \mathbb{R}^{2(K+1)}$ are defined by the constraints in Eq. (13), and inequality relation "$\leq$" is interpreted element-wise on vectors. Next, we define vector $\bar{b} = (b, 1)$, where b is in Eq. (16a), and dual variable $\lambda_i \in \mathbb{R}^{2(K+1)}_{\geq 0}$. Then, Constraint (15c) can be re-written as two constraints:

$$h_i^\top \lambda_i \leq b_i + \beta_i, \tag{18a}$$

$$H_i^\top \lambda_i = \bar{b}. \tag{18b}$$

Since both constraints are linear in the decision variables λ_i, b, and β_i, the optimization problem in Theorem 2 can be written as an LP, which we formalize in Theorem 3.

Theorem 3 (*PWC-SBF Dual LP*). *An optimal solution* $(b^{*1}, \beta^{*1}, \eta^{*1}, \lambda_1^*, \ldots, \lambda_K^*)$ *to the following LP coincides with an optimal solution* $(b^{*2}, \beta^{*2}, \eta^{*2}, p_1^*, \ldots, p_K^*)$ *to the optimization problem in Theorem 2 on the SBF decision variables* b, β, η. *That is,* $(b^{*1}, \beta^{*1}, \eta^{*1}) = (b^{*2}, \beta^{*2}, \eta^{*2})$:

$$\min \eta + \beta N$$

subject to

$$\begin{aligned}
&0 \leq b_i \leq 1 && \forall i \in \{1, \ldots, K\},\\
&b_i \leq \eta && \forall i : X_i \cap X_0 \neq \emptyset,\\
&h_i^\top \lambda_i \leq b_i + \beta_i && \forall i \in \{1, \ldots, K\},\\
&H_i^\top \lambda_i = \bar{b} && \forall i \in \{1, \ldots, K\},\\
&\lambda_i \geq 0 && \forall i \in \{1, \ldots, K\},\\
&0 \leq \beta_i \leq \beta && \forall i \in \{1, \ldots, K\}.
\end{aligned}$$

Proof. Theorem 3 follows as an application of the following lemma.

Lemma 1 (*Zero Duality Gap*). *Consider the following two optimization problems with decision variables* $z = (b_1, \ldots, b_K, \beta_i)$ *and* λ_i *and feasible transition kernel set* $\mathcal{P}_i = \{p_i : H_i p_i \leq h_i\}$

$$\begin{aligned}
\min_{z} \quad & \beta_i \\
\text{s.t.} \quad & \bar{b}^\top p_i \leq b_i + \beta_i \qquad \forall p_i \in \mathcal{P}_i
\end{aligned} \tag{19}$$

and

$$\begin{aligned}
\min_{z,\lambda_i} \quad & \beta_i \\
\text{s.t.} \quad & h_i^\top \lambda_i \leq b_i + \beta_i \\
& H_i^\top \lambda_i = \bar{b}, \\
& \lambda_i \geq 0.
\end{aligned} \tag{20}$$

Let z_1^* *and* (z_2^*, λ_i^*) *be optimal solutions to the Problems* 19 *and* (20), *respectively. Then,* $z_1^* = z_2^*$ *holds.*

Proof. The constraint $\bar{b}^T p_i \leq b_i + \beta_i$, for all $p_i \in \mathcal{P}_i$ can be written as an inner optimization problem

$$\begin{pmatrix} \max\limits_{p_i \in \mathcal{P}_i} & \bar{b}^T p_i \\ \text{s.t.} & H_i p_i \leq h_i \end{pmatrix} \leq b_i + \beta_i.$$

Due to strong duality of LP (Boyd & Vandenberghe, 2004, Chapter 5), we can substitute the inner problem for its equivalent (asymmetric) dual problem

$$\begin{pmatrix} \min\limits_{\lambda_i \geq 0} & h_i^\top \lambda_i \\ \text{s.t.} & H_i^\top \lambda_i = \bar{b} \end{pmatrix} \leq b_i + \beta_i.$$

Since the inner problem is a minimization, it can be lifted out into the outer problem to become the constraints in Problem (20). □

It now follows that by Lemma 1, the optimization problem in Theorem 3 is an equivalent dual formulation, i.e., with zero duality gap, of the primal maximization problem in Theorem 2. □

Hence, $B(x)$ constitutes a proper stochastic barrier certificate, which guarantees probability of safety $P_s(X_s, X_0, N) \geq 1 - (\eta + \beta N)$.

Computational complexity. The time complexity of a standard primal linear program is $\mathcal{O}(n^2\, m)$ where n is the number of decision variables and m is the number of constraints (Boyd & Vandenberghe, 2004). The program in Theorem 3 has $n = 2K^2 + 2K$ decision variables and $m = K^2 + 6K + L + 1$ where $L = |\{i : X_i \cap X_0 \neq \emptyset\}|$ is the number of regions intersecting with the initial set. Combining the complexity of a linear program with the number of decision variables and constraints for our dual linear program, we get $\mathcal{O}(n^2\, m) = \mathcal{O}(K^6)$. For many applications this may be prohibitive, hence, in the following subsections, we introduce two iterative approaches that lead to an improved runtime complexity.

5.2.2. LP counter-example guided inductive synthesis

Here, we introduce another PWC-SBF synthesis method that is exact and computationally more efficient than the dual LP approach. The method is based on splitting the minimax problem in Theorem 2 into two separate LPs. One LP generates a candidate PWC-SBF that optimizes the unsafety probability given a set of finite feasible distributions $\tilde{\mathcal{P}}_i \subset \mathcal{P}_i$, and the other LP generates distribution witnesses (counter-examples) that violate the safety probability guarantee of the candidate PWC-SBF. Then, the witnesses are added to $\tilde{\mathcal{P}}_i$, and the process repeats until no more counter-example can be generated. We dub this method as LP Counter-Example Guided Inductive Synthesis (LP-CEGIS) for PWC-SBF.

Algorithm 1 LP-CEGIS for PWC-SBF

Input: Initial set X_0, partition $X = \{X_i\}_{i=1}^K$, time horizon N, and feasible transition kernel sets $\{\mathcal{P}_i\}_{i=1}^K$.
Output: Optimal PWC-SBF B^* and safety probability bound $\underline{P}_s^*$.

$\{\tilde{\mathcal{P}}_i \leftarrow \text{SAMPLEDIST}(\mathcal{P}_i)\}_{i=1}^K$ ▷ Initialize $\tilde{\mathcal{P}}_i$
$\beta^* \leftarrow 0$ ▷ Initialize β^*
$\{\beta_i \leftarrow 1\}_{i=1}^K$ ▷ Initialize β_i
while $\beta^* < \max \beta_i$ **do**
  $b^*, \eta^*, \beta^* \leftarrow \text{PWBSYNTHLP}(X_0, N, X, \{\tilde{\mathcal{P}}_i\}_{i=1}^K)$
  for $i \leftarrow 1$ to K **do**
    $p_i, \beta_i \leftarrow \text{COUNTEREXDISTLP}(b^*, \mathcal{P}_i, i)$
    $\tilde{\mathcal{P}}_i \leftarrow \tilde{\mathcal{P}}_i \cup \{p_i\}$ ▷ add counterexamples
return $B^* \leftarrow (b^*, 1)$ and $\underline{P}_s^* \leftarrow 1 - (\eta^* + N\beta^*)$

Algorithm 2 $\text{PWBSYNTHLP}(X_0, N, X, \{\tilde{P}_i\}_{i=1}^K)$

Input: Initial set X_0, partition $X = \{X_i\}_{i=1}^K$, time horizon N, and finite set of distributions $\{\tilde{\mathcal{P}}_i\}_{i=1}^K$.
Output: Optimal vector $b^* = (b_1, \dots, b_K)^*$ and scalars η^* and β^* that minimize unsafety probability w.r.t. the given finite distribution sets $\tilde{P}_1, \dots, \tilde{P}_K$.

