

# Lipschitz Based Lower Bound Construction for Surrogate Optimization

Mohammadsina Almasi<sup>a</sup>, Hadis Anahideh<sup>a</sup>, Jay M. Rosenberger<sup>b</sup>,

<sup>a</sup>*University of Illinois at Chicago, 842 W Taylor Street, Chicago, 60607, IL, USA*

<sup>b</sup>*University of Texas at Arlington, P.O. Box 19017, Arlington, 76019, TX, USA*

---

## Abstract

Bounds play a vital role in guiding optimization algorithms by enhancing convergence, improving solution quality, and quantifying optimality gaps. While Lipschitz-based lower bounds are well-established, their effectiveness is often constrained by the function's topological properties. To address these limitations, we propose an approach that integrates nonlinear distance metrics with surrogate approximations, yielding more adaptive and accurate lower bounds. A key aspect of our methodology lies in the flexibility of the chosen distance metric, which can be adapted to various function behaviors. In particular, we explore sublinear and superlinear metrics under the Hölder continuity assumption, demonstrating their capacity to capture local function characteristics beyond the scope of conventional linear bounds. Empirical evaluations on diverse benchmark test problems show that our approach surpasses both standard Lipschitz-based and statistical methods in producing high-quality lower bounds. We further employ these refined bounds as an acquisition function within surrogate optimization, a common technique for expensive black-box problems wherein a surrogate model approximates the true objective function. By leveraging these bounds to balance exploration and exploitation, we effectively prioritize evaluations in regions with high potential for improvement. Overall, our framework not only offers more accurate and flexible lower bound estimates, but also acts as a robust acquisition strategy that expedites convergence to near-optimal solutions in black-box optimization.

*Keywords:*

Surrogate Optimization, Lipschitz Continuity, Lower Bound Construction, Nonlinear Distance Metrics

---

## 1. Introduction

*Global optimization* problems are ubiquitous in various fields of science and engineering, ranging from aerospace and mechanical engineering to finance and logistics [25, 53]. These problems often involve complex objective functions, nonlinear constraints, and high-dimensional search spaces, making them particularly challenging to solve [22, 49]. Consequently, developing efficient opti-

mization algorithms that can find a global optimum is vital for many practical applications.

One approach to solving global optimization problems is to leverage lower<sup>1</sup> bounds to guide the optimization process and narrow down the search space. Lower bounds play a crucial role in ensuring convergence to a global minimum within a specified tolerance [38, 80]. They are particularly useful in constructing efficient *branch-and-bound* algorithms, where techniques such as *interval arithmetic* [28, 39] and *statistical lower bounds* [31] have proven effective in obtaining tight bounds that enhance computational efficiency.

Beyond their role in structured algorithms, including branch-and-bound, lower bounds are also instrumental in guiding search strategies when function evaluations are costly or limited. Various optimization frameworks rely on lower bounds to improve search efficiency by avoiding unnecessary evaluations in unpromising regions of the search space. One of the main domains that can significantly benefit from lower bound properties in effectively searching the solution space is *surrogate optimization (SO)*. The premise of surrogate optimization is that the cost to evaluate a point  $\mathbf{x}$  in the objective function  $f$  is expensive [29, 57, 72]. Specifically, a typical surrogate optimization approach repeatedly iterates over the following three steps.

- For a given set of *evaluated points*  $\mathbf{x}^1, \dots, \mathbf{x}^N$  for which the function values  $f(\mathbf{x}^1), \dots, f(\mathbf{x}^N)$  are known, construct a surrogate model.
- Given the surrogate model and certain locations of the evaluated points, select a set of candidate points.
- Evaluate the selected candidate points in the objective function  $f$ .

In such settings, lower bounds provide valuable information for balancing exploration (selecting points in regions with few or only distant evaluated points) and exploitation (selecting promising points based upon the surrogate) trade-off, helping to prioritize the candidate selection in the second step of the SO algorithm for the function evaluations in the third.

In this research, we propose an estimated lower bound function on the objective function  $f$  that draws upon *Lipschitz continuity* [9] and *statistical inference* [12]. Estimating the Lipschitz constant can be challenging, particularly when the function is not known explicitly and can only be accessed through an expensive experiment or estimated using a surrogate model [65, 30]. In recent years, several techniques have been proposed for estimating the Lipschitz constant using surrogate models and linear distance metric functions [43, 44, 51]. These techniques aim to improve the accuracy and applicability of Lipschitz estimation in various domains and scenarios. Motivated by this line of work, we leverage local information from evaluated points to guide the optimization process. Specifically, by adjusting the estimate of the Lipschitz constant, the

---

<sup>1</sup>While the focus of this paper is on minimization problems and the use of lower bounds, the same discussion can be extended to maximization problems and the use of upper bounds.

algorithm can balance between local exploitation and global exploration, effectively narrowing down the search space based on information from nearby evaluated points. This approach is akin to utilizing nearest neighbor information to inform decisions about where to sample next, thereby enhancing the efficiency of the optimization process.

The estimated lower bound function developed in this research has a similar structure to an estimated Lipschitz lower bound function, which uses an estimated Lipschitz constant and the distance to the nearest evaluated point, but also makes use of the surrogate model described in the aforementioned first step of the general surrogate optimization approach. Note that the surrogate model enhances the traditional approach by incorporating information from all evaluated points rather than focusing solely on the nearest evaluated point. This comprehensive integration provides a more global perspective on the function’s behavior, allowing for a richer and more robust understanding of the underlying dynamics. By aggregating data from the entire set of evaluated points, the surrogate model refines the local estimation of the Lipschitz constant and contributes to a more accurate and resilient estimation of the lower bound in complex optimization landscapes. Finally, instead of only considering a Euclidean distance metric, as in a traditional estimated Lipschitz lower bound estimate, we explore alternative distance metrics, including *superlinear* and *sublinear* metrics, to provide either more conservative or tighter estimated bounds, respectively. In this context, the distribution of evaluated points is both uncertain and dynamic; its frequency evolves with the continuous addition of new points. This variability, combined with the inherent challenges in precisely estimating actual distances, significantly affects the estimated bound, making it highly sensitive to the chosen distance metric. Consequently, it becomes necessary to carefully determine whether a tighter, more aggressive bound or a more conservative bound is most appropriate. The theoretical proofs, particularly Theorem 5.2, explicitly support this point by demonstrating that a superlinear distance metric yields more conservative bounds when evaluated points are sufficiently spaced, whereas a sublinear distance metric produces tighter, more aggressive bounds. This critical distinction guides the selection of the distance metric based on the desired level of conservatism in the bound estimation process.

The remainder of this paper is organized as follows: Section 2 reviews recent advancements in lower bound and Lipschitz constant estimation and their role in surrogate optimization. Section 3 outlines our contributions and the research gap addressed. Section 4 introduces the essential theoretical concepts, while Section 5 details our methodology for estimating the Lipschitz constant via surrogate approximations using various distance metrics and constructing improved lower bounds. Section 6 presents experimental results across optimization benchmarks, including its use as an acquisition function in surrogate optimization. Finally, Section 7 concludes with key insights and future directions.

## 2. Related Literature

Lower bounds play a crucial role in global optimization algorithms by guiding the search for optimal solutions [24, 33], accelerating convergence [40], and helping to avoid suboptimal regions [55, 82]. Beyond accelerating the search, lower bounds also enable the computation of optimality gaps (i.e., an upper bound on the difference between the current best solution and the true global optimum [47, 54]) and can serve as stopping criteria in practice once a sufficiently small gap is reached. Moreover, they are central to identifying or pruning regions of the feasible domain; for instance, in branch-and-bound frameworks, lower bounds help prune unpromising nodes and focus computational effort on regions more likely to contain a global optimum [21, 83]. Such pruning has been extensively explored for combinatorial optimization problems as well, where these bounds act as a filter to eliminate branches unlikely to yield better solutions [59, 70, 71].

Various techniques have been developed for constructing lower bounds on the optimal objective value, such as *convex relaxation* [48, 63], *interval analysis* [14, 50], *statistical inference* [66], and *Lipschitz continuity* [45]. Among these, using the Lipschitz constant for lower-bound construction is a well-established technique that improves computational efficiency in a variety of optimization contexts, such as branch-and-bound algorithms [13, 83], Lipschitzian optimization [61, 62], and black-box optimization [51, 64].

However, estimating a Lipschitz constant in many black-box optimization scenarios with highly non-uniform or complex functions may require the use of surrogate models to mitigate the cost of function evaluations [18, 60]. In such settings, using a linear distance metric like the Euclidean distance to assess similarity between sampled points often yields less accurate Lipschitz constants estimates and ultimately overly conservative bounds [35, 46]. This limitation reduces the practical accuracy of Lipschitz constant estimates derived from surrogate models. To address this issue, recent studies have adopted Hölder continuity, which extends the traditional Lipschitz framework by incorporating non-linear distance metrics, thereby offering a more flexible and precise approach to capturing complex function behavior [2, 78]. This flexibility is particularly helpful in scenarios without global smoothness of a function but where local function structure can still be exploited.

Recent advancements integrate function smoothness assumptions into surrogate modeling frameworks, thus merging Lipschitz or Hölder continuity with data-driven modeling [23, 79]. Notably, the choice of distance metric, which is often assumed to be the  $L^2$ -norm, can also be adapted to different norms (e.g., superlinear, linear, sublinear) to better capture region-specific function variation [81]. Although references that explicitly test multiple distance metrics are scarce, the rationale is that non-Euclidean metrics may yield less conservative bounds, especially when the function’s behavior varies dramatically across dimensions or exhibits sharp local peaks [1, 32].

Given the expensive evaluations often seen in black-box settings (e.g., engineering simulations or high-fidelity models), SO offers a cost-effective alternative



by substituting a computationally cheap approximation  $\hat{f}$  in place of the true objective [29, 57, 72]. Within SO, the use of lower bounds is twofold:

(i) Pruning or skipping evaluations in regions predicted (by the surrogate) to have little chance of improving the objective, thus accelerating convergence [56, 73]. (ii) Guiding acquisition functions as the mechanisms that select where to sample next by considering not just the predicted mean and uncertainty, but also a data-driven estimate of how low the function could potentially be [20, 74].

A recently developed research area applies Lipschitz-based acquisition functions in SO, directly translating Lipschitz continuity assumptions into selection criteria for promising sampling locations [10, 41]. However, when functions exhibit high variability or partial smoothness, Lipschitz or Hölder-based approaches can offer better quality bounds [3, 7], improving the exploration-exploitation trade-off. For instance, if certain subregions have gentler slopes, a more flexible bounding strategy can push the search aggressively there without imposing an overly conservative constant estimate. Recent efforts also explore data-driven Lipschitz estimates to guide where the model is likely to see significant improvement [17, 42].

Despite these advancements, there remains a critical need for methods that (i) further refine local smoothness properties (potentially via alternative distance metrics) and (ii) integrate such adaptive bounding within SO in a seamless, data-driven manner. Although some researchers have ventured into localized Lipschitz/Hölder strategies [15, 36], they often still rely on the  $L^2$ -norm or assume a single exponent  $\alpha$ , which may be suboptimal in problems with markedly different behaviors across the domain.

This gap motivates the approach proposed in this research. Our method aims to (i) explore nonlinear distance metrics beyond the Euclidean norm for enhanced bound accuracy, and (ii) employ these adaptive bounds within the sampling core of surrogate optimization to balance exploration and exploitation in expensive black-box problems. By tailoring local smoothness estimates to real-time data from previously evaluated points, our approach more accurately captures regional variability than traditional global bounds, thereby yielding more efficient and robust search strategies.

### 3. Contribution

While previous studies have dedicated efforts to constructing lower bounds using Lipschitz constant estimations, their effectiveness and accuracy across different problem domains remain underexplored, a gap this study addresses through a systematic theoretical and empirical analysis.

In particular, existing studies often focus on linear or piecewise linear approximations for Lipschitz constant estimations [4, 51], neglecting the potential benefits of nonlinear techniques, such as *nonlinear distance metrics*. Exploring these alternative metrics in Lipschitz constant estimation can enhance the accuracy and efficiency of lower bounds, particularly for functions with nonlinear characteristics or complex problem domains. Consequently, optimization algorithms guided by these nonlinear lower bounds can explore the search space

more efficiently, avoiding unnecessary evaluations in unpromising regions and focusing on potentially superior solution subspaces.

