BOUNDARY UNLOCKING VALUE FOR SAFE BAYESIAN OPTIMIZATION UNDER MONOTONE SAFETY CONSTRAINTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Many safety-constrained optimization problems involve both a system configuration and an ordered intensity. When safety risk increases monotonically with intensity, each configuration is associated with an unknown maximum safe intensity. Under a limited evaluation budget, efficient optimization requires identifying which safe experiments are most valuable for subsequent optimization. Existing safe Bayesian optimization methods often guide query selection using criteria based on uncertainty, optimism, or information gain, but these criteria do not directly quantify the optimization value that may result from resolving different safety boundaries. The monotone configuration–intensity structure makes such value tractable to quantify: the feasibility uncertainty of each configuration can be represented by a latent maximum safe intensity, and each possible boundary location determines an accessible intensity range and the best objective value attainable within it. Building on this observation, we introduce Boundary Unlocking Value (BUV), which combines posterior boundary uncertainty with optimistic objective gains to quantify the exploration potential of each configuration. We further develop BUV-SafeBO, which evaluates candidate points within the certified safe set based on how informative they are about the safety frontiers of high-BUV configurations. It also incorporates expected improvement at each candidate to balance future safe expansion with immediate objective optimization. Experiments on multiple structured safe optimization benchmarks show that BUV-SafeBO achieves competitive optimization performance while maintaining high safe-query rates.

1 INTRODUCTION

Many safety-constrained optimization problems naturally involve two types of decision variables with distinct roles: configuration variables that determine how the system operates, and ordered intensity variables that control the magnitude of system actuation. Typical examples include electrode configurations and stimulation amplitudes in neuromodulation (Capogrosso et al., 2013; Sui et al., 2015; 2018; Wei et al., 2025), treatment choices and drug dosages in clinical applications (Losalka & Scarlett, 2023; 2024), and controller parameters and actuation magnitudes in robotic control (Berkenkamp et al., 2016; 2023; Rothfuss et al., 2023; Brunke et al., 2022). In these problems, for a fixed configuration, increasing the intensity typically leads to a monotonic increase in safety risk, so that each configuration is associated with an unknown maximum safe intensity. In contrast to general safe optimization problems, where the unknown feasible region may be high-dimensional, unordered, or even disconnected, this monotone structure compresses the feasibility uncertainty of each configuration into a one-dimensional latent safety boundary (Losalka & Scarlett, 2023; 2024). When the number of configurations is large and the experimental budget is limited, the optimizer must not only identify high-performing safe configuration–intensity pairs, but also determine how the limited budget for safe exploration should be allocated across configurations.

Safe Bayesian optimization provides a systematic framework for such problems by modeling the unknown objective and safety constraints with probabilistic surrogate models (Shahriari et al., 2016) and restricting evaluations to regions certified as safe with high confidence (Sui et al., 2015; Berkenkamp et al., 2023). SafeOpt and StageOpt established confidence-bound-based paradigms

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

for safe-set expansion (Sui et al., 2015; 2018), while subsequent methods such as GoOSE and information-theoretic safe exploration further focused safety learning on goal-relevant or highly informative regions to reduce unnecessary exploration (Turchetta et al., 2019; Bottero et al., 2022; 2026). Recent work has also begun to explicitly exploit configuration–intensity structure. M-SafeUCB leverages monotonicity between intensity and safety risk to improve the efficiency of safety-boundary learning (Losalka & Scarlett, 2023), while M-SafeOpt further incorporates objective information to eliminate clearly suboptimal configurations and expands only those configurations that may yield better actions under optimistic estimates (Losalka & Scarlett, 2024). However, when multiple configurations remain promising for further expansion, an important experimental budget-allocation problem remains: which configuration's safety boundary should the next safe experiment prioritize learning about?

Existing methods often prioritize safe expansion according to safety uncertainty or optimistic improvement potential (Sui et al., 2015; 2018; Losalka & Scarlett, 2024), but these criteria do not directly answer the question above. A boundary may be highly uncertain, yet have limited exploration value if further expansion would only expose low-value actions. Conversely, a boundary with relatively low uncertainty may deserve higher priority if it has a high probability of unlocking substantially better actions. Likewise, two configurations may both contain potentially better solutions under optimistic upper bounds, while one can reach a high-value region only under a low-probability boundary realization and the other can achieve a more moderate but more reliable improvement with higher posterior probability. Thus, a configuration that remains potentially useful is not necessarily the one that should be explored first. What matters is both the likelihood of further safe expansion and the objective improvement it could unlock. The monotone structure makes this value explicitly quantifiable: each possible boundary location corresponds to a specific maximum accessible intensity and, consequently, to the best attainable objective value within the corresponding range. This allows different configurations to be ranked by combining the posterior probability of each potential boundary location with the objective improvement it could provide relative to the currently certified boundary.

Building on this observation, we represent the maximum safe intensity of each configuration as a latent boundary and construct a surrogate distribution over its location using posterior samples from the safety GP. We introduce Boundary Unlocking Value (BUV), which combines this distribution with optimistic objective gains beyond the currently certified boundary. By accounting for both the likelihood of further safe expansion and the objective value it could unlock, BUV characterizes the exploration potential of different configurations. Building on BUV, we propose BUV-SafeBO to translate configuration-level boundary value into safe query selection. The algorithm combines BUV-weighted mutual information with expected improvement to select queries within the certified safe set, balancing future safe expansion with immediate objective optimization.

Our main contributions are:

- We introduce BUV to quantify the potential optimization value beyond the certified safe boundary of each configuration under a monotone configuration–intensity structure.
- We develop BUV-SafeBO, which uses boundary value to guide information acquisition while pursuing objective improvement within the currently certified safe region.
- We evaluate BUV-SafeBO across multiple tasks against competitive baselines. The experiments demonstrate that our method achieves competitive optimization performance among the compared methods while maintaining high safe-query rates.

2 Related Work

2.1 Safe Bayesian Optimization

Safe Bayesian optimization aims to optimize an expensive black-box objective while ensuring that every evaluation satisfies unknown safety constraints throughout the learning process. SafeOpt established a confidence-bound-based framework that expands a certified safe region from an initial safe set while searching for high-performing actions (Sui et al., 2015). SafeOpt-MC extended this framework to separate objectives and multiple safety constraints (Berkenkamp et al., 2023), while StageOpt decoupled safe-region expansion from utility maximization into two stages (Sui et al.,

2018). In the related setting of safe active learning, GP-based methods combine safe-region identification with entropy-based sampling (Schreiter et al., 2015). Subsequent work has sought to make safe exploration more selective and sample-efficient. GoOSE performs goal-oriented safe exploration by acquiring only the safety information required to evaluate targets proposed by an external optimization procedure (Turchetta et al., 2019). Information-theoretic methods instead select certified-safe queries according to how informative they are about the unknown safety constraint, and can combine safety exploration with standard Bayesian optimization acquisitions (Bottero et al., 2022; 2026). Transductive active learning similarly selects observations from an accessible region to reduce uncertainty about prediction targets that may lie outside it, with applications to safe Bayesian optimization (Hübotter et al., 2024). Other work has addressed scalability, for example by reducing high-dimensional optimization to sequences of one-dimensional subproblems (Kirschner et al., 2019) or by combining dimensionality reduction with local optimistic safe exploration (Wei et al., 2025). These methods improve safe-set expansion, task-directed safety learning, information acquisition, and scalability. However, relatively little work explicitly values unresolved safety uncertainty according to the downstream objective opportunities that resolving it may unlock. In general safe Bayesian optimization, such a valuation is not straightforward: safety uncertainty may correspond to a potentially complex unknown feasible set, so its optimization consequence depends on both the possible feasible regions and the objective values accessible within them. In our monotone configuration–intensity setting, the feasible set of each configuration is instead completely characterized by a single latent safety boundary, which makes it possible to associate possible boundary realizations directly with their downstream optimization value (further discussed in Appendix A.1.1).

2.2 Monotonicity in Safe Bayesian Optimization

Monotonicity with respect to a designated safety variable provides additional structure for safe optimization. M-SafeUCB assumes that the relevant function is monotone in a safety variable and exploits this structure to simplify safe exploration, with theoretical guarantees on safety, regret, and approximation of the safe boundary (Losalka & Scarlett, 2023). M-SafeOpt considers separate objective and safety functions, requiring monotonicity only of the safety function with respect to a single safety variable (Losalka & Scarlett, 2024). By exploiting this ordering together with objective information, it can avoid unnecessary exploration of configurations that are already provably suboptimal and concentrate exploration on configurations that may still contain improved safe actions. In contrast, our work builds on the same monotone configuration–intensity structure but uses it for a different purpose. While existing monotonicity-aware methods mainly exploit this ordering to facilitate safe-boundary learning and avoid unnecessary expansion, we use the latent boundary representation to explicitly compare configurations according to the downstream optimization value of potential expansion beyond their currently certified safe boundaries.

3 BUV-SafeBO Method

The overall framework of BUV-SafeBO is illustrated in Fig. 1. Figure 1(a) shows the monotone increase of safety risk with intensity and the currently certified safe points. Figure 1(b) illustrates how configuration-specific boundary distributions and optimistic objective gains are combined to compute the Boundary Unlocking Value (BUV). Figure 1(c) combines BUV with mutual information to quantify the exploration value of candidate points in the certified safe set, and further incorporates expected improvement to select the next query.

3.1 Problem Formulation and Safe Set

Let $x = (c, I)$ denote an action consisting of a configuration $c \in \mathcal{C}$ and an ordered intensity $I \in \mathcal{I}$. Given initial observations $\mathcal{D}_0$ and a query budget T, we seek to maximize an unknown objective $f(x)$ while satisfying the safety constraint $g(x) \geq 0$ at every query, and return the best observed safe action. Here, $g(x)$ denotes the safety margin. For each fixed configuration, we assume that safety deteriorates monotonically with intensity,

$$I_i < I_j \quad \Longrightarrow \quad g(c, I_i) \geq g(c, I_j), \tag{1}$$

while no monotonicity assumption is imposed on f. At iteration t, we update independent Gaussian process (GP) models for f and g using the accumulated observations $\mathcal{D}_t$ to obtain posterior means

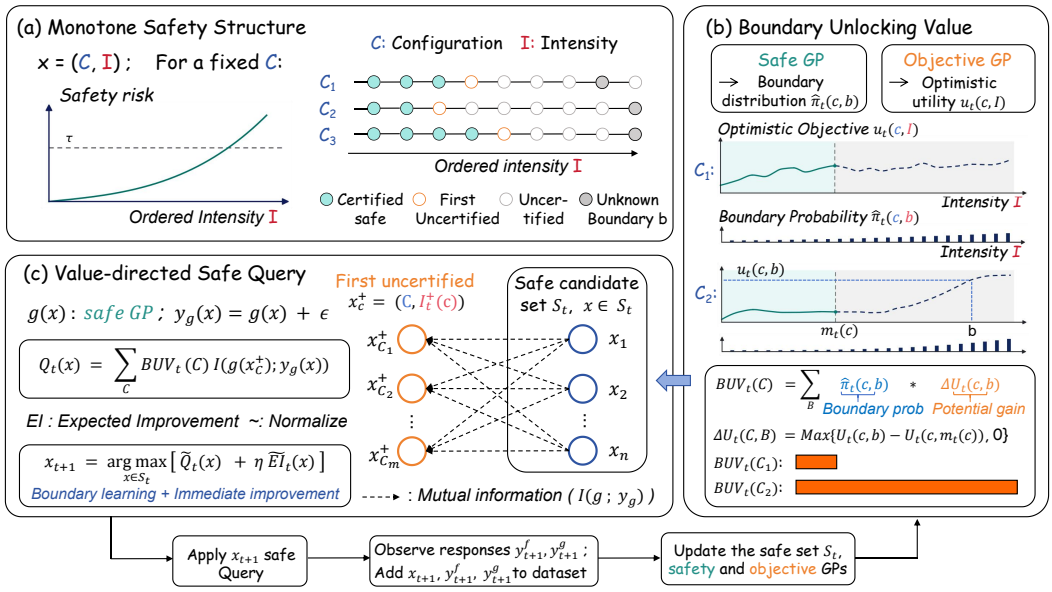


Figure 1: Overview of the BUV-SafeBO framework. (a) Monotone safety induces a certified safe prefix for each configuration. (b) BUV combines posterior boundary probabilities with optimistic objective gains to quantify the value of further safe expansion. (c) The next certified-safe query is selected using BUV-weighted mutual information together with expected improvement. The observations then update the GP models and safe set for the next iteration.