$b^*, \beta^*, \eta^* \leftarrow \arg\min_{b,\beta,\eta} \eta + N \cdot \beta$

subject to:

$$0 \le b_i \le 1 \qquad \forall i \in \{1, \dots, K\}$$
$$b_i \le \eta \qquad \forall i : X_i \cap X_0 \neq \emptyset$$
$$\sum_{j=1}^{K} b_j \cdot p_{ij} + p_{iu} \le b_i + \beta \qquad \forall i \text{ and } \forall p_i \in \tilde{\mathcal{P}}_i$$

return b^*, η^*, β^*

The LP-CEGIS algorithm is shown in Alg. 1, which relies on subroutines in Algs. 2 and 3. The main algorithm first sets up the counter-example distribution sets $\tilde{\mathcal{P}}_i$ by sampling a feasible distribution from each $\mathcal{P}_i$. Then, based on $\tilde{\mathcal{P}}_i$, it synthesizes a candidate optimal vector b^* that minimizes the unsafety probability with its corresponding scalar β^* using the subroutine PWBSYNTHLP. As shown in Alg. 2, this subroutine is an LP that captures the min component (outer optimization) of the minimax problem in Corollary 1. Next, given b^*, for each partition region X_i, an optimal distribution p_i that maximizes its corresponding martingale gap β_i is computed using subroutine COUNTEREXDISTLP and then added to the set of witnesses $\tilde{\mathcal{P}}_i$. As shown in Alg. 3, COUNTEREXDISTLP is also an LP, and it captures the max component (inner optimization) of the minimax problem. Note that the function Vertices($\mathcal{P}_i$) returns the (finite) set of vertices of simplex $\mathcal{P}_i$. If the obtained martingale gap β_i is greater than β^* for some region X_i, it means that the candidate b^* is not optimal and there exists at least a distribution that violates the probabilistic guarantee of the candidate. Hence, the process repeats with the updated witnesses until $\beta_i \le \beta^*$ for all $i \in \{1, \dots, K\}$.

The following theorem guarantees that the LP-CEGIS algorithm terminates in finite time with an optimal solution.

Theorem 4 (*LP-CEGIS for PWC-SBF*). *Alg. 1 terminates in finite time with PWC-SBF B^* that is an optimal solution to the problem in Theorem 2.*

Algorithm 3 $\text{COUNTEREXDISTLP}(b, \mathcal{P}_i, i)$

Input: Vector of constants $b = (b_1, \dots, b_k)$, a set of feasible distributions $\mathcal{P}_i$ as defined in equation (13), and region index i.
Output: Counterexample distribution $p_i = (p_{i1}, \dots, p_{iK}, p_{iu})$ that maximizes martingale gap β_i with respect to b.

$p_i, \beta_i \leftarrow \arg\max_{p_i, \beta_i} \beta_i$

subject to:

$$\sum_{j=1}^{K} b_j \cdot p_{ij} + p_{iu} = b_i + \beta_i$$
$$p_i \in \text{Vertices}(\mathcal{P}_i)$$

return p_i, β_i

Proof. The strategy of the proof is as follows: (i) we first prove convergence in finite number of iterations of Alg. 1 using standard results from linear programming and then, (ii) we prove that the algorithm converges to the optimum using a proof by contradiction.

To prove convergence of Alg. 1 in finite number of iteration, we start by observing that the constraint $\sum_{j=1}^K b_j \cdot p_{ij} + p_{iu} \le b_i + \beta_i$ for all $p_i \in \mathcal{P}_i$ is true if and only if $\sum_{j=1}^K b_j \cdot p_{ij} + p_{iu} \le b_i + \beta_i$ for all $p_i \in \text{Vertices}(\mathcal{P}_i)$ (Ben-Tal & Nemirovski, 1998). Hence for the β_i computed using Alg. 3, it holds that $\sum_{j=1}^K b_j \cdot p_{ij} + p_{iu} \le b_i + \beta_i$ for all $p_i \in \mathcal{P}_i$. Next, let (b^*, η^*, β^*) denote the optimal solution of Alg. 2 with respect to a given set of distributions $\{\tilde{\mathcal{P}}_i\}_{i=1}^K$. Notice that β^* is a (non-strict) lower bound for β_i for all $i \in \{1, \dots, K\}$, provided that β_i is computed from b^*, since $\tilde{\mathcal{P}}_i \subset \mathcal{P}_i$. Then, there are two cases:

- If for some $i \in \{1, \dots, K\}$ it holds that $\beta^* < \beta_i$, then there exists a distribution $p_i \in \mathcal{P}_i \setminus \tilde{\mathcal{P}}_i$ such that $\sum_{j=1}^K b_j^* \cdot p_{ij} + p_{iu} > b_i^* + \beta^*$. As a consequence, if Alg. 3 returns a β_i such that $\beta^* < \beta_i$ then the accompanying distribution $p_i \in \text{Vertices}(\mathcal{P}_i)$ is a *previously unseen* counter-example for the condition $\sum_{j=1}^K b_j^* \cdot p_{ij} + p_{iu} \le b_i^* + \beta^*$ for all $p_i \in \mathcal{P}_i$.
- If instead $\beta^* = \max \beta_i$, then it holds that $\sum_{j=1}^K b_j^* \cdot p_{ij} + p_{iu} \le b_i^* + \beta^*$ for all $p_i \in \mathcal{P}_i$.

As a result, to conclude it is enough to note that since $\mathcal{P}_i$ is a simplex, it has a finite number of vertices.

To prove that Alg. 1 converges to the optimal solution, we can employ a proof by contradiction. Assume that $\beta^* = \max \beta_i$ and the resulting β^* is not optimal. Then, there must exists a distribution $p_i \in \mathcal{P}_i$ such that $\text{PWBSYNTHLP}(X_0, N, X, \{\tilde{\mathcal{P}}_i\}_{i=1}^K \cup p)$ returns a smaller β^* compared to $\text{PWBSYNTHLP}(X_0, N, X, \{\tilde{\mathcal{P}}_i\}_{i=1}^K)$, but this is a contradiction as $\{\tilde{\mathcal{P}}_i\}_{i=1}^K \subseteq \{\tilde{\mathcal{P}}_i\}_{i=1}^K \cup p$. □

Computational complexity. We note that the LP in subroutine COUNTEREXDIST (Agl. 3) can be solved more efficiently using a procedure called O-maximization as proposed in Givan et al. (2000). The time complexity of this procedure is $\mathcal{O}(K \log K)$, whereas a standard LP algorithm runs in $\mathcal{O}(K^3)$ time. In the worst-case, all the vertices of the simplex $\mathcal{P}_i$ are explored for all $i \in \{1, \dots, K\}$, which has time complexiy $\mathcal{O}(K^2)$. It is further noted that Alg. 1 keeps track of the history of counter-examples p_is added at each iteration, making the approach potentially memory intensive.