In this research, we present a new method to develop an estimated lower bound function  $\hat{f}_{lb}$  that is expected to no greater than a function  $f$  over the domain of  $f$ , even though  $f$  has only been evaluated on a small discrete set of points  $\mathbf{x}^1, \dots, \mathbf{x}^N$ . The estimated lower bound function makes use of the structure of an existing surrogate  $\hat{f}$  and Hölder-style nonlinear distance metrics. To our knowledge, only Bian et al. [10] have considered surrogates to estimate lower bounds, but unlike this research, they did not consider nonlinear distance metrics. In addition, we leverage local information from evaluated points to capture non-uniform smoothness or varying sensitivity of the function across different regions of the search space. Specifically, by adjusting the estimate of the Lipschitz constant, the algorithm can balance between local exploitation and global exploration, effectively narrowing down the search space based on information from nearby evaluated points. We prove, under certain assumptions, that the lower bounds estimated using superlinear ( $\hat{f}_{lb}^{sup}(\mathbf{x})$ ), linear ( $\hat{f}_{lb}^{lin}(\mathbf{x})$ ), and sublinear ( $\hat{f}_{lb}^{sub}(\mathbf{x})$ ) distance metrics follow a specific order:  $\hat{f}_{lb}^{sup}(\mathbf{x}) < \hat{f}_{lb}^{lin}(\mathbf{x}) < \hat{f}_{lb}^{sub}(\mathbf{x})$ . This ordering provides insights into the tightness and conservatism of the estimated lower bounds. To study the quality of the estimated lower bounds, we introduce a new evaluation metric that considers both point-wise evaluations—comparing reliability at individual data points—and comparative analyses across different estimates. Using the evaluation metric, we show that the estimated lower bounds developed in this research outperform a traditional Lipschitz estimated lower bound and a statistical lower bound based upon Working–Hotelling [58].

Furthermore, we apply the proposed lower-bound estimation method within a surrogate optimization framework, incorporating nonlinear distance metrics to refine the selection of candidate points. By choosing distance metrics that align with the topological characteristics of the objective function, we enhance both convergence speed and solution quality. Our empirical results demonstrate that this approach outperforms standard SO baselines, including Upper Confidence Bound (UCB) and Expected Improvement (EI). These findings highlight the potential of Lipschitz-based lower bounds as a more effective and adaptive tool for guiding the search process in complex optimization problems.

## 4. Background

### 4.1. Lipschitz and Hölder Continuity:

Lipschitz continuity is a fundamental mathematical concept, providing a bound on how rapidly a function can change over its domain, ensuring controlled variations. Common practices in optimization frameworks often try to adapt Lipschitz continuity as a global assumption. Specifically, consider a real-valued function  $f : \mathbb{R}^d \rightarrow \mathbb{R}$ , there exists a constant  $L$  such that for any two points  $\mathbf{x}^i$  and  $\mathbf{x}^j$  in the domain, the following inequality holds:

$$|f(\mathbf{x}^i) - f(\mathbf{x}^j)| \leq L \|\mathbf{x}^i - \mathbf{x}^j\| \quad (1)$$

where  $\|\mathbf{x}^i - \mathbf{x}^j\|$  denotes the Euclidean distance (or  $L^2$ -norm) between the points [61, 77].

Another generalization of Lipschitz continuity is the Hölder condition, which avoids reliance on a strictly uniform linear norm. This alternative framework relaxes the rigidity of the Lipschitz requirement by introducing an exponent that better accommodates varied functional changes. Specifically, a function  $f$  satisfies the Hölder condition if there exist constants  $L > 0$  and  $\alpha \in (0, 1]$  such that for every  $\mathbf{x}^i, \mathbf{x}^j$  in its domain,

$$|f(\mathbf{x}^i) - f(\mathbf{x}^j)| \leq L \|\mathbf{x}^i - \mathbf{x}^j\|^\alpha \quad (2)$$

This formulation offers a more adaptable mechanism for bounding function variations, particularly when the function does not exhibit uniform smoothness [67].

By integrating the Hölder condition into lower-bound estimation, we can improve the flexibility of our framework, allowing for alternative distance metrics that better capture function behavior. Specifically, the superlinear, linear, and sublinear distance metrics discussed in Section 5.2 can be seen as practical implementations of the Hölder condition, where different values of  $\alpha$  determine the appropriate rate of function variation.

#### 4.2. Surrogate Optimization (SO):

SO methods are widely used global optimization methods when the true objective function,  $f$ , is either unavailable or computationally expensive to evaluate across the entire input space [26]. The general procedure usually follows Algorithm 1.

---

#### Algorithm 1 Surrogate Optimization Algorithm

---

- 1: Sample initial design space  $\mathcal{I} = \{\mathbf{x}^i \in \mathbb{R}^d \mid \forall i = 1, \dots, N\}$
  - 2: Evaluate initial data set  $\mathcal{I}$ ,  $f(\mathbf{x}^i), \forall \mathbf{x}^i \in \mathcal{I}$
  - 3: **while** Termination criteria are unsatisfied **do**
  - 4:     Construct a surrogate model on  $|\mathcal{I}|$  evaluated points,  $\hat{f}$
  - 5:     Use acquisition function to search for new candidate points  $P$  based upon  $\hat{f}$  and  $\mathcal{I}$
  - 6:     Evaluate selected candidate points,  $P$ ,  $f(\mathbf{x}^k), \forall \mathbf{x}^k \in P$
  - 7:     Update the collection of already evaluated data points,  $\mathcal{I} = \mathcal{I} \cup P$
  - 8: **end while**
  - 9: Return the BSMS,  $\mathbf{x} \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{I}} f(\mathbf{x})$
- 

SO methods begin by sampling an initial design space,  $\mathcal{I} = \{\mathbf{x}^i \in \mathbb{R}^d \mid \forall i = 1, \dots, N\}$  and evaluating the function values at these points,  $f(\mathbf{x}^i)$ , to construct an initial dataset. This dataset serves as the foundation for building a surrogate model,  $\hat{f}$ , which approximates the objective function using the available evaluations.

Once the surrogate model is built, an acquisition function is used to guide the selection of new candidate points,  $P$ , that balance exploitation—leveraging information from the surrogate—and exploration—considering the spatial distribution of evaluated points. Two commonly used acquisition functions that we

consider as baselines in this study include Expected Improvement (EI) and Upper Confidence Bound (UCB). EI selects the next evaluation point based on the expected gain over the current best-observed value, leveraging both exploitation and exploration by favoring points with high predictive uncertainty and potential improvement. Mathematically, EI evaluates the probability-weighted improvement over the best-known function value [84]. On the other hand, UCB balances exploration and exploitation by constructing an upper confidence bound around the surrogate model’s prediction, typically selecting points with the highest upper bound value to ensure sufficient exploration [76]. Both methods rely on the statistical properties of the surrogate model, often Gaussian processes, to make informed decisions about where to sample next. The selected candidate points,  $P$ , are then evaluated using the true objective function, and the newly obtained function values,  $f(\mathbf{x}^k), \forall \mathbf{x}^k \in P$  are added to the dataset, updating  $\mathcal{I}$ . This iterative process continues until a termination criterion is met, which can be either a predefined maximum number of expensive function evaluations or the expected improvement of the best sampled mean solution (BSMS) [5]. By iteratively refining the surrogate model and updating the evaluated dataset, SO aims to efficiently approximate the global optimum of the objective function while minimizing the number of costly evaluations. The algorithm ensures that the search process is both computationally feasible and strategically guided toward optimal solutions.

SO can benefit significantly from lower bounds in guiding the selection of new evaluation points, improving both sampling efficiency and search effectiveness. By incorporating lower bounds, SO can better prioritize candidate points, ensuring that regions with higher potential for improvement are explored while avoiding unnecessary evaluations in less promising areas. In this study, we develop an adaptive lower-bound estimation method tailored for SO, integrating alternative distance metrics with a surrogate model-driven strategy to estimate function variation dynamically, leading to a more flexible and adaptive lower-bound formulation. By incorporating superlinear, linear, and sublinear distance metrics, we refine the accuracy of lower-bound estimates, making them more representative of the true function landscape.

#### 4.3. Statistical Lower Bounds:

While Lipschitz-based bounds leverage function smoothness to impose deterministic constraints, statistical lower bounds provide an alternative approach rooted in data-driven estimation techniques. One such method is the Working-Hotelling (WH) confidence bound, which constructs simultaneous confidence intervals to establish lower-bound estimates for function values across different points in the search space [58]. The WH bound is formulated as:

$$\hat{f}(\mathbf{x}) \pm W \hat{s}_{\hat{f}(\mathbf{x})} \quad (3)$$

where  $\hat{f}(\mathbf{x})$  represents the estimated mean response at point  $\mathbf{x}$ ,  $\hat{s}_{\hat{f}(\mathbf{x})}$  is the standard error of  $\hat{f}(\mathbf{x})$ , and  $W$  is a constant determined by the number of levels and

the desired confidence level. Unlike Lipschitz bounds, which impose a spatially uniform constraint on function variation through a fixed constant  $L$ , the WH approach derives its bounds relative to the estimated function values rather than a geometric assumption of function behavior [12, 66]. This key difference allows statistical lower bounds to be constructed without explicitly requiring distance-based smoothness assumptions, making them applicable across a broader range of function types.

However, statistical lower bounds can be more conservative than Lipschitz-based approaches, as they depend on variance estimates from the surrogate model. Additionally, while Lipschitz bounds inherently adapt to local function smoothness through distance-based formulations, WH bounds focus on capturing variability across a dataset rather than explicitly modeling function variation over spatial distances.

In this study, we compare WH-based lower bounds with Lipschitz-derived lower bounds using different distance metrics, examining their ability to provide informative estimates for function behavior. Our analysis highlights how incorporating alternative distance metrics in Lipschitz-based lower bounds can outperform traditional statistical bounds in optimization scenarios, particularly when function smoothness properties can be effectively captured.

Moreover, the proposed bound is likely to outperform the statistical bound because the reference concerning the center of the data considered in statistical bounds may not be appropriate for an optimization framework. Sampled data within an optimization algorithm often has poorly distributed evaluated points, rendering the center of data an ill-equipped descriptor of certain regions [52, 85].

## 5. Methodology

In this study, we investigate the use of nonlinear distance metrics to estimate the Lipschitz constant and construct adaptive lower bounds for the objective function. Our approach leverages alternative distance formulations to improve the accuracy and flexibility of Lipschitz-based lower-bound estimation, enabling more effective guidance in optimization algorithms.

### 5.1. Problem Formulation

Let  $f : \mathbb{R}^d \rightarrow \mathbb{R}$  represent the objective function to be minimized (or maximized) in an optimization problem, let  $\mathcal{S}$  represents the *solution space*, which is a set of possible values for the decision variables, and let  $\Omega \subseteq \mathcal{S}$  be the *feasible region*. Specifically, a point  $\mathbf{x} \in \mathcal{S}$  satisfies the constraints of the optimization problem if and only if  $\mathbf{x} \in \Omega$ . The point  $\mathbf{x}^* \in \Omega \subseteq \mathcal{S}$  is a *global optimizer* if and only if  $\mathbf{x}^*$  satisfies the property that  $f(\mathbf{x}^*) \leq f(\mathbf{x})$ , for all feasible  $\mathbf{x} \in \Omega \subseteq \mathcal{S}$ . A *lower bound function*, denoted as  $f_{lb} : \mathbb{R}^d \rightarrow \mathbb{R}$ , satisfies the following condition for all feasible  $\mathbf{x} \in \Omega \subseteq \mathcal{S}$ :

$$f_{lb}(\mathbf{x}) \leq f(\mathbf{x}) \quad (4)$$

Function  $f$  is said to be *Lipschitz continuous* if there exists a constant  $l_f \geq 0$  such that for any two points  $\mathbf{x}^i$  and  $\mathbf{x}^j$  in the domain  $\mathbb{R}^d$ , the following inequality

holds:

$$|f(\mathbf{x}^i) - f(\mathbf{x}^j)| \leq l_f \|\mathbf{x}^i - \mathbf{x}^j\| \quad (5)$$

where  $\|\cdot\|$  denotes the Euclidean norm ( $L^2$ -norm). The Lipschitz constant  $l_f$  provides an upper bound on the slope or rate of change of the function  $f$ , as presented below:

$$l_f = \sup_{\mathbf{x}^i, \mathbf{x}^j \in \mathcal{S}^2} \frac{|f(\mathbf{x}^i) - f(\mathbf{x}^j)|}{\|\mathbf{x}^i - \mathbf{x}^j\|}. \quad (6)$$

To define a lower bound at a given point  $\mathbf{x}^j$ , we compare it to a reference point  $\mathbf{x}^i$  (which we shall elaborate on later in this section), within the domain. Rather than treating the lower bound as an approximation of  $f(\mathbf{x}^j)$ , it is constructed relative to the function value at a nearby point  $\mathbf{x}^i \in \mathcal{S}$ . The lower bound is then given by:

$$f_{lb}(\mathbf{x}^i) = f(\mathbf{x}^j) - l_f \|\mathbf{x}^i - \mathbf{x}^j\|. \quad (7)$$

Calculating  $l_f$  using Equation (6) can be challenging, particularly when the function  $f$  is unknown or can only be evaluated at a limited set of points. The standard formulation assumes that function variation is directly proportional to Euclidean distance ( $\|\mathbf{x}^i - \mathbf{x}^j\|$ ), which may not always be appropriate, especially for functions with non-uniform smoothness or varying sensitivity across different regions of the search space. To address this limitation, we generalize the Lipschitz constant formulation in Equation (5) to *Hölder constant* [78] as:

$$|f(\mathbf{x}^i) - f(\mathbf{x}^j)| \leq l_f \|\mathbf{x}^i - \mathbf{x}^j\|^p \quad (8)$$

where  $p \in (0, 1]$ . The choice of  $p$  allows for different function variations. Further details on the construction and selection of suitable distance metrics will be discussed in Section 5.2.

