Control Engineering Practice 175 (2026) 107060

Contents lists available at ScienceDirect

Control Engineering Practice

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

SC-BO: Stability-certified Bayesian optimization for fixed-order LTI controllers

Marta Manzoni *, Marco Lovera

Dipartimento di Scienze e Tecnologie Aerospaziali, Politecnico di Milano, Via La Masa 34, Milano, 20156, Italy

ARTICLE INFO

Keywords:
Bayesian optimization
Gaussian processes
Fixed-order controller synthesis
Robust stability
Disk margins
LTI controllers

ABSTRACT

Improving closed-loop control performance from data is challenging when stability must be guaranteed throughout the optimization process. Bayesian optimization provides a sample-efficient framework for this setting by selecting informative evaluations of the closed-loop system. Safe Bayesian Optimization (SafeBO) methods extend this framework by incorporating safety constraints during learning, but they do not directly provide formal certificates of closed-loop robust stability. This limits their use in applications where controllers must satisfy explicit stability-margin requirements. In this paper, we introduce Stability-Certified Bayesian Optimization (SC-BO), a framework for fixed-order Linear Time-Invariant (LTI) controllers that enforces robust stability within the optimization loop through disk-margin certification. The method alternates between two steps: (i) a Bayesian optimization step that proposes candidate controller parameters to improve closed-loop performance, and (ii) a robust-stability enforcement step, in which the proposed controller is certified against a prescribed disk-margin condition and, when necessary, projected onto the certified set by solving a convex semidefinite program. Numerical results on a high-fidelity quadrotor simulator show that SC-BO improves closed-loop performance while ensuring that all evaluated controllers satisfy the prescribed robust-stability requirement. The comparison with representative SafeBO baselines highlights the role of the certification/projection layer: in the considered application, SC-BO maintains competitive performance while preventing stability violations by construction.

1. Introduction

The design of reliable and high-performance control systems remains a central challenge in modern engineering, where achieving the desired trade-off between robustness and performance requires careful controller design.

Modern control theory offers a broad range of model-based approaches for controller design, including H_∞ control (Skogestad & Postlethwaite, 2005), Linear Quadratic Regulation (Anderson & Moore, 2007), and Model Predictive Control (Rawlings et al., 2020). Their effectiveness, however, relies on accurate mathematical models of the plant, typically derived from first principles or identified from data. Although reliable identification techniques exist (Ljung, 1998; Pintelon & Schoukens, 2012), they often involve simplifying assumptions that fail to capture the complexity of real systems, potentially leading to suboptimal controller performance.

These limitations have motivated data-driven controller design methods that synthesize controllers directly from experimental data, without relying solely on an explicit plant model. Existing approaches include reinforcement learning (Mate et al., 2023; McClement et al., 2022), evolutionary methods (Joseph et al., 2022; Rodríguez-Molina et al., 2020), and genetic algorithms (Davidor, 1991). Although effective for complex, nonlinear, and high-dimensional problems, these methods often require extensive trial-and-error and many experimental evaluations, limiting their use when experiments are costly, time-consuming, or safety-critical.

Other data-driven control design techniques include inversion-based control (Novara & Formentin, 2017), virtual reference feedback tuning (Campi et al., 2002), and correlation-based tuning (Van Heusden et al., 2011). While effective in several applications, these methods may struggle to explicitly handle input and output constraints, and their performance can be sensitive to the choice of reference models or design parameters, particularly when prior system knowledge is limited.

In recent years, Bayesian Optimization (BO) (Mockus, 2005) has emerged as a sample-efficient framework for improving closed-loop performance from experimental data, particularly when system evaluations are costly. However, its use in safety-critical systems remains challenging, since candidate controllers proposed during the optimization process may violate operational constraints or compromise closed-loop stability.

* Corresponding author.
E-mail addresses: marta.manzoni@polimi.it (M. Manzoni), marco.lovera@polimi.it (M. Lovera).

https://doi.org/10.1016/j.conengprac.2026.107060
Received 31 December 2025; Received in revised form 28 April 2026; Accepted 8 May 2026
Available online 22 May 2026

To mitigate these risks, SafeBO methods incorporate probabilistic safety constraints into the optimization process. A representative approach is SafeOpt (Sui et al., 2015), with subsequent extensions addressing multiple safety constraints and improved computational efficiency (Berkenkamp et al., 2023; Turchetta et al., 2019). These methods have also been demonstrated on real systems, including safe quadrotor controller tuning (Berkenkamp et al., 2016). However, SafeOpt requires an initial controller known to satisfy all safety constraints and restricts exploration to regions that are safely reachable from this point, which may lead to suboptimal local solutions. More recent methods, such as GoSafe (Baumann et al., 2021) and GoSafeOpt (Sukhija et al., 2023), alleviate this limitation by enabling exploration of disconnected safe regions, but still require an initial safe point.

Despite these advances, SafeBO methods cannot provide formal guarantees on closed-loop stability. Safety is typically treated as an unknown constraint learned from data, whereas closed-loop stability is a global property of the system that cannot be expressed as a simple pointwise constraint on controller parameters.

1.1. Related work

Several works have addressed stability in practical BO applications through heuristic mechanisms. For example, Khosravi et al. (2020) restrict the controller gains to predefined ranges known to preserve stability. This approach, however, assumes prior knowledge of the stable region, which is not always available, and determining it through numerical simulations can be computationally demanding. Other methods infer potentially unstable behavior directly from measured responses. In Khosravi et al. (2019), SafeBO is applied to a heat-pump controller using a safety function defined on the system output, where large output amplitudes are treated as unsafe indicators. Similarly, Khosravi et al. (2022) tune the gains of a PID cascade controller by detecting potential instabilities from the frequency content of the position error through Fast Fourier Transform analysis. This allows critical gain values to be identified experimentally and was validated on a linear axis drive. Related ideas also appear in Rodríguez-Molina et al. (2020), where multi-objective BO is used to optimize performance and robustness jointly, with gain and delay margins estimated from experimental data to approximate the corresponding Pareto front.

These approaches are attractive because they are fully model-free and can be applied when accurate modeling is difficult or costly. However, they address stability only indirectly and do not provide formal guarantees on closed-loop robustness. This limitation is particularly relevant in certification-oriented domains, such as aerospace, where controllers are required to satisfy explicit robustness requirements, commonly expressed through stability margins such as gain and phase margins (Federal Aviation Administration, 2023). These requirements cannot be certified from experimental observations alone: closed-loop stability is a property of the plant–controller interconnection and therefore requires a model-based certificate.

Conversely, robust control methods, such as H_∞ synthesis and related LMI-based approaches, provide systematic tools for enforcing stability and robustness requirements from a nominal model and an uncertainty description (Skogestad & Postlethwaite, 2005; Zhou et al., 1996). Their performance, however, depends on the accuracy of the model used for synthesis. In particular, if the model is adequate for robustness certification but not sufficiently accurate for direct performance optimization, purely model-based synthesis can be conservative from a performance viewpoint. Recent results have shown that a robust controller can first be synthesized with H_∞ and then further refined through BO, leading to improved performance (Manzoni & Lovera, 2025). This suggests that the combination of model-based robustness certification and data-driven performance optimization can be advantageous when demanding performance targets must be met without sacrificing certified stability. This motivates the need for a BO framework that combines data-driven performance optimization with model-based robustness certification.

To the best of our knowledge, a BO framework that ensures all evaluated controllers satisfy prescribed robustness margins while optimizing closed-loop performance from data is still missing. This gap motivates the stability-certified BO framework proposed in this work.

1.2. Contributions

In this paper, we propose Stability-Certified Bayesian Optimization (SC-BO), a framework for fixed-order LTI controller synthesis that combines data-driven performance optimization with model-based robust-stability certification. The method targets settings in which robust stability can be certified using a simplified nominal model and a disk-margin uncertainty description, while closed-loop performance is more accurately evaluated on a higher-fidelity simulator or on the physical system. The overall workflow of SC-BO is illustrated in Fig. 1.
The main contributions are:

- a BO-based framework in which every evaluated controller is certified to satisfy a prescribed disk-margin robust-stability requirement;
- the integration of a certification/projection layer into the BO loop, so that uncertified proposals are projected before evaluation;
- validation on a high-fidelity quadrotor simulator;
- comparison with representative SafeBO methods, including SafeOpt and GoSafeOpt, together with analyses of computational cost, penalty selection, and model-sensitivity effects.

Key advantages of the proposed framework include:

- **Certified robust stability**: every evaluated controller satisfies the prescribed disk-margin condition.
- **No initial stable controller required**: the initial controller is also certified or projected before evaluation.
- **Decoupled exploration and certification**: BO can propose candidates over the full parameter domain, while the certification/projection layer ensures that only certified controllers are evaluated.

The paper is organized as follows. Section 2, formalizes the problem addressed in this work. Section 3 derives a disk-margin-based condition for certifying robust stability. Section 4 introduces the fundamentals of BO and SafeBO, while Section 5 introduces the proposed SC-BO algorithm. Numerical results are reported in Sections 6, and 7 concludes the paper.

1.3. Notation

We denote by $\mathbb{R}$ ($\mathbb{R}_{>0}$, $\mathbb{R}_{\geq 0}$) the set of (positive, nonnegative) real numbers. $\mathbb{R}^n$ denotes the n-dimensional real coordinate space and $\mathbb{R}^{m\times n}$ the set of $m \times n$ real matrices. Let $\mathcal{L}_2^n$ be the space of square-integrable functions from $\mathbb{R}_{\geq 0}$ to $\mathbb{R}^n$, and let $\mathcal{L}_{2e}^n$ denote the space of functions from $\mathbb{R}_{\geq 0}$ to $\mathbb{R}^n$ that are square-integrable on every finite interval $[0, T]$ for all $T > 0$.

2. Problem setup

In this section, we formalize the control problem considered in this work. The goal is to automatically synthesize a fixed-order LTI controller that improves closed-loop performance using data from a high-fidelity simulator or from experiments, while ensuring robust closed-loop stability with respect to a prescribed model uncertainty description.