5.2.3. Projected gradient descent

While LP-CEGIS is faster than the dual approach, it can be memory intensive. To alleviate this and allow better scalability,

we present a third method for computing a PWC-SBF, namely, a gradient descent-based approach.

With an abuse of notation, let $\beta_i(b)$ be martingale gap of region X_i for a given PWC-SBF defined by vector $b = (b_1, \ldots, b_K)$. Specifically,

$$\beta_i(b) = \sup_{p_i \in \mathcal{P}_i} \max\left\{0,\ \sum_{j=1}^{K} b_j \cdot p_{ij} + p_{iu} - b_i\right\}, \tag{21}$$

and $\beta(b) = \max_i \beta_i(b)$. Similarly, we denote

$$\eta(b) = \max_{i: X_i \cap X_0 \neq \emptyset} b_i. \tag{22}$$

Then, we define loss (objective) function

$$\mathcal{L}(b) = \eta(b) + N \cdot \beta(b). \tag{23}$$

This loss function indeed describes the objective function of the minimax problem in Theorem 2. Hence, by minimizing $\mathcal{L}(b)$, we solve the minimax problem. Below, we show that $\mathcal{L}(b)$ is convex, and hence, we can use a gradient descent-based method for its optimization. More precisely, our approach is based on projected subgradient descent, since the elements of b are constrained to $[0, 1]$ and the maximization and supremum in Eq. (21) are not smooth, admitting only subgradients.

Theorem 5 (*Convexity of $\mathcal{L}(b)$*). *The objective function $\mathcal{L}(b)$ in Eq. (23) is convex in b.*

Proof. The proof follows a standard structure from disciplined convex programming, where functions are composed under convexity-preserving operations. We start by proving that $\beta_i(b)$ is convex in b. To this end, observe that $\sum_{j=1}^{K} b_j \cdot p_{ij} + p_{iu} - b_j$ is convex in b, invariant to the value p_i. Next, $\max(0, \cdot)$ is a convexity-preserving function, thus $\beta_i(b)$ is convex in b. The finite maximization for both η and β are once again convexity-preserving operations and finally, addition is convexity-preserving. This concludes the proof that $\mathcal{L}(b)$ is convex in b. □

Notice that the computation of $\beta_i(b)$ in Eq. (21) is equivalent to the inner optimization problem of LP-CEGIS. Hence, the CounterExDistLP routine in Alg. 3 (based on O-maximization method) can be used for efficient computation of subgradients of β_i. One subgradient for $\mathcal{L}(b)$ is the following

$$\nabla_b \mathcal{L}(b) = \vec{1}_\eta(b) + N \cdot \vec{1}_\beta(b), \tag{24}$$

where $\vec{1}_\eta(b) = \vec{1}_i$ is a one-hot vector with the 1 being in element $i = \text{argmax}_{i: X_i \cap X_0 \neq \emptyset} b_i$, and

$$\vec{1}_\beta(b) = \nabla_b \beta_l(b) = p_l^* - \vec{1}_l, \tag{25}$$

where $l = \arg\max_l \beta_l(b)$, and $p_l^* = \arg\max_{p_l \in \mathcal{P}_l} \sum_{j=1}^{K} b_j \cdot p_{lj} + p_{lu} - b_l$.

A challenge with applying subgradient methods in practice is that small step sizes are required due to the non-smoothness, slowing down the convergence. We propose to ameliorate this by substituting the maximization in $\mathcal{L}(b)$ with an L_p-norm where $1 < p < \infty$. Specifically, let $\tilde{\eta}(b) = (b_{i_1}, \ldots, b_{i_m})$ be a vector of PWC-SBF values corresponding to the regions that overlap with X_0, and $\tilde{\beta}(b) = (\beta_1(b), \ldots, \beta_K(b))$ be a vector of $\beta(b)_i$ values. Then, the proposed loss function is

$$\tilde{\mathcal{L}}(b) = \|\tilde{\eta}(b)\|_p + N\|\tilde{\beta}(b)\|_p, \tag{26}$$

and the corresponding gradient is

$$\nabla_b \tilde{\mathcal{L}}(b) = \sum_{k=1}^{m} \left(\frac{\tilde{\eta}(b)_k}{\|\tilde{\eta}(b)\|_p}\right)^{p-1} \vec{1}_{i_k} + \sum_{i=1}^{K} \left(\frac{\tilde{\beta}_i(b)}{\|\tilde{\beta}(b)\|_p}\right)^{p-1} \nabla_b \beta_i(b). \tag{27}$$

Algorithm 4 GD for PWC-SBF

Input: Initial set X_0, partition $X = \{X_i\}_{i=1}^K$, time horizon N, and feasible transition kernel sets $\{\mathcal{P}_i\}_{i=1}^K$.
Output: Optimal PWC-SBF B^* and safety probability bound $\underline{P}_s^*$

```
b^(1) ← (p_1u, ..., p_Ku)                          ▷ Initialize b
b_best ← COPY(b)                                    ▷ Initialize b_best
L_best ← ∞
k ← 1                                               ▷ Iteration number
while not converged do
    g ← ∇_{b^(k-1)} L̃(b^(k-1))                      ▷ According to Eq. (27)
    α_k ← STEPSIZE(k, g)
    b^(k) ← b^(k-1) − α_k g                         ▷ Gradient descent step
    for i ← 1 to K do                               ▷ Project onto [0, 1]
        b_i^(k) ← max(0, min(1, b_i^(k)))
    η_best, β_best ← ||η̃(b^(k))||_∞, ||β̃(b^(k))||_∞
    if η_best + N β_best < L_best then
        b_best ← COPY(b^(k))
        L_best ← η_best + N β_best
    k ← k + 1
return B* ← (b_best, 1) and P_s* ← 1 − L_best
```

The benefit of this loss is that L_p-norms are smooth, and that $\tilde{\mathcal{L}}(b) \geq \mathcal{L}(b)$ for every $p_i \in \mathcal{P}_i$ and all b. Furthermore, the relative magnitude is bounded, $\|y\|_p / \|y\|_\infty \leq \sqrt[p]{r}$ for any $y \in \mathbb{R}^r$. With the L_p-norm, we may tune the over-approximation to a trade-off between smoothness and tighter approximating the L_∞-norm. The modified loss, through smoothness, also addresses the issue that each $\mathcal{P}_i$ often is sparse, i.e., $\overline{p}_{ij} \approx 0$ for many j. Therefore, $\nabla_b \mathcal{L}(b)$ is often 0 for most regions. We remark that the modified loss $\tilde{\mathcal{L}}(b)$ is not smooth due to the maximization and supremum in the computation of $\beta_i(b)$.

The projected subgradient descent procedure is described in Alg. 4. For each iteration, the algorithm calculates the gradient according to Eq. (27) and an appropriate decreasing step size α, followed by the gradient step. Since the PWC-SBF must reside in $[0, 1]^K$ and the gradient step may violate this, it is projected back onto the admissible space. Finally, since subgradient methods are not a steepest descent method, the algorithm keeps track of the best observed loss.

To guarantee convergence of gradient descent (Alg. 4) towards the optimum, it is paramount that the objective is convex. Theorem 5 shows that $\mathcal{L}(b)$ is convex. From this, it immediately follows that $\tilde{\mathcal{L}}(b)$ is also convex.

Corollary 2. *The modified objective $\tilde{\mathcal{L}}(b)$ is convex in b.*

Proof. The proof follows from the fact that η and β are convex in b. Further, L_p-norms are convexity-preserving operations (Boyd & Vandenberghe, 2004). □

We may characterize the non-asymptotic convergence of Alg. 4, i.e. that it finds an ϵ-optimal b within a finite number of steps.