μ_t^f, μ_t^g and standard deviations σ_t^f, σ_t^g (Rasmussen & Williams, 2006). Define the safety lower confidence bound as

$$\ell_t^g(x) = \mu_t^g(x) - \beta_g \sigma_t^g(x), \tag{2}$$

where $\beta_g > 0$ controls the confidence-bound width. The maximum model-certified safe intensity for configuration c is

$$m_t^{\mathrm{LCB}}(c) = \max\{I \in \mathcal{I} : \ell_t^g(c, I') \geq 0,\ \forall I' \leq I\}. \tag{3}$$

Let $L_t(c)$ denote the largest intensity previously evaluated for configuration c that was certified safe before evaluation and yielded a nonnegative observed safety response. Whenever these certifications are valid, $(c, L_t(c))$ is safe, and monotonicity implies that all lower intensities for the same configuration are also safe. We define the current safe boundary as

$$m_t(c) = \max\left\{m_t^{\mathrm{LCB}}(c), L_t(c)\right\}, \tag{4}$$

and the corresponding safe candidate set as

$$\mathcal{S}_t = \{(c, I) : I \leq m_t(c)\}. \tag{5}$$

3.2 Boundary Unlocking Value

The value of expanding the current safe region depends on both how far safety may extend and how much objective improvement the newly accessible region may offer. BUV combines these two factors using the safety and objective GP posteriors, respectively, as illustrated in Fig. 1(b).

Boundary distribution. Under the monotonicity assumption, the safe intensities of each configuration form a prefix characterized by its unknown maximum safe intensity B_c. A realization $B_c = b$ implies that all candidate intensities up to b are safe. To construct a surrogate distribution over boundary locations, we draw R joint posterior functions $g_t^{(r)}$ from the safety GP using posterior function sampling (Wilson et al., 2020; 2021). In our implementation, we sample the finite-feature GP weights, sharing each weight draw across all configurations and intensities. Since the safety GP

is not explicitly constrained to be monotone, its posterior samples may exhibit multiple sign changes across intensity levels. We therefore define each sampled boundary as the endpoint of the longest safe prefix:

$$B_c^{(r)} = \max\left\{b \in \mathcal{I} : g_t^{(r)}(c, I) \geq 0,\ \forall I \leq b\right\}. \tag{6}$$

This mapping ensures that each induced boundary represents a safe prefix, without imposing monotonicity on the GP posterior itself. We use the resulting distribution as a tractable surrogate for uncertainty about the unknown safety boundary when computing BUV. Specifically, the induced boundary distribution is estimated by

$$\widehat{\pi}_t(c, b) = \frac{1}{R}\sum_{r=1}^{R} \mathbf{1}\left\{B_c^{(r)} = b\right\}. \tag{7}$$

Here, $\widehat{\pi}_t(c, b)$ is the empirical probability that a sampled safe prefix ends at b, providing a surrogate distribution for the latent boundary B_c.

Optimistic Objective Value. We next quantify the objective value that would become accessible under each possible boundary. Define the optimistic objective value (Srinivas et al., 2012)

$$u_t(c, I) = \mu_t^f(c, I) + \beta_f \sigma_t^f(c, I), \tag{8}$$

and the best optimistic utility within the safe prefix induced by boundary b as

$$U_t(c, b) = \max_{I \in \mathcal{I},\, I \leq b} u_t(c, I). \tag{9}$$

The prefix maximum is required because the objective function is not assumed to be monotone in intensity. Relative to the current safe boundary $m_t(c)$, the potential optimistic gain associated with boundary b is

$$\Delta U_t(c, b) = \max\left\{U_t(c, b) - U_t(c, m_t(c)),\ 0\right\}. \tag{10}$$

BUV. Combining the posterior probability of each possible boundary with the optimistic objective gain it could make accessible, we define the **Boundary Unlocking Value (BUV)** of configuration c as

$$\mathrm{BUV}_t(c) = \sum_{b \in \mathcal{I},\ b > m_t(c)} \widehat{\pi}_t(c, b)\, \Delta U_t(c, b). \tag{11}$$

BUV is therefore a posterior-weighted optimistic gain over possible safety-boundary realizations. It assigns high value to configurations that are both likely to admit further safe expansion and capable of unlocking high-value objective regions beyond their currently certified boundary. If no higher candidate intensity remains, or if all possible extensions yield zero optimistic gain, then $\mathrm{BUV}_t(c) = 0$. Further interpretation of BUV is provided in Appendix A.1.

3.3 Value-Directed Safe Query Selection

BUV identifies configurations with valuable unresolved safe extensions, but does not specify which certified-safe point should be queried next. Inspired by information-based acquisition in Bayesian optimization and safe exploration (Hennig & Schuler, 2012; Hernández-Lobato et al., 2014; Bottero et al., 2022), we evaluate how informative each candidate in the certified safe set is about the safety boundaries of these configurations.

Information targets. Directly evaluating information gain about the latent boundary B_c requires integrating over boundary distributions induced by possible future observations. We instead use a local information target associated with the next potential safe expansion. Specifically, we define the first uncertified intensity as

$$I_t^+(c) = \min\{I \in \mathcal{I} : I > m_t(c)\}, \qquad x_c^+ = (c, I_t^+(c)). \tag{12}$$

Under monotone safety structure, the first uncertified intensity is the transition point between certified and uncertified regions. Its safety observation determines whether the boundary can move forward. Therefore, we use its information as a tractable approximation to boundary information. The extension to continuous intensities is discussed in Appendix A.4.

Algorithm 1 BUV-SafeBO

Input: Initial dataset $\mathcal{D}_0$, query budget T, target limit M, weight η

for $t = 0, \dots, T-1$ **do**
  Update objective and safety GPs using $\mathcal{D}_t$
  Determine $m_t(c)$ and construct the certified safe set $\mathcal{S}_t$
  Sample posterior safety functions and estimate boundary probabilities $\widehat{\pi}_t(c, b)$
  Compute optimistic prefix utilities $U_t(c, b)$ and gains $\Delta U_t(c, b)$
  $\mathrm{BUV}_t(c) \leftarrow \sum_{b > m_t(c)} \widehat{\pi}_t(c, b)\, \Delta U_t(c, b)$
  Construct $\mathcal{T}_t$ from the top-M unresolved configurations with positive BUV scores
  Determine the first uncertified point x_c^+ for each $c \in \mathcal{T}_t$
  $Q_t(x) \leftarrow \sum_{c \in \mathcal{T}_t} \mathrm{BUV}_t(c)\, I_t(c; x), \quad x \in \mathcal{S}_t$
  Compute $\mathrm{EI}_t(x)$ and normalize $Q_t(x)$ and $\mathrm{EI}_t(x)$
  Set $\mathcal{S}_t^{\mathrm{query}}$ to the unevaluated points in $\mathcal{S}_t$ if any remain; otherwise use $\mathcal{S}_t$
  $x_{t+1} \leftarrow \arg\max_{x \in \mathcal{S}_t^{\mathrm{query}}} \left[\widetilde{Q}_t(x) + \eta\, \widetilde{\mathrm{EI}}_t(x) \right]$
  Query x_{t+1} and observe y_{t+1}^f, y_{t+1}^g
  $\mathcal{D}_{t+1} \leftarrow \mathcal{D}_t \cup \{(x_{t+1}, y_{t+1}^f, y_{t+1}^g)\}$
end for

Output: Best observed safe action in $\mathcal{D}_T$

Value-weighted frontier information. Motivated by information-based safe exploration (Bottero et al., 2022), for $x \in \mathcal{S}_t$ we evaluate

$$I_t(c; x) = \mathrm{I}\big(g(x_c^+); y_g(x) \mid \mathcal{D}_t\big)\,. \tag{13}$$

Under the Gaussian surrogate and independent Gaussian observation noise, this quantity has the analytic form (Krause et al., 2008)

$$\rho_{t,c}^2(x) = \frac{\kappa_t^g(x_c^+, x)^2}{\big[\sigma_t^g(x_c^+)\big]^2 \big(\big[\sigma_t^g(x)\big]^2 + \sigma_{n,g}^2\big)}, \qquad I_t(c; x) = -\tfrac{1}{2} \log\big(1 - \rho_{t,c}^2(x)\big). \tag{14}$$

We use a positive observation-noise variance in this calculation and set $I_t(c; x) = 0$ when the frontier has zero posterior variance. The derivation is given in Appendix A.2.

For computational efficiency, we retain up to M unresolved configurations with the largest positive BUV values, forming $\mathcal{T}_t$. We then aggregate the frontier information across these configurations and weight each term by its BUV. The exploration score is

$$Q_t(x) = \sum_{c \in \mathcal{T}_t} \mathrm{BUV}_t(c) I_t(c; x). \tag{15}$$

This score separates boundary value from query informativeness: BUV determines which safety frontiers are more valuable to explore, while mutual information determines which certified-safe queries are most informative about them. As illustrated in Fig. 1(c), posterior dependence allows a query to provide information about frontier points belonging to other configurations. Thus, $Q_t(x)$ favors safe candidates that are informative about high-BUV frontiers.