We assume that a nominal model of the plant is available, obtained either from first-principles modeling or from system identification. This model, together with a disk-margin uncertainty description, is used for robust-stability certification. The plant used to evaluate closed-loop performance may differ from this nominal model because of unmodeled dynamics, parameter variations, or higher-order effects. To account for this mismatch in the stability certification step, we introduce an uncertain plant model based on disk margins. We then define the controller structure and formulate the resulting constrained optimization problem.

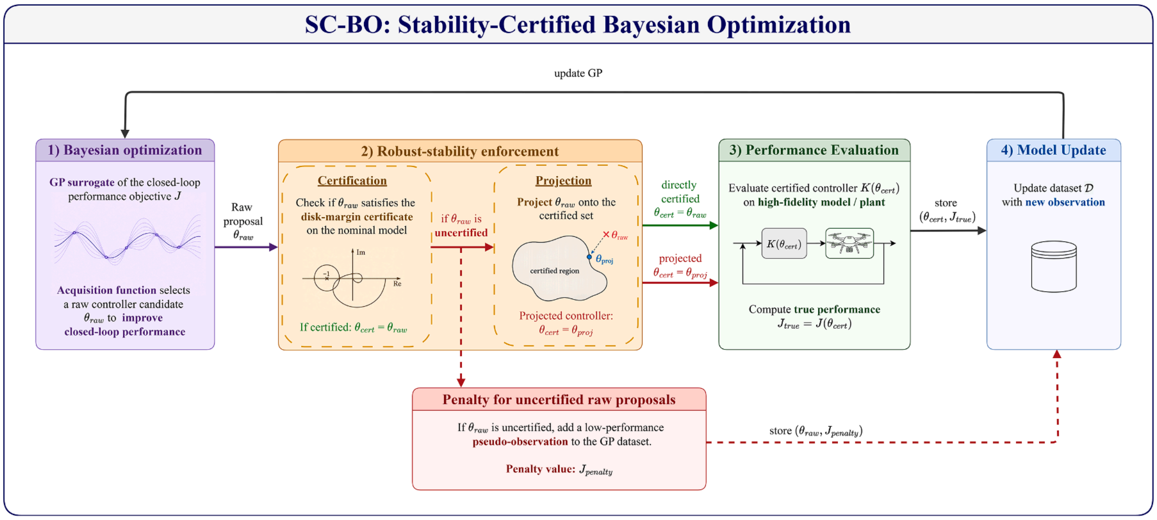

Fig. 1. Illustrative workflow of the proposed SC-BO method. Bayesian optimization first proposes a raw controller candidate. The candidate is then checked against the prescribed disk-margin robust-stability certificate using the nominal certification model. If the certificate is satisfied, the controller is evaluated directly; otherwise, it is projected onto the certified set before evaluation. The high-fidelity simulator or plant is used only to compute the closed-loop performance of the certified controller. When the raw proposal is uncertified, a penalized pseudo-observation is also added to the Gaussian Process dataset to discourage repeated sampling of uncertified regions.

2.1. Uncertain plant model

Consider a nominal LTI plant P described by

$$\begin{bmatrix} \dot{x}_p(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_p & B_p \\ C_p & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ u(t) \end{bmatrix}, \tag{1}$$

where $x_p(t) \in \mathbb{R}^{n_p}$ is the plant state, $y(t) \in \mathbb{R}^{n_y}$ the measured output, and $u(t) \in \mathbb{R}^{n_u}$ the control input at time t. Although the plant model used for certification is LTI, this setting is consistent with flight-control practice, where controllers are commonly designed and certified around linearized operating conditions. Nevertheless, the underlying certification framework also extends to plants with nonlinearities described by integral quadratic constraints (Junnarkar et al., 2024).

Classical gain and phase margins are widely used robustness measures in control engineering and are aligned with certification-oriented aerospace practice, where robustness requirements are commonly expressed through stability margins (Federal Aviation Administration, 2023). However, gain and phase margins quantify perturbations separately, whereas simultaneous gain and phase variations may occur in practice. A system with large individual gain and phase margins may still lose stability under combined variations (Doyle, 1996). Disk margins (Seiler et al., 2020) address this limitation by providing a unified robustness measure that is directly related to classical gain and phase margins while accounting for simultaneous gain and phase perturbations. They also provide a broad uncertainty parametrization, not tied to a single source of modeling error: other uncertainty classes, including parametric uncertainty, can often be recast, or conservatively overbounded, in disk-margin form.

In this work, we augment the nominal plant with an uncertain input block and assess robust stability over all admissible perturbations associated with a disk margin $D(\alpha, \sigma)$:

$$D(\alpha,\sigma) = \left\{ \frac{1 + \frac{1-\sigma}{2}\delta}{1 - \frac{1+\sigma}{2}\delta} : \delta \in \mathbb{C},\ |\delta| < \alpha \right\},$$

where $\alpha \in \mathbb{R}_{\geq 0}$ determines the disk radius and $\sigma \in \mathbb{R}$ its skewness. This disk characterizes robustness to channel-wise input uncertainties whose frequency response lies within the complex disk. Equivalently, disk margins can be represented through an input perturbation $\Delta_p(s)$ satisfying $\|\Delta_p\|_\infty < \alpha$:

$$H_{\Delta_p}(s) = \left(I + \frac{1-\sigma}{2}\Delta_p(s) \right)\left(I + \frac{1+\sigma}{2}\Delta_p(s) \right)^{-1}.$$

The nominal plant is recovered for $\Delta_p(s) = 0$, which gives $H_{\Delta_p}(s) = I$.

In block-diagram form, shown in Fig. 2, the uncertain plant $\tilde{P}_{\Delta_p}$ from input $\tilde{u}$ to output y is

$$\begin{aligned} y &= Pu, & u &= v_p + \frac{1-\sigma}{2} w_p, \\ v_p &= \tilde{u} + \frac{1+\sigma}{2} w_p, & w_p &= \Delta_p v_p, \end{aligned} \tag{2}$$

where $v_p \in \mathcal{L}_{2e}^{n_u}$ is the input to the uncertainty Δ_p, and $w_p \in \mathcal{L}_{2e}^{n_u}$ is its output.

This formulation naturally extends to non-LTI uncertainties by considering Δ_p to be $\mathcal{L}_2$ norm-bounded by α, defining the uncertain plant family

$$\mathcal{M}_\alpha = \{ \tilde{P}_{\Delta_p} \mid \|\Delta_p\|_{\mathcal{L}_2} < \alpha \}.$$

2.2. Controller model

The uncertain plant is regulated by a fixed-order LTI controller C of the form

$$\begin{bmatrix} \dot{x}_c(t) \\ \tilde{u}(t) \end{bmatrix} = \begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix} \begin{bmatrix} x_c(t) \\ y(t) \end{bmatrix}, \tag{3}$$

where $x_c(t) \in \mathbb{R}^{n_c}$ is the controller state, $y(t) \in \mathbb{R}^{n_y}$ the measured plant output, and $\tilde{u}(t) \in \mathbb{R}^{n_u}$ the control input applied to the uncertain plant. The controller order n_c is fixed a priori, while the state-space matrices (A_c, B_c, C_c, D_c) are free design variables.

The controller C is said to achieve the disk margin $D(\alpha, \sigma)$ if it stabilizes all plants in the uncertainty set $\mathcal{M}_\alpha$.

2.3. Problem statement

The goal is to construct a controller C in (3) by iteratively using closed-loop performance data from a high-fidelity simulator or from experiments, while ensuring that every controller actually evaluated satisfies the prescribed robust-stability requirement. To this end, the controller is parameterized by a vector θ collecting all free entries of its state-space realization, and we write $C(\theta)$. Formally, $\theta \in \Theta$ is defined as

$$\theta \triangleq \mathrm{vec}\left(\begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix}\right), \tag{4}$$

where vec(·) denotes column-wise vectorization.

The resulting problem is formulated as:

$$\begin{aligned} &\max_{\theta\in\Theta} \quad J(\theta) \\ &\text{subject to} \quad F(\tilde{P}_{\Delta_p}, C(\theta)) \text{ is stable for all } \tilde{P}_{\Delta_p} \in \mathcal{M}_\alpha, \end{aligned} \tag{5}$$

where $F(\tilde{P}_{\Delta_p}, C(\theta))$ denotes the closed-loop interconnection between the uncertain plant and the controller.

The scalar function $J(\theta)$ measures closed-loop performance over a finite horizon $[0, T]$. It may encode standard control objectives such as tracking accuracy, control effort, energy consumption, or weighted combinations thereof. We write

$$J(\theta) = \int_0^T r\big(x_p(t), u(t)\big)\, dt, \tag{6}$$

where $r(\cdot,\cdot)$ is chosen so that larger values of $J(\theta)$ correspond to better closed-loop performance. In the setting considered here, $J(\theta)$ is treated as a black-box function: it is evaluated from closed-loop simulations or experiments, rather than optimized from the nominal model used for certification.

3. Disk margin-based stability certification

This section derives a certification condition to verify whether a given LTI controller of the form (3) satisfies the prescribed disk margin $D(\alpha, \sigma)$ for the nominal plant P. The derivation builds on the framework introduced by Junnarkar et al. (2025), originally developed for neural-network controllers, and specializes it to the LTI controller setting considered here. This yields a formulation that is more directly aligned with practical control-design applications, such as aerospace and flight control, where LTI controllers remain widely used. To this end, the uncertain closed-loop system is rewritten in a standard LTI–uncertainty interconnection form, which enables robust-stability certification through quadratic constraints.