Proposition 2. *Let ϵ denote the desired error, the gradient bound $G \geq \sup_{b \in [0,1]^K} \|\nabla_b \tilde{\mathcal{L}}(b)\|_2$, and the upper bound of the initial distance to the optimal barrier b^* by $R \geq \|b^{(1)} - b^*\|_2$. Then there exist integers N_1 and N_2 such that $\alpha_k \leq \epsilon / G$ for all $k > N_1$ and*

$$\sum_{k=1}^{N_2} \alpha_k \geq \frac{1}{\epsilon}\left(R^2 + G^2 \sum_{k=1}^{N_1} \alpha_k^2\right).$$

Then for any $k > N = \max(N_1, N_2)$ it holds that

$$\tilde{\mathcal{L}}(b_{\text{best}}) - \tilde{\mathcal{L}}(b^*) \leq \epsilon. \tag{28}$$

Table 1
Benchmark results for the *piecewise constant* and *continuous* SBF methods. The three PWC-SBF synthesis methods are: dual LP, LP-CEGIS and GD. These are compared against state-of-the-art methods: SOS and NBF. K denotes the size of partition of X_s, τ_p is the computation time for the bounds on the transition kernel, $\underline{P}_s$ is the obtained lower bound on the safety probability for $N = 10$ time steps, τ_o is the computation time for the SBF synthesis, and Deg is the degree of the SOS polynomial for SBF.

			Piecewise Constant Stochastic Barriers						Continuous Stochastic Barriers*				
			Dual LP		LP-CEGIS		GD		NBF		SOS		
Model	K	$\tau_p(s)$	$\underline{P}_s$	$\tau_o(s)$	$\underline{P}_s$	$\tau_o(s)$	$\underline{P}_s$	$\tau_o(s)$	$\underline{P}_s$	$\tau_o(s)$	Deg	$\underline{P}_s$	$\tau_o(s)$
Linear	64	0.02	0.992	0.52	0.992	0.04	0.952	0.04	0.585	3850.93	4	0.582	0.014
2D	225	0.31	0.998	164.60	0.998	0.44	0.973	0.20	0.940	3991.47	8	0.582	0.265
Convex	900	8.85	0.999	1087.78	0.999	17.93	0.990	7.22	0.961	4025.67	30	0.978	151.16
	2500	41.44	0.999	2897.77	0.999	88.45	0.998	52.78	0.976	4085.65	36	0.992	458.21
Linear	900	5.04	0.494	1197.99	0.494	3.79	0.494	3.52	0.792	3546.69	12	0.010	0.02
2D	1225	8.20	0.800	1389.78	0.800	7.22	0.800	6.64	0.844	3579.58	20	0.010	11.08
Non-Convex	1444	9.18	0.921	1545.45	0.921	11.65	0.921	10.12	0.855	3589.13	24	0.023	37.89
	2926	47.98	0.927	3161.56	0.927	98.36	0.927	20.86	0.928	3599.85	26	0.034	62.88
	5890	179.94	0.929	8191.65	0.929	458.44	0.929	133.84	0.929	3675.77	30	0.075	196.85
	11 818	477.45	–	TO	0.936	1875.49	0.936	842.67	0.931	3744.23	34	–	OM
	24 336	987.65	–	TO	0.938	4441.55	0.938	3099.30	0.936	4234.56	36	–	OM
Pendulum	120	6.37	1.00	0.51	0.99	5.84	0.989	3.75	0.999	4242.89	4	0.999	7.71
2D	240	18.33	1.00	6.08	1.00	14.88	0.990	9.99	0.999	4457.82	4	0.999	34.96
	480	37.84	1.00	29.39	1.00	43.42	0.999	17.88	0.999	4675.12	4	0.999	187.60
Unicycle	1250	1103.42	0.750	1000.19	0.750	26.37	0.750	5.68			2	0.00	3110.21
4D	1800	1756.25	0.975	1719.58	0.975	92.26	0.975	25.78			4	0.00	5451.19
	2400	2001.11	0.998	2548.56	0.998	145.45	0.998	55.59			6	–	OM
Quadrotor	7865	80.80	0.770	9906.67	0.770	1174.49	0.770	2589.56			2	0.584	0.20
6D	15 625	160.61	–	TO	0.901	3788.98	0.901	3258.87			8	0.900	8628.58
Convex	46 656	458.59	–	TO	–	TO	0.912	9542.75			12	–	OM
Quadrotor	15 625	188.10	–	TO	0.670	3845.25	0.670	3478.31			4	0.00	4.91
6D	31 250	395.59	–	TO	0.810	9548.78	0.810	5878.28			8	0.00	8715.54
Non-Convex	46 656	506.99	–	TO	–	TO	0.900	9789.54			12	–	OM
Quadrotor	65 536	845.44	–	TO	–	TO	0.500	19377.90			2	0.00	14830.23
8D	128 000	2530.74	–	TO	–	TO	0.560	39132.59			4	–	OM

* For linear systems that use SOS optimization, the state space is not partitioned. Further, synthesis methods for SOS and NBF do not use bounds on the transition kernel.

The ϵ-optimal convergence follows directly from a result for convergence with strictly decreasing and non-summable step sizes in Boyd et al. (2003). A conservative bound on R is K and on G is $m + NK\sqrt{2}$. To conclude, Alg. 4 converges to a desired precision in a finite number of steps (Boyd et al., 2003).

Hyperparameters. The key hyperparameters for running (Alg. 4) are:

- maximum number of iterations,
- initial learning rate, which defines the update step,
- *decay* parameter, which fine-tunes convergence, and
- momentum, which is used to accelerate convergence.

The maximum number of iterations must be sufficiently high to allow convergence, while the initial learning rate generally performs well with its default value. Reducing the learning rate slows convergence, whereas increasing it too much can lead to divergence until the decay sufficiently reduces it. The decay parameter plays a crucial role, as it must be neither too large nor too small to ensure stability. Finally, momentum is essential for accelerating convergence while preventing overstepping. In our implementation the default values are 10,000 for max iterations, 0.01 for learning rate, 0.9999 for the decay parameter, and 0.9 for the momentum.

Computational complexity. The complexity per iteration of the projected subgradient descent (Alg. 4), utilizing the O-maximization procedure to compute β_i, is $\mathcal{O}(K \log K)$. We emphasize that much of the computation can be parallelized.

6. Evaluations

To show the power of PWC-SBFs and efficacy of our proposed synthesis methods, we study their performance on a set of seven benchmark problems, consisting of stochastic systems with linear and nonlinear dynamics and varying dimensionality, from 2D to 8D. We also compare the performance of PWC-SBFs against state-of-the-art continuous SBF techniques, namely SOS and NBF.

Our implementations of all methods, except NBF, including the three proposed PWC-SBF synthesis methods, are in `Julia`. The code is publicly available for download on GitHub.[2] All the computations for the benchmarks were performed on a computer with 3.9 GHz 8-Core CPU and 128 GB of memory.

The stochastic systems we considered are:

- 2D linear stochastic system with a (1) convex X_s, and (2) non-convex X_s,
- 2D pendulum Neural Network Dynamic Model (NNDM) adapted from Mazouz et al. (2022),
- 4D nonlinear unicycle model under a hybrid feedback control law,
- 6D linearized lateral and vertical quadrotor guidance model with a hybrid controller with a (1) convex X_s, and (2) non-convex X_s,
- 8D linearized lateral and longitudinal quadrotor guidance model with a hybrid controller.