Immediate improvement and query execution. To balance exploration beyond the current certified boundary with objective improvement within the certified safe set, we combine the safe-expansion information score $Q_t(x)$ with expected improvement (EI) (Jones et al., 1998; Brochu et al., 2010), computed relative to the best observed feasible objective value:

$$\mathrm{EI}_t(x) = \mathbb{E}_t[(f(x) - f_t^\star)_+]\,, \tag{16}$$

where $f_t^\star$ denotes the best observed objective value among previously evaluated safe points. We normalize $Q_t(x)$ and $\mathrm{EI}_t(x)$ over the current safe candidate set and define the final acquisition function as

$$\alpha_t(x) = \widetilde{Q}_t(x) + \eta\, \widetilde{\mathrm{EI}}_t(x), \qquad \eta > 0, \tag{17}$$

where tildes denote normalization to $[0, 1]$ over the current safe candidate set, and η controls the trade-off between future safe expansion and immediate objective improvement. We restrict query selection to previously unevaluated points in the certified safe set whenever such points remain; otherwise, repeated queries are allowed. Let $\mathcal{S}_t^{\text{query}}$ denote this query candidate set. The next query is selected as

$$x_{t+1} \in \arg\max_{x \in \mathcal{S}_t^{\text{query}}} \alpha_t(x). \tag{18}$$

After selecting x_{t+1} from the certified safe candidate set and observing y_{t+1}^f and y_{t+1}^g, we update

$$\mathcal{D}_{t+1} = \mathcal{D}_t \cup \{(x_{t+1}, y_{t+1}^f, y_{t+1}^g)\}. \tag{19}$$

The new observations are then used to update both GP surrogates, the certified safe set, and the BUV scores, potentially changing which safe extensions are considered most valuable. This closes the feedback loop illustrated in Fig. 1, and the process is repeated until the query budget is exhausted. The best observed safe action is returned at termination. Algorithm 1 summarizes the complete optimization loop.

4 Experiments

We evaluate BUV-SafeBO on spinal cord stimulation, quadrotor trajectory tracking, and drug–dose optimization to assess sample efficiency and empirical safety. Ablation studies examine the contributions of its components. Implementation details and additional experiments are provided in Appendices B and C.

Baselines. We compare BUV-SafeBO with six methods: HdSafeBO (Wei et al., 2025), M-SafeOpt (Losalka & Scarlett, 2024), M-SafeUCB (Losalka & Scarlett, 2023), SafeOpt (Sui et al., 2015), ISE (Bottero et al., 2022), and ITL (Hübotter et al., 2024). These methods cover high-dimensional safe optimization, monotonicity-aware exploration, confidence-bound-based safe expansion, and information-directed sampling. Within each benchmark, all methods use a shared GP modeling framework and receive the same initial observations and additional query budget. Baseline adaptations and method-specific safety rules are described in the Appendix B.1.

Evaluation metrics. We report three metrics across all benchmarks. Normalized progress measures the improvement in the best true-safe objective value found so far relative to the best initial safe value, normalized by the gap between that initial value and the reference safe optimum. The safe-query rate is the proportion of queries satisfying the safety constraint. Cumulative safety violation sums the magnitudes of negative safety margins over the query sequence. For benchmarks with multiple targets, results are first averaged across targets within each seed and then summarized across seeds. Curves show means, with shaded regions indicating one standard error across seeds.

4.1 Spinal Cord Stimulation

We use the spinal cord stimulation simulation benchmark introduced by Wei et al. (2025). Each action combines a configuration of a 16-contact electrode array with a stimulation current intensity. The dataset contains 21,800 configurations and ten intensity levels, yielding 218,000 candidate actions. Increasing current can increase both target and non-target activation, with excessive activation posing a safety risk. The objective rewards target-muscle activation relative to neighboring non-target muscles, while safety requires the maximum normalized activation across modeled muscles not to exceed 0.95. The objective and safety functions, along with the target-muscle specifications, are detailed in Appendix B.2.1.

We evaluate two target muscles using five algorithm seeds per target, with 200 initial observations and 800 additional queries per run. As shown in Fig. 2, BUV-SafeBO achieves the highest final mean normalized progress among the compared methods while maintaining a high safe-query rate and low cumulative safety violation.

4.2 Quadrotor Trajectory Tracking

We consider a quadrotor trajectory-tracking benchmark constructed using Safe-Control-Gym (Yuan et al., 2022). Each action combines a configuration of four controller gains with a trajectory inten-

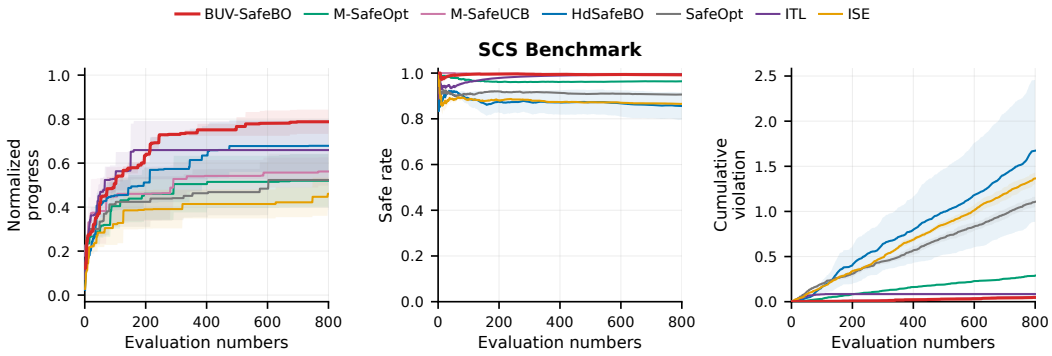


Figure 2: Comparison on the spinal cord stimulation benchmark. Results are averaged over two targets and five seeds. Shaded regions indicate one standard error across seeds.

sity specifying the number of figure-eight cycles requested within six seconds. The candidate space contains 256 configurations and 20 intensity levels, yielding 5,120 actions. Higher intensity increases the demands on thrust and attitude control, making safety constraint violations more likely. Using the benchmark's tracking-error and control-effort measurements, we construct an objective that balances trajectory intensity, tracking accuracy, and control effort. Safety accounts for position, altitude, pitch, requested thrust, and episode completion. The objective and safety functions, along with the controller settings, are detailed in Appendix B.2.2.

We evaluate five settings covering plant-parameter mismatch and observation noise. Each run starts from 256 verified safe observations and permits up to 800 additional queries. Each setting is evaluated using 20 algorithm seeds. The results are shown in Fig. 3. BUV-SafeBO achieves near-optimal normalized progress with fewer queries than most baselines while maintaining a high safe-query rate and low cumulative safety violation.

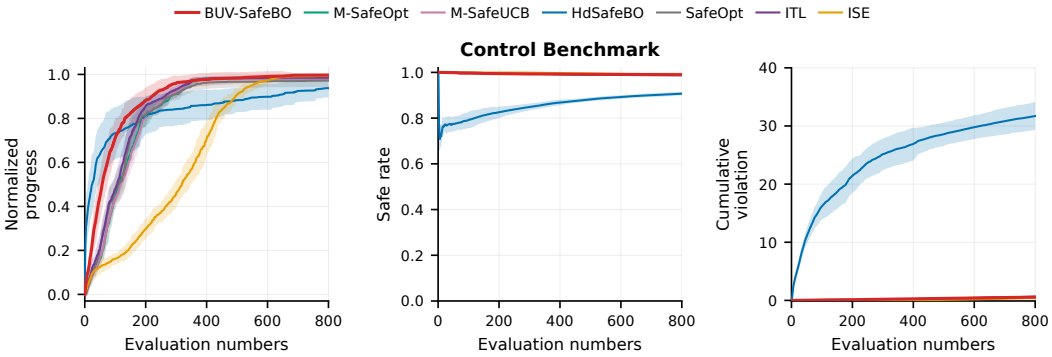


Figure 3: Comparison on the quadrotor trajectory-tracking benchmark across five settings and 20 seeds per setting. Shaded regions indicate one standard error. Except for HdSafeBO, the safe-query rate and cumulative violation curves largely overlap.

4.3 Drug–Dose Optimization on LINCS

We evaluate drug–dose optimization using HMS LINCS 20343 response data (Kalocsay et al., 2023). Each action combines a compound with an ordered dose, with 34 compounds and nine dose levels yielding 306 candidate actions. The objective is to maximize growth suppression in the target tumor cell line, measured using normalized growth-rate inhibition (GR) responses (Hafner et al., 2016; 2017). Safety is defined by requiring the mean GR response of the non-malignant refer-

ence cell line MCF10A to remain at least 0.5. Higher doses can suppress growth in both cell lines, creating a trade-off between target-cell inhibition and preservation of reference-cell growth. The objective and safety functions and benchmark settings are detailed in Appendix B.2.3.

We evaluate 33 target cell lines using 20 algorithm seeds per target, with 34 initial safe observations and up to 272 additional queries per run. As shown in Fig. 4, BUV-SafeBO achieves a favorable balance between optimization progress and empirical safety. Although HdSafeBO and ITL attain higher final normalized progress, they exhibit substantially lower safe-query rates and greater cumulative safety violations.

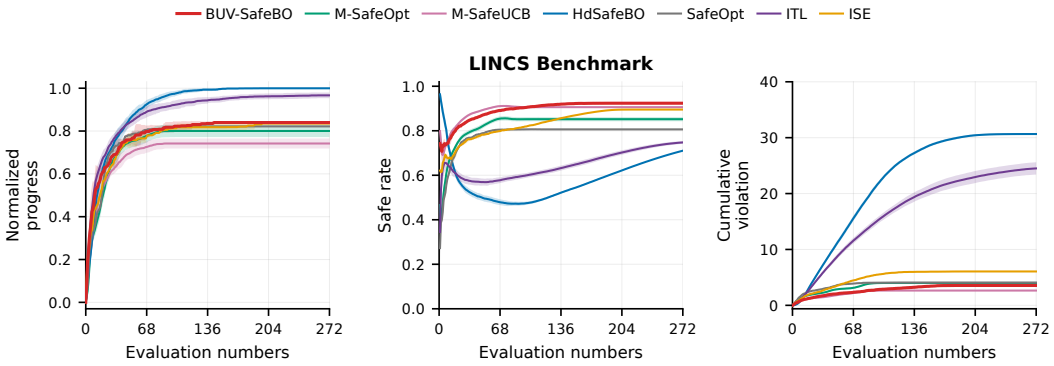


Figure 4: Comparison on the LINCS drug–dose benchmark. Results are averaged over 33 targets and 20 seeds. Shaded regions indicate one standard error across seeds.

4.4 Ablation Studies

We compare three variants: BUV-Uncertainty replaces BUV with boundary entropy for target selection and weighting; BUV-DirectFrontier queries certified frontiers of high-BUV configurations without MI; and BUV-EI selects queries using EI alone. We focus on SCS and quadrotor control to compare sample efficiency over a common budget of 800 queries; LINCS permits only 272 additional queries. Table 1 reports mean normalized progress over 800 query opportunities (AUC_{800}). BUV-SafeBO achieves a higher mean AUC than all three ablation variants on both benchmarks. Variant details and full results are provided in Appendix C.2.

Table 1: Ablation results measured by AUC_{800}, the average normalized progress over 800 query opportunities (mean $\pm$ standard deviation). Bold indicates the highest mean in each row.