3.1. Reformulation of the uncertain plant

The uncertain plant $\tilde{P}_{\Delta_p}$ introduced in Section 2.1 can be rewritten as a nominal LTI system interconnected with an uncertainty block Δ_p, consistent with the standard robust control framework. Starting from the relations in (2), the nominal dynamics can be reorganized as a single state-space model with inputs $\tilde{u}$ and w_p, and outputs y and v_p. Using the definitions $u = v_p + \frac{1-\sigma}{2} w_p$ and $v_p = \tilde{u} + \frac{1+\sigma}{2} w_p$ one obtains $u = w_p + \tilde{u}$, which can be substituted into the nominal plant Eq. (1) to yield

$$\begin{bmatrix} \dot{x}_p(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_p & B_p & B_p \\ C_p & 0 & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ w_p(t) \\ \tilde{u} \end{bmatrix}.$$

Explicitly including v_p as an output using its definition from (2) results in

$$\begin{aligned} &\begin{bmatrix} \dot{x}_p(t) \\ v_p(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_p & B_p & B_p \\ 0 & \frac{1+\sigma}{2} I & I \\ C_p & 0 & 0 \end{bmatrix} \begin{bmatrix} x_p(t) \\ w_p(t) \\ \tilde{u}(t) \end{bmatrix}, \\ &w_p(t) = \Delta_p v_p(t). \end{aligned} \tag{7}$$

The interconnection is assumed to be well-posed, meaning that for any input $\tilde{u} \in \mathcal{L}_{2e}^{n_u}$ and any initial condition $x_p(0) = x_{p_0}$, there exists a unique trajectory $(x_p, v_p, w_p, y) \in \mathcal{L}_{2e}$, satisfying (7).

This reformulation allows the complete closed-loop system to be represented as an LTI system connected to an uncertainty block, as shown in Fig. 3. Such a structure allows the use of standard robust control tools to certify disk-margin stability.

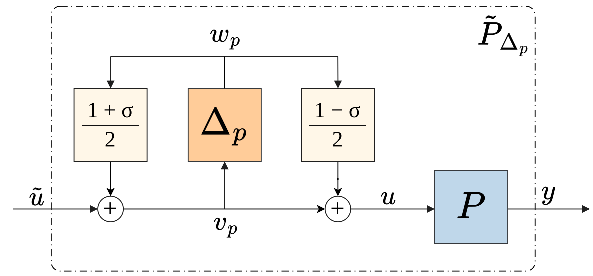

Fig. 2. Block diagram of the uncertain plant model. The nominal plant P is augmented with disk-margin uncertainty at the plant input.

3.2. Integral quadratic constraints for Δ_p

To describe the uncertainty block Δ_p in a form suitable for stability analysis, the relationship between its inputs and outputs is expressed through Integral Quadratic Constraints (IQCs) (Megretski & Rantzer, 2002; Seiler, 2014).

In this work, the uncertainty block Δ_p is modeled as a diagonal operator with $\mathcal{L}_2$ gain bounded by α. Under this assumption, Δ_p satisfies a static IQC of the form

$$\int_0^T \begin{bmatrix} v_p(t) \\ w_p(t) \end{bmatrix}^\top \underbrace{\begin{bmatrix} \alpha^2 \Lambda_p & 0 \\ 0 & -\Lambda_p \end{bmatrix}}_{M} \begin{bmatrix} v_p(t) \\ w_p(t) \end{bmatrix} dt \geq 0, \tag{8}$$

for all time horizons $T \geq 0$ and for all diagonal, positive definite matrices $\Lambda_p \in \mathbb{R}^{n_u \times n_u}$ ($\Lambda_p \succeq 0$). Here, $v_p \in \mathcal{L}_{2e}^{n_u}$ denotes the input signal to Δ_p, while $w_p = \Delta_p(v_p)$ is the corresponding output.

This formulation corresponds to a static IQC, which is sufficient to describe norm-bounded uncertainties. More general cases, such as dynamic uncertainties, can also be represented within the same framework by introducing suitable dynamic filters (Megretski & Rantzer, 2002).

3.3. Robust stability certification

The closed-loop system is obtained by interconnecting the uncertain plant $\tilde{P}_{\Delta_p}$, defined in (7), with the LTI controller in (3). Let the stacked state collecting the plant and controller state be $x(t) = [x_p(t)^\top \; x_c(t)^\top]^\top \in \mathbb{R}^{n_p+n_c}$ and denote $w_p, v_p \in \mathbb{R}^{n_u}$ the uncertainty input and output signals, as introduced for the plant. The closed-loop dynamics can be written compactly as

$$\begin{aligned} &\begin{bmatrix} \dot{x}(t) \\ v_p(t) \end{bmatrix} = \begin{bmatrix} A_{cl} & B_{cl} \\ C_{cl} & D_{cl} \end{bmatrix} \begin{bmatrix} x(t) \\ w_p(t) \end{bmatrix}, \\ &w_p(t) = \Delta_p v_p(t). \end{aligned} \tag{9}$$

The closed-loop matrices, expressed in terms of the plant and controller matrices, are affine in the controller parameters and given explicitly by

$$A_{cl} = \begin{bmatrix} A_p + B_p D_c C_p & B_p C_c \\ B_c C_p & A_c \end{bmatrix}, \qquad B_{cl} = \begin{bmatrix} B_p \\ 0 \end{bmatrix},$$

$$C_{cl} = \begin{bmatrix} D_c C_p & C_c \end{bmatrix}, \qquad D_{cl} = \frac{1+\sigma}{2} I.$$

This formulation represents an uncertain LTI system where the uncertainty Δ_p satisfies a static IQC. Consequently, stability can be certified via the following lemma, which provides a matrix inequality condition in terms of the closed-loop parameters and the matrix M.

Lemma 1. *Let the nominal plant be given as in (1) and consider the disk margin parameterized by (α, σ). The controller defined in (3) satisfies the specified disk margin if there exist a matrix $P \in \mathbb{R}^{(n_p+n_c)\times(n_p+n_c)}$ with $P \succ 0$ and a diagonal $\Lambda_p \in \mathbb{R}^{n_u \times n_u}$ with $\Lambda_p \succeq 0$, such that*

$$\begin{bmatrix} A_{cl}^\top P + P A_{cl} & P B_{cl} \\ B_{cl}^\top P & 0 \end{bmatrix} + \begin{bmatrix} C_{cl} & D_{cl} \\ 0 & I \end{bmatrix}^\top M \begin{bmatrix} C_{cl} & D_{cl} \\ 0 & I \end{bmatrix} \preceq 0, \tag{10}$$

where M encodes the static IQC satisfied by the diagonal uncertainty Δ_p, as defined in (8).

Proof. Since the closed-loop system can be represented as an LTI interconnection with an uncertainty characterized by a static IQC, the result follows directly by applying Lemma 1 of Junnarkar et al. (2024). □

Remark 1. The inequality in (10) is quadratic in the controller matrices due to the term involving M. This issue will be addressed in Section 5.

4. Fundamentals of Bayesian optimisation

This section reviews the Bayesian optimization framework used in the proposed method. In particular, it introduces Gaussian Process (GP) modeling of unknown performance functions, standard BO, and its safety-constrained extension (SafeBO).

4.1. Gaussian processes

GPs offer a flexible, non-parametric probabilistic framework for modeling unknown functions (Williams & Rasmussen, 2006). They define a probability distribution over functions such that, for any finite collection of input points, the corresponding function values follow a multivariate Gaussian distribution. A GP is fully specified by two components: a mean function $\mu(\theta) : \Theta \to \mathbb{R}$ and a positive-definite covariance (kernel) function $k(\theta, \theta') : \Theta \times \Theta \to \mathbb{R}$, which encodes the similarity or correlation between inputs. Denoting the unknown function of interest as $J(\theta) : \Theta \to \mathbb{R}$, with $\Theta \subseteq \mathbb{R}^{n_\theta}$ the n_θ-dimensional input space, the GP prior is expressed as $J(\theta) \sim \mathcal{GP}(\mu(\theta), k(\theta, \theta'))$, where $\mu(\theta) = \mathbb{E}[J(\theta)]$ is the prior mean and $k(\theta, \theta') = \mathbb{E}[(J(\theta) - \mu(\theta))(J(\theta') - \mu(\theta'))]$ is the prior variance.

When noisy measurements $\hat{J}(\theta_i) = J(\theta_i) + \nu$, with $\nu \sim \mathcal{N}(0, \sigma_\nu^2)$, are available for $i = 1, \dots, n$, the GP framework provides a closed-form expression for the posterior distribution over functions. Let the dataset of observations be $\mathcal{D}_n = \{(\theta_i, \hat{J}(\theta_i))\}_{i=1}^n$. The posterior mean and covariance at a new input θ are then given by

$$\begin{aligned} \mu_n(\theta) &= \mu(\theta) + k_n(\theta)^\top (K_n + \sigma_\nu^2 I_n)^{-1} \hat{J}_n, \\ \Sigma_n(\theta, \theta') &= k(\theta, \theta') - k_n(\theta)^\top (K_n + \sigma_\nu^2 I_n)^{-1} k_n(\theta'). \end{aligned} \tag{11}$$

Here, $k_n(\theta) = [k(\theta, \theta_1), \dots, k(\theta, \theta_n)]^\top$ gives the covariances between the new input and each of the observed points. $\hat{J}_n = [\hat{J}(\theta_1), \dots, \hat{J}(\theta_n)]^\top$ contains the observations, $K_n \in \mathbb{R}^{n \times n}$ is the kernel matrix with entries $k(\theta_i, \theta_j)$, and I_n is the n-dimensional identity matrix.

4.2. Bayesian optimization

BO is a probabilistic framework for efficient global optimization of an unknown function $J(\theta)$, especially when evaluations are costly, such as in physical experiments or high-fidelity simulations (Mockus, 2005; Shahriari et al., 2015). The goal is to maximize $J(\theta)$ while minimizing the number of evaluations, which can be formally written as

$$\max_{\theta \in \Theta} J(\theta). \tag{12}$$

BO constructs a probabilistic surrogate, usually a GP, to model the objective function and quantify uncertainty. This surrogate informs the choice of new evaluation points via an acquisition function that trades off exploration and exploitation. At iteration n, the next input is selected as:

$$\theta_n = \arg\max_{\theta \in \Theta} \alpha_n(\theta). \tag{13}$$

Commonly used acquisition functions include Expected Improvement (EI) (Mockus, 2005), Probability of Improvement (PI), and GP-Upper Confidence Bound (GP-UCB) (Srinivas et al., 2012).

Once $J(\theta_n)$ is observed, the GP posterior is updated, and the iterative process continues until a convergence criterion is met or until a maximum number of iterations $N_{\max}$ is reached.