While the Lipschitz and more general Hölder conditions are known to apply to any continuous function, this research makes two additional general assumptions.

**Assumption 1.** *[Bounded Domain] The domain  $\mathcal{S}$  is bounded, meaning that there exists a finite upper bound on the distance between any two points within the domain:*

$$\exists M > 0 \quad s.t. \quad \forall \mathbf{x}^i, \mathbf{x}^j \in \mathcal{S}, \quad \|\mathbf{x}^i - \mathbf{x}^j\| < M. \quad (9)$$

This assumption ensures that distances between points in the domain remain finite, which is a necessary condition for deriving meaningful lower-bound estimates.

Because this research is motivated by SO, we let the set  $\mathcal{I} \subset \mathcal{S}$  represent a small finite set of evaluated points within the solution space  $\mathcal{S}$ .

**Assumption 2.** *[Minimally Exploratory Sampling (MES)] In surrogate optimization, a finite set of evaluated points  $\mathcal{I} = \{\mathbf{x}^i \mid i = 1, \dots, N\}$  is given, with*

function values known at these locations. A subsequent iteration of the optimization algorithm selects a finite set of unevaluated points from a set  $\mathcal{U} \subset \mathcal{S} \setminus \mathcal{I}$  for evaluation. To ensure effective exploration, the selected points must maintain a minimum distance from the evaluate points. That is,

$$\exists \varepsilon > 0 \quad \text{s.t.} \quad \forall \mathbf{x}^i \in \mathcal{I}, \mathbf{x}^j \in \mathcal{U}, \quad \|\mathbf{x}^i - \mathbf{x}^j\| > \varepsilon. \quad (10)$$

While the Hölder condition restricts  $p < 1$ , under Assumptions 1 and 2, we prove Proposition 1, which allows exponent  $p > 0$ .

**Proposition 1.** *Suppose that  $f$  satisfies the Lipschitz condition (5),  $\mathcal{S}$  is bounded (9), and  $\mathcal{I}$  and  $\mathcal{U}$  satisfy the MES condition (10). Then, for any  $p > 0$ , and any points  $\mathbf{x}^i \in \mathcal{I}$  and  $\mathbf{x}^j \in \mathcal{U}$ , there exists a constant  $l_f > 0$  such that Equation (8) is true.*

The proof of Proposition 1 is given in the appendix Section 10.1. Proposition 1 allows us to extend the formulation of the lower bound function, as presented in Equation (7), by incorporating a general nonlinear distance function  $\varphi(\cdot)$ , given a set of evaluated points  $\mathcal{I}$  and unevaluated points  $\mathcal{U}$ :

$$f_{lb}(\mathbf{x}^j) = f(\mathbf{x}^i) - l_f \varphi(\mathbf{x}^i, \mathbf{x}^j) \leq f(\mathbf{x}^j), \quad \forall \mathbf{x}^i \in \mathcal{I}, \mathbf{x}^j \in \mathcal{U}, \quad (11)$$

where  $l_f$  is given by

$$l_f = \sup_{\mathbf{x}^i \in \mathcal{I}, \mathbf{x}^j \in \mathcal{U}} \frac{f(\mathbf{x}^i) - f(\mathbf{x}^j)}{\varphi(\mathbf{x}^i, \mathbf{x}^j)}. \quad (12)$$

However, there are two drawbacks to the lower bound in equation (11). As with the Lipschitz constant from equation (6), the constant  $l_f$  in equation (12) may be challenging to evaluate, especially in a surrogate optimization setting in which  $f$  is expensive to evaluate. Moreover, the function value at the reference point,  $f(\mathbf{x}^i)$ , comes only from a single point  $\mathbf{x}^i$  in the set of evaluated points  $\mathcal{I}$ . To overcome the latter drawback, we make use a surrogate model  $\hat{f}$ , which is trained on multiple points in  $\mathcal{I}$  and common in many optimization settings, such as surrogate optimization.

To estimate the Lipschitz constant  $\hat{l}_f$  in our proposed estimated lower bound function,  $\hat{f}_{lb}$ , we first solve a linear programming (LP) model using the set of evaluated points  $\mathcal{I}$ . Specifically, let  $\{\mathcal{I}_1, \dots, \mathcal{I}_K\}$  be a partition of the set  $\mathcal{I}$ . For each  $\mathbf{x}^i \in \mathcal{I}$ , let  $\mathcal{I}^i$  be the subset within its partition, and let  $\bar{\mathcal{I}}^i$  be the complementary subset excluding  $\mathcal{I}^i$ , i.e.,  $\bar{\mathcal{I}}^i = \mathcal{I} \setminus \mathcal{I}^i$ . For each subset  $\mathcal{I}^i$ , we define a surrogate model  $\hat{f}_{\bar{\mathcal{I}}^i}$  trained on the complementary subset  $\bar{\mathcal{I}}^i$ , ensuring that  $\mathbf{x}^i$  is not included in the training set of  $\hat{f}_{\bar{\mathcal{I}}^i}$ . This step prevents biased estimation of the Lipschitz constant that could arise from interpolating the function at known points.

To leverage information from evaluated points and enhance the accuracy of the estimated Lipschitz constant, we define  $\bar{\mathbf{x}}^i$  as the nearest evaluated point to  $\mathbf{x}^i$  within  $\bar{\mathcal{I}}^i$ , based on the generalized distance metric  $\varphi$ :

$$\bar{\mathbf{x}}^i \in \arg \min_{\mathbf{x} \in \bar{\mathcal{I}}^i} \varphi(\mathbf{x}^i, \mathbf{x}). \quad (13)$$

Let  $\eta_i$  be a nonnegative slack variable representing the difference between the estimated lower bound function at  $\mathbf{x}^i$  and its true evaluated function  $f(\mathbf{x}^i)$ . Given the true evaluated function  $f(\mathbf{x}^i)$ , the surrogate prediction of the point  $\hat{f}_{\bar{\mathcal{I}}^i}(\mathbf{x}^i)$ , and the distance  $\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)$  to the nearest point  $\bar{\mathbf{x}}^i$  in  $\bar{\mathcal{I}}^i$ , for each evaluated point  $\mathbf{x}^i \in \mathcal{I}$ , the LP formulation, as given in Equation (14), determines the constant  $\hat{l}_f$  that minimizes the worst-case deviations (maximum)  $\eta_i$  for all evaluated points:

$$\min_{\eta} \max_{i=1, \dots, N} \eta_i \quad (14a)$$

s.t.

$$\hat{f}_{\bar{\mathcal{I}}^i}(\mathbf{x}^i) - \hat{l}_f \varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) + \eta_i = f(\mathbf{x}^i), \quad \forall \mathbf{x}^i \in \mathcal{I}, \quad (14b)$$

$$\eta_i \geq 0, \quad \forall i = 1, \dots, N. \quad (14c)$$

With the estimated Lipschitz constant  $\hat{l}_f$ , we estimate a lower bound of the function  $f$  at any unevaluated point  $\mathbf{x} \in \mathcal{U}$  as:

$$\hat{f}_{lb}(\mathbf{x}) = \hat{f}(\mathbf{x}) - \hat{l}_f \varphi(\mathbf{x}, \bar{\mathbf{x}}) \approx f_{lb}(\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{U}, \quad (15)$$

where  $\bar{\mathbf{x}}$  is a nearest evaluated point to  $\mathbf{x}$  in  $\mathcal{I}$  based upon the distance metric  $\varphi$ ; that is,

$$\bar{\mathbf{x}} \in \arg \min_{\bar{\mathbf{x}} \in \mathcal{I}} \varphi(\mathbf{x}, \bar{\mathbf{x}}). \quad (16)$$

Algorithm 2 outlines the steps involved in estimating the Lipschitz constant using distance metrics and surrogate models.

---

**Algorithm 2** Lipschitz Constant Estimation Using Distance Metrics

---

- 1: Partition  $\mathcal{I}$  into  $\{\mathcal{I}_1, \dots, \mathcal{I}_K\}$
  - 2: **for**  $k = 1, \dots, K$  **do**
  - 3:   Train a surrogate model  $\hat{f}_{\bar{\mathcal{I}}_k}$  on the data  $(\mathbf{x}^i, f(\mathbf{x}^i)) \forall \mathbf{x}^i \in \bar{\mathcal{I}}_k$ .
  - 4:   **for**  $\mathbf{x}^i \in \mathcal{I}_k$  **do**
  - 5:     Calculate  $\hat{f}_{\bar{\mathcal{I}}_k}(\mathbf{x}^i)$ .
  - 6:     Find  $\bar{\mathbf{x}}^i$  from Equation (13).
  - 7:     Calculate  $\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)$
  - 8:   **end for**
  - 9: **end for**
  - 10: Solve the linear program (LP) using Equation (14) to determine  $\hat{l}_f$ .
  - 11: Return the estimated Lipschitz constant  $\hat{l}_f$ .<sup>2</sup>
- 

The algorithm iterates over the  $K$  subsets within the partition  $\mathcal{I}_1, \dots, \mathcal{I}_K$ . This partitioning allows us to train a surrogate model  $\hat{f}_{\bar{\mathcal{I}}_k}$  on  $(\mathbf{x}^i, f(\mathbf{x}^i))$ ,  $\forall \mathbf{x}^i \in \bar{\mathcal{I}}_k$ , which is the complementary subset of  $\mathcal{I}_k$  i.e.  $\bar{\mathcal{I}}_k = \mathcal{I} \setminus \mathcal{I}_k$ . Then for each point  $\mathbf{x}^i \in \mathcal{I}_k$ , the surrogate model is used to predict the function value at  $\mathbf{x}^i$ , resulting in  $\hat{f}_{\bar{\mathcal{I}}_k}(\mathbf{x}^i)$ . Next, the algorithm identifies the closest point to  $\mathbf{x}^i$  in  $\bar{\mathcal{I}}_k$ .

---

<sup>2</sup>Throughout the remainder of the paper,  $\hat{l}$  refers to  $\hat{l}_f$ .



This step determines the reference point  $\bar{\mathbf{x}}^i$  that is used in the distance metric calculations. The distance metric value  $\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)$  is computed, representing the relationship between the current point  $\mathbf{x}^i$  and the reference point  $\bar{\mathbf{x}}^i$ . The process continues for each point in each subset of the partition, resulting in a set of distance metric values for each point. Finally, the algorithm solves the linear program (LP) given in Equation (14). Upon solving the LP, the algorithm returns the estimated Lipschitz constant  $\hat{l}_f$ .

Algorithm 2 is similar to *cross-validation (CV)*, which is commonly employed in statistics and machine learning to mitigate overfitting and assess the variance of surrogate models. CV is particularly valuable in surrogate optimization, where datasets are often limited in size [6]. By incorporating an approach similar to CV into the estimated Lipschitz constant process, we can account for the uncertainty in the surrogate model’s predictions and improve the robustness of the lower bound construction. Specifically, CV in its traditional machine learning setting trains multiple models on different subsets of the data and validates them on the complementary datasets. Similarly, in the LP, multiple models are trained, but the Lipschitz constant is estimated using evaluated points in complementary datasets  $\tilde{\mathcal{I}}$  as shown in Equation (14b).

## 5.2. Distance Metrics and Similarity Measurements

Traditional Lipschitz-based methods typically assume a Euclidean distance metric, which imposes a linear relationship between function changes and spatial separation. However, this assumption may be overly restrictive, especially for functions exhibiting non-uniform smoothness or variable scaling behaviors.

To address this limitation, our study explores alternative distance metrics that allow for adaptive function variation estimates, enabling more accurate and flexible lower-bound constructions. Specifically, we consider three classes of distance metrics—superlinear, linear, and sublinear—each offering a different degree of sensitivity to function behavior. Consider a given point-wise distance-based metric  $\varphi(\mathbf{x}^i, \mathbf{x}^j)$  expressed in the following form:

$$\varphi(\mathbf{x}^i, \mathbf{x}^j) = \alpha \|\mathbf{x}^i - \mathbf{x}^j\|^p \quad (17)$$

Depending on the value of  $p$ , different classes of distance metrics emerge, allowing for adaptive control of function variation estimates.