Benchmark	BUV-SafeBO	BUV-Uncertainty	BUV-DirectFrontier	BUV-EI
SCS	$\mathbf{0.6876 \pm 0.0500}$	0.6446 ± 0.0510	0.6648 ± 0.0697	0.6555 ± 0.0882
Quadrotor	$\mathbf{0.8863 \pm 0.0259}$	0.8091 ± 0.0272	0.8544 ± 0.0235	0.8522 ± 0.0190

5 Conclusion

In this paper, we introduced the BUV-SafeBO algorithm for safe optimization with a monotone configuration–intensity structure. Its central component, Boundary Unlocking Value, combines a surrogate boundary distribution with optimistic objective gains to quantify the potential value of further safe expansion. BUV-SafeBO translates this value into query selection through BUV-weighted mutual information and expected improvement within the certified safe set. Experiments on spinal cord stimulation, quadrotor control, and drug–dose optimization demonstrate competitive optimization performance with high safe-query rates. Ablation results further support the contribution of boundary-value weighting and information-guided query selection to sample efficiency.

AI USE STATEMENT

During the preparation of this work, we used generative AI tools to provide feedback on research methodology and experimental design, and to assist with the interpretation and presentation of experimental results. Generative AI tools were not used to formulate the core research question, generate the central methodological idea, or determine the experimental hypotheses. We additionally used generative AI tools to assist with scientific writing and editing, as well as with portions of code development and debugging.

All methodological decisions, mathematical derivations, experimental protocols, code implementations, and final conclusions were reviewed and verified by the authors. All experimental results were obtained by running the authors' research code and were checked against the underlying data and evaluation procedures. References, factual statements, and technical content suggested with the assistance of generative AI were verified by the authors against the corresponding original sources. The authors take full responsibility for the final content of this work, including any text, claims, code, or other research artifacts produced or modified with the assistance of generative AI.

REPRODUCIBILITY STATEMENT

The BUV-SafeBO formulation and query-selection procedure are described in Section 3. Supporting derivations and the assumptions underlying the conditional safety analysis are provided in Appendices A.1, A.2, and A.3. Experimental protocols and evaluation metrics are provided in Section 4. Appendix B.1 documents surrogate modeling, hyperparameters, safety settings, acquisition computation, and query protocols. Benchmark-specific definitions and settings are detailed in Appendices B.2.1, B.2.2, and B.2.3. Additional experiments on the Sim4Life dataset, ablation studies, and hyperparameter sensitivity analyses are presented in Appendices C.1.1, C.2, and C.3.

REFERENCES

Felix Berkenkamp, Angela P. Schoellig, and Andreas Krause. Safe controller optimization for quadrotors with Gaussian processes. In *2016 IEEE International Conference on Robotics and Automation*, pp. 491–496, 2016. doi: 10.1109/ICRA.2016.7487170.

Felix Berkenkamp, Andreas Krause, and Angela P. Schoellig. Bayesian optimization with safety constraints: Safe and automatic parameter tuning in robotics. *Machine Learning*, 112:3713–3747, 2023. doi: 10.1007/s10994-021-06019-1.

Alessandro G. Bottero, Carlos E. Luis, Julia Vinogradska, Felix Berkenkamp, and Jan Peters. Information-theoretic safe exploration with Gaussian processes. In *Advances in Neural Information Processing Systems*, volume 35, pp. 30707–30719, 2022.

Alessandro G. Bottero, Carlos E. Luis, Julia Vinogradska, Felix Berkenkamp, and Jan Peters. Information-theoretic safe Bayesian optimization. *Journal of Machine Learning Research*, 27 (145):1–42, 2026.

Eric Brochu, Vlad M. Cora, and Nando de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010. URL `https://arxiv.org/abs/1012.2599`.

Lukas Brunke, Melissa Greeff, Adam W. Hall, Zhaocong Yuan, Siqi Zhou, Jacopo Panerati, and Angela P. Schoellig. Safe learning in robotics: From learning-based control to safe reinforcement learning. *Annual Review of Control, Robotics, and Autonomous Systems*, 5:411–444, 2022. doi: 10.1146/annurev-control-042920-020211.

Marco Capogrosso, Nikolaus Wenger, Stanisa Raspopovic, Pavel Musienko, Janine Beauparlant, Lorenzo Bassi Luciani, Grégoire Courtine, and Silvestro Micera. A computational model for epidural electrical stimulation of spinal sensorimotor circuits. *Journal of Neuroscience*, 33(49): 19326–19340, 2013. doi: 10.1523/JNEUROSCI.1688-13.2013.

Marc Hafner, Mario Niepel, Mirra Chung, and Peter K. Sorger. Growth rate inhibition metrics correct for confounders in measuring sensitivity to cancer drugs. *Nature Methods*, 13(6):521–527, 2016. doi: 10.1038/nmeth.3853.

Marc Hafner, Mario Niepel, Kartik Subramanian, and Peter K. Sorger. Designing drug-response experiments and quantifying their results. *Current Protocols in Chemical Biology*, 9(2):96–116, 2017. doi: 10.1002/cpch.19.

Philipp Hennig and Christian J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13(57):1809–1837, 2012.

José Miguel Hernández-Lobato, Matthew W. Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems*, volume 27, pp. 918–926, 2014.

Jonas Hübotter, Bhavya Sukhija, Lenart Treven, Yarden As, and Andreas Krause. Transductive active learning: Theory and applications. In *Advances in Neural Information Processing Systems*, volume 37, pp. 124686–124755, 2024.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998. doi: 10.1023/A:1008306431147.

Marian Kalocsay, Matthew J. Berberich, Robert A. Everley, Maulik K. Nariya, Mirra Chung, Benjamin Gaudio, Chiara Victor, Gary A. Bradshaw, Robyn J. Eisert, Marc Hafner, Peter K. Sorger, Caitlin E. Mills, and Kartik Subramanian. Proteomic profiling across breast cancer cell lines and models. *Scientific Data*, 10:514, 2023. doi: 10.1038/s41597-023-02355-0.

Johannes Kirschner, Mojmír Mutný, Nicole Hiller, Rasmus Ischebeck, and Andreas Krause. Adaptive and safe Bayesian optimization in high dimensions via one-dimensional subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3429–3438. PMLR, 2019.

Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(8):235–284, 2008.

Arpan Losalka and Jonathan Scarlett. Benefits of monotonicity in safe exploration with Gaussian processes. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pp. 1304–1314. PMLR, 2023.

Arpan Losalka and Jonathan Scarlett. No-regret algorithms for safe Bayesian optimization with monotonicity constraints. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 3232–3240. PMLR, 2024.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Jonas Rothfuss, Christopher Koenig, Alisa Rupenyan, and Andreas Krause. Meta-learning priors for safe Bayesian optimization. In *Proceedings of the 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 237–265. PMLR, 2023.

Jens Schreiter, Duy Nguyen-Tuong, Mona Eberts, Bastian Bischoff, Heiner Markert, and Marc Toussaint. Safe exploration for active learning with Gaussian processes. In *Machine Learning and Knowledge Discovery in Databases*, volume 9286 of *Lecture Notes in Computer Science*, pp. 133–149. Springer, 2015. doi: 10.1007/978-3-319-23461-8_9.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2016. doi: 10.1109/JPROC.2015.2494218.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012. doi: 10.1109/TIT.2011.2182033.

Yanan Sui, Alkis Gotovos, Joel W. Burdick, and Andreas Krause. Safe exploration for optimization with Gaussian processes. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 997–1005. PMLR, 2015.

Yanan Sui, Vincent Zhuang, Joel W. Burdick, and Yisong Yue. Stagewise safe Bayesian optimization with Gaussian processes. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4781–4789. PMLR, 2018.

Matteo Turchetta, Felix Berkenkamp, and Andreas Krause. Safe exploration for interactive machine learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yunyue Wei, Zeji Yi, Hongda Li, Saraswati Soedarmadji, and Yanan Sui. Safe Bayesian optimization for the control of high-dimensional embodied systems. In *Proceedings of the 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4771–4792. PMLR, 2025.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Efficiently sampling functions from Gaussian process posteriors. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10292–10302. PMLR, 2020.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Pathwise conditioning of Gaussian processes. *Journal of Machine Learning Research*, 22(105):1–47, 2021.

Zhaocong Yuan, Adam W. Hall, Siqi Zhou, Lukas Brunke, Melissa Greeff, Jacopo Panerati, and Angela P. Schoellig. Safe-Control-Gym: A unified benchmark suite for safe learning-based control and reinforcement learning in robotics. *IEEE Robotics and Automation Letters*, 7(4):11142–11149, 2022. doi: 10.1109/LRA.2022.3196132.

ZMT Zurich MedTech AG. Sim4Life, n.d. URL `https://sim4life.swiss/`. Accessed: September 26, 2026.

APPENDIX

A ADDITIONAL METHOD DETAILS AND ANALYSIS

A.1 BOUNDARY REPRESENTATION AND BUV

A.1.1 WHY MONOTONE STRUCTURE SIMPLIFIES VALUE ASSESSMENT

For a fixed configuration, consider K candidate intensities $\mathcal{I} = \{I_1 < \cdots < I_K\}$. Without structural restrictions, each candidate may be safe or unsafe, giving up to 2^K possible safety-label patterns. Under monotone safety, a safe intensity implies that all lower intensities are safe. Only $K + 1$ patterns remain: the empty prefix and the prefixes ending at $I_1, \ldots, I_K$. Thus, one boundary variable is sufficient to represent the feasible intensities of each configuration.

This representation directly connects safety uncertainty to accessible objective value. A boundary realization $B_c = b$ identifies the accessible prefix, whose optimistic utility is $U_t(c, b)$. Weighting its gain over the current certified prefix by the boundary probability yields BUV. No monotonicity assumption on the objective is needed, because U_t uses a prefix maximum.

For general feasible regions, an analogous value assessment must account for the joint feasibility of candidate actions and the objective values available within each possible region. Such a quantity can still be approximated by posterior sampling, but a single scalar boundary generally cannot describe the region. The benefit of monotonicity is therefore a compact representation that makes boundary-dependent gains straightforward to evaluate. The count of safety-label patterns illustrates this representational difference; it does not imply that a general method must enumerate all 2^K patterns.

A.1.2 ESTIMATING THE BOUNDARY SURROGATE

Each posterior safety-function sample is mapped to its longest safe prefix, as defined in the main text. The fraction of samples ending at b gives the empirical distribution $\widehat{\pi}_t(c, b)$. A shared GP weight draw across configurations and intensities retains the dependence represented by the finite-feature model, rather than treating safety at different candidates as independent.

Since the GP is not constrained to be monotone, this mapping need not retain every safe intensity predicted by a sample. For example, signs $(+, +, -, +)$ yield a prefix ending at the second intensity. The empirical distribution is consequently a GP-induced boundary surrogate, rather than a posterior explicitly conditioned on monotonicity. Increasing the number of samples reduces Monte Carlo variability for this surrogate, but does not remove the structural approximation.

For completeness, let $\perp$ represent an empty prefix when a sample predicts that the lowest intensity is unsafe. Such samples remain included in the total sample count R and contribute zero extension gain.

A.1.3 INTERPRETING THE BUV SCORE

For a configuration with a nonempty certified prefix, define the sampled extension gain as

$$V_t^{(r)}(c) = \begin{cases} U_t(c, B_c^{(r)}) - U_t(c, m_t(c)), & B_c^{(r)} > m_t(c), \\ 0, & \text{otherwise, including an empty prefix.} \end{cases} \tag{20}$$

Because $U_t(c, b)$ is a prefix maximum, it is nondecreasing in b, so the extension gain in the first case is nonnegative.