4.2.1. Safe Bayesian optimization

SafeBO (Sui et al., 2015) extends standard BO to problems in which safety constraints must be respected during the optimization process. The objective is to maximize a performance function $J(\theta)$ subject to unknown safety constraints $g_i(\theta) : \Theta \to \mathbb{R}$, $i = 1, \dots, m$. This leads to the constrained optimization problem

$$\max_{\theta \in \Theta} J(\theta) \quad \text{subject to } g_i(\theta) \geq 0, \quad i = 1, \dots, m. \tag{14}$$

Both the objective and the safety functions are accessible only through noisy evaluations and are modeled using probabilistic surrogates, typically GPs. Querying the system at θ yields observations $\hat{J}(\theta) = J(\theta) + \nu$ and $\hat{g}_i(\theta) = g_i(\theta) + \nu_i$, where ν and ν_i denote measurement noise.

To ensure safety during optimization, evaluations are restricted to parameter values that satisfy the safety constraints with high confidence according to the surrogate models. SafeBO methods therefore require an initial safe set $S_0 \subseteq \Theta$, known a priori to satisfy all constraints. Starting from S_0, the algorithm selects new evaluation points that are predicted to remain safe while improving performance and, when possible, expanding the region that can be explored safely. Different SafeBO methods differ in how this safe exploration is carried out, for example in the way they explore or connect safe regions, but they all rely on the same basic principle of learning safety constraints from data while restricting evaluations to points that are safe with high probability.

5. Stability-certified Bayesian optimization for fixed-order LTI controllers

In this section, we introduce SC-BO, the proposed framework for improving the closed-loop performance of fixed-order LTI controllers while enforcing a prescribed disk-margin robust-stability requirement. The key idea is to separate the data-driven performance optimization from the model-based stability certification. Performance is treated as a black-box quantity evaluated through closed-loop simulations or experiments, whereas robust stability is verified using the nominal model and the associated disk-margin uncertainty description. The algorithm follows an iterative two-step loop, alternating between:

- a *performance improvement step*, in which BO proposes controller parameters aimed at improving the closed-loop performance metric;
- a *robust-stability enforcement step*, in which the proposed controller is checked against the prescribed disk-margin requirement and, if needed, projected via a convex semidefinite program onto the set of controllers for which this requirement can be certified.

This two-step structure differs from standard constrained BO, where constraints are usually modeled as unknown scalar functions learned from data. Here, the robust stability requirement is not learned from experimental observations; it is checked through a model-based certification problem. Therefore, BO is used only to guide performance improvement, while the certification/projection layer enforces robust stability at every evaluation.

The remainder of this section presents the ingredients of SC-BO. First, the projection problem is formulated from the robust-stability condition in Lemma 1. Then, a change of variables is introduced to obtain a tractable convex reformulation. Finally, these components are combined into the complete SC-BO algorithm. The projection problem, the change of variables, and the resulting convex reformulation follow the projection- based framework of Junnarkar et al. (2025), specialized here to the fixed-order LTI controller setting.

5.1. Projection problem

The projection problem is formulated from the robust stability condition introduced in Lemma 1. Given a candidate controller θ' proposed by the BO step, the goal is to find a nearby controller realization $\theta = \{A_c, B_c, C_c, D_c\}$ that satisfies the prescribed disk-margin stability condition. This leads to the projection problem

$$\begin{aligned} \min_{\theta, P, \Lambda_p} \quad & \|\theta' - \theta\| \\ \text{s.t.} \quad & P \succ 0 \\ & \Lambda_p \succeq 0, \quad \Lambda_p \text{ diagonal} \\ & \mathcal{L}(A_{cl}, B_{cl}, C_{cl}, D_{cl}, P, M, \Lambda_p) \preceq 0 \text{ for } \theta. \end{aligned} \tag{15}$$

Here, $\mathcal{L}(\cdot)$ denotes the matrix inequality in (10) associated with the disk-margin certification condition.

When the controller matrices $\theta = \{A_c, B_c, C_c, D_c\}$ are fixed (*e.g.*, when verifying robust stability for a given controller), this condition is an Linear Matrix Inequality (LMI) in the certificate variables P and Λ_p, and can therefore be used to verify robust stability of a given controller. By contrast, when the controller matrices are also treated as decision variables, the problem becomes non-convex because of bilinear couplings between the θ and P (*e.g.*, terms like $A_c^\top P$) and quadratic terms involving θ and Λ_p. As a result, the projection problem in (15) is not directly computationally tractable. To recover a computationally tractable projection step, a common approach is to fix P and Λ_p to their most recent feasible values and perform the projection over the corresponding certified controller set:

$$\Theta(P, \Lambda_p) \triangleq \{\theta \mid \mathcal{L}(A_{cl}, B_{cl}, C_{cl}, D_{cl}, P, M, \Lambda_p) \preceq 0\}.$$

In practice, this procedure is often used as part of a successive convexification scheme, where P and Λ_p are updated after each projection step to expand the stabilizing set.

5.2. Convex reformulation

To address the non-convex coupling between the controller parameters θ and the storage matrix P, a change of variables can be applied, following the procedure presented in Junnarkar et al. (2024). This transformation builds on the approach originally introduced in Scherer et al. (2002) for $\mathcal{H}_\infty$ control.

This transformation reformulates the robust-stability condition in Lemma 1 into an equivalent matrix inequality that is jointly convex in a new set of variables $\hat{\theta} = \{S, R, N_A\}$ and P, while keeping Λ_p fixed during the projection step. This leads to the following equivalent formulation.

Lemma 2. *Consider the nominal plant defined in (1) and a desired disk margin $D(\alpha, \sigma)$. Let $\Lambda_p = \tilde{L}_{\Delta_p}^\top \tilde{L}_{\Delta_p}$, where $\Lambda_p \succeq 0$ and diagonal. If there exist $\hat{\theta} = \{S, R, N_A\}$ with $S \succ 0$, $R \succ 0$, and satisfying $\begin{bmatrix} R & I \\ I & S \end{bmatrix} \succ 0$ such that condition (16) holds, then there exists a corresponding controller parameterization $\theta = \{A_c, B_c, C_c, D_c\}$ that can be reconstructed from $\hat{\theta}$ and guarantees that the closed-loop system satisfies the prescribed disk margin.*

$$\left[\begin{array}{cc|c|c} \mathrm{He}(A_p R + B_p N_{A21}) & A_p + B_p N_{A22} C_p + N_{A11}^\top & B_p & \alpha N_{A21}^\top \tilde{L}_{\Delta_p}^\top \\ N_{A11} + A_p^\top + C_p^\top N_{A22}^\top B_p^\top & \mathrm{He}(S A_p + N_{A12} C_p) & S B_p & \alpha C_p^\top N_{A22}^\top \tilde{L}_{\Delta_p}^\top \\ \hline B_p^\top & B_p^\top S & -\Lambda_p & \frac{\alpha(1+\sigma)}{2} \tilde{L}_{\Delta_p}^\top \\ \hline \alpha \tilde{L}_{\Delta_p} N_{A21} & \alpha \tilde{L}_{\Delta_p} N_{A22} C_p & \frac{\alpha(1+\sigma)}{2} \tilde{L}_{\Delta_p} & -I \end{array}\right] \preceq 0. \tag{16}$$

The set of parameters $\hat{\theta}$ that satisfies the conditions of Lemma 2 for a fixed Λ_p is denoted by

$$\hat{\Theta}(\Lambda_p) \triangleq \{\, \hat{\theta} \mid \text{Lemma 2 holds for } \Lambda_p \,\}.$$

Each $\hat{\theta} \in \hat{\Theta}(\Lambda_p)$ defines, through a specific change of variables, an equivalent controller parameterization θ. The relationship between the auxiliary variables (S, R, N_A) and the actual controller matrices (A_c, B_c, C_c, D_c) is established by the following transformation:

$$N_A = \begin{bmatrix} S A_p R & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} U & S B_p \\ 0 & I \end{bmatrix} \begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix} \begin{bmatrix} V^\top & 0 \\ C_p R & I \end{bmatrix},$$

where U and V are matrices satisfying $V U^\top = I - RS$ (for instance, they can be obtained via a singular value decomposition).

Given this transformation, a controller θ reconstructed from $\hat{\theta}$ satisfies the prescribed disk margin for the given Λ_p. The corresponding Lyapunov matrix P certifying stability can be expressed as:

$$P \triangleq \begin{bmatrix} I & S \\ 0 & U^\top \end{bmatrix} \begin{bmatrix} R & I \\ V^\top & 0 \end{bmatrix}^{-1}.$$

Therefore, whenever the conditions of Lemma 2 are met, it follows that the set $\Theta(P, \Lambda_p)$, containing all controllers that satisfy the disk-margin inequality for these matrices, is non-empty.

The mapping between θ and $\hat{\theta}$ is bidirectional: one can also recover $\hat{\theta}$ from a given θ, P, and Λ_p using the same equations.

5.3. Convex projection problem

The change of variables introduced in Lemma 2 leads to a formulation where all matrix inequalities are affine in the transformed parameters $\hat{\theta}$. This allows the projection step to be expressed as a convex semidefinite optimization problem, formulated as:

$$\begin{aligned} \min_{\hat{\theta}} \quad & \|\hat{\theta}' - \hat{\theta}\| \\ \text{s.t.} \quad & S \succ 0, \quad R \succ 0 \\ & \begin{bmatrix} R & I \\ I & S \end{bmatrix} \succ 0 \\ & \mathcal{L}_{\mathrm{cvx}}(S, R, N_A, \Lambda_p) \preceq 0, \end{aligned} \tag{17}$$

where $\mathcal{L}_{\mathrm{cvx}}(\cdot)$ represents the convex matrix inequality derived from condition (16).

In this formulation, the projection can be solved efficiently with standard semidefinite programming solvers. The resulting problem projects $\hat{\theta}'$ onto the set $\hat{\Theta}(\Lambda_p)$ while ensuring that the reconstructed controller satisfies the prescribed disk margin. Once the optimal $\hat{\theta}$ is found, the corresponding controller matrices θ can be recovered through the inverse of the variable transformation introduced earlier.