**Definition 1.** A *superlinear distance metric*  $\varphi(\mathbf{x}^i, \mathbf{x}^j)$  is a type of distance metric that follows Equation (17) where  $\mathbf{x}^i$  and  $\mathbf{x}^j$  are input vectors,  $\alpha$  is a scaling factor, and  $p$  is a positive exponent greater than 1. This implies that the distance metric grows faster than linearly, making it more sensitive to larger separations between points. Such metrics are particularly useful in capturing higher-order variations in function smoothness.

Similarly, we define:

- *Linear distance metric*: When  $p = 1$ , the distance metric retains the standard Euclidean form, enforcing a linear relationship between function variation and spatial separation.

- *Sublinear distance metric:* When  $p < 1$ , the distance metric grows more slowly than Euclidean distance, making it suitable for smooth or gradually varying functions where overestimating local changes may be detrimental.

While we primarily focus on the power-law form of Equation (17), it is important to note that alternative forms of distance metrics can be explored, depending on the specific function properties and optimization framework. This flexibility enables more accurate and adaptive Lipschitz constant estimation, thereby improving lower-bound formulations in surrogate and black-box optimization.

By analyzing different distance metrics, we can effectively measure the similarity or distance between input vectors and utilize them for estimating the Lipschitz constant. Each metric possesses unique characteristics that influence how well it captures the linearity properties of the function, making the choice of distance metric dependent on the specific requirements of the problem. The superlinear distance metric accommodates nonlinear relationships and allows for a faster growth rate than the linear metric, making it suitable for modeling complex interactions between input vectors. In contrast, the sublinear distance metric assigns less weight to larger distances, making it particularly useful for sparse and high-dimensional data where extreme values should not dominate. The linear distance metric, commonly based on Euclidean distance, provides a straightforward and computationally efficient baseline, though its linear nature limits its ability to capture nonlinearity in function behavior. By leveraging these distance metrics and their respective properties, we can efficiently approximate the Lipschitz constant, which serves as a crucial measure of the smoothness and regularity of the objective function. This, in turn, is essential for optimization problems, as it constrains the variation of function values across the search space, ensuring more stable and reliable optimization performance.

The parameters of the distance metric definitions play a crucial role and can lead to different outcomes. While the right-hand side (RHS) of the LP in Equation (14b) remains the same for all distance metrics, the left-hand side (LHS) differs in terms of the distance metric  $\varphi$ . Consequently, when considering three distance metrics, a superlinear one, a linear one, and a sublinear one, we obtain three distinct estimated Lipschitz constant values, namely  $l^{sup}$ ,  $l^{lin}$ , and  $l^{sub}$  from the LP. Theorem 5.1 provides a closed-form solution for the estimated Lipschitz constant  $l$  based on the LP in Equation (14), which we will explain in detail below.

In this study, we will explore three different distance metrics, each with its own set of parameters and properties. It is important to note that practitioners may not necessarily use or define all these distance metrics, but our comparison aims to provide some guidance for their selection.

**Theorem 5.1.** *When solving the LP (in line 10) of Algorithm 2, there exists a data point  $\mathbf{x}^i \in \mathcal{I}$ , where  $\eta_i = 0$  in the LP solution, which maximizes the*

fraction:

$$\hat{l} = \max_{\mathbf{x}^i \in \mathcal{I}} \frac{\hat{f}_{\bar{\mathcal{I}}^i}(\mathbf{x}^i) - f(\mathbf{x}^i)}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)} \quad (18)$$

where  $\bar{\mathbf{x}}^i$  represents the closest point to  $\mathbf{x}^i$  from the complementary data set  $\bar{\mathcal{I}}^i$  given by Equation (13). This maximization provides an optimal solution for the estimated Lipschitz constant  $\hat{l}$ .

The proof is provided in appendix Section 10.2. With Theorem 5.1 established, let  $\tilde{\mathbf{x}} \in \mathcal{I}$  be the *ratio maximizer* of Equation (18). Specifically, the ratio maximizer  $\tilde{\mathbf{x}}$  is given by Equation (19),

$$\tilde{\mathbf{x}} = \arg \max_{\mathbf{x}^i \in \mathcal{I}} \frac{\hat{f}_{\bar{\mathcal{I}}^i}(\mathbf{x}^i) - f(\mathbf{x}^i)}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)} \quad (19)$$

Additionally, let  $\bar{\bar{\mathcal{I}}}$  be the complementary subset of  $\tilde{\mathbf{x}}$ , and let  $\bar{\bar{\mathbf{x}}} \in \bar{\bar{\mathcal{I}}}$  be the nearest point to  $\tilde{\mathbf{x}}$  in  $\bar{\bar{\mathcal{I}}}$  as defined in Equation (13).

*Remark 1.* For a given set of evaluated points  $\mathcal{I}$ , in many situations with two (or more) distance metrics,  $\varphi^i$  and  $\varphi^j$ , the ratio maximizers,  $\tilde{\mathbf{x}}^i$  and  $\tilde{\mathbf{x}}^j$ , are the same point. This is primarily because the distance metrics (the denominator of Equation (19)) are both increasing in the Euclidean distance, and the error in the surrogate  $\hat{f}_{\bar{\mathcal{I}}^i}(\mathbf{x}^i) - f(\mathbf{x}^i)$  (the numerator in Equation (19)) is independent of the distance metric. Consequently, when executing Algorithm 2 for  $\varphi^i$  and  $\varphi^j$  but with all other inputs being equal, the solutions to Equation (19) for both executions are frequently the same point.

To establish a rigorous foundation for the results presented in this research, several intermediary lemmas and theoretical derivations have been developed. For clarity and conciseness, these supporting results, along with their formal statements, are provided in Appendix 10. These lemmas play a crucial role in the justification of key assumptions and in the derivation of Theorem 5.2 in Section 5.3 as well as its proof provided in appendix Section 10.9. With these theoretical foundations in place, we now proceed to utilize the estimated Lipschitz constant  $\hat{l}$  to compute estimated lower bounds on the objective function for unevaluated points.

### 5.3. Bound Construction

Once the estimated Lipschitz constant  $\hat{l}$  has been obtained using Algorithm 2 we can use Equation (15) to estimate the lower bound function of  $f$  over a set of unevaluated data points  $\mathcal{U}$ . Let  $\rho$  be defined as:

$$\rho(\mathbf{x}) = \hat{l}\varphi(\mathbf{x}, \bar{\mathbf{x}}), \forall \mathbf{x} \in \mathcal{U} \quad (20)$$

Combining Equations (15) and (20), we have the following estimated lower bound function:

$$\hat{f}_{lb}(\mathbf{x}) = \hat{f}(\mathbf{x}) - \rho(\mathbf{x}), \forall \mathbf{x} \in \mathcal{U} \quad (21)$$

**Theorem 5.2.** *Let  $\mathcal{I}$  be a given set of evaluated data points and  $\mathcal{U}$  be a set of unevaluated data points. Consider two distance metric  $\varphi^{(p^i)}$  and  $\varphi^{(p^j)}$  with exponents  $p^i$  and  $p^j$ , where  $p^i > p^j$ . Suppose Algorithm 2 is used with  $\varphi^{(p^i)}$  and  $\varphi^{(p^j)}$  and all other inputs being the same to determine estimated lower bounds  $\hat{f}_{lb}^{(p^i)}$  and  $\hat{f}_{lb}^{(p^j)}$ , respectively. For  $\hat{f}_{lb}^{(p^j)}$ , let  $\tilde{\mathbf{x}}^j$  and  $\bar{\mathbf{x}}^j$  be the ratio maximizer and the nearest points to it in its complementary subset, respectively. Then, we have the following result:*

$$\hat{f}_{lb}^{(p^i)}(\mathbf{x}) < \hat{f}_{lb}^{(p^j)}(\mathbf{x}), \forall \mathbf{x} \in \{\mathbf{x} \in \mathcal{U} \mid \|\mathbf{x} - \bar{\mathbf{x}}^j\| > \|\tilde{\mathbf{x}}^j - \bar{\mathbf{x}}^j\|\} \quad (22)$$

The proof of the theorem is provided in appendix Section 10.9. The theorem is notable because it shows that using superlinear distance metrics yields more conservative estimated lower bounds for unevaluated points that are sufficiently far away from evaluated points, while using a sublinear distance metric yields a more aggressive estimated lower bound function. We will discuss this point with empirical results in Section 5.7. It is particularly interesting though that the Euclidean distance between the ratio maximizer and its nearest neighbor in the complementary subset for the distance metric of with the smaller exponent  $p^j$  is all that is needed to determine how far these unevaluated points need to be from the evaluated points. Moreover, while the conditions of the theorem include  $p^i > p^j$ , the Euclidean distance between the unevaluated and evaluated points in Equation (22) is independent of the value of the larger exponent  $p^i$ .

#### 5.4. Lower Bound Quality

The quality of the estimated lower bound functions constructed using Lipschitz estimates is an essential aspect of assessing their accuracy and usefulness in practical applications. One way to evaluate the quality of the estimated lower bound functions is through *point-wise evaluation*. This approach focuses on assessing the accuracy of the estimated lower bound function at individual data points. Given a set of data points  $\mathcal{I}$ , we can compare the true function values  $f(\mathbf{x}^i)$  with the corresponding estimated lower bounds  $\hat{f}_{lb}(\mathbf{x}^i)$  obtained using Lipschitz estimates. For each point  $\mathbf{x}^i \in \mathcal{I}$ , we compute the absolute difference between the true function value and the estimated lower bound:  $|f(\mathbf{x}^i) - \hat{f}_{lb}(\mathbf{x}^i)|$ .

By comparing the estimated lower bound functions generated by different estimators or parameter configurations, we can identify which approach produces more aggressive and conservative estimated lower bound functions. This comparative analysis allows us to select the most suitable Lipschitz estimator or parameter setting for a specific application, optimizing the quality of the estimated lower bound functions.

Certain adjustments can be considered to obtain improved estimated lower bound functions, one of which is enhancing the accuracy of the surrogate model used to predict outputs for unevaluated observations. The reliability of the surrogate model  $\hat{f}$  in approximating  $f$  depends on several key factors, including model selection; indeed, the type of surrogate model such as polynomial regression, radial basis functions, or neural networks significantly impacts the fidelity of  $\hat{f}$  to  $f$ . Each model type offers distinct strengths and weaknesses

depending on the function’s behavior and the problem space’s dimensionality. Moreover, the quality and coverage of the training data are crucial, as sparse or unevenly distributed samples can lead to poor approximations in underrepresented regions. In cases where the true function  $f$  exhibits high nonlinearity, discontinuities, or other complexities, more sophisticated or customized modeling approaches may be required to capture its nuances effectively [26, 37].

When evaluating the quality of the estimated lower bound functions, the objective is to minimize the deviation from the true function values. Equation (21) highlights two key aspects to consider to achieve this objective:

(i) The accuracy and proximity of the surrogate model predictions to the true function values: The closer the predictions are to the true function values, the smaller the difference between the prediction and the true function value will be.

(ii) The choice of  $\rho(\mathbf{x})$  for each data point: Smaller  $\rho(\mathbf{x})$  leads to smaller differences between the surrogate model predictions and the estimated lower bound function.

It is important to note that improving the accuracy of the surrogate model is beyond the scope of this research. However, based on how the function  $\rho$  is determined, some general guidelines can be considered to influence the function  $\rho$ . To minimize  $\rho$ , it is necessary to minimize each component of the expression in Equation (20). Therefore, it is crucial to investigate the behavior of  $\rho$  on evaluated and unevaluated points individually.

### 5.5. Evaluation Metric

To assess the effectiveness of different distance metrics in lower bound function estimation, we introduce an evaluation metric for selecting the winning distance metric at a given sampled point. The selection criterion is based on the tightness and validity of the estimated lower bound function each distance metric generates. Specifically, let  $\mathbf{x} \in \mathcal{S}$  be a sampled point, and consider two estimated lower bound functions  $\hat{f}_{lb}^{\varphi_1}$  and  $\hat{f}_{lb}^{\varphi_2}$  derived from distance metrics  $\varphi_1$  and  $\varphi_2$ . Assuming  $\hat{f}_{lb}^{\varphi_1}(\mathbf{x}) \neq \hat{f}_{lb}^{\varphi_2}(\mathbf{x})$ , we declare  $\varphi_1$  the *winning distance metric* at  $\mathbf{x}$  if any of the conditions in Equation (23) are true.

$$\hat{f}_{lb}^{\varphi_2}(\mathbf{x}) < \hat{f}_{lb}^{\varphi_1}(\mathbf{x}) \leq f(\mathbf{x}), \quad (23a)$$

$$\hat{f}_{lb}^{\varphi_1}(\mathbf{x}) \leq f(\mathbf{x}) < \hat{f}_{lb}^{\varphi_2}(\mathbf{x}), \quad (23b)$$

$$f(\mathbf{x}) < \hat{f}_{lb}^{\varphi_1}(\mathbf{x}) < \hat{f}_{lb}^{\varphi_2}(\mathbf{x}). \quad (23c)$$

Since lower bound estimates can either overestimate or underestimate the true function value  $f(\mathbf{x})$ , our metric evaluates three distinct scenarios. First, it determines whether the distance metric using Algorithm 2 produces an estimated lower bound function that is a *valid lower bound* at  $\mathbf{x}$ . If both distance metrics meet this criterion, the metric selects the one that yields an estimated lower bound function with the smallest gap between the estimated lower bound

at  $\mathbf{x}$  and  $f(\mathbf{x})$  as the *winning distance metric*, as defined in Equation (23a). However, a distance metric may be *aggressive* if it attempts to satisfy Equation (23a) but, in doing so, might fail to satisfy Equation (23b). Conversely, a *conservative* bound prioritizes satisfying Equation (23b) and potentially Equation (23c), ensuring robustness at the cost of possibly overestimating  $f(\mathbf{x})$ . In the second scenario, where one distance metric estimates a valid lower bound while the other fails to do so, the metric selects the valid lower bound as in Equation (23b). Finally, when neither distance metric yields a valid lower bound, the metric still determines the winning distance metric by selecting the one that yields an estimated lower bound with the smallest gap, as described in Equation (23c).