Substituting the empirical boundary frequencies defined in the main text into the BUV definition gives

$$\mathrm{BUV}_t(c) = \frac{1}{R} \sum_{r=1}^{R} V_t^{(r)}(c). \tag{21}$$

Thus, BUV is the sample-average optimistic extension gain under the empirical boundary surrogate. A large potential gain contributes little if the boundary surrogate assigns low probability to reach-

ing the corresponding region; extensive safety uncertainty contributes little if the newly accessible intensities offer no improvement over the current prefix.

The reference is the current certified utility of the same configuration. BUV is therefore an optimistic configuration-level extension score, rather than the expected global improvement from a single query. MI and EI subsequently account for query informativeness and immediate objective improvement, respectively.

A.2 Rationale and Derivation of Gaussian Mutual Information

A.2.1 From Posterior Dependence to Information

The acquisition uses an observation at a certified-safe point x to learn about the safety margin at the frontier x_c^+. Under the safety GP, posterior covariance describes the dependence between these two points. This dependence allows a safe query to provide information about an uncertified frontier, including one belonging to a different configuration.

Conditioned on $\mathcal{D}_t$ and the current GP parameters, let $Z_c = g(x_c^+)$ denote the latent frontier safety margin and $Y_x = y_g(x) = g(x) + \epsilon$ the candidate observation, where $\epsilon \sim \mathcal{N}(0, \sigma_{n,g}^2)$ is independent of the latent function and previous observation noise. Define

$$v_c = [\sigma_t^g(x_c^+)]^2, \qquad v_x = [\sigma_t^g(x)]^2 + \sigma_{n,g}^2, \qquad k_{cx} = \kappa_t^g(x_c^+, x). \tag{22}$$

For $v_c > 0$, their squared posterior correlation is

$$\rho_{t,c}^2(x) = \frac{k_{cx}^2}{v_c v_x}. \tag{23}$$

Under the Gaussian surrogate, their joint posterior distribution is

$$\begin{pmatrix} Z_c \\ Y_x \end{pmatrix} \Big| \, \mathcal{D}_t \sim \mathcal{N}\left(\begin{pmatrix} \mu_t^g(x_c^+) \\ \mu_t^g(x) \end{pmatrix}, \begin{pmatrix} v_c & k_{cx} \\ k_{cx} & v_x \end{pmatrix} \right). \tag{24}$$

For $v_c > 0$, Gaussian conditioning gives

$$\mathrm{Var}(Z_c \mid Y_x, \mathcal{D}_t) = v_c - \frac{k_{cx}^2}{v_x} = v_c\big(1 - \rho_{t,c}^2(x)\big), \qquad \rho_{t,c}^2(x) = \frac{k_{cx}^2}{v_c v_x}. \tag{25}$$

Here, $\rho_{t,c}^2(x)$ is the squared posterior correlation between the latent frontier margin and the noisy candidate observation. Unlike covariance alone, it accounts for their marginal variances and the observation noise.

Equation (25) shows that $\rho_{t,c}^2(x)$ is exactly the fraction of frontier variance removed by observing Y_x under the current model. A larger squared correlation therefore implies a greater fractional reduction in frontier uncertainty. Both positive and negative correlations can be informative, while increasing the observation-noise variance reduces the information available from a query when the latent posterior is held fixed.

A.2.2 Mutual Information as Entropy Reduction

The differential entropy of a scalar Gaussian with variance $v > 0$ is $\frac{1}{2}\log(2\pi e v)$. Since the conditional variance in Eq. (25) does not depend on the realized observation, the expected entropy reduction is

$$I_t(c; x) = H(Z_c \mid \mathcal{D}_t) - \mathbb{E}_{Y_x \mid \mathcal{D}_t}[H(Z_c \mid Y_x, \mathcal{D}_t)] \tag{26}$$

$$= \frac{1}{2}\log \frac{v_c}{v_c - k_{cx}^2/v_x} = -\frac{1}{2}\log(1 - \rho_{t,c}^2(x)). \tag{27}$$

Here, H denotes differential entropy and logarithms are natural. This recovers the MI formula in the main text. It establishes a direct connection between posterior dependence and information acquisition: MI is a monotone function of the squared correlation and quantifies the expected reduction in frontier entropy.

With $\sigma_{n,g}^2 > 0$, positive semidefiniteness of the latent posterior covariance implies

$$k_{cx}^2 \le v_c[\sigma_t^g(x)]^2, \qquad 0 \le \rho_{t,c}^2(x) \le \frac{[\sigma_t^g(x)]^2}{[\sigma_t^g(x)]^2 + \sigma_{n,g}^2} < 1. \tag{28}$$

The MI is therefore finite. When $v_c = 0$, the frontier value is already determined under the posterior, so its MI is zero and the entropy formula is not evaluated.

For the same amount of information, learning about the frontier of a configuration with a higher BUV contributes more to the exploration score. BUV thus specifies which configurations offer more valuable potential extensions, while MI measures how informative each safe query is about their frontiers.

This criterion measures information about the selected frontier safety margins, rather than the entire boundary distribution. Reducing frontier uncertainty can support safe expansion, but does not necessarily certify the frontier immediately: certification also depends on the updated posterior mean and safety confidence bound.

A.3 Conditional Safety Analysis

The safety of an executed query follows from the validity of the certified set, rather than from the BUV or MI scores. To make this dependence explicit, assume that the true safety function is nonincreasing in intensity and that initial historical safety certificates are valid. Define the joint lower-bound event over the optimization horizon as

$$\mathcal{E}_T = \bigcap_{t=0}^{T-1} \bigcap_{x \in \mathcal{C} \times \mathcal{I}} \{g(x) \ge \ell_t^g(x)\}. \tag{29}$$

Earlier certificates must either be covered by the same type of event or be justified by the initial safety assumption. All certificates concern the same safety function and threshold. Empty model-certified or historical prefixes contribute no points to the safe set.

Conditional safety statement. On $\mathcal{E}_T$, every executed query selected from $\mathcal{S}_t^{\text{query}} \subseteq \mathcal{S}_t$ is safe.

Justification. For the model-certified prefix, $I \le m_t^{\text{LCB}}(c)$ implies $\ell_t^g(c, I) \ge 0$, and hence $g(c, I) \ge 0$ on $\mathcal{E}_T$. For the historical prefix, the point defining $L_t(c)$ is safe by its initial certificate or, inductively, because it was selected from a previously valid safe set. Monotonicity then gives $g(c, I) \ge g(c, L_t(c)) \ge 0$ for every $I \le L_t(c)$. The union of these two safe prefixes is the prefix ending at $m_t(c) = \max\{m_t^{\text{LCB}}(c), L_t(c)\}$. Therefore $\mathcal{S}_t$ is safe and so is the next selected query, completing the induction. The nonnegative observed-response condition used in $L_t(c)$ only filters previously certified points; it is not itself a proof of safety under observation noise.

If a separate confidence analysis establishes $\Pr(\mathcal{E}_T) \ge 1 - \delta$ and initial certificates are valid surely, the same probability bound applies to the safety of all executed queries. A fixed multiplier β_g alone does not establish this joint coverage across candidates and iterations. The statement above is therefore conditional on certification validity. It does not require the boundary surrogate to be exact, because boundary samples affect query ranking while execution is restricted by the certified set. If that set is empty, no query is authorized by this rule.

A.4 Discrete and Continuous Intensities

A.4.1 BUV with Continuous Intensities

The main algorithm uses a finite ordered intensity set. For a continuous intensity interval $[I_{\min}, I_{\max}]$, suppose $g(c, \cdot)$ is continuous and nonincreasing and its safe set is nonempty. The safe intensities then form a closed prefix with endpoint

$$B_c^{\text{cont}} = \max\{I \in [I_{\min}, I_{\max}] : g(c, I) \ge 0\}. \tag{30}$$

For continuous $u_t(c, \cdot)$, define the optimistic utility accessible up to boundary b as

$$U_t^{\text{cont}}(c, b) = \max_{I \in [I_{\min}, b]} u_t(c, I). \tag{31}$$

Given a boundary distribution $\Pi_t(c, \mathrm{d}b)$ and a certified endpoint $m \in [I_{\min}, I_{\max}]$, the continuous analogue of BUV is

$$\mathrm{BUV}_t^{\mathrm{cont}}(c) = \int_{(m, I_{\max}]} \left[U_t^{\mathrm{cont}}(c, b) - U_t^{\mathrm{cont}}(c, m)\right] \Pi_t(c, \mathrm{d}b). \tag{32}$$

This replaces the discrete probability-weighted sum with an integral over possible boundary locations. The interpretation remains unchanged: each boundary realization contributes the optimistic gain it makes accessible beyond the currently certified prefix, weighted by its probability. The integral can be approximated by averaging these gains over sampled boundaries. A finite intensity grid provides a practical approximation, as used in the main algorithm.

A.4.2 Choosing a Frontier Information Target

On a discrete grid, the first uncertified intensity is the next candidate above the certified endpoint: if $m_t(c) = I_k$ and $k < K$, then $I_t^+(c) = I_{k+1}$. This rule remains applicable when the physical intensity is continuous but the optimizer operates on a finite candidate grid.

In a continuous interval, there is no smallest intensity strictly greater than $m_t(c)$. A continuous extension must therefore specify how far beyond the certified endpoint to place its information target. One option is to introduce a positive offset $\Delta_t(c) > 0$ and set

$$I_t^+(c) = \min\left\{m_t(c) + \Delta_t(c), I_{\max}\right\}, \qquad m_t(c) < I_{\max}. \tag{33}$$

The offset may, for example, represent a prescribed physical intensity increment. The resulting $x_c^+ = (c, I_t^+(c))$ is a local information target beyond the certified endpoint, rather than a uniquely defined first uncertified point. Choosing the offset is an additional design decision for a continuous extension; the implemented algorithm uses the next grid intensity instead. If $m_t(c) = I_{\max}$, no further frontier target exists.

Grid spacing or target offset determines the extent of the prospective extension being examined and affects its posterior dependence with safe query points. In either case, the frontier remains an information target: actual evaluations are restricted to the certified safe set.

B Experimental Details

B.1 Implementation Details

This section describes the implementation details for spinal cord stimulation (SCS), quadrotor trajectory tracking, and LINCS drug–dose optimization, including surrogate modeling, hyperparameter settings, acquisition computation, query protocols, and baseline adaptations. Benchmark-specific candidate spaces, objective and safety functions, and evaluation settings are provided in Appendix B.2.

GP models and preprocessing. For the finite-feature models, quadrotor inputs combine four controller gains with trajectory intensity. LINCS inputs combine physicochemical descriptors of each compound with its ordered dose rank. SCS inputs combine electrode-state encodings with the ordered stimulation-amplitude level. Intensity coordinates are normalized, and the feature representation includes configuration features, intensity terms, their interactions, and random Fourier features.