Remark 2. The computational cost of certification and projection is governed by the size of the underlying semidefinite program, which depends on the plant and controller order, the number of uncertainty channels, and the complexity of the chosen certificate.

5.4. SC-BO algorithm

The ingredients developed above, namely the disk-margin certification test, the change of variables, and the convex projection step, are combined into the proposed Stability-Certified Bayesian Optimization (SC-BO) algorithm, summarized in Algorithm 1. The method alternates between two stages: a BO step for closed-loop performance improvement and a robust-stability enforcement step. At each iteration, BO proposes a candidate controller parametrization θ' aimed at improving the performance objective. The candidate is then checked against the disk-margin certification condition of Lemma 1. If the condition is satisfied, the controller is accepted directly; otherwise, it is projected onto the certified set by solving the convex problem in (17). Therefore, only controllers satisfying the prescribed disk-margin requirement are evaluated.

Algorithm 1 Stability-certified Bayesian optimization (SC-BO).

1: **Input:** nominal plant (A_p, B_p, C_p, D_p), arbitrary controller θ_0, disk margin $D(\alpha, \sigma)$, GP prior $\mathcal{M}_0$.
2: Initialize $\Lambda_p = I$, $P = I$, and $\theta = \theta_0$
3: **for** $i = 1, \ldots, N_{\max}$ or until convergence **do**
  ▷ Bayesian optimization step
4:   $\theta' \leftarrow$ BayesianOptimizationStep$(\mathcal{M}_{i-1})$
  ▷ Robust-stability enforcement step
5:   **if** $\exists\, P', \Lambda'_p : \theta'$ satisfies disk-margin condition **then**
6:     $\theta, P, \Lambda_P \leftarrow \theta', P', \Lambda'_P$
7:   **else**
8:     $\hat{\theta}' \leftarrow$ TransformToThetaHat(θ', P)
9:     $\hat{\theta} \leftarrow$ ThetaHatProjection$(\hat{\theta}', \hat{\Theta}(\Lambda_p))$
10:     $P \leftarrow$ RecoverP$(\hat{\theta})$
11:     $\theta \leftarrow$ ThetaProjection(θ', P, Λ_p)
12:   **end if**
  ▷ Performance evaluation and model update
13:   $\hat{J}(\theta) \leftarrow$ EvaluatePerformance(θ)
14:   **if** θ' violates the disk-margin condition **then**
15:     $\hat{J}(\theta') \leftarrow J_{\text{penalty}}$
16:     $\mathcal{M}_i \leftarrow$ Update$(\mathcal{M}_{i-1}, \{(\theta, \hat{J}(\theta)), (\theta', \hat{J}(\theta'))\})$
17:   **else**
18:     $\mathcal{M}_i \leftarrow$ Update$(\mathcal{M}_{i-1}, (\theta, \hat{J}(\theta)))$
19:   **end if**
20: **end for**

The algorithm can be initialized from a controller θ_0 selected using prior knowledge or chosen arbitrarily, for example, as a zero-gain controller (line 2). Unlike SafeBO methods, SC-BO does not require the initial controller to be known a priori to be safe or stabilizing. Before starting the optimization loop, the controller is verified for robust stability and, if the condition is not certified, it is projected onto the certified robust-stability set. Hence, the first controller evaluated by SC-BO is guaranteed to satisfy the disk-margin requirement, while the subsequent optimization is not restricted to a safe or stable region grown from the initial point.

For simplicity, the certificate variables P and Λ_p may be initialized as identity matrices (line 2). Alternatively, if a model-based controller and a corresponding feasible certificate are available, they can be used to initialize the algorithm. This can provide a more informed starting point.

At each iteration, BO proposes a candidate controller θ' aimed at improving the performance objective (line 4). The candidate is first tested directly against the disk-margin certification condition (line 5). If feasible certificate variables (P', Λ'_p) exist, the controller is accepted, and the certificate is updated accordingly. Hence, projection is applied only when direct certification fails.
When θ' is not directly certifiable, the projection step is performed using the most recent feasible certificate information. First, θ' is mapped to the transformed variable $\hat{\theta}'$ using the change of variables in Lemma 2 (line 8). The semidefinite program in (17) is then solved with Λ_p fixed to its most recent feasible value, yielding a projected point $\hat{\theta} \in \hat{\Theta}(\Lambda_p)$ (line 9). A consistent Lyapunov matrix P is recovered from $\hat{\theta}$ (line 10), and the final controller θ is obtained by solving a projection problem analogous to (15) over the set $\Theta(P, \Lambda_p)$ of controllers certified by the recovered certificate (line 11). This final projection is performed in the original controller-parameter space, so that the executed controller is chosen close to the original BO proposal while satisfying the disk-margin certificate.

Remark 3. Within each projection problem, Λ_p is kept fixed to its most recent feasible value. This preserves convexity and keeps the projection problem computationally tractable, but can be conservative because the projected set may be smaller than the set obtained by optimizing over Λ_p. At the algorithmic level, however, Λ_p is not fixed once and for all; alternation on Λ_p is introduced during training: whenever a BO proposal is directly certified, the corresponding feasible certificate is retained and used in subsequent iterations (line 5). This allows the certificate to adapt during the optimization while keeping each individual projection step convex.

Remark 4. The present implementation specializes the certification layer to disk-margin robust stability. The same architecture could in principle accommodate other model-based closed-loop requirements, provided that they admit a tractable joint certification/projection formulation, for example through IQC, LMI, or dissipativity-based conditions. Possible extensions include certified L_2-gain bounds for disturbance attenuation, passivity-type requirements, and more general dissipativity-based performance specifications. Multiple certified requirements could also be enforced jointly when they can be expressed within a common convex certification framework.

5.4.1. Role of uncertified controllers

Raw BO proposals that violate the disk-margin condition are not evaluated. Nevertheless, they provide useful information for the surrogate model. If such proposals were simply discarded, the GP posterior could retain high uncertainty in regions that repeatedly generate uncertified controllers, causing the acquisition function to keep sampling similar candidates.

To avoid this behavior, each uncertified raw proposal θ' is added to the GP dataset with an artificial penalty value J_{penalty} (line 15–16), while the projected controller θ is evaluated using the true performance function (line 13). Since the optimization is formulated as a maximization problem, J_{penalty} is chosen below the objective values typically obtained by certified controllers, so that uncertified proposals are interpreted by the surrogate as low-performance observations. In practice, this value should be selected relative to the scale of the observed objective values, rather than as an arbitrary absolute constant. The sensitivity of the results to this choice is evaluated empirically in Section 6

6. Numerical experiments

We evaluate the proposed method through simulations on a quadrotor platform. In SC-BO, the performance-improvement step is implemented using the LineBO algorithm (Kirschner et al., 2019), which is suited to moderately high-dimensional black-box optimization. The proposed method is compared with representative SafeBO algorithms, namely SafeOpt and GoSafeOpt.

6.1. System description

In this section, we describe the quadrotor models used in the numerical experiments. We present both the high-fidelity model used for performance evaluation and the nominal model employed for certification and projection.

6.1.1. Quadrotor model

The considered platform is an ANT-X[1] fixed-pitch quadrotor, and the experiments focus on the roll dynamics. Closed-loop performance is evaluated by simulation on a high-fidelity plant model which is taken to represent the true system. The dynamics is described by a fourth-order continuous-time model identified from experimental input-output data via the Predictor-Based Subspace IDentification (PBSID) method

[1] https://antx.it/

(Van Wingerden & Verhaegen, 2008). The identified model is

$$\dot{x}_4(t) = \begin{bmatrix} -0.2217 & -27.1432 & 0.1981 & 8.0469 \\ 0.0515 & -0.0594 & -16.539 & -16.5410 \\ 3.0884 & 0.5869 & 7.2893 & 167.8874 \\ -2.2418 & -0.4339 & -4.5071 & -126.9662 \end{bmatrix} x_4(t) + \begin{bmatrix} 4.4283 \\ 19.3749 \\ -61.7074 \\ 44.6587 \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} -2.6164 & 0.1034 & -0.0733 & -0.1177 \end{bmatrix} x_4(t), \tag{18}$$

where $x_4(t) \in \mathbb{R}^4$ is the high-fidelity state vector, $u(t) \in \mathbb{R}$ is the roll-moment input L, and $y(t) \in \mathbb{R}$ is the measured roll rate p. The input is saturated according to the physical actuator limit $|L| \leq 0.152$ Nm.

To reflect realistic conditions, the model in (18) is assumed to be unavailable for controller design. Instead, robust-stability certification is performed on a reduced-order nominal model, while performance is evaluated on the high-fidelity model. The reduced-order nominal model is a second-order continuous-time model, also identified from experimental data via the PBSID method:

$$\dot{x}_2(t) = \begin{bmatrix} -0.3511 & -27.1526 \\ -0.1372 & -0.0690 \end{bmatrix} x_2(t) + \begin{bmatrix} 7.0133 \\ 23.1516 \end{bmatrix} u(t),$$
$$y(t) = \begin{bmatrix} -2.6164 & 0.1034 \end{bmatrix} x_2(t), \tag{19}$$

where $x_2(t) \in \mathbb{R}^2$ is the reduced-order state vector. The input $u(t)$ and output $y(t)$ have the same meaning as in (18).

The PBSID procedure also provides covariance estimates for the identified state-space matrices. These estimates are used for disk-margin selection and for the sensitivity analyses reported later.

6.1.2. Disk margin selection

The disk-margin parameters are chosen to cover the mismatch between the reduced-order nominal model used for certification and the higher-fidelity model used for performance evaluation. We use a symmetric disk, i.e., $\sigma = 0$, since the identification data did not suggest a systematic bias toward gain increase or decrease. The radius is set to $\alpha = 0.353$, corresponding to a minimum gain margin of approximately 3 dB and a minimum phase margin of approximately 20°.