A summary of the results is shown in Table 1. For all case studies, $N = 10$ time steps. `TO` and `OM` denote time-out ($\tau > 45 \times 10^3 s$) and out-of-memory, respectively. Due to limitations pertaining to memory or hybrid control, the NBF is only run for the 2D linear system and the pendulum NNDM. It is further denoted that all NBF experiments are run for 150 epochs, with 400 iterations per epoch. Each NBF architecture consists of 2 hidden layers with 32 ReLu activated neurons per layer. For a relevant comparison, it is noted that NBF uses the same initial partitioning as PWC. Finally, it is noted that the GD approach uses a decay rate of 0.999, and a termination condition when the difference in max β is less than $1e^{-6}$ among 50 iterations. We provide detailed discussions on the experimental setup and obtained results below. For case studies with non-convex X_s, we use the notion of *obstacle* to refer to the unsafe sets. All the obstacles are hyper-rectangles defined by a center point $c = (c_1, \ldots, c_n) \in \mathbb{R}^n$ and the half-length along each dimension denoted by ϵ_i. Formally, $\mathcal{O}(c, \epsilon) = \{(x_1, \ldots, x_n) \in \mathbb{R}^n : |x_i - c_i| \leq \epsilon_i \;\; \forall 1 \leq i \leq n\}$.

[2] https://github.com/aria-systems-group/StochasticBarrier.jl

6.1. 2D linear system

We considered a linear stochastic system in $\mathbb{R}^2$ with dynamics

$$\mathbf{x}(k+1) = 0.5I \cdot \mathbf{x}(k) + \mathbf{v},$$

where I is the identity matrix, and $\mathbf{v} \sim \mathcal{N}(0, 10^{-2}I)$. The initial set $X_0 = [-0.8, -0.6] \times [-0.2, 0.0]$. We consider two cases for X_s, both defined based on the set $X_D = [-1, 0.5] \times [-0.5, 0.5]$ as shown in Fig. 1(a):

(1) Convex safe set: $X_s^{conv} = X_D$, and
(2) Non-convex safe set: $X_s^{ncon} = X_D \setminus \bigcup_{i=1}^{2} \mathbb{O}_i(c_i, \epsilon_i)$, where $c_1 = (-0.55, +0.30)$, $\epsilon_1 = (0.02, 0.02)$, $c_2 = (-0.55, -0.15)$, $\epsilon_2 = (0.02, 0.02)$.

Convex X_s^{conv} case. Observe in Table 1 that all three PWC-SBF methods outperform the state-of-the-art continuous SBF techniques SOS and NBF. It takes a degree 36 polynomial SBF to obtain a result similar to that of the Piecewise Constant (PWC) methods using just $K = 64$, despite the absence of obstacles (convexity of X_s). This is mainly attributed to the fact that initial set is not configured in a symmetric fashion, making curve fitting difficult for even simple 2-dimensional problems. The main limitation of NBF in this experiment pertains to the training time. Note that it takes the continuous SBF methods 3–4 orders of magnitude larger in computation time to arrive to a similar value of $\underline{P}_s$ as the PWC methods. In addition, the continuous SBF methods never reached $\underline{P}_s = 0.999$. Among the PWC-SBF methods, Dual LP and LP-CEGIS always obtain the same $\underline{P}_s$, as both are exact optimal methods. However, it takes LP-CEGIS at least an order of magnitude less time to arrive to this optimal value. That is due to the large number of the decision variables in the Dual LP, which increases as K increases. The GD approach is the fastest method; however, its hyperparameters are tricky to design. For this case study, GD terminated before convergence to the true optimal $\underline{P}_s$.

Non-convex X_s^{ncon} case. For this case, the power of the PWC-SBFs are even more evident (see Fig. 1). Due to the fact that the obstacles are within the safe set, the SOS approach has difficulties fitting a function. This forces $B(x) \geq 1$ in the entire state space, as is evident from Fig. 1(b). This phenomenon does not improve much for a significant increase in the degree of the SOS polynomial, as it is an inherent limitation of the approach. On the other hand, our methods effectively tackle this problem by assigning $b_i = 1$ to the regions that overlap with the obstacles, and optimize over the remainder of the state space. This results in the PWC as depicted in Fig. 1(d). The difference is likewise highly notable in terms of the safety probability $\underline{P}_s$, where SOS can only guarantee safety probability of 0.075 for the polynomial degree 30, where two of PWC-SBF methods provide up to 0.938 safety probability. Note that for large values of K, the dual LP is unable to terminate, due to the scope of the required convex optimization. A similar fact is observed for high degree SOS barrier polynomials. The NBF method performs well in terms of safety probability, at lower number of partitions even better than PWC SBF. Nonetheless, the PWC SBFs are superior in terms of computation time.

6.2. 2D pendulum NNDMs

In this case study, we consider the NNDM from Mazouz et al. (2022) with dynamics $\mathbf{x} = f^{NN}(x) + \mathbf{v}$, where f^{NN} is a NN trained on a Pendulum agent with a fixed mass m and length l in closed-loop with a NN controller. The actuator is limited by $u \in [-1, 1]$ and the dynamics of the pendulum are

$$\dot{\theta}_{k+1} = \dot{\theta}_k + \frac{3g}{2l}\sin(\theta_k)\delta t^2 + \frac{3}{ml^2}u\delta t^2,$$

$$\theta_{k+1} = \theta_k + \dot{\theta}_{k+1}\delta t$$

For the details on the trained model, we refer to Mazouz et al. (2022). For this system, noise $\mathbf{v} \sim \mathcal{N}(0, 10^{-4}I)$. The safe set is $\theta \in [-\frac{\pi}{15}, \frac{\pi}{15}]$ and $\dot{\theta} \in [-1, 1]$ and the initial set is $\theta, \dot{\theta} \in [-0.01, 0.01]$.

We note that since for this particular case study the dynamics are given by a NN, the SOS approach also uses the partitioning of size K, where local linear relaxations of f^{NN} are performed. For more information on this SOS formulation of the SBF see Mazouz et al. (2022). As can be observed from Table 1, the PWC, SOS and NBF methods perform similarly in terms of safety probability. It is noted that SOS performs well due to the fact that the initial set is centered in the state space. In Fig. 2, we show the plots of the SOS, NBF and PWC (dual approach) SBFs. In Fig. 2(a), we observe that the obtained SOS polynomial is an over-fit, as the values near the boundary of the safe set by-far exceed one. We observe a similar yet less excessive pattern for the NBF in Fig. 2(b). For the PWC SBF the barrier in the domain is less then or equal to 1. Among the PWC SBF methods, a general trend is observed: the dual LP is the slowest algorithm and the GD the fastest. It is also worth noting that that the GD method terminates with a $\underline{P}_s$ that is slightly below the optimal value, indicating the difficulty of designing the termination criterion.