Independent GP surrogates for the objective and safety functions are updated from accumulated observations at each iteration. The feature representation and regularization remain fixed, while response scales and residual noise estimates are updated. Predictions are transformed back to original response units before safety certification. Confidence bounds use latent posterior uncertainty, while prospective observation variances additionally include the estimated observation noise.

Hyperparameters and safety settings. Across all three benchmarks, BUV-SafeBO uses $R = 64$ posterior function samples, at most $M = 3$ information targets, $\eta = 1$, and $\beta_f = 2$. Baseline implementations follow their original papers and available open-source code, with adaptations to the benchmark domains and evaluation protocols. Table 2 reports the benchmark-specific safety multipliers β_g, which act directly on posterior standard deviations, together with the method-specific safety-accessibility rules. HdSafeBO instead uses an optimistic safety filter with a positive standard-deviation coefficient of 4.

Table 2: Safety confidence multipliers β_g and accessibility rules. HdSafeBO uses an optimistic filter rather than an LCB.

Method	SCS	Quadrotor	LINCS	Safety rule
BUV-SafeBO	4	16	3	LCB prefix + historical certificates
ISE	2	16	2	Pointwise LCB
M-SafeOpt	3	16	3	Pointwise LCB + safe anchors
M-SafeUCB	5	16	5	Confidence-threshold boundary
SafeOpt	2	16	2	Strictly positive LCB
ITL	2	2	2	Intersected confidence bounds
HdSafeBO	—	—	—	Optimistic filter: $\mu_g(x) + 4\sigma_g(x) \geq 0$

Boundary sampling and acquisition. Posterior safety-weight draws are shared across configurations and intensities. Each latent function sample is mapped to its longest safe prefix, yielding the empirical boundary distribution $\widehat{\pi}_t(c, b)$ used in Eq. (11). Samples with an empty safe prefix remain included in the total sample count R and contribute zero extension gain. Information targets are selected from configurations with a nonempty, incomplete certified prefix and positive BUV. Queries remain within the certified safe set. Information scores and EI are normalized separately by subtracting their 5th percentiles, dividing by their 95th–5th percentile ranges, and clipping to $[0, 1]$. Constant scores map to zero. Normalization precedes exclusion of previously observed actions. The EI incumbent is the largest observed objective with a nonnegative observed safety response. Without eligible information targets, BUV-SafeBO selects using normalized EI.

Query protocol. Initial observations are matched across methods and excluded from the additional-query budget. For BUV-SafeBO, query selection is restricted to previously unevaluated points in the certified safe set whenever such points remain; otherwise, repeated queries are allowed. Query responses are retrieved from the precomputed dataset, so repeated queries to the same action return identical responses. Apart from the initial observations, algorithms access response values only at queried actions. Every executed query counts toward the additional-query budget, and safety rates are computed over all executed queries, including repetitions.

B.2 Benchmark Details

This section supplements the main experiments with benchmark-specific response definitions, candidate spaces, and evaluation settings. Surrogate modeling and acquisition implementation are described in Appendix B.1.

B.2.1 Spinal Cord Stimulation

Dataset and candidate space. We use the spinal cord stimulation simulation benchmark introduced by Wei et al. (2025). The dataset provides predicted muscle activation responses from a spinal cord model incorporating finite-element electric-field calculations. Each action combines an electrode configuration c, specifying the arrangement of cathodes and anodes on a 16-contact array, with an ordered stimulation intensity I. The candidate space contains 21,800 configurations and ten intensity levels per configuration, yielding 218,000 actions. Queries retrieve the precomputed responses without additional measurement noise.

Objective and safety functions. Let $a_j(c, I) \in [0, 1]$ denote the normalized activation of muscle j. Following the source benchmark, the objective for target muscle k is its selectivity index:

$$f_k(c, I) = a_k(c, I) - \frac{1}{|\mathcal{N}_k|} \sum_{j \in \mathcal{N}_k} a_j(c, I), \tag{34}$$

where $\mathcal{N}_k$ is the set of neighboring non-target muscles used to evaluate selectivity. This objective rewards target activation while penalizing the mean activation of non-target muscles. Increasing intensity can recruit both groups, so greater intensity need not improve selectivity.

Safety follows the source benchmark's maximum-activation criterion (Wei et al., 2025). Under our zero-threshold convention, the safety function is

$$g(c, I) = 0.95 - \max_j a_j(c, I). \tag{35}$$

Thus, an action is safe when no modeled muscle has normalized activation above 0.95. This is a simulation-benchmark criterion for limiting excessive activation.

Targets and evaluation protocol. The dataset includes six muscle groups: iliopsoas (IL), vastus lateralis and rectus femoris (VL&RF), tibialis anterior (TA), biceps femoris and gluteus maximus (BF&GM), semitendinosus (ST), and gastrocnemius (GA). We evaluate VL&RF and GA as separate optimization targets over the same candidate space. Changing the target changes f_k, while the safety function remains shared. Each target uses 200 initial observations and a budget of 800 additional queries, with five algorithm seeds. Initial observations are matched across methods within each paired run. Results are averaged across the two targets within each seed and then summarized across seeds.

B.2.2 Quadrotor Trajectory Tracking

Task and candidate space. We construct the trajectory-tracking task using Safe-Control-Gym (Yuan et al., 2022). The controller configuration is $c = (K_{p,x}, K_{d,x}, K_{p,h}, K_{d,h})$, containing horizontal and vertical proportional and derivative gains:

$$\begin{aligned} K_{p,x} &\in \{0.12, 0.20, 0.30, 0.40\}, \\ K_{d,x} &\in \{0.15, 0.20, 0.25, 0.30\}, \\ K_{p,h} &\in \{0.625, 0.9375, 1.25, 1.5625\}, \\ K_{d,h} &\in \{0.25, 0.375, 0.50, 0.625\}. \end{aligned} \tag{36}$$

Their Cartesian product gives 256 configurations. Intensity I specifies the number of figure-eight reference-trajectory cycles requested within six seconds. We discretize $I \in [0.5, 3.0]$ into 20 equally spaced levels, yielding $5{,}120$ candidate actions per setting. Increasing intensity requires faster traversal of the reference path and greater acceleration, increasing the demands on thrust and attitude control.

Objective function. The benchmark records provide the position-tracking root-mean-square error $E(c, I)$ in meters and the normalized control effort $U(c, I)$. The latter is the root-mean-square applied thrust divided by the actuator upper limit. To balance trajectory intensity, tracking accuracy, and control effort, we construct the following objective using these quantities:

$$f(c, I) = \frac{I}{3} \exp\left[-\left(\frac{E(c, I)}{0.10\,\mathrm{m}} \right)^2 \right] - 0.05\, U(c, I). \tag{37}$$

The factor $I/3$ normalizes intensity by its maximum value. The exponential factor discounts the intensity reward as tracking error increases, using $0.10\,\mathrm{m}$ as the error scale; at this error, the reward is reduced to e^{-1} of its value under perfect tracking. The coefficient 0.05 weights the control-effort penalty. Consequently, an increase in safely achievable intensity does not necessarily improve the objective.

Safety function. Safety is assessed over the recorded rollout using horizontal position, altitude, pitch, requested thrust, and episode completion. Let $p_x(t)$, $h(t)$, and $\theta(t)$ denote horizontal position, altitude, and pitch, respectively. Let $u_{\mathrm{cmd}}(t)$ be the requested thrust before actuator clipping, and let $u_{\min}$ and $u_{\max}$ be the actuator limits. The safety function is

$$\begin{aligned} g(c, I) = \min\Bigg\{ s_{\mathrm{end}}, \min_{t \in \mathcal{T}_{\mathrm{roll}}} \min\Bigg[& 1 - \frac{|p_x(t)|}{1.5\,\mathrm{m}}, \frac{h(t) - 0.1\,\mathrm{m}}{0.9\,\mathrm{m}}, \\ & \frac{1.9\,\mathrm{m} - h(t)}{0.9\,\mathrm{m}}, 1 - \frac{|\theta(t)|}{30^\circ}, \frac{u_{\max} - u_{\mathrm{cmd}}(t)}{u_{\max} - u_{\min}} \Bigg] \Bigg\}. \end{aligned} \tag{38}$$

Here, $\mathcal{T}_{\mathrm{roll}}$ includes all recorded time steps, including the initial state. The position constraints require $|p_x(t)| \le 1.5\,\mathrm{m}$ and $h(t) \in [0.1, 1.9]\,\mathrm{m}$, while the pitch magnitude is limited to 30°. The altitude margins are normalized by $0.9\,\mathrm{m}$, half the permitted altitude range, and the requested-thrust margin is normalized by the actuator range $u_{\max} - u_{\min}$. The latter constraint requires the requested thrust not to exceed $u_{\max}$. We set $s_{\mathrm{end}} = 1$ for a completed episode and $s_{\mathrm{end}} = -1$ for an early-terminated episode. Thus, $g(c, I) \ge 0$ requires episode completion and satisfaction of all specified constraints at every recorded time step.

Task settings and evaluation protocol. We evaluate five settings: three with mass and pitch-inertia mismatch relative to the nominal controller model, and two with noisy observations. In the noisy settings, independent Gaussian perturbations are added to objective and safety responses. Both responses use a standard deviation of 0.01 in one setting and 0.02 in the other. Perturbations are sampled once per action, remain fixed during optimization, and are shared across methods.

Each run starts from 256 verified safe observations, one at the lowest intensity for each configuration. The additional-query budget is 800, and each setting is evaluated using 20 algorithm seeds.

B.2.3 Drug–Dose Optimization on LINCS

Dataset and candidate space. We use the drug-dose response data from HMS LINCS 20343, which has also been used in breast cancer drug-sensitivity research (Kalocsay et al., 2023). Each action consists of a compound c and an ordered dose I. The evaluation table contains 34 compounds with nine dose levels each, yielding 306 candidate actions for each target cell line. The algorithm uses dose rank as its intensity coordinate; physical concentrations are retained as dataset metadata.

Responses are represented by normalized growth-rate inhibition (GR) values (Hafner et al., 2016; 2017). We use the mean recorded GR response for each compound–dose–cell-line combination.

Objective and safety functions. The task is to suppress growth in the target tumor cell line while keeping the mean GR response of the reference cell line MCF10A at or above the benchmark safety threshold. Increasing the dose can suppress growth in both the target and reference cell lines, potentially lowering the reference-cell GR response below the safety threshold. Let $\overline{\mathrm{GR}}_k(c, I)$ denote the mean GR response of target cell line k. The objective is

$$f_k(c, I) = 1 - \overline{\mathrm{GR}}_k(c, I), \tag{39}$$

so lower target-cell GR corresponds to greater growth suppression and a higher objective value. We use the non-malignant human mammary epithelial cell line MCF10A as the reference for the safety proxy. Its mean GR response must remain at least 0.5, giving

$$g(c, I) = \overline{\mathrm{GR}}_{\mathrm{MCF10A}}(c, I) - 0.5. \tag{40}$$

Targets and evaluation protocol. We evaluate 33 fixed tumor-cell targets as separate optimization tasks. The target-specific objective changes across tasks, while the candidate space and MCF10A safety responses remain shared. Each run starts from 34 safe observations, one at the lowest dose for each compound, and permits up to 272 additional queries. We use 20 algorithm seeds per target. For each seed, results are first averaged equally over the 33 targets and then summarized across seeds.