The value of α was selected from a frequency-domain mismatch analysis between the second-order nominal model and the fourth-order high-fidelity model. The mismatch was expressed as the equivalent disk perturbation associated with the input-uncertainty model used in the certification step, and $\alpha = 0.353$ was chosen to conservatively cover the observed mismatch around the crossover-frequency range most relevant for closed-loop stability. To further assess whether this margin is robust to identification uncertainty, the covariance returned by the PBSID procedure was also used to generate representative samples of the fourth-order model, and the same frequency-domain check was repeated on these samples.

6.1.3. Controller parametrization

We consider a continuous-time LTI controller of order $n_c = 2$ for the roll-rate control loop. Since the plant is single-input single-output, the controller matrices in (3) satisfy $A_c \in \mathbb{R}^{2\times2}$, $B_c \in \mathbb{R}^{2\times1}$, $C_c \in \mathbb{R}^{1\times2}$, and $D_c \in \mathbb{R}$. All free entries are collected in the parameter vector $\theta \in \mathbb{R}^9$ defined in (4), which fully specifies the controller. Therefore, the optimization is performed over a 9-dimensional search space.

6.2. Implementation choices

This section summarizes the main implementation choices adopted in the numerical experiments. Unless otherwise stated, the same settings are used for all methods in the comparison.

Performance objective. The chosen performance metric penalizes tracking error and control effort. Since the optimization problem is formulated as a maximization problem, we define

$$J(\theta) = -\frac{1}{N} \sum_{k=0}^{N-1} \left[w_p \left(p_k - p_{r,k}\right)^2 + w_u u_k^2 \right], \tag{20}$$

where p_k is the measured roll angular rate, $p_{r,k}$ is the reference roll rate, and u_k is the roll-moment input. The weights $w_p > 0$ and $w_u > 0$ tune the trade-off between tracking accuracy and control effort. The negative sign makes larger values of $J(\theta)$ correspond to better closed-loop performance.

Although the metric in (20) is quadratic in the measured trajectory variables, the mapping $\theta \mapsto J(\theta)$ is not available in closed form, since it is evaluated from closed-loop simulations on the high-fidelity plant model. Thus, the performance-improvement step remains a black-box optimization problem.

Kernel selection. The objective function is modeled using a GP with zero-mean prior. We employ a Matérn kernel with smoothness parameter $\nu = 3/2$ (Williams & Rasmussen, 2006), which provides a good balance between model flexibility and smoothness for noisy yet differentiable functions. Automatic relevance determination is used to assign individual length scales to each input dimension. The same kernel structure is used for all GP models in the comparison, including the objective and constraint models used by SafeOpt and GoSafeOpt.

Kernel hyperparameters. For all methods, kernel hyperparameters are initialized before the start of the BO optimization loop by maximizing the marginal likelihood on a preliminary dataset generated by Latin Hypercube Sampling (LHS) (McKay et al., 2000). For SC-BO, we use $N_{\text{init}} = 36$ initial samples; uncertified raw proposals in this dataset are not evaluated directly, but are added to the GP dataset with the penalty value described above, while the corresponding projected controllers are evaluated. For SafeOpt and GoSafeOpt, the certified/safe region is very small relative to the selected controller-parameter domain. Therefore, a larger preliminary LHS candidate set of approximately $N_{\text{init}} = 200$ points is generated to obtain at least 15 safe initial controllers and initialize the algorithms in an informative way. During the optimization runs, the GP hyperparameters are re-estimated every 10 iterations; for GoSafeOpt, this update is applied to both the reward and safety GP models.

Safety constraint. Unsafe behavior in quadrotor systems is often associated with high angular velocities, which can also be an indication of instability. Following common practice in safe learning for aerial vehicles (Berkenkamp et al., 2023), safety is enforced by limiting the roll-rate magnitude. The safety function is defined as $g(\theta) = \omega_{\max} - \max_{0\leq k\leq N} p_k$, where p_k denotes the roll angular rate at time k, and $\omega_{\max}$ is the maximum admissible value. For the ANT-X quadrotor, experimental evidence indicates that roll rates above approximately 10.5 rad/s compromise safe operation. A conservative threshold of $\omega_{\max} = 8$ rad/s is therefore adopted.

Penalty selection. For all SC-BO experiments, the penalty assigned to uncertified raw proposals was chosen from a preliminary SC-BO run using

$$J_{\text{penalty}} = J_{\min}^{\text{pre}} - \eta\left(J_{\max}^{\text{pre}} - J_{\min}^{\text{pre}}\right), \tag{21}$$

where $J_{\min}^{\text{pre}}$ and $J_{\max}^{\text{pre}}$ are the minimum and maximum objective values observed for evaluated certified controllers in the preliminary run, and $\eta > 0$ controls the penalty strength.
With $\eta = 1$, this rule gave $J_{\text{penalty}} \approx -31.5$. We therefore used the rounded value $J_{\text{penalty}} = -30$ in the reported experiments. The sensitivity of the results to this choice is assessed in the ablation study reported in Section 6.

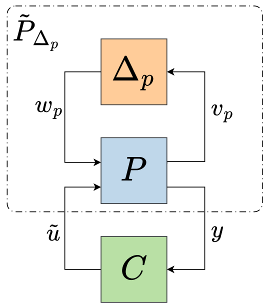

Fig. 3. Closed-loop interconnection used for robust-stability analysis. The nominal plant P is interconnected with the uncertainty block Δ_p, which represents the disk-margin perturbation, and with the controller C.

Initial controller. All methods are initialized from the same controller parameterization to ensure a fair comparison. For the SafeBO baselines, namely SafeOpt and GoSafeOpt, this initial controller must satisfy the imposed safety constraint and is used to define the initial safe set from which safe exploration starts. In contrast, SC-BO does not require an *a priori* safe set or an initial set of stabilizing controllers.

Reference trajectory. The closed-loop response is evaluated using a double-step reference in roll angular rate. The reference starts from zero, rises to 1 rad/s and is held for 1 s, then switches to −1 rad/s for another 1 s, before returning to zero.

6.3. Results and discussion

All numerical experiments are performed using continuous-time plant and controller models propagated with exact Zero-Order Hold (ZOH) discretization. Over each sampling interval $[t_k, t_{k+1})$, the controller input is kept constant at the measured output $y(t_k)$, while the plant input is kept constant at the computed control input $u(t_k)$. The corresponding state updates are computed analytically using the matrix exponential and the ZOH integral. The sampling time is $\Delta t = 0.001$ s, and the roll-moment input is saturated within $[-0.152, 0.152]$ Nm to respect the actuator limits.

The certification and projection problems in SC-BO are formulated in CVXPY and solved using MOSEK. All computations are executed in Python on a personal computer equipped with an Intel Core i7 processor running at 2.30 GHz and 16 GB of RAM.

6.3.1. Comparison with SafeBO methods

The proposed framework is compared against SafeOpt and GoSafeOpt. All methods are run for a maximum of $N_{\max} = 200$ iterations, and the reported statistics are computed over five independent runs per method. For SafeOpt and GoSafeOpt, robust stability violations are assessed a posteriori by checking whether the evaluated controller satisfies the same disk-margin certificate used by SC-BO. For SC-BO, the roll-rate safety metric is monitored a posteriori.

Fig. 4 shows the evolution of the best cost over iterations. The plotted value is the positive cost, i.e., $-J(\theta)$, so lower values correspond to better performance. SC-BO achieves the lowest final cost. SafeOpt exhibits limited improvement, consistently with its safe-set expansion mechanism: exploration is restricted to the region safely reachable from the initial controller, and in several runs the safe set stops expanding early. GoSafeOpt improves over SafeOpt by allowing exploration beyond a single connected safe region, but its performance remains limited by the learned safety constraint. In contrast, SC-BO separates performance optimization from robust-stability enforcement, allowing the BO step to explore the full controller-parameter domain while ensuring that only certified controllers are evaluated.

The time-domain responses in Fig. 5 further illustrate the performance differences. To make the transient behavior clearly visible, only the first 0.5 s of the response are shown. For each method, the plotted controller is the best-performing controller obtained among the runs; the initial controller is included as a reference. The controller obtained with SC-BO tracks the roll-rate reference with a faster transient, reduced overshoot, and no oscillatory behavior. GoSafeOpt also improves the initial controller, but its response is slower. SafeOpt remains more conservative and exhibits a slower transient with larger tracking error and overshoot.

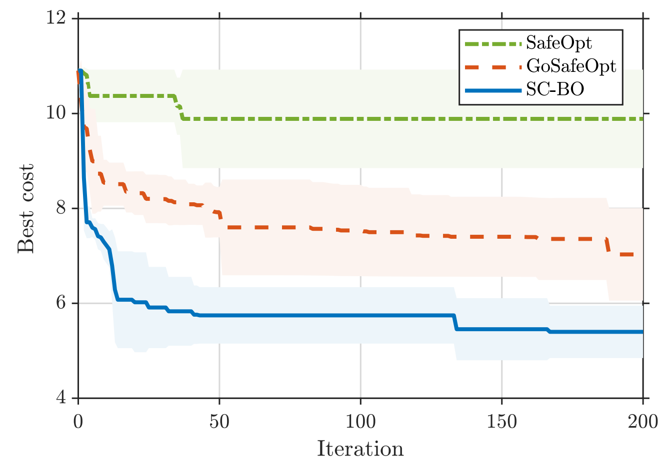

Fig. 4. Best cost over iterations. The curves show the mean over independent runs, and the shaded regions denote one standard deviation. Lower values correspond to better closed-loop performance.

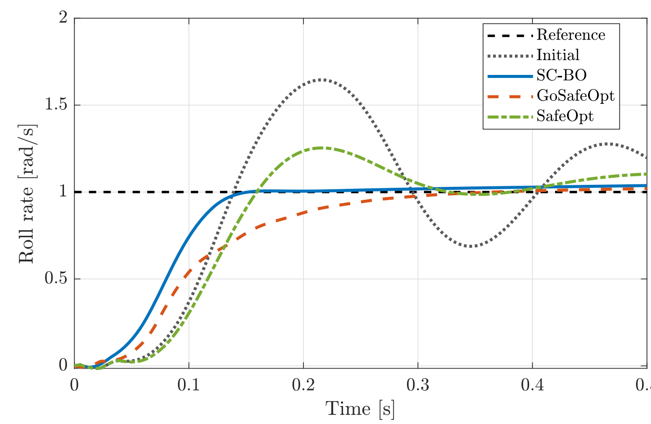

Fig. 5. Zoom of the first 0.5 s of the roll-rate response for the best controller obtained by each method, compared with the initial controller and the reference trajectory.