### 5.6. Data Distribution

The data distribution plays a crucial role in determining the lower bound estimates. The distribution can be divided into two distinct sections: evaluated and unevaluated data points. The distribution characteristics of these sections directly affect the computation of the estimated Lipschitz constant, as they influence the distance and  $\varphi$  involved in the process.

**Evaluated Data Distribution:** The distribution of the evaluated points  $\mathcal{I}$  and their pairwise distances impact the calculation of the estimated Lipschitz constant. Specifically, by combining Equations (18) and (19), we have the following equation of the estimated Lipschitz constant:

$$\hat{l} = \frac{\hat{f}_{\mathcal{I}}(\tilde{\mathbf{x}}) - f(\tilde{\mathbf{x}})}{\varphi(\tilde{\mathbf{x}}, \tilde{\mathbf{x}})} \quad (24)$$

Observe that the larger the distance (distance metric values) between the ratio maximizer and its nearest point in the complementary set, the smaller the estimated Lipschitz constant becomes. Consequently, a well-spread data set, where the evaluated points are distributed across a wider range, usually leads to a smaller estimated Lipschitz constant.

**Unevaluated Data Distribution:** Consider a finite subset of unevaluated  $\mathcal{U}' \subset \mathcal{U}$ . The distribution of the data points in  $\mathcal{U}'$  also plays a significant role in the tightness of the estimated lower bounds. The pairwise distances between the unevaluated points  $\mathcal{U}'$  and their corresponding closest point in the evaluated set  $\mathcal{I}$  are used to calculate  $\rho(\mathbf{x})$  as shown in Equation (20). Using the estimated Lipschitz constant  $\hat{l}$  obtained from the evaluated dataset  $\mathcal{I}$ , a tighter lower bound estimate would be created by subtracting a smaller  $\rho(\mathbf{x})$  in Equation (21). Let us consider two unevaluated points  $\mathbf{x}^i, \mathbf{x}^j \in \mathcal{U}'$ . Referring to the general definition of a distance metric in Equation (17), we observe that  $\varphi(\mathbf{x}, \bar{\mathbf{x}})$  is monotonically increasing with respect to the Euclidean distance between the two points. Consequently, we can make the following inference:

$$\|\mathbf{x}^i - \bar{\mathbf{x}}^i\| > \|\mathbf{x}^j - \bar{\mathbf{x}}^j\| \implies \varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) > \varphi(\mathbf{x}^j, \bar{\mathbf{x}}^j) \implies \rho(\mathbf{x}^i) > \rho(\mathbf{x}^j) \quad (25)$$

Consequently, unevaluated data points that have a smaller Euclidean distance from their closest evaluated point will result in tighter lower bound estimates.

### 5.7. Distance Metric Effects on Lower Bound Construction Through Empirical Examples

The choice of distance metric significantly impacts the quality of the lower bound estimates obtained for a given dataset. Variations in  $\varphi(\mathbf{x}, \bar{\mathbf{x}})$  values influence the accuracy of the lower bound estimates on the function values, which can arise either from different distance metric selections or variations in data point distribution, as previously discussed in Section 5.6. The effectiveness of a distance metric in guiding estimated lower-bound construction is inherently linked to the smoothness and structure of the underlying function.

This motivates the need for alternative distance metrics beyond the conventional Euclidean form. Specifically, the superlinear, linear, and sublinear distance metrics, as introduced in Section 5.2, provide different growth behaviors that may better align with function characteristics. Our hypothesis is that selecting a distance metric that matches the function’s inherent linearity properties results in tighter and more accurate estimated lower bound function.

Three empirical tests and sensitivity analyses have been conducted to evaluate the performance and general applicability of different distance metrics in optimization. We expect to observe a direct relationship between function characteristics and the suitability of a given distance metric, where the quality of the estimated lower bound function improves when the underlying function type aligns with the selected metric. Additionally, traditional estimated lower-bound construction methods should exhibit inferior performance compared to the proposed distance metric-based approaches. The baseline methodologies considered in this study are the Standard Lipschitz constant estimation method (denoted as Lipschitz in the tables) as described in Section 4.1 and Working-Hotelling as explained in Section 4.3. For the Standard Lipschitz constant estimation method, we estimate constant  $L$  in Equation (1) using Equation (14) with  $\hat{f}_{\mathcal{I}^i}(\mathbf{x}^i) = f(\bar{\mathbf{x}}^i)$ , providing a more data-driven estimation method. Similar bounds are widely employed in probabilistic branch-and-bound optimization frameworks [34].

The specific distance metric definitions in Table 1 are carefully selected to highlight the necessary linearity characteristics of the distance metric. While they differ from the general form in Equation (17), they effectively capture the essential behavior relevant to our analysis. Each includes a product of a constant  $m$  and the Euclidean distance. The superlinear distance metric accentuates small distances by incorporating an exponential term, allowing it to emphasize close similarities between data points. The linear distance metric, on the other hand, directly scales the distance between points. Lastly, the sublinear distance metric applies a logarithmic transformation to the distance, which effectively downplays larger distances.

By using these specific definitions, we can demonstrate the distinctive behaviors of the different distance metric types and their impact on the evaluation of  $\varphi$ . This facilitates a more intuitive understanding of the characteristics of each distance metric and its suitability for specific problems.

Table 1: Distance Metric Definitions

Superlinear distance metric	$\varphi(\mathbf{x}^i, \mathbf{x}^j) = \exp(m\ \mathbf{x}^i - \mathbf{x}^j\ ) - 1$
Linear distance metric	$\varphi(\mathbf{x}^i, \mathbf{x}^j) = m\ \mathbf{x}^i - \mathbf{x}^j\ $
Sublinear distance metric	$\varphi(\mathbf{x}^i, \mathbf{x}^j) = \ln(m\ \mathbf{x}^i - \mathbf{x}^j\  + 1)$

To approximate the response surface of the objective function, as defined in Equation (14), we use a Radial Basis Function (RBF) surrogate model, a well-established approach in the literature [27, 68]. By analyzing three representative examples, we assess the effectiveness of different distance metrics in capturing function linearity and constructing reliable estimated lower bound function. To ensure consistency across tests, we define the domain  $\mathcal{S}$  as the interval  $[-6, 6]$  and use 10 data points, with seven forming the evaluated set  $\mathcal{I}$  and three unevaluated points from  $\mathcal{U} \subset \mathcal{S} \setminus \mathcal{I}$ , strategically selected to test the surrogate models and distance metrics. This standardized setup enables a fair comparison of results across linear, sublinear, and superlinear functions.

**Linear Function.** Consider the function  $f$  as a piecewise linear function given by:

$$f(\mathbf{x}) = \min\{|\mathbf{x} - 4|, |\mathbf{x} + 4|\} \quad (26)$$

Figure 1 presents a visualization of the piecewise linear test function and the estimated lower bounds on the three unevaluated points. The left section depicts the function, defined in Equation (26) over the domain  $[-6, 6]$ , where evaluated points (blue squares) and the unevaluated points (orange circles) are highlighted. The right section compares lower bound estimates at the unevaluated points, illustrating the effect of different distance metrics. The “ $f(\mathbf{x})$ ” represents actual function values, while the “ $\hat{f}(\mathbf{x})$ ” corresponds to the estimates from the RBF surrogate model. When the surrogate closely aligns with the true function, these markers overlap. The same structured visualization is used across different test functions, including sublinear and superlinear cases, ensuring consistency in evaluating lower bound estimation performance.

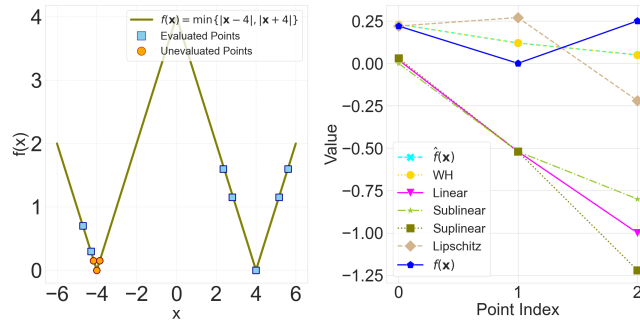


Figure 1: [Left] Defined piecewise linear function. [Right] Comparison of lower bound approximations for the unevaluated points using different distance metrics.

As depicted in Figure 1, we can observe that the estimated lower bound function from linear distance metric yields the closest lower bound estimates for



	Linear	Sublinear	Superlinear	WH	Lipschitz	Total
<b>Linear</b>	<b>*</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>8</b>
<b>Sublinear</b>	1	*	2	2	2	7
<b>Superlinear</b>	1	1	*	2	2	6
<b>Working - Hotelling</b>	1	1	1	*	2	5
<b>Standard Lipschitz</b>	1	1	1	1	*	4
<b>Total</b>	4	5	6	7	8	*

Table 2: Distance metric lower bound comparison for linear test function

the piecewise linear function, by being slightly tighter than that of the sublinear distance metric on the lowest two unevaluated points. We can also observe the ordering of the estimated lower bound functions from Theorem 5.2 on the third lowest unevaluated point because it is sufficiently far away from the evaluated points, while the lowest two unevaluated are very close to an evaluated point. Meanwhile the Standard Lipschitz method fails to determine a valid lower bound estimate in two of the three unevaluated points. In addition, Table 2 indicates the cumulative number of times each distance metric outperforms the other distance metrics based upon the metric described in section 5.5. Table 2 also indicates that the linear distance metric produces a better lower bound estimate in eight out of twelve instances compared to the other distance metrics for the unevaluated points in the domain.

**Sublinear Function.** Consider the piecewise sublinear function  $f$  defined as:

$$f(\mathbf{x}) = \min\{\ln(|\mathbf{x} - 4| + 1), \ln(|\mathbf{x} + 4| + 1)\} \quad (27)$$

In Figure 2 and Table 3, we observe that the sublinear distance metric produces the most accurate lower bound estimates for the sublinear function as it is designed to capture sublinear behavior. Observe the ordering of the estimated lower bound functions from Theorem 5.2 is apparent at all three unevaluated points.

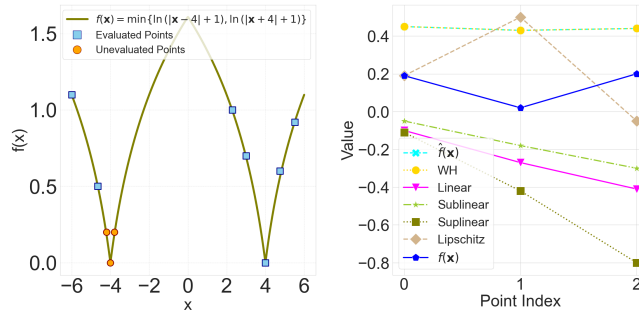


Figure 2: [Left] Defined piecewise sublinear function. [Right] Comparison of lower bound approximations for the unevaluated points using different distance metrics.

**Superlinear Function.** Let us consider a piecewise-defined superlinear function given by:

	Linear	Sublinear	Superlinear	WH	Lipschitz	Total
Linear	*	0	3	3	2	8
<b>Sublinear</b>	<b>3</b>	<b>*</b>	<b>3</b>	<b>3</b>	<b>2</b>	<b>11</b>
Superlinear	0	0	*	3	2	5
Working - Hotelling	0	0	0	*	1	1
Standard Lipschitz	1	1	1	2	*	5
Total	4	1	7	11	7	*

Table 3: Distance metric lower bound comparison for sublinear test function

$$f(\mathbf{x}) = \min\{\exp(|\mathbf{x} - 4|) - 1, \exp(|\mathbf{x} + 4|) - 1\} \quad (28)$$

As anticipated, the superlinear distance metric is expected to yield a superior lower bound estimate compared to other distance metrics due to its ability to capture superlinear relationships between data points. Superlinear distance metrics, such as the one defined here, are known for their capability to capture complex nonlinear patterns more effectively than linear and sublinear distance metrics. This is particularly evident when examining specific points, as illustrated in Figure 3. Moreover, because all three unevaluated points are so close to an evaluated point, Theorem 5.2 and its ordering of the estimated lower bound functions do not apply.