C Additional Experimental Results

C.1 Additional Benchmark Results

C.1.1 Spinal Cord Stimulation on Sim4Life

We additionally evaluate BUV-SafeBO on a stimulation–response dataset constructed using the Sim4Life simulation software (ZMT Zurich MedTech AG, n.d.). As illustrated in Fig. 5, the construction pipeline starts from a spinal cord anatomical model and a 16-contact paddle lead, computes electric fields for different electrode configurations, simulates afferent fiber recruitment using NEURON, and maps recruitment responses to lower-limb muscle activation. The dataset contains 171 electrode configurations with 100 current intensities each, yielding 17,100 candidate actions. Each action is associated with activation responses for 14 muscles, covering seven muscles on each side.

We use the same configuration–intensity formulation as in Section 4.1. The objective is target-muscle selectivity, as defined in Eq. 34, and the safety function follows Eq. 35, requiring the maximum normalized muscle activation not to exceed 0.95. This experiment evaluates the method on responses generated by a different simulation pipeline and a finer intensity grid than the HdSafeBO SCS benchmark.

We evaluate four target muscles using five algorithm seeds per target. All methods receive the same 200 initial observations and a budget of 800 additional queries per run. Queries are evaluated using

the precomputed simulation responses, and algorithms can access response values only at queried actions. We first average results across targets within each seed and then summarize performance across seeds. The results are shown in Fig. 6. BUV-SafeBO reaches high normalized progress with fewer queries than the compared baselines and maintains near-optimal progress thereafter. Although M-SafeUCB achieves better safety performance, BUV-SafeBO maintains a higher safe-query rate and lower cumulative violation than the remaining baselines, demonstrating a favorable balance between sample efficiency and empirical safety. Sim4Life uses a fine grid of 100 intensity levels per configuration. Some configurations may reach the activation threshold at relatively low intensities, leaving many higher-intensity candidates unsafe. This may contribute to the lower safe-query rate when errors in safety-boundary estimation cause such candidates to be incorrectly certified as safe.

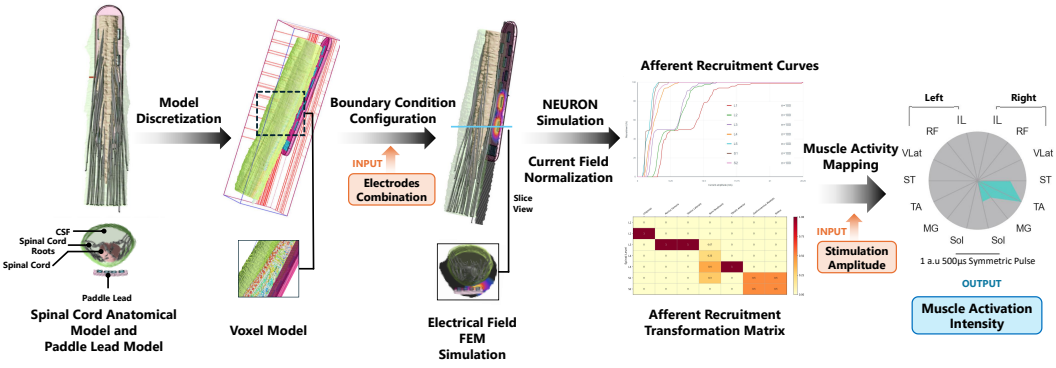


Figure 5: Construction pipeline of the Sim4Life stimulation–response dataset. Electrode configurations and stimulation intensities are mapped to lower-limb muscle activation through electric-field simulation, afferent recruitment modeling, and muscle activity mapping.

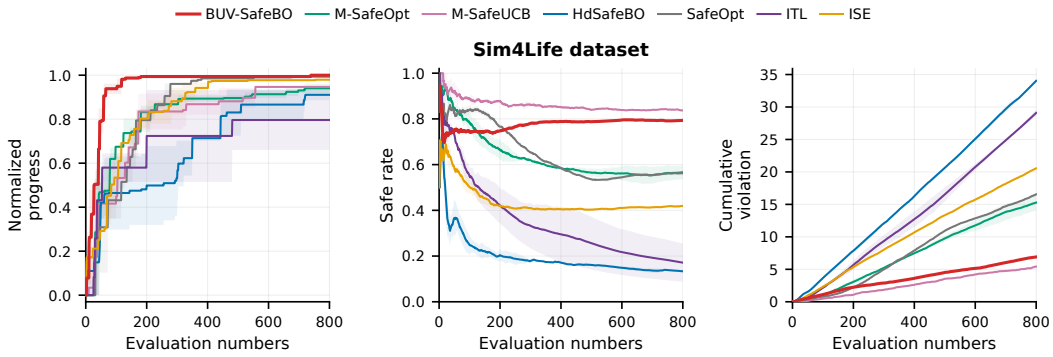


Figure 6: Comparison on the Sim4Life spinal cord stimulation benchmark. Results are averaged over four targets and five seeds. Shaded regions indicate one standard error across seeds.

C.2 Ablation Details and Results

We conduct ablation studies on the quadrotor trajectory-tracking and spinal cord stimulation benchmarks to examine the roles of boundary-value weighting and information-guided safe exploration. These benchmarks share a budget of 800 additional queries; LINCS uses up to 272 additional queries and is not included in this comparison. We compare the full BUV-SafeBO acquisition strategy with three variants. All variants use the same GP modeling framework, safety-certification procedure, initial observations, query budgets, and rules for excluding previously evaluated actions as the full method. Figure 7 presents the corresponding optimization progress, safe-query rates, and cumulative safety violations.

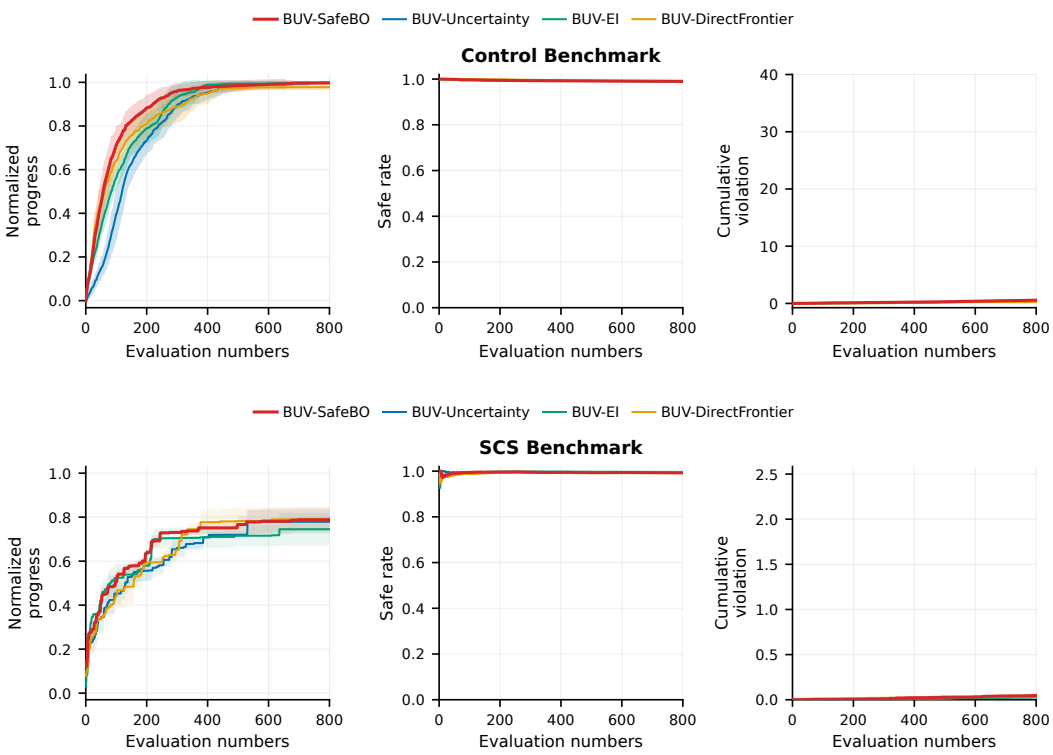


Figure 7: Ablation studies on quadrotor trajectory tracking (top) and spinal cord stimulation (bottom). We compare BUV-SafeBO with uncertainty-weighted MI + EI, Direct BUV, and EI only. Curves show mean performance, and shaded regions indicate one standard error under the aggregation protocol used in the main experiments.

BUV-Uncertainty: uncertainty-weighted MI + EI. This variant replaces BUV with the entropy of the surrogate boundary distribution:

$$H_t(c) = -\sum_b \widehat{\pi}_t(c,b) \log \widehat{\pi}_t(c,b), \tag{41}$$

where the sum includes all boundary outcomes represented by the surrogate distribution. We use $H_t(c)$ both to select up to M unresolved configurations with the highest positive scores and to weight their frontier information. Denoting the resulting target set by $\mathcal{T}_t^H$, the exploration score becomes

$$Q_t^H(x) = \sum_{c \in \mathcal{T}_t^H} H_t(c)\, I_t(c;x). \tag{42}$$

We retain the normalization procedure, EI term, and weight η used by the full method. This comparison tests whether accounting for potential objective gains improves exploration over prioritizing boundary uncertainty alone.

BUV-DirectFrontier: direct configuration-level selection. This variant retains the BUV scores but replaces the combined MI–EI acquisition with direct configuration-level selection. At each iteration, it selects the configuration with the largest BUV among those containing at least one admissible query candidate, and queries the highest-intensity candidate within that configuration. Both selections are restricted to the current query candidate set $\mathcal{S}_t^{\text{query}}$:

$$\begin{aligned} c_t^\star &\in \arg\max_{c:\, \exists I,\, (c,I) \in \mathcal{S}_t^{\text{query}}} \mathrm{BUV}_t(c), \\ I_t^\star &= \max\{I : (c_t^\star, I) \in \mathcal{S}_t^{\text{query}}\}. \end{aligned} \tag{43}$$

The next query is $x_{t+1} = (c_t^\star, I_t^\star)$. This comparison evaluates direct expansion of high-BUV configurations against the full strategy, which considers query informativeness and immediate objective improvement.

BUV-EI: next-query selection using EI only. This variant removes the exploration score $Q_t(x)$ from the acquisition function and selects queries solely by expected improvement:

$$x_{t+1} \in \arg\max_{x \in \mathcal{S}_t^{\text{query}}} \text{EI}_t(x). \tag{44}$$

The EI definition, GP modeling framework, safety-certification procedure, and query-eligibility rules remain unchanged, while the weight η is omitted. This variant can still expand the certified safe set through the safety observations collected at its selected queries, but it does not explicitly value information for future safe expansion. This comparison assesses the benefit of BUV-weighted exploration beyond EI-based selection within the currently certified safe region.