Table 1
Performance comparison over independent runs. Values are reported as mean ± standard deviation. Stability violations denote evaluated controllers that do not satisfy the disk-margin certificate. Safety violations denote violations of the roll-rate constraint.

Method	Best cost	Stability violations	Safety violations
SC-BO	5.399 ± 0.552	0.00 ± 0.00	0.00 ± 0.00
GoSafeOpt	7.031 ± 0.974	198 ± 3.033	6.50 ± 9.73
SafeOpt	9.887 ± 1.038	42.00 ± 81.68	1.00 ± 0.71

Table 1 summarizes the final performance and constraint violations. SC-BO obtains the lowest best cost and, by construction, no stability violations are observed. In contrast, for SafeOpt and GoSafeOpt several evaluated controllers fail the post-hoc stability check.

Table 2 reports the average computational cost per iteration. For SafeOpt, timing statistics are computed only over the iterations before repeated identical proposals indicate that the safe-set expansion has

Table 2
Average computation times per iteration. Values are reported in seconds as mean ± standard deviation over five independent runs. "Certification & projection" denotes the full certification/projection routine, while "MOSEK solver only" reports only the time spent by MOSEK inside the numerical solve.

Time [s]	SC-BO	GoSafeOpt	SafeOpt
BO candidate selection	0.004 ± 0.000	0.932 ± 0.137	0.036 ± 0.005
Certification & projection	15.176 ± 1.114	–	–
MOSEK solver only	0.023 ± 0.006	–	–
Simulation	0.228 ± 0.001	0.276 ± 0.014	0.187 ± 0.015
GP update & hyper. opt	0.205 ± 0.010	0.183 ± 0.072	0.397 ± 0.142
Full iteration	15.614 ± 1.118	1.341 ± 0.168	0.65 ± 0.146

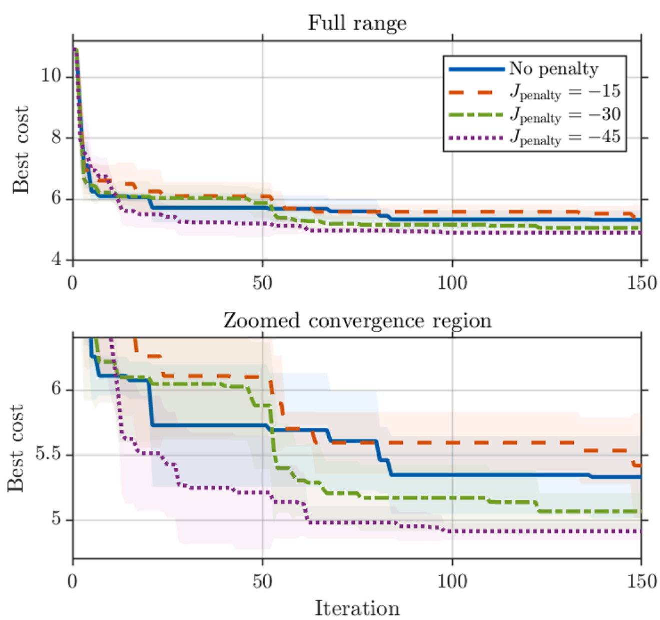

Fig. 6. Penalty ablation study for SC-BO. The curves report the mean best cost over five seeds, while the shaded regions denote one standard deviation. The top panel shows the full cost range, including the initial controller performance, whereas the bottom panel zooms into the convergence region. Lower values indicate better closed-loop performance.

stalled. The certification/projection layer dominates the runtime of SC-BO. However, the solver-only timing shows that the SDP solution with MOSEK is fast; most of the cost comes from the high-level CVXPY-based implementation. Thus, the reported time reflects the current implementation rather than an intrinsic limitation of the proposed certification approach. A more efficient implementation could formulate the fixed conic problems directly through a lower-level solver interface, such as MOSEK FUSION or another low-level optimizer API.

6.3.2. Sensitivity to penalty selection

We assess the effect of the artificial penalty assigned to raw BO proposals that do not satisfy the disk-margin certification condition. The ablation is performed over five seeds, considering no penalty and three penalty values, $J_{\text{penalty}} \in \{-15, -30, -45\}$. Each run is carried out for 150 BO iterations, and Table 3 reports the final best cost as mean ± standard deviation over the seeds. The corresponding convergence curves are shown in Fig. 6.

The results show that the penalty value affects the final performance. The no-penalty case and the weak penalty $J_{\text{penalty}} = -15$ yield higher final costs, indicating that uncertified regions are not sufficiently discouraged. More conservative penalties improve performance, with $J_{\text{penalty}} = -30$ and $J_{\text{penalty}} = -45$ achieving lower final costs. In this study,

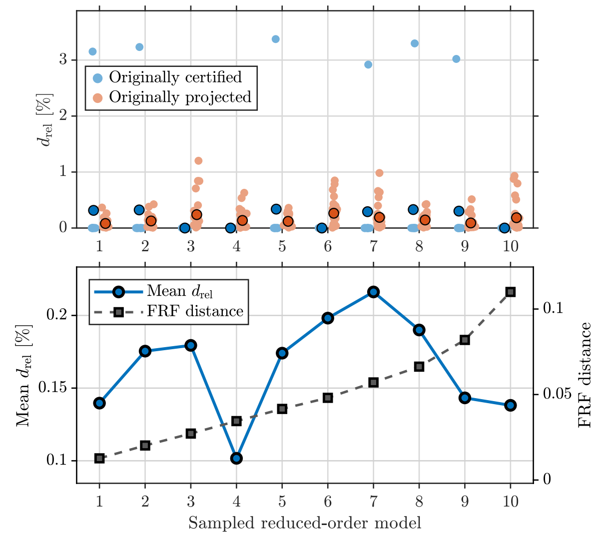

Fig. 7. Sensitivity of the stability-enforcement layer to reduced-order model uncertainty. The upper panel reports the relative variation (d_{rel}) of the executed controller parameters when the certification/projection step is repeated using each sampled reduced-order model. Results are shown separately for controller candidates that were directly certified with the reference model and for candidates that required projection in the original SC-BO run. The lower panel compares the mean relative sensitivity with the frequency-response distance of each sampled reduced-order model from the reference reduced-order model.

Table 3
Ablation study on the penalty assigned to raw uncertified BO proposals. Values are reported as mean ± standard deviation over five seeds.

J_{penalty}	None	-15	-30	-45
Final cost	5.33 ± 0.32	5.31 ± 0.42	5.07 ± 0.14	4.91 ± 0.07

$J_{\text{penalty}} = -45$ gives the best result, while the value used in the main experiments, $J_{\text{penalty}} = -30$, remains close in performance.

6.3.3. Sensitivity of the projection step to reduced-order model uncertainty

We assess the sensitivity of the stability-enforcement layer to uncertainty in the reduced-order model used for certification. Starting from the covariance returned by the PBSID identification procedure, we generate 1000 sampled second-order models and select 10 representative samples spanning different levels of frequency-response deviation from the reference reduced-order model. For each selected model, the certification/projection step is repeated on 40 controller candidates extracted from a previous SC-BO run: 10 candidates that were directly certified with the reference model and 30 candidates that required projection. This evaluates both the robustness of direct certification and the sensitivity of the projected controller. For each sampled model, the controller returned by the perturbed certification/projection step is compared with the controller obtained using the reference reduced-order model. The relative sensitivity is computed as the relative Euclidean distance between the two executed controller-parameter vectors. The distribution of the resulting relative sensitivities and their relation with the selected model perturbations are shown in Fig. 7.

The results in Table 4 show that controllers directly certified with the reference reduced-order model remain certified in 94% of the tested cases. For candidates that required projection in the original SC-BO run, projection is triggered for all sampled reduced-order models, but the resulting projected controllers remain close to the reference projections, with a mean relative sensitivity of 0.16% and a maximum of 1.20%. This behavior is also visible in Fig. 7, where the relative controller-parameter variations remain small across the selected reduced-order model perturbations. This indicates that, for the tested model perturbations, the

Table 4
Sensitivity of the certification/projection layer to perturbations of the identified reduced-order model.

Set	N	Cert. [%]	Proj. [%]	$\bar{d}_{\mathrm{rel}}$ [%]	$d_{\mathrm{rel}}^{\max}$ [%]
Certified	100	94.0	6.0	0.19	3.37
Projected	300	0.0	100.0	0.16	1.20

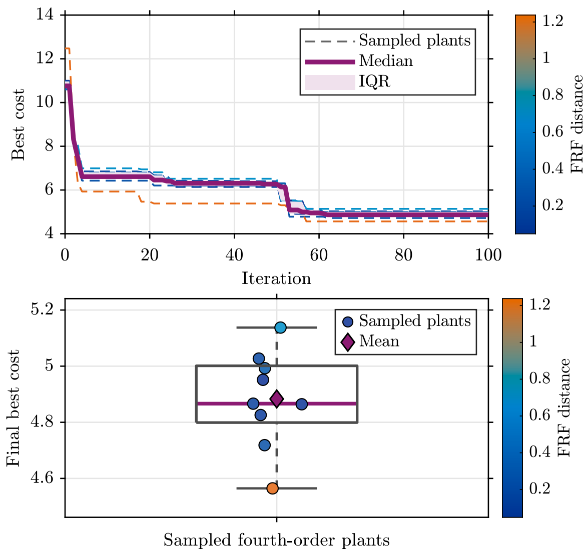

Fig. 8. Sensitivity of SC-BO to variations of the high-fidelity fourth-order simulation model. The upper panel shows the best cost over the optimization iterations for the selected sampled plants; individual dashed curves correspond to different fourth-order plants and are colored according to their frequency-response distance from the reference high-fidelity model. The median best-cost trajectory is shown with a solid line, and the shaded region denotes the interquartile range across sampled plants. The lower panel summarizes the distribution of the final best cost using a boxplot with individual sampled plants colored by frequency-response distance; the diamond marker denotes the mean final cost. Lower values indicate better closed-loop performance.

projection layer is not highly sensitive to the identified reduced-order model uncertainty.