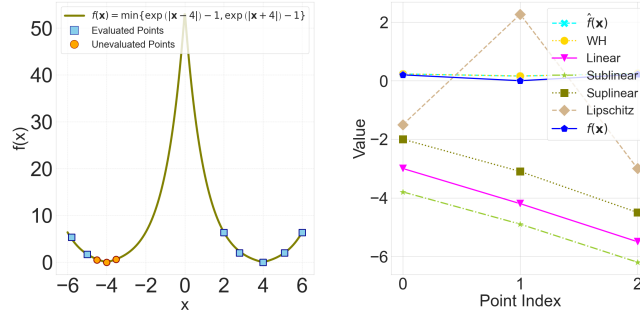


Figure 3: [Left] Defined piecewise superlinear function. [Right] Comparison of lower bound approximations for the unevaluated points using different distance metrics.

	Linear	Sublinear	Superlinear	WH	Lipschitz	Total
Linear	*	3	0	3	1	7
Sublinear	0	*	0	3	1	4
<b>Superlinear</b>	<b>3</b>	<b>3</b>	<b>*</b>	<b>3</b>	<b>1</b>	<b>10</b>
Working - Hotelling	0	0	0	*	1	1
Standard Lipschitz	2	2	2	2	*	8
Total	5	8	2	11	4	*

Table 4: Distance metric lower bound comparison for superlinear test function

In conclusion, Table 4 demonstrates the significant advantage of the superlinear distance metric over other distance metrics and approaches in generating high-quality estimated lower bound functions. It is noteworthy that the superlinear distance metric consistently produces lower bound estimates with a

narrower disparity in ten out of twelve instances compared to the other distance metrics, indicating its superior performance in capturing the characteristics of the underlying function.

Ultimately, we can assert that these observations validate our assumption that there is a direct relationship between the characteristics of the underlying function, such as linearity, and the suitability of a certain distance metric for producing accurate estimated lower bound functions. The linear distance metric is well-suited for approximating the linear behavior of the function, as it provides a more accurate estimated lower bound function compared to other distance metrics that may not effectively capture the linearity of the function.

While the sublinear distance metric may generally perform well in capturing sublinear behavior, there can still be cases where it fails to accurately provide lower bound estimates for certain instances of a sublinear function [8, 16]. This can be the result of certain traits of the function or distance metric restrictions. For instance, a function may exhibit irregularities, such as sharp jumps or spikes, or it may have local maxima or minima in certain locations, making it difficult for a sublinear distance metric to estimate the lower bound function with reliability, resulting in invalid estimates.

As discussed in Section 5.7, it can be observed that the  $\varphi$  values generated by the superlinear distance metric exhibit greater consistency within their range. Consequently, the superlinear distance metric demonstrates the ability to generate a tighter estimated lower bound function for superlinear functions compared to other distance metrics and methods. It is important to note that the Standard Lipschitz method and the Working-Hotelling method fail to find high-quality estimated lower bound functions due to their limited flexibility in capturing the nonlinearity aspects of the underlying functions.

## 6. Experiments and Results

### 6.1. Experimental Setup

In this section, we conduct an experimental analysis of our proposed methodology for estimating the Lipschitz constant and constructing estimated lower bound functions using different distance metrics. Furthermore, we assess effectiveness of using the estimated lower bound functions as acquisition functions to guide the search process within a surrogate optimization framework. For this purpose, we employ ten benchmark functions from the SFU optimization test problems library [69]. Table 5 outlines the details of each test function, specifying their dimensionality, the respective domain, and the known global minimum. Moreover, to approximate the response surface of the objective function in both cases, we utilize an RBF surrogate model, following a similar approach as described in Section 5.7. For all the experiments in this section a single workstation with a 2 GHz quad-core Intel Core i5 CPU, Python 3.11.5, and Gurobi Optimizer version 9.5.2 is used to solve the LP in Equation (14) and to run the experiments.

Test Function	Dimension	Domain	Global Minimum
Cross-in-Tray (C.I.T.)	2	$[-10, 10]^2$	-2.06261
Rosenbrock	4	$[-5, 10]^4$	0
Rastrigin	4	$[-5.12, 5.12]^4$	0
Sphere	4	$[-5.12, 5.12]^4$	0
Ackley	4	$[-32.768, 32.678]^4$	0
Holder Table	2	$[-10, 10]^2$	-19.2085
Shubert	2	$[-5.12, 5.12]^2$	-186.7309
Branin	2	$[-5, 10]^2$	0.397887
Six Hump Camel	2	$[-3, 3]^2$	-1.0316
Goldstein-Price	2	$[-2, 2]^2$	3

Table 5: Test Functions Definition

### 6.2. Lower Bound Estimation

We conduct 30 random seeded runs for each test function in Table 5, and we determine the best distance metric for each run based on the metric defined by Equation (23). We also report the mean and standard deviation (STD) of the total number of times the winning distance metric has won (out of 500) over 30 different runs at each test problem. In each of the 30 runs, we generate 150 points uniformly in the corresponding domain, as shown in Table 5. We use 100 points for the evaluated set  $\mathcal{I}$  to estimate the Lipschitz constant  $\hat{l}_f$ . The partition of  $\mathcal{I}$  is just the set of 100 individual points; that is,  $\mathcal{I}^i = \{\mathbf{x}^i\}$ ,  $\forall i = 1, \dots, |\mathcal{I}|$ . We use the remaining 50 points, to evaluate the winning distance metric.

Table 6 shows the final results for each of the 10 test functions after 30 runs. The winning distance metric has been highlighted in green for each test function. Some distance metrics may tie for the highest score after 30 runs, reflecting the same tightness. We assign a win to each of these distance metrics, which may result in more than 30 wins across methods. We indicate the scores of the tied distance metric with an asterisk (\*).











Test Function	Distance Metric	Wins	STD	Mean	Illustration
Cross in Tray	Linear	8	22.80	116.30	
	Sublinear	12	27.99	121.83	
	Superlinear	4	29.07	84.900	
	WH	2	15.79	99.770	
	Lipschitz	4	38.84	77.200	
Rosenbrock	Linear	23*	12.99	144.07	
	Sublinear	6	20.21	124.00	
	Superlinear	1*	22.12	93.670	
	WH	1	19.25	102.50	
	Lipschitz	0	8.860	35.770	
Rastrigin	Linear	15	14.47	134.47	
	Sublinear	7*	22.55	113.30	
	Superlinear	6	23.50	119.13	
	WH	3*	16.94	105.37	
	Lipschitz	0	6.220	27.730	
Sphere	Linear	15	14.79	135.43	
	Sublinear	6	22.39	109.53	
	Superlinear	6	22.74	119.70	
	WH	3	15.84	108.63	
	Lipschitz	0	5.360	26.700	
Ackley	Linear	16	16.82	140.37	
	Sublinear	13	21.28	141.33	
	Superlinear	0	12.83	39.570	
	WH	0	14.39	93.900	
	Lipschitz	1	16.02	84.830	
Holder Table	Linear	7*	8.540	126.47	
	Sublinear	10*	26.96	105.40	
	Superlinear	17*	25.01	134.27	
	WH	1	12.79	108.40	
	Lipschitz	0	3.740	25.467	
Shubert	Linear	5*	7.930	126.93	
	Sublinear	11*	26.16	109.63	
	Superlinear	16*	28.25	134.50	
	WH	0	14.08	102.43	
	Lipschitz	0	4.210	26.500	
Branin	Linear	6	13.06	126.47	
	Sublinear	9	29.71	105.40	
	Superlinear	15	29.18	134.27	
	WH	0	19.17	108.40	
	Lipschitz	0	5.710	25.467	
Camel	Linear	4	8.050	128.70	
	Sublinear	11	32.42	115.63	
	Superlinear	15	30.39	130.53	
	WH	0	14.33	97.530	
	Lipschitz	0	4.850	27.600	
Goldstein	Linear	1	6.970	124.73	
	Sublinear	11	34.31	107.87	
	Superlinear	18	31.22	139.17	
	WH	0	12.96	95.300	
	Lipschitz	0	9.900	32.930	

Table 6: Test functions with the Superlinear distance metric as the winning distance metric

Based on the results, the superlinear distance metric outperforms the linear and sublinear distance metrics individually in generating estimated lower bound functions for half of the benchmark functions as presented in Table 6. This matches the number of times where the linear and sublinear distance metrics together won.

The comparative results consistently demonstrate that distance metricized Lipschitz constant estimation outperforms other methods in constructing estimated lower bound functions, including WH and the Standard Lipschitz technique, yielding tighter and more accurate estimated lower bound functions.

In test functions where the global optima are located in flat regions such as Shubert, Branin, Six-Hump Camel, and Goldstein-Price, the superlinear dis-

tance metric tends to generate more conservative estimated lower bound functions, which outperform the estimated lower bound functions generated by other distance metrics. Similarly, for test functions where the global optima occurs on the boundaries of the function, such as the Hölder Table, the conservative superlinear distance metric demonstrates greater flexibility than the estimated lower bound functions generated by other distance metrics in capturing those minimum values.

In functions that exhibit a consistent pattern of change in the function value (either increasing or decreasing), the estimated lower bound functions obtained by the linear and sublinear distance metrics are outperform other estimated lower bound functions. However, when it comes to establishing an estimated lower bound function, the sublinear distance metric is more aggressive than the linear distance metric. This means that for functions with characteristics similar to sublinear functions, such as the Rosenbrock function, the linear distance metric outperforms the sublinear distance metric in terms of the number of valid lower bound estimates produced (satisfying Equation (23a)). Although the sublinear distance metric generates lower bound estimates for fewer points, those lower bound estimates are tighter compared to the linear distance metric’s lower bound estimates for the same points. Consequently, the linear distance metric’s estimated lower bound function can be considered more conservative, while the sublinear distance metric is more aggressive and provides tighter estimated lower bound functions.

Furthermore, the standard deviation (STD) of the total number of times the winning distance metric has won for each test function reveals that the linear distance metric exhibits less volatility in generating estimated lower bound functions across the 30 executions with different random data distributions. This characteristic can be attributed to the complexity associated with the properties of the test functions and their sensitivity to variations in data distribution.

### 6.3. Surrogate Optimization Results

Acquisition functions are fundamental in surrogate optimization, guiding the selection of new evaluation points while balancing exploration and exploitation [75]. In this study, we incorporate our proposed approach as an acquisition function and compare its performance against two widely used baselines: Expected Improvement (EI) and Upper Confidence Bound (UCB), as discussed in Section 4.2. The surrogate optimization experiments are conducted on the same benchmark functions listed in Table 5, ensuring consistent domain constraints and dimensionality across all tests. Each optimization problem is executed for 50 independent runs, each with a computational budget of 100 iterations, and results are aggregated across all runs to ensure statistical robustness.

For candidate selection, we employ the Sobol sequence to generate low-discrepancy candidate sets, ensuring an efficient and well-distributed sampling of the search space [11]. This approach provides a structured method for candidate point generation, enhancing the consistency and reliability of the optimization process. To evaluate performance, we present plots depicting the true function value  $f(\mathbf{x})$  of the best-known solution at each iteration, with standard deviation

represented as error bars. Additionally, we provide boxplots to illustrate the distribution of the best-known solution values across all runs, further validating the comparative effectiveness of different acquisition functions.

As illustrated in Figure 4, our proposed methodology consistently outperforms EI and UCB in efficiently identifying near-optimal solutions across all benchmark functions. Unlike EI and UCB, which rely on probabilistic estimates, our approach explicitly computes a lower bound estimate of the function value at each candidate point, making selection decisions based on the most promising lower bound estimate. This systematic adaptation to the function’s underlying structure, achieved through dynamic distance metric scaling, enables more reliable and efficient convergence. Notably, in particularly challenging test functions such as the Holder Table, Ackley, and Shubert functions, where the presence of steep valleys, multiple local minima, and complex surface variations significantly hinder convergence, our method rapidly identifies promising regions. As a result, it converges to the global minimum substantially faster, even within a constrained computational budget.

A closer examination of these results reveals that functions exhibiting sharp discontinuities and irregular landscapes—such as deep drops leading to valleys with numerous local minima or global optima positioned at the domain’s borders or center—pose a major challenge for conventional acquisition functions. These functions often cause optimizers to get trapped in local optima, preventing convergence within the available computational budget. In contrast, our methodology effectively detects shifts in function behavior early in the optimization process, efficiently focusing the search space as early as 20 iterations. This early adaptation results in a significant and rapid reduction in the true function value of the best-known solution, as demonstrated in Figures 4e-4g.