6.3. 4D unicycle

In this case study, we consider a wheeled mobile robot with the dynamics of a unicycle

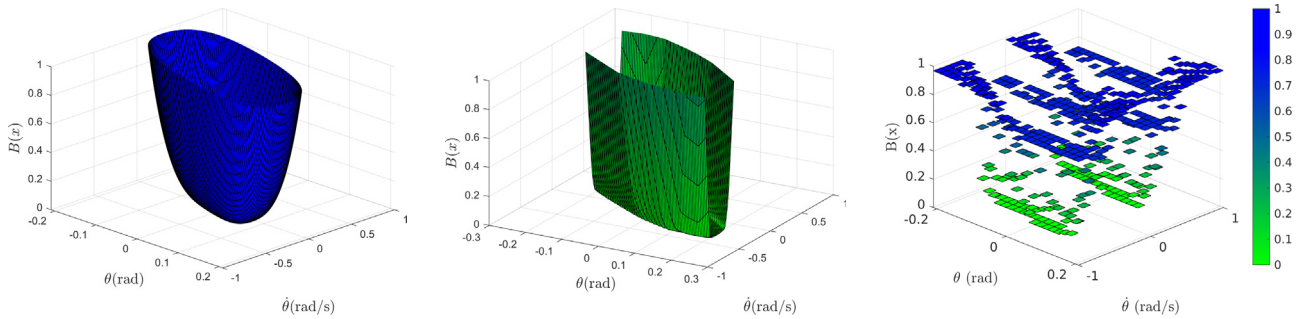
$$\dot{x} = v\cos\theta, \quad \dot{y} = v\sin\theta, \quad \dot{\theta} = \omega, \quad \dot{v} = a,$$

where $x \in [-1.0, 0.5]$ and $y \in [-0.5, 1.0]$ are the Cartesian position, $\theta \in [-1.75, 0.5]$ is the orientation with respect to the x-axis, and $v \in [-0.5, 1.0]$ is the speed. The inputs are steering rate ω and acceleration a. We design a feedback linearization controller according to De Luca et al. (2000) coupled with an LQR stabilizing controller, making this a hybrid model. We use $\Delta t = 0.01$ to obtain a discrete time dynamics, using the Euler method. We add noise $\mathbf{v} \sim \mathcal{N}(0, 10^{-4}I)$ to capture the time-discretization error that is inherent to the Euler method.

Fig. 3 shows 10^5 Monte Carlo simulations of the trajectories of the unicycle from the initial set defined by a ball centered at $(-0.5, -0.5, 0, 0)$ with radius 0.01. These simulated trajectories suggest that the system is stable under the hybrid control law. This is in-line with the results in Table 1 for the PWC methods, especially for $K = 2400$, for which $\underline{P}_s = 0.998$. Note that the SOS approach highly suffers for this nonlinear case study, with $\underline{P}_s = 0$, despite very long computation times. This further emphasizes the efficacy of our methods. Observe that the GD approach terminates with the optimal $\underline{P}_s$ value in this case study. We finally note that since the controller is hybrid, current NBF formulations cannot be employed for this problem.

6.4. 6D and 8D quadcopter systems

To evaluate scalability of the proposed PWC-SBF methods, we considered a set of high-dimensional systems. Specifically, we considered a quadrotor guidance model where the dynamics consist of lateral and vertical dynamics and the longitudinal and spin variables are constrained, and then lateral and longitudinal motion model by fixing the vertical and spin variables.



(a) Degree-8 SOS-polynomial (b) NBF with $|K| = 480$. (c) PWC (dual LP) with $|K| = 480$.

Fig. 2. SBF plots for the pendulum NNDM case study.

6.4.1. 6D lateral-vertical model

The lateral closed-loop state space model is defined as

$$\Delta\dot{y}_E = \Delta v, \qquad \Delta\dot{v} = g\Delta\phi,$$
$$\Delta\dot{\phi} = \Delta p, \qquad \Delta\dot{p} = \frac{1}{I_x}\Delta L_c,$$

where g is gravity, and I_x is the inertia about the x-axis. State $y_E \in [-0.5, 2.0]$ is the $y-$position, and $v \in [-1.0, 1.0]$ is the corresponding velocity. The roll angle $\phi \in [-0.1, 0.1]$, the roll rate $p \in [-0.1, 0.1]$, and L_c is the roll control moment. The control law follows directly from

$$\Delta L_c = -k_1\Delta p - k_2\Delta\phi - k_3\Delta v - k_3k_4\Delta y_E + k_3k_4\Delta y_{E,r},$$

where k_i denotes the gain for $i \in \{1, \ldots, 4\}$, and $\Delta y_{E,r}$ is the reference position. The lateral model is appended with a guidance model for vertical motion given by

$$\Delta\dot{z}_E = \Delta w, \qquad \Delta\dot{w} = \frac{1}{m}\Delta Z_c,$$

where m denotes the quadrotor mass, $z_E \in [-0.5, 3.0]$ denotes the z-position and $w \in [-0.5, 1.5]$ is the yaw rate. The control law ΔZ_c is defined as

$$\Delta Z_c = -k_1\Delta z_E - k_2\Delta w - Fr$$

where Fr denotes the reference signal for the vertical dynamics. The full dynamics constitutes a 6D linear model. Using $\Delta t = 0.01$, we obtained a discrete-time dynamics via the Euler method, to which we added noise $\mathbf{v} \sim \mathcal{N}(0, 10^{-4}I)$. The designed control laws stabilize the quadrotor around the equilibrium point $x_{\text{eq}} = (1, 0, 0, 0, 2, 0)$. Monte Carlo simulations of 10^5 trajectories of this model is shown in Fig. 4. The 3D workspace (environment) is shown in Fig. 4(c). We considered two safe sets:

(1) Convex safe set: X_s^{conv} is defined by the given ranges of the variables above,
(2) Non-convex safe set: $X_s^{ncon} = X_s^{conv} \setminus \bigcup_{i=1}^{2} \mathbb{O}(c_i, \epsilon_i)$, where $c_1 = (1, 0, 0, 0, 1, 0)$, $\epsilon_1 = 0.01 \cdot \mathbb{1}_{6\times 1}$ $c_2 = (1, 0, 0, 0, 2.75, 0)$, $\epsilon_2 = 0.01 \cdot \mathbb{1}_{6\times 1}$. Fig. 4(c) depicts the obstacles.

Convex X_s^{conv} case. Table 1 shows that the two PWC methods (namely, LP-CEGIS and GD) outperform SOS, both in terms of computation time, as well as safety probability certification. It is important to note that systems with linear dynamics and convex X_s are an ideal setup for SOS optimization. However, it has a lot of difficulty with the dimensionality of the system as the results suggest. For this case study, an SOS polynomial of degree greater than 8 runs out of memory. Finally, we were not able to run NBF for this problem since it cannot handle hybrid controllers and suffers from scalability.

Non-convex X_s^{ncon} case. In this case, it is even more evident that PWC-SBFs can be more powerful than continuous SBFs. The SOS method provides $\underline{P}_s = 0.00$, whereas the GD method gets to $\underline{P}_s = 0.900$. Observe that the dual LP method times out, which is also true for the LP-CEGIS method when $K > 4.5 \times 10^4$. The GD approach is able to handle this, showing the capability of PWC SBFs in scalability.