C.3 Hyperparameter Sensitivity

C.3.1 Experimental Protocol

We examine the sensitivity of BUV-SafeBO to the target limit M, the weight η, and the number of posterior function samples R, together with β_g and β_f in the safety lower confidence bound and optimistic objective value. The reference setting is

$$(M, \eta, R, \beta_g, \beta_f) = (3, 1, 64, 4, 2).$$

We use all six SCS target muscles and paired seeds $\{3, 10, 18\}$. Each run starts from 200 historical observations and receives 800 additional queries. Initial observations, the candidate domain, and the feature representation are shared across settings for each target–seed pair.

e additional

Progress AUC is the average normalized progress over the 800 additional queries. We first average results across the six targets within each seed, then report means $\pm$ one standard error across the three seeds; curve shading uses the same standard error. Paired changes are computed from the corresponding seed-wise differences.

C.3.2 Target Limit M

The target limit M controls how many unresolved configurations contribute to the exploration score Q_t. Every iteration in this sweep uses exactly M information targets. Table 3 and Figure 8 show that a small target set works well. $M = 1$ gives the highest mean AUC and final progress, at 0.7258 and 0.8340, compared with 0.7012 and 0.8138 for $M = 3$.

Increasing M to 32 or 96 does not improve mean progress. At $M = 96$, the safe-query rate also decreases to 98.757% and cumulative violation increases to 0.0848. These results favor concentrating information acquisition on a small set of high-BUV frontiers. We retain $M = 3$ as the common reference, with $M = 1$ providing a competitive alternative.

Table 3: Sensitivity to the target limit M, with $(\eta, R, \beta_g, \beta_f) = (1, 64, 4, 2)$.

M	Progress AUC	Final progress	Safe queries (%)	Cum. violation
1	0.7258 ± 0.0519	0.8340 ± 0.0588	99.347 ± 0.070	0.0460 ± 0.0048
3	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
8	0.7040 ± 0.0463	0.8075 ± 0.0388	99.118 ± 0.173	0.0639 ± 0.0136
32	0.6833 ± 0.0343	0.8013 ± 0.0353	99.076 ± 0.054	0.0641 ± 0.0033
96	0.6862 ± 0.0274	0.8113 ± 0.0251	98.757 ± 0.171	0.0848 ± 0.0079

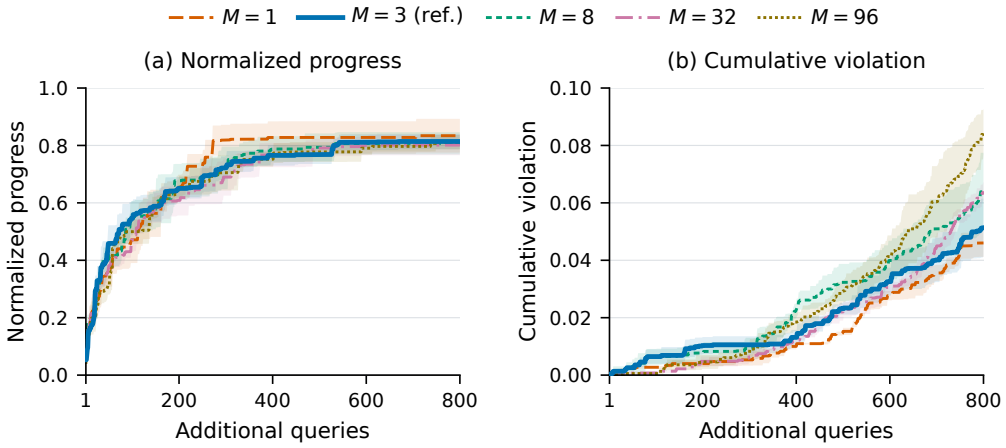


Figure 8: Effect of the target limit M on normalized progress and cumulative safety violation during optimization.

C.3.3 Weight η

The weight η scales the normalized expected-improvement term in $\alpha_t(x) = \widetilde{Q}_t(x) + \eta\widetilde{\mathrm{EI}}_t(x)$. At $\eta = 0$, acquisition uses only $\widetilde{Q}_t$ whenever information targets are available.

As shown in Table 4 and Figure 9, adding EI markedly improves progress. Mean AUC rises from 0.3478 at $\eta = 0$ to 0.7012 at $\eta = 1$, while final progress rises from 0.4292 to 0.8138. The smaller positive weights $\eta = 0.15$ and 0.5 give intermediate performance. Among the tested values, $\eta = 1$ has the best mean performance on all four reported metrics. Increasing η to 2 produces similar results, with a paired AUC change of -0.0120 ± 0.0099. This supports using $\eta = 1$ to combine safe expansion with improvement within the current safe set.

Table 4: Sensitivity to the weight η, with $(M, R, \beta_g, \beta_f) = (3, 64, 4, 2)$.

η	Progress AUC	Final progress	Safe queries (%)	Cum. violation
0	0.3478 ± 0.0260	0.4292 ± 0.0265	98.861 ± 0.093	0.0684 ± 0.0061
0.15	0.5740 ± 0.0658	0.7081 ± 0.0450	98.937 ± 0.021	0.0790 ± 0.0083
0.5	0.6272 ± 0.0302	0.7641 ± 0.0595	99.069 ± 0.046	0.0678 ± 0.0024
1	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
2	0.6892 ± 0.0171	0.7979 ± 0.0192	99.188 ± 0.130	0.0565 ± 0.0067

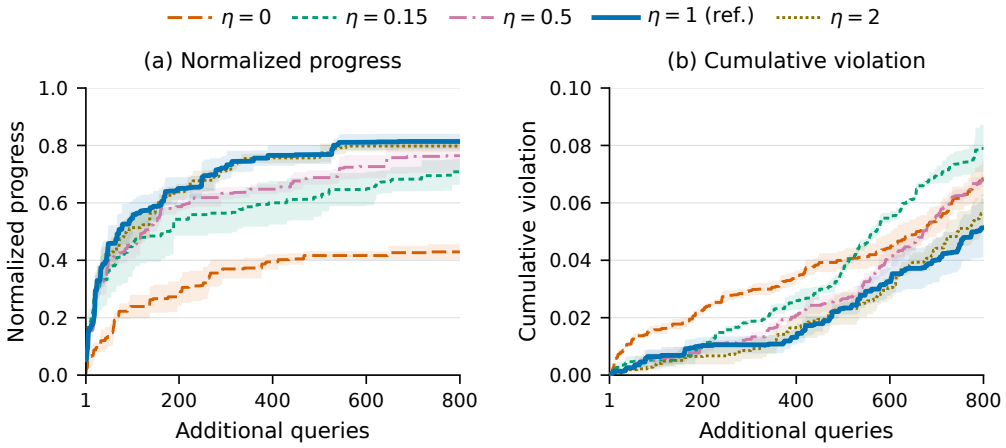


Figure 9: Effect of the weight η on normalized progress and cumulative safety violation during optimization.

C.3.4 Number of Posterior Function Samples R

The number of posterior function samples R determines how many joint functions are drawn from the safety GP to estimate the boundary distribution. Each draw is shared across configurations and intensities. Table 5 shows that online performance changes modestly across the tested values. $R = 16$ gives the highest mean progress, but its safe-query rate is lower than that of $R = 64$ by 0.181 ± 0.061 percentage points. $R = 128$ yields a small mean AUC gain of 0.0090 ± 0.0156 over $R = 64$, whereas $R = 256$ gives lower mean progress. $R = 64$ has the highest mean safe-query rate, and $R = 32$ has the lowest mean cumulative violation.

Table 5: Sensitivity to the number of posterior function samples R, with $(M, \eta, \beta_g, \beta_f) = (3, 1, 4, 2)$.

R	Progress AUC	Final progress	Safe queries (%)	Cum. violation
16	0.7119 ± 0.0443	0.8449 ± 0.0226	99.118 ± 0.025	0.0644 ± 0.0041
32	0.6729 ± 0.0474	0.7965 ± 0.0425	99.264 ± 0.090	0.0482 ± 0.0068
64	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
128	0.7102 ± 0.0129	0.8234 ± 0.0052	99.278 ± 0.076	0.0529 ± 0.0060
256	0.6833 ± 0.0271	0.7892 ± 0.0208	99.201 ± 0.039	0.0550 ± 0.0025

C.3.5 Parameter β_g

The parameter β_g controls the width of the safety lower confidence bound $\mu_t^g - \beta_g \sigma_t^g$. Table 6 varies β_g with $\beta_f = 2$. Increasing β_g from 2 to 4 raises the safe-query rate from 95.750% to 99.299% and reduces cumulative violation from 0.3964 to 0.0513. Mean AUC also increases, from 0.6769 to 0.7012.

At $\beta_g = 6$, the safe-query rate reaches 99.868%, while mean AUC decreases to 0.6581. The mean fraction of candidates certified safe falls from 28.319% to 14.378% and 5.752% as β_g increases from 2 to 4 to 6. The tighter certification reduces violations but also limits the region available for optimization.

Table 6: Sensitivity to β_g at fixed $\beta_f = 2$, with $(M, \eta, R) = (3, 1, 64)$.

β_g	Progress AUC	Final progress	Safe queries (%)	Cum. violation
2	0.6769 ± 0.0209	0.8308 ± 0.0284	95.750 ± 0.060	0.3964 ± 0.0077
4	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
6	0.6581 ± 0.0423	0.7603 ± 0.0411	99.868 ± 0.007	0.0091 ± 0.0020

C.3.6 Parameter β_f

The parameter β_f controls the contribution of posterior uncertainty to the optimistic objective value $\mu_t^f + \beta_f \sigma_t^f$ used in BUV. At fixed $\beta_g = 4$, the three values of β_f give similar mean AUC, ranging from 0.6969 to 0.7046 (Table 7). $\beta_f = 4$ achieves the highest mean final progress, while $\beta_f = 2$ gives the highest mean safe-query rate and lowest mean cumulative violation.

Table 7: Sensitivity to β_f at fixed $\beta_g = 4$, with $(M, \eta, R) = (3, 1, 64)$.

β_f	Progress AUC	Final progress	Safe queries (%)	Cum. violation
1	0.6969 ± 0.0394	0.8373 ± 0.0345	99.188 ± 0.171	0.0609 ± 0.0186
2	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
4	0.7046 ± 0.0351	0.8387 ± 0.0325	99.174 ± 0.080	0.0549 ± 0.0024

C.3.7 Joint Comparison of M and η

Table 8 compares the reference $(M, \eta) = (3, 1)$ with the additional joint setting $(96, 0.15)$. The reference achieves higher mean progress and safe-query rate, with lower cumulative violation. Its paired gains are 0.1973 ± 0.0281 in AUC and 0.2409 ± 0.0572 in final progress. These improvements hold across all three seeds after averaging the target muscles. This comparison favors the reference combination; the separate sweeps above show how changing each parameter contributes to performance.

Table 8: Joint comparison of M and η, with $(R, \beta_g, \beta_f) = (64, 4, 2)$.

(M, η)	Progress AUC	Final progress	Safe queries (%)	Cum. violation
$(3, 1)$	0.7012 ± 0.0267	0.8138 ± 0.0268	99.299 ± 0.039	0.0513 ± 0.0101
$(96, 0.15)$	0.5040 ± 0.0499	0.5729 ± 0.0495	98.208 ± 0.036	0.1079 ± 0.0023