Similarly, in functions characterized by a mild and gradual descent toward the global minimum, such as Branin, Sphere, and Rosenbrock, where the absence of a pronounced pattern and the presence of topological irregularities—such as those seen in the Camel function—complicate the identification of the global optima, our approach continues to demonstrate superior performance. By efficiently navigating these complex landscapes, our methodology consistently outperforms both EI and UCB, as evidenced in Figures 4i, 4h, 4b, and 4d.

Figure 5 presents the boxplots of the area under the curve (AUC) of the true function value for the best-known solution, aggregated across all 50 runs for each test function and acquisition function methodology in the surrogate optimization experiments. These boxplots illustrate the distribution of AUC values across 50 independent random-seeded runs, providing insight into the consistency and robustness of each method. A lower and more stable AUC distribution indicates a more effective acquisition function in steering the optimization process toward optimal solutions in the minimization setting.

The results further confirm that our structured distance metric framework consistently achieves lower and more stable AUC values compared to traditional acquisition functions, as shown in Figure 5. This highlights the efficiency of our approach in facilitating more effective search space exploration and accelerating convergence to high-quality solutions, offering a robust alternative to

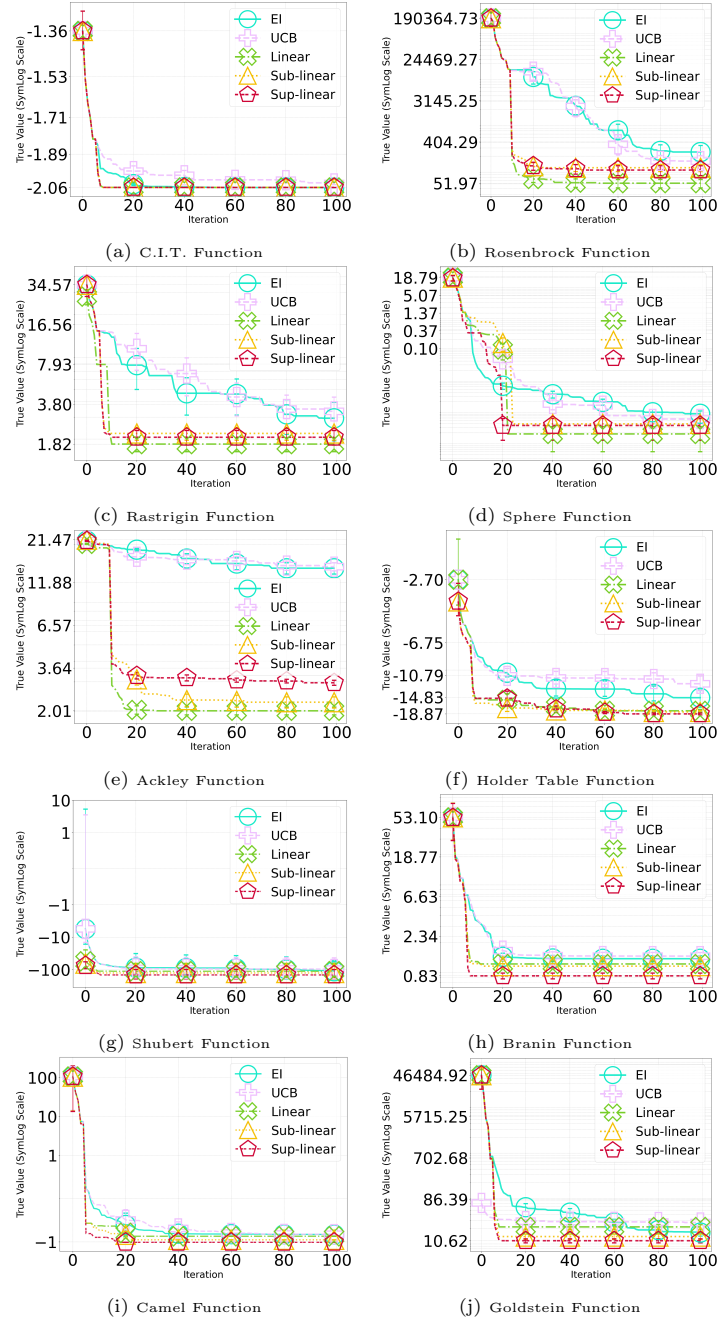


Figure 4: Comparison of the true function value of the best-known solution over 100 iterations, aggregated across 50 runs for each test function.



conventional methodologies in surrogate optimization. Additionally, the winning methodology in Figure 5 is highlighted in green, demonstrating that the distance metric-based approach generally yields lower and more compact AUC values, reflecting reduced variance and a stronger alignment with the global optima of the test functions.

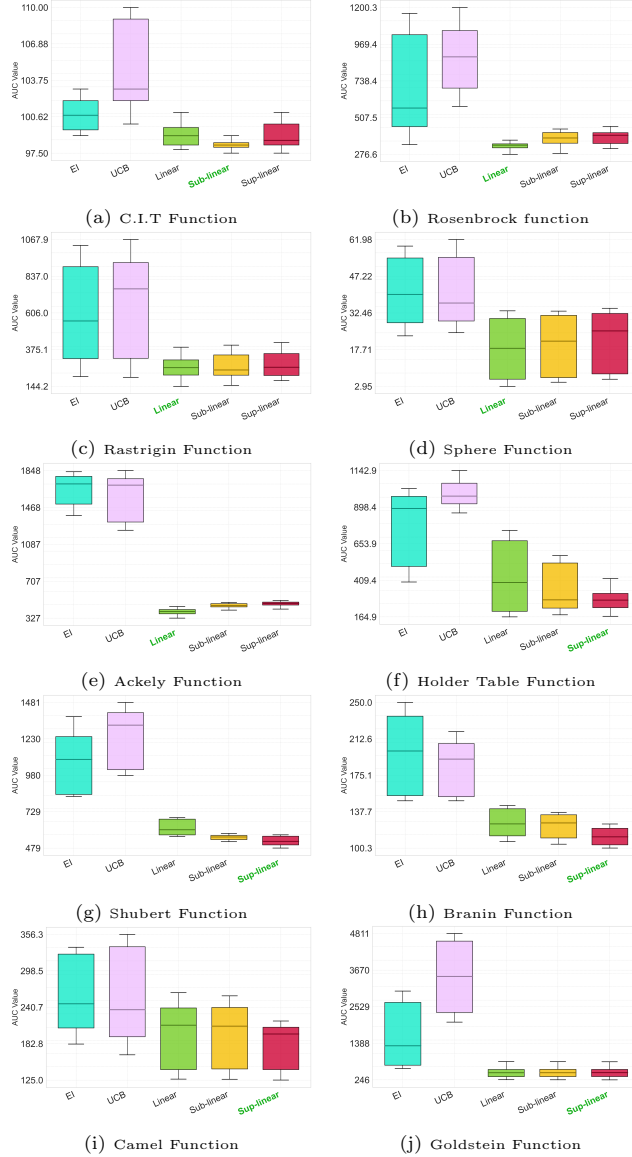


Figure 5: Comparison of AUC boxplots illustrating the distribution of the true function values of the best-known solutions across different acquisition functions applied to ten benchmark test functions in the minimization-based surrogate optimization setting.

## 7. Conclusion and Future Direction

In this paper, we have presented a novel approach for estimating the Lipschitz constant and constructing estimated lower bound functions using surrogate models and alternative nonlinear distance metrics. Specifically, our method utilizes a surrogate model to approximate the response surface of the objective function and compares the performance of three types of distance metrics: superlinear, sublinear, and linear to estimate a Lipschitz constant. We have developed theoretical results showing how superlinear distance metrics lead to more conservative estimated lower bound functions. Through experiments on various benchmark test functions, we have evaluated the effectiveness the method for each of the nonlinear distance metrics by measuring the gap between the lower bound estimates and the true function value for a set of unevaluated observations. Additionally, we have identified the best distance metric for each run based on this metric. Moreover, we have evaluated the performance of our methodology in using distance metrics to construct estimated lower bound functions in surrogate optimization settings across various test functions as an acquisition function within this setting to guide the search process of a global optima of the function.

Our findings demonstrate the superiority of our proposed method in constructing estimated lower bound functions when compared to alternative approaches such as the Working-Hotelling statistical method and the Standard Lipschitz method as well as outperforming commonly used acquisition functions in surrogate optimization such as EI and UCB. We have observed that the superlinear distance metric generally provides more conservative and versatile results across a broader range of function types, while the linear distance metric tends to be more aggressive, yielding tighter estimated lower bound functions for certain function types. Consequently, we have concluded that the choice of distance metric should depend on the characteristics of the objective function and the desired trade-off between tightness and conservatism.

In conclusion, our proposed method offers a novel and effective approach for constructing estimated lower bound functions, which can be directly applied in various domains such as surrogate optimization. By leveraging surrogate models and comparing superlinear and sublinear distance metrics as well as linear ones, we have demonstrated superior performance over existing methods. With future enhancements and applications, we anticipate that our method will contribute to advancements in optimization techniques and find practical utility in a wide range of domains.

## 8. Acknowledgements

This research is partially funded by National Science Foundation Awards CHE-2108767 and CMMI-1926792.

## 9. Statements and Declarations

The authors have no relevant financial or non-financial interests to disclose.

The authors have no competing interests to declare that are relevant to the content of this article.

All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

The authors have no financial or proprietary interests in any material discussed in this article.

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

## 10. Appendix

The appendix provides additional details on the theoretical contributions, including the proofs of theorems, lemmas, and the proposition. It also contains supplementary results that further support the findings presented in the main text.

### 10.1. Proposition 1

*Proof of Proposition 1:* From Equations (9), and (10), we have the following:

$$\varepsilon < \|\mathbf{x}^i - \mathbf{x}^j\| < M \quad \forall \mathbf{x}^i \in \mathcal{I}, \mathbf{x}^j \in \mathcal{U} \quad (29)$$

Since  $p > 0$ ,

$$\varepsilon^p < \|\mathbf{x}^i - \mathbf{x}^j\|^p < M^p. \quad (30)$$

Moreover, by Equation (5), we have the following:

$$|f(\mathbf{x}^i) - f(\mathbf{x}^j)| \leq l \|\mathbf{x}^i - \mathbf{x}^j\| \times \frac{\|\mathbf{x}^i - \mathbf{x}^j\|^p}{\|\mathbf{x}^i - \mathbf{x}^j\|^p} \quad (31)$$

$$= \left( \frac{l \|\mathbf{x}^i - \mathbf{x}^j\|}{\|\mathbf{x}^i - \mathbf{x}^j\|^p} \right) \|\mathbf{x}^i - \mathbf{x}^j\|^p < \frac{lM}{\varepsilon^p} \|\mathbf{x}^i - \mathbf{x}^j\|^p \quad (32)$$

Consequently, for any constant  $l_f \geq \frac{lM}{\varepsilon^p}$ , Equation (8) is true.  $\square$

### 10.2. Theorem 5.1

*Proof of Theorem 5.1:* We aim to prove that there exists a data point  $\mathbf{x}^i \in \mathcal{I}$  such that  $\eta_i = 0$  in an optimal solution to LP (14), and it maximizes the fraction in Equation (18). Let  $\tilde{\eta}$  be a variable representing the maximum of all  $\eta_i$  in Equation (14). We then rewrite the LP formula as follows:

$$\min \tilde{\eta} : \quad (33a)$$

s.t.

$$-\hat{l}_f \varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) + \eta_i = f(\mathbf{x}^i) - \hat{f}_{\mathcal{I}^i}(\mathbf{x}^i) \quad \forall i = 1, \dots, N \quad (33b)$$

$$\eta_i \leq \tilde{\eta} \quad \forall i = 1, \dots, N \quad (33c)$$

$$\eta_i \geq 0 \quad \forall i = 1, \dots, N \quad (33d)$$

The remainder of this proof uses the strong duality theorem in linear programming [19], which states that for any optimal solution to a linear program, there must exist an optimal dual solution that is feasible to the dual linear programming formulation, and the optimal solution and the dual optimal solution satisfy complementary slackness conditions. Consequently, the structure of this proof is by contradiction and shows that if all  $\eta_i$  are positive, then no such dual solution can be constructed. Let  $y_i$  and  $\pi_i$  be the dual variables of Equations (33b) and (33c), respectively. To establish the existence of a data point  $\mathbf{x}^i$  such that  $\eta_i = 0$ , we introduce the linear programming complementary slackness conditions as follows:

$$(y_i - \pi_i)\eta_i = 0 \quad \forall i = 1, \dots, N \quad (34a)$$

$$(\tilde{\eta} - \eta_i)\pi_i = 0 \quad \forall i = 1, \dots, N \quad (34b)$$

We now write the dual linear programming formulation:

$$\max \quad \sum_{i=1}^N (f(\mathbf{x}^i) - \hat{f}_{\mathcal{I}^i}(\mathbf{x}^i))y_i \quad (35a)$$

s.t.

$$y_i - \pi_i \leq 0 \quad \forall i = 1, \dots, N \quad (\text{constraint for } \eta_i) \quad (35b)$$

$$\sum_{i=1}^N -\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)y_i = 0 \quad (\text{constraint for } \hat{l}_f) \quad (35c)$$

$$\sum_{i=1}^N \pi_i = 1 \quad \forall i = 1, \dots, N \quad (\text{constraint for } \tilde{\eta}) \quad (35d)$$

$$\pi_i \geq 0 \quad \forall i = 1, \dots, N \quad (35e)$$

Now, assume to the contrary that  $\eta_i > 0$  for all  $i = 1, \dots, N$ . According to Equation (34a), we have  $y_i = \pi_i$  for all  $i = 1, \dots, N$ . Therefore, by constraints (35d) and (35e),

$$y_i \geq 0, \forall i = 1, \dots, N, \quad \text{and} \quad \sum_{i=1}^N y_i = 1 \quad (36a)$$

Considering the values of  $\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)$  are always positive, we have:

$$\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) > 0 \Rightarrow -\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) < 0 \quad \forall i = 1, \dots, N \quad (37a)$$

$$(37b)$$

Combining (36a) and (37a), we have the following.

$$\sum_{i=1}^N -\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) y_i < 0 \quad (38a)$$

However, this contradicts Equation (35c).