6.5. 8D lateral-longitudinal dynamics

Here, we aim to test the performance boundary of PWC-SBFs in terms of scalability. We consider an 8D model by combining the lateral dynamics of the quadcopter with its longitudinal and fixing vertical and spin variables. In a similar fashion as lateral, the longitudinal guidance is constructed under the dynamics

$$\Delta\dot{x}_E = \Delta u, \qquad \Delta\dot{u} = -g\Delta\delta,$$
$$\Delta\dot{\theta} = \Delta q, \qquad \Delta\dot{q} = \frac{1}{I_y}\Delta M_c,$$

where g is gravity, and I_y is the inertia about the y-axis. State $x_E \in [-0.5, 4.0]$ is the $x-$position, and $u \in [-0.5, 1.5]$ is the corresponding velocity. The pitch angle $\theta \in [-0.1, 0.1]$, the pitch rate $q \in [-0.1, 0.1]$, and M_c is the pitch control moment. The control law follows directly from

$$\Delta M_c = -l_1\Delta q - l_2\Delta\theta - l_3\Delta u - l_3l_4\Delta x_E + l_3l_4\Delta x_{E,r},$$

where l_i denotes the gain for $i \in \{1, \ldots, 4\}$, and $\Delta x_{E,r}$ is the reference position.

The results are shown in the bottom 2 rows of Table 1. For this system, the SOS approach is unable to provide any safety

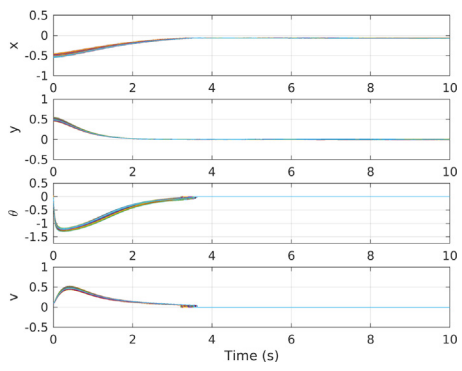
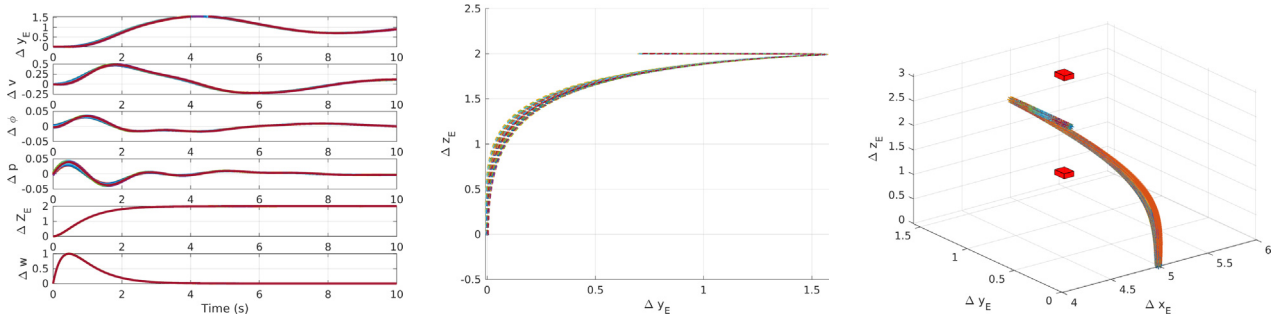


Fig. 3. Monte Carlo simulation (10^5 instances) for the unicycle model with $t = 10$ s.



(a) State Evolution (b) y-z Plane (c) 3D Workspace

Fig. 4. Monte Carlo simulations (10^5 instances) for the 6D quadcopter model with lateral-vertical dynamics for 10 s. Trajectories are visualized in (a) state vs. time plot, (b) y-z Plane, and (c) 3D workspace. The red boxes in (c) are the obstacles. The control laws stabilize the quadrotor around the equilibrium point $x_{\text{eq}} = (1, 0, 0, 0, 2, 0)$.

guarantees, i.e., $\underline{P}_s = 0$. The dimensionality naturally has a major effect on the performance of the SOS optimizer. The dual LP and LP-CEGIS likewise suffer, pertaining to the dimensionality and the partitioning of the state space. Nonetheless, the GD approach is still able to handle the large scale optimization, and can provide a lower safety probability of $\underline{P}_s = 0.560$. While this is not near the statistical safety threshold obtained through simulation ($\tilde{P}_{s,sim} \approx 0.95$), it clearly shows that the proposed PWC-SBF method outperforms the current state-of-the-art SBF methods.

7. Conclusion

In this work, we introduce a formulation for piecewise (PW) stochastic barrier functions (SBFs). We discuss the flexibility they provide at the cost of increased computational cost. To lay the groundwork, we focus on their simplest form, namely, PW constant (PWC) functions. We show that the synthesis of optimal PWC SBFs reduces to a constrained minimax optimization problem that includes bilinear terms. We further show that PWC SBFs can obtain the same or even higher probabilistic safety guarantees than continuous SBFs. We also propose three efficient computational methods for solving the minimax problem by relaxing bilinear terms. This is achieved by (i) dual formulation in form of a linear program (LP), (ii) splitting the problem into two separate iterative LPs, where one synthesizes candidate function and the other provide counter examples, and finally (iii) designing a convex loss function that precisely captures the objective function of the minimax problem, enabling the use of gradient descent method. To show the expressivity of PWC SBFs and scalability of the proposed computational methods, we provide extensive evaluations on a range of benchmark problems. We also compare performance of the proposed methods against state-of-the-art SOS optimization and NBF learning. The results clearly show the efficacy and superiority of PWC SBFs. The PWC SBFs provide a new approach to ensuring safety for stochastic systems.

This study raises several interesting research questions for future investigations. One is how to perform adaptive refinement of the pieces of the PWC SBF to reduce computational overhead. Another interesting question is on the complexity of PW linear (and nonlinear) SBFs and whether they provide more expressivity (better safety guarantees) than PWC SBFs. Finally, an interesting direction is PWC-SBF-based control synthesis.

References

Abate, A., Prandini, M., Lygeros, J., & Sastry, S. (2008). Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, *44*(11), 2724–2734.

Adams, S., Lahijanian, M., & Laurenti, L. (2022). Formal control synthesis for stochastic neural network dynamic models. *IEEE Control Systems Letters*, *6*, 2858–2863.

Ahmadi, A. A., & Majumdar, A. (2014). DSOS and SDSOS optimization: LP and SOCP-based alternatives to sum of squares optimization. In *2014 48th annual conference on information sciences and systems* (pp. 1–5). IEEE.

Alur, R., Bodik, R., Juniwal, G., Martin, M. M., Raghothaman, M., Seshia, S. A., Singh, R., Solar-Lezama, A., Torlak, E., & Udupa, A. (2013). *Syntax-guided synthesis*. IEEE.

Ames, A. D., Coogan, S., Egerstedt, M., Notomista, G., Sreenath, K., & Tabuada, P. (2019). Control barrier functions: Theory and applications. In *2019 18th European control conference* (pp. 3420–3431). IEEE.

Ames, A. D., Grizzle, J. W., & Tabuada, P. (2014). Control barrier function based quadratic programs with application to adaptive cruise control. In *53rd IEEE conference on decision and control* (pp. 6271–6278). IEEE.

Ames, A. D., Xu, X., Grizzle, J. W., & Tabuada, P. (2016). Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, *62*(8), 3861–3876.

Anand, M., Lavaei, A., & Zamani, M. (2022). From small-gain theory to compositional construction of barrier certificates for large-scale stochastic systems. *IEEE Transactions on Automatic Control*, *67*(10), 5638–5645.

Barbosa, H., Reynolds, A., Larraz, D., & Tinelli, C. (2019). Extending enumerative function synthesis via SMT-driven classification. In *2019 formal methods in computer aided design* (pp. 212–220). IEEE.

Ben-Tal, A., & Nemirovski, A. (1998). Robust convex optimization. *Mathematics of Operations Research*, *23*(4), 769–805.