6.3.4. *Sensitivity to high-fidelity plant variations*

We also assess whether the observed SC-BO performance depends on the specific fourth-order model used for objective evaluation. Starting from the covariance returned by the PBSID identification procedure, we generate 1000 sampled fourth-order models and select 10 representative samples. The complete SC-BO procedure is then repeated on each selected plant, while keeping the reduced-order model used for certification and projection fixed. This analysis does not constitute a hardware validation, but it evaluates whether the method remains effective when the plant used for performance evaluation varies around the identified high-fidelity model. This is consistent with the intended hardware setting, where the certification model is fixed and the objective evaluations would be obtained from the physical system. The corresponding convergence behavior and final-cost distribution are reported in Fig. 8.

Results are summarized in Table 5. Across the ten sampled plants, SC-BO consistently improves closed-loop performance. The final best cost is 4.88 ± 0.17, with values ranging from 4.56 to 5.14. No roll-rate safety violations are observed, and all evaluated controllers remain certified with respect to the fixed reduced-order model. The variation in final cost therefore reflects changes in the high-fidelity plant used to evaluate performance, not a failure of the certification/projection layer. As shown in Fig. 8, the best-cost trajectories converge to a similar range across the selected fourth-order plants, and the final costs remain concentrated despite the different frequency-response distances from the reference high-fidelity model.

Table 5
Sensitivity of SC-BO to variations of the high-fidelity fourth-order simulation model. Statistics are computed over 10 representative sampled plants.

Successful runs	Final best cost	Final best cost range	Stability violations	Safety violations
10	4.88 ± 0.17	[4.56, 5.14]	0	0

7. Conclusion

This paper presented SC-BO, a Bayesian optimization framework for fixed-order LTI controllers with certified disk-margin robust stability. The proposed method separates data-driven performance optimization from model-based stability certification: BO proposes candidate controller parameters, while a certification/projection layer ensures that only controllers satisfying the prescribed disk-margin requirement are evaluated.

The approach was evaluated on a quadrotor roll-rate control problem. The results show that SC-BO improves closed-loop performance while avoiding stability violations with respect to the prescribed disk-margin certificate. In contrast, the considered SafeBO baselines enforce safety constraints during learning but do not explicitly certify robust stability; therefore, most of the evaluated controllers fail the same disk-margin check. Additional analyses showed that the penalty assigned to uncertified raw proposals affects the optimization behavior, that the projection step is not highly sensitive to perturbations of the reduced-order certification model, and that the method remains effective across representative variations of the high-fidelity simulation model.

Future work will focus on hardware validation, more efficient solver implementations, and extensions of the certification layer to additional closed-loop requirements, such as disturbance-attenuation or dissipativity-based specifications.

CRediT authorship contribution statement

Marta Manzoni: Writing – review & editing, Writing - original draft, Visualization, Validation, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization; **Marco Lovera:** Writing – review & editing, Supervision, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Anderson, B. D. O., & Moore, J. B. (2007). Optimal control: Linear quadratic methods. Courier Corporation.

Baumann, D., Marco, A., Turchetta, M., & Trimpe, S. (2021). GoSafe: Globally optimal safe robot learning. In *2021 IEEE International conference on robotics and automation (ICRA)* (pp. 4452–4458). IEEE.

Berkenkamp, F., Krause, A., & Schoellig, A. P. (2023). Bayesian optimization with safety constraints: Safe and automatic parameter tuning in robotics. *Machine Learning, 112*(10), 3713–3747.

Berkenkamp, F., Schoellig, A. P., & Krause, A. (2016). Safe controller optimization for quadrotors with Gaussian processes. In *2016 IEEE International conference on robotics and automation (ICRA)* (pp. 491–496). IEEE.

Campi, M. C., Lecchini, A., & Savaresi, S. M. (2002). Virtual reference feedback tuning: A direct method for the design of feedback controllers. *Automatica, 38*(8), 1337–1346.

Davidor, Y. (1991). Genetic algorithms and robotics: A heuristic strategy for optimization (vol. 1). World Scientific Publishing Company.

Doyle, J. (1996). Robust and optimal control. In *Proceedings of 35th IEEE conference on decision and control* (pp. 1595–1598). IEEE (*vol. 2*).

Federal Aviation Administration (2023). 14 CFR Part 25: Airworthiness standards: Transport category airplanes. Code of Federal Regulations.

Joseph, S. B., Dada, E. G., Abidemi, A., Oyewola, D. O., & Khammas, B. M. (2022). Metaheuristic algorithms for PID controller parameters tuning: Review, approaches and open problems. *Heliyon*, *8*(5), e09399.

Junnarkar, N., Arcak, M., & Seiler, P. (2024). Synthesizing neural network controllers with closed-loop dissipativity guarantees. arXiv:2404.07373.

Junnarkar, N., Arcak, M., & Seiler, P. (2025). Stability margins of neural network controllers. In *2025 American control conference (ACC)* (pp. 1355–1360). IEEE.

Khosravi, M., Behrunani, V., Smith, R. S., Rupenyan, A., & Lygeros, J. (2020). Cascade control: Data-driven tuning approach based on Bayesian optimization. *IFAC-PapersOnLine*, *53*(2), 382–387.

Khosravi, M., Eichler, A., Schmid, N., Smith, R. S., & Heer, P. (2019). Controller tuning by Bayesian optimization an application to a heat pump. In *2019 18th European control conference (ECC)* (pp. 1467–1472). IEEE.

Khosravi, M., König, C., Maier, M., Smith, R. S., Lygeros, J., & Rupenyan, A. (2022). Safety-aware cascade controller tuning using constrained Bayesian optimization. *IEEE Transactions on Industrial Electronics*, *70*(2), 2128–2138.

Kirschner, J., Mutny, M., Hiller, N., Ischebeck, R., & Krause, A. (2019). Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In *International conference on machine learning* (pp. 3429–3438). PMLR.

Ljung, L. (1998). System identification. In *Signal analysis and prediction* (pp. 163–173). Boston, MA: Birkhäuser Boston.

Manzoni, M., & Lovera, M. (2025). Data-driven task-specific optimization of H_∞ controllers via Bayesian optimization. *IFAC-PapersOnLine*, *59*(20), 835–840.

Mate, S., Pal, P., Jaiswal, A., & Bhartiya, S. (2023). Simultaneous tuning of multiple PID controllers for multivariable systems using deep reinforcement learning. *Digital Chemical Engineering*, *9*, 100131.

McClement, D. G., Lawrence, N. P., Backström, J. U., Loewen, P. D., Forbes, M. G., & Gopaluni, R. B. (2022). Meta-reinforcement learning for the tuning of PI controllers: An offline approach. *Journal of Process Control*, *118*, 139–152.

McKay, M. D., Beckman, R. J., & Conover, W. J. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, *42*(1), 55–61.

Megretski, A., & Rantzer, A. (2002). System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control*, *42*(6), 819–830.

Mockus, J. (2005). The Bayesian approach to global optimization. In *System modeling and optimization: Proceedings of the 10th IFIP conference New York city, USA, August 31–September 4, 1981* (pp. 473–481). Springer.

Novara, C., & Formentin, S. (2017). Data-driven inversion-based control of nonlinear systems with guaranteed closed-loop stability. *IEEE Transactions on Automatic Control*, *63*(4), 1147–1154.

Pintelon, R., & Schoukens, J. (2012). System identification: A frequency domain approach. John Wiley & Sons.

Rawlings, J. B., Mayne, D. Q., & Diehl, M. M. (2020). Model predictive control: Theory, computation, and design. 2nd ed. Nob Hill Publishing. Santa Barbara, CA.

Rodríguez-Molina, A., Mezura-Montes, E., Villarreal-Cervantes, M. G., & Aldape-Pérez, M. (2020). Multi-objective meta-heuristic optimization in intelligent control: A survey on the controller tuning problem. *Applied Soft Computing*, *93*, 106342.

Scherer, C., Gahinet, P., & Chilali, M. (2002). Multiobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, *42*(7), 896–911.

Seiler, P. (2014). Stability analysis with dissipation inequalities and integral quadratic constraints. *IEEE Transactions on Automatic Control*, *60*(6), 1704–1709.

Seiler, P., Packard, A., & Gahinet, P. (2020). An introduction to disk margins [lecture notes]. *IEEE Control Systems Magazine*, *40*(5), 78–95.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & De Freitas, N. (2015). Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, *104*(1), 148–175.

Skogestad, S., & Postlethwaite, I. (2005). Multivariable feedback control: Analysis and design. john Wiley & sons.

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. W. (2012). Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, *58*(5), 3250–3265.

Sui, Y., Gotovos, A., Burdick, J., & Krause, A. (2015). Safe exploration for optimization with Gaussian processes. In *International conference on machine learning* (pp. 997–1005). PMLR.

Sukhija, B., Turchetta, M., Lindner, D., Krause, A., Trimpe, S., & Baumann, D. (2023). GoSafeOpt: Scalable safe exploration for global optimization of dynamical systems. *Artificial Intelligence*, *320*, 103922.

Turchetta, M., Berkenkamp, F., & Krause, A. (2019). Safe exploration for interactive machine learning. *Advances in neural information processing systems*, *vol. 32*.

Van Heusden, K., Karimi, A., & Bonvin, D. (2011). Data-driven model reference control with asymptotically guaranteed stability. *International Journal of Adaptive Control and Signal Processing*, *25*(4), 331–351.

Van Wingerden, J.-W., & Verhaegen, M. (2008). Subspace identification of MIMO LPV systems: The PBSID approach. In *2008 47th IEEE conference on decision and control* (pp. 4516–4521). IEEE.

Williams, C. K. I., & Rasmussen, C. E. (2006). Gaussian processes for machine learning (vol. 2). MIT press Cambridge, MA.

Zhou, K., Doyle, J. C., & Glover, K. (1996). Robust and optimal control (vol. 40). Prentice hall New Jersey.