Bertsekas, D. P., & Shreve, S. E. (2004). *Stochastic optimal control: the discrete-time case*: *Vol. 5*, Athena Scientific.

Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Boyd, S., Xiao, L., & Mutapcic, A. (2003). Subgradient methods. In *Lecture notes of ee392o*. Stanford University, Autumn Quarter.

Cauchi, N., Laurenti, L., Lahijanian, M., Abate, A., Kwiatkowska, M., & Cardelli, L. (2019). Efficiency through uncertainty: Scalable formal synthesis for stochastic hybrid systems. In *Proceedings of the 22nd ACM international conference on hybrid systems: computation and control* (pp. 240–251).

Dawson, C., Qin, Z., Gao, S., & Fan, C. (2022). Safe nonlinear control using robust neural lyapunov-barrier functions. In *Conference on robot learning* (pp. 1724–1735). PMLR.

De Luca, A., Oriolo, G., & Vendittelli, M. (2000). Stabilization of the unicycle via dynamic feedback linearization. *IFAC Proceedings Volumes*, *33*(27), 687–692.

Ding, M., Lin, K., Lin, W., & Ding, Z. (2022). A novel counterexample-guided inductive synthesis framework for barrier certificate generation. In *2022 IEEE 33rd international symposium on software reliability engineering* (pp. 263–273). IEEE.

Edwards, A., Peruffo, A., & Abate, A. (2024). Fossil 2.0: Formal certificate synthesis for the verification and control of dynamical models. In *Proceedings of the 27th ACM international conference on hybrid systems: computation and control* (pp. 1–10).

Givan, R., Leach, S., & Dean, T. (2000). Bounded-parameter Markov decision processes. *Artificial Intelligence*, *122*(1–2), 71–109.

Guiochet, J., Machin, M., & Waeselynck, H. (2017). Safety-critical advanced robots: A survey. *Robotics and Autonomous Systems*, *94*, 43–52.

Jagtap, P., Soudjani, S., & Zamani, M. (2020). Formal synthesis of stochastic systems via control barrier certificates. *IEEE Transactions on Automatic Control*, *66*(7), 3097–3110.

Kushner, H. J. (1967). *Stochastic stability and control*: *Technical report*, Brown Univ Providence RI.

Laurenti, L., & Lahijanian, M. (2023). Unifying safety approaches for stochastic systems: From barrier functions to uncertain abstractions via dynamic programming. arXiv preprint arXiv:2310.01802.

Laurenti, L., Lahijanian, M., Abate, A., Cardelli, L., & Kwiatkowska, M. (2020). Formal and efficient synthesis for continuous-time linear stochastic hybrid processes. *IEEE Transactions on Automatic Control*, *66*(1), 17–32.

Lee, H., Kim, H., & Kim, H. J. (2016). Planning and control for collision-free cooperative aerial transportation. *IEEE Transactions on Automation Science and Engineering*, *15*(1), 189–201.

Maluf, D. A., Gawdiak, Y. O., & Bell, D. G. (2005). On space exploration and human error-a paper on reliability and safety. In *Proceedings of the 38th annual Hawaii international conference on system sciences* (p. 79). IEEE.

Mathiesen, F. B., Calvert, S. C., & Laurenti, L. (2022). Safety certification for stochastic systems via neural barrier functions. *IEEE Control Systems Letters*, *7*, 973–978.

Mazouz, R., Muvvala, K., Ratheesh Babu, A., Laurenti, L., & Lahijanian, M. (2022). Safety guarantees for neural network dynamic systems via stochastic barrier functions. *Advances in Neural Information Processing Systems*, *35*, 9672–9686.

Nejati, A., & Zamani, M. (2023). Data-driven synthesis of safety controllers via multiple control barrier certificates. *IEEE Control Systems Letters*, *7*, 2497–2502.

Peruffo, A., Ahmed, D., & Abate, A. (2021). Automated and formal synthesis of neural barrier certificates for dynamical models. In *International conference on tools and algorithms for the construction and analysis of systems* (pp. 370–388). Springer.

Prajna, S. (2006). Barrier certificates for nonlinear model validation. *Automatica*, *42*(1), 117–126.

Prajna, S., & Jadbabaie, A. (2004). Safety verification of hybrid systems using barrier certificates. In *International workshop on hybrid systems: computation and control* (pp. 477–492). Springer.

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

Prajna, S., & Rantzer, A. (2005). On the necessity of barrier certificates. *IFAC Proceedings Volumes*, *38*(1), 526–531.

Santoyo, C., Dutreix, M., & Coogan, S. (2021). A barrier function approach to finite-time stochastic system verification and control. *Automatica*, *125*, Article 109439.

Shalev-Shwartz, S., Shammah, S., & Shashua, A. (2017). On a formal model of safe and scalable self-driving cars. arXiv preprint arXiv:1708.06374.

Skovbekk, J., Laurenti, L., Frew, E., & Lahijanian, M. (2023). Formal abstraction of general stochastic systems via noise partitioning. *IEEE Control Systems Letters*, *7*, 3711–3716.

Zhang, H., Weng, T.-W., Chen, P.-Y., Hsieh, C.-J., & Daniel, L. (2018). Efficient neural network robustness certification with general activation functions. In *Advances in neural information processing systems*.

Žikelić, Đ., Lechner, M., Henzinger, T. A., & Chatterjee, K. (2023). Learning control policies for stochastic systems with reach-avoid guarantees. In *Proceedings of the AAAI conference on artificial intelligence*: *Vol. 37*, (pp. 11926–11935).



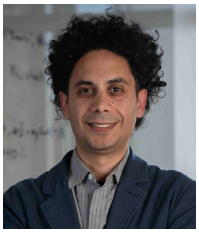
Rayan Mazouz received his B.Sc. and M.S.c degrees in Aerospace Engineering from Delft University of Technology. Currently, he is persuing a Ph.D. at the University of Colorado in Boulder. His research primarily focuses on the formal verification and control synthesis of stochastic dynamical systems.



Frederik Baymler Mathiesen obtained his B.Sc. (2019) and M.Sc. (2021) degrees in Computer Science at Aalborg University, Denmark. He is currently pursuing a Ph.D. at Delft Center for Systems and Control, Delft University of Technology. The primary focus of his research concerns formal methods and machine learning for stochastic dynamical systems.



Luca Laurenti is a tenure track assistant professor at the Delft Center for Systems and Control at TU Delft and head of the Research on Robust and Intelligent Autonomous Systems Lab at AI4I. He received his Ph.D. from the Department of Computer Science, University of Oxford (UK), where he was a member of the Trinity College. Luca has a background in stochastic systems, control theory, formal methods, and artificial intelligence. His research work focuses on developing data-driven systems that are provably robust to interactions with a dynamic and uncertain world.



Morteza Lahijanian is an Associate Professor in the Aerospace Engineering Sciences department, an affiliated faculty at the Computer Science department and Robotics Program, and the director of the Assured, Reliable, and Interactive Autonomous (ARIA) Systems group at the University of Colorado Boulder.

His research interests span the areas of control theory, stochastic hybrid systems, formal methods, machine learning, and game theory with applications in robotics, particularly, motion planning, strategy synthesis, model checking, and human–robot interaction.

Dr. Lahijanian's awards include Ella Mae Lawrence R. Quarles Physical Science Achievement Award, Jack White Engineering Physics Award, outstanding junior faculty award, and NSF GK-12 Fellowship.