Now that we have established the existence of a data point  $\mathbf{x}^i \in \mathcal{I}$  where  $\eta_i = 0$  in a solution to the LP (14), our goal is to demonstrate that the fraction in Equation (18) achieves its maximum value at this point  $\mathbf{x}^i$ . Assume  $\tilde{\mathbf{x}}$  represents the point where  $\eta = 0$ , which we refer to as the *ratio maximizer*. Recalling Equation (14b) we have:

$$\hat{l}_f = \frac{\hat{f}_{\mathcal{I}^i}(\mathbf{x}^i) - f(\mathbf{x}^i)}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)} + \frac{\eta_i}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)}, \quad \forall i = 1, \dots, N. \quad (39)$$

Since we know that  $\eta_i \geq 0$ ,  $\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i) > 0$ , and  $\eta = 0$  for the ratio maximizer, it follows that:

$$\frac{\eta_i}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)} \geq 0 \implies \hat{l}_f = \frac{\hat{f}_{\mathcal{I}^i}(\tilde{\mathbf{x}}) - f(\tilde{\mathbf{x}})}{\varphi(\tilde{\mathbf{x}}, \tilde{\mathbf{x}})} \geq \frac{\hat{f}_{\mathcal{I}^i}(\mathbf{x}^i) - f(\mathbf{x}^i)}{\varphi(\mathbf{x}^i, \bar{\mathbf{x}}^i)} \quad \forall i = 1, \dots, N. \quad (40)$$

Thus, the claim holds true.  $\square$

For the remainder of this section, we assume the following. Let  $\mathcal{I}$  be a given set of evaluated data points and  $\mathcal{U}$  be a set of unevaluated data points. Consider two distance metric  $\varphi^{(p^i)}$  and  $\varphi^{(p^j)}$  with exponents  $p^i$  and  $p^j$  and constant coefficients  $\alpha^i$  and  $\alpha^j$ . Suppose Algorithm 2 is used with  $\varphi^{(p^i)}$  and  $\varphi^{(p^j)}$  and all other inputs being the same to determine estimated Lipschitz constants,  $\hat{l}^i$  and  $\hat{l}^j$ , ratio maximizers,  $\tilde{\mathbf{x}}^i$  and  $\tilde{\mathbf{x}}^j$ , complementary subsets,  $\tilde{\mathcal{I}}^i$  and  $\tilde{\mathcal{I}}^j$ , nearest points in the complementary sub sets,  $\tilde{\mathbf{x}}^i$  and  $\tilde{\mathbf{x}}^j$ , functions  $\rho^{(p^i)}$  and  $\rho^{(p^j)}$ , and estimated lower bound functions  $\hat{f}_{lb}^{(p^i)}$  and  $\hat{f}_{lb}^{(p^j)}$ , respectively.

### 10.3. Lemma 10.1

**Lemma 10.1.** *If  $p^i = p^j$ , then  $\rho^{(p^i)} = \rho^{(p^j)}$ .*

*Proof of Lemma 10.1.* By recalling Equations (18), we have:

$$\hat{l}^i = \max_{\mathbf{x}^k \in \mathcal{I}} \frac{\hat{f}_{\bar{\mathcal{I}}^k}(\mathbf{x}^k) - f(\mathbf{x}^k)}{\alpha^i \|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^{p^i}} \quad (41)$$

$$\hat{l}^j = \max_{\mathbf{x}^k \in \mathcal{I}} \frac{\hat{f}_{\bar{\mathcal{I}}^k}(\mathbf{x}^k) - f(\mathbf{x}^k)}{\alpha^j \|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^{p^j}} \quad (42)$$

$$(43)$$

Since  $p^i = p^j$  and  $\alpha^i$  and  $\alpha^j$  are constants,

$$\alpha^i \hat{l}^i = \max_{\mathbf{x}^k \in \mathcal{I}} \frac{\hat{f}_{\bar{\mathcal{I}}^k}(\mathbf{x}^k) - f(\mathbf{x}^k)}{\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^{p^i}} = \alpha^j \hat{l}^j \quad (44)$$

Hence,  $\forall x \in \mathcal{U}$ ,  $\rho^{(p^i)}(x) = \hat{l}^i \varphi^i(\mathbf{x}, \bar{\mathbf{x}}) = \alpha^i \hat{l}^i \|\mathbf{x} - \bar{\mathbf{x}}\|^{p^i} = \alpha^j \hat{l}^j \|\mathbf{x} - \bar{\mathbf{x}}\|^{p^i} = \hat{l}^j \varphi^j(\mathbf{x}, \bar{\mathbf{x}}) = \rho^{(p^j)}(\mathbf{x})$ .  $\square$

From Lemma 10.3, we observe that the function  $\rho$  is independent of the constant coefficient  $\alpha$ . Consequently, for the remainder of the section, we assume that the constant coefficients  $\alpha^i = \alpha^j = 1$ . Moreover, without loss of generality, we assume  $p^i > p^j$ .

#### 10.4. Corollary 10.2

As noted in Remark 1, in many situations, the ratio maximizers are the same. Corollary 10.2 shows that this can imply an ordering of the estimated Lipschitz constants.

**Corollary 10.2.** *If the ratio maximizers are the same,  $\tilde{\mathbf{x}}^i = \tilde{\mathbf{x}}^j$ , and  $\|\tilde{\mathbf{x}}^i - \bar{\tilde{\mathbf{x}}}^i\| > 1$ , then  $\hat{l}^i < \hat{l}^j$ .*

*Proof of Corollary 10.2.* Since  $\tilde{\mathbf{x}}^i = \tilde{\mathbf{x}}^j$  we have:

$$\hat{l}^i \varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i) = \hat{f}_{\bar{\mathcal{I}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i) = \hat{f}_{\bar{\mathcal{I}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j) = \hat{l}^j \varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i) \quad (45)$$

Now we can conclude:

$$\|\tilde{\mathbf{x}}^i - \bar{\tilde{\mathbf{x}}}^i\| > 1, p^i > p^j \Rightarrow \varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i) > \varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i) \Rightarrow \hat{l}^i < \hat{l}^j \quad (46)$$

$\square$

#### 10.5. Lemma 10.3

By Equation (18) and the definitions of the ratio maximizers, we have the following inequalities:

$$\hat{l}^i = \frac{\hat{f}_{\bar{\mathcal{I}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i)} \geq \frac{\hat{f}_{\bar{\mathcal{I}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^j, \bar{\tilde{\mathbf{x}}}^j)} \quad (47)$$

$$\hat{l}^j = \frac{\hat{f}_{\bar{\mathcal{I}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\tilde{\mathbf{x}}}^j)} \geq \frac{\hat{f}_{\bar{\mathcal{I}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\tilde{\mathbf{x}}}^i)} \quad (48)$$

Now we show  $\|\tilde{\mathbf{x}}^i - \bar{\tilde{\mathbf{x}}}^i\| \leq \|\tilde{\mathbf{x}}^j - \bar{\tilde{\mathbf{x}}}^j\|$ .

**Lemma 10.3.**  $\|\tilde{\mathbf{x}}^i - \tilde{\mathbf{x}}^i\| \leq \|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\|$ .

*Proof of Lemma 10.3.* Assume to the contrary that  $\|\tilde{\mathbf{x}}^i - \tilde{\mathbf{x}}^i\| > \|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\|$ .

Consider the function  $\frac{\varphi^{(p^i)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)}$  as follow:

$$\frac{\varphi^{(p^i)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)} = \|\mathbf{x}^i - \tilde{\mathbf{x}}^i\|^{p^i - p^j} \quad (49)$$

Since  $p^i > p^j$ , this is an increasing distance function. Because  $\frac{\varphi^{(p^i)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\mathbf{x}^i, \tilde{\mathbf{x}}^i)}$  is increasing, we know that:

$$\frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} > \frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \quad (50)$$

Multiplying both sides by  $\frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)}$

$$\frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} > \frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i) \times (\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j))}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i) \times \varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \quad (51)$$

From Equation (48),  $\frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \geq \frac{\hat{f}_{\tilde{\mathcal{T}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}$ , so multiplying both sides by  $\frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}$ , we have:

$$\frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \times \frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \geq \frac{\hat{f}_{\tilde{\mathcal{T}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \times \frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \quad (52)$$

$$\frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i) \times (\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j))}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i) \times \varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \geq \frac{\hat{f}_{\tilde{\mathcal{T}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \quad (53)$$

Combining Equations (51) and (53) contradicts Equation (47), which states that  $\frac{\hat{f}_{\tilde{\mathcal{T}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)} \geq \frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^i)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)}$ . Hence  $\|\tilde{\mathbf{x}}^i - \tilde{\mathbf{x}}^i\| \leq \|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\|$ .  $\square$

#### 10.6. Lemma 10.4

**Lemma 10.4.** *The surrogate model prediction error for  $\tilde{\mathbf{x}}^i$  is no more than the prediction error for  $\tilde{\mathbf{x}}^j$ ; that is:*

$$\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j) \geq \hat{f}_{\tilde{\mathcal{T}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)$$

*Proof of Lemma 10.4.* We rewrite  $\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)$  as below:

$$\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j) = \frac{\hat{f}_{\tilde{\mathcal{T}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \times \varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j) \quad (54)$$

Recalling from Equation (48):

$$\frac{\hat{f}_{\bar{\mathcal{L}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j)} \times \varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j) \geq \frac{\hat{f}_{\bar{\mathcal{L}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\mathbf{x}}^i)} \times \varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j) \quad (55)$$

Since  $\|\tilde{\mathbf{x}}^i - \bar{\mathbf{x}}^i\| \leq \|\tilde{\mathbf{x}}^j - \bar{\mathbf{x}}^j\|$  from Lemma 10.3, and  $\varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})$  is increasing, we have  $\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j) \geq \varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\mathbf{x}}^i)$ , so  $\frac{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^i, \bar{\mathbf{x}}^i)} \geq 1$ . Hence:

$$\hat{f}_{\bar{\mathcal{L}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j) \geq \hat{f}_{\bar{\mathcal{L}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i) \quad (56)$$

□

#### 10.7. Lemma 10.5

**Lemma 10.5.**

$$\forall \mathbf{x} \in \mathcal{U} \text{ such that } \|\mathbf{x} - \bar{\mathbf{x}}\| > \left( \frac{\hat{l}^j}{\hat{l}^i} \right)^{\frac{1}{p^i - p^j}} \implies \rho^{(p^i)}(\mathbf{x}) > \rho^{(p^j)}(\mathbf{x}). \quad (57)$$

*Proof of Lemma 10.5.* Recalling Equation (18), for  $\varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}})$  and  $\varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})$  we have:

$$\frac{\hat{l}^i \varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}})}{\hat{l}^j \varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})} = \frac{\hat{l}^i \|\mathbf{x} - \bar{\mathbf{x}}\|^{p^i}}{\hat{l}^j \|\mathbf{x} - \bar{\mathbf{x}}\|^{p^j}} = \frac{\hat{l}^i}{\hat{l}^j} \|\mathbf{x} - \bar{\mathbf{x}}\|^{p^i - p^j} \quad (58)$$

Moreover, Equation (58) is a distance metric of the form (17), which passes through value 1 at point  $\mathbf{x}'$  where  $\|\mathbf{x}' - \bar{\mathbf{x}}'\| = \left( \frac{\hat{l}^j}{\hat{l}^i} \right)^{\frac{1}{p^i - p^j}}$ . Thus we have the following:

$$\forall \mathbf{x} \in \mathcal{U} \text{ s.t. } \|\mathbf{x} - \bar{\mathbf{x}}\| > \left( \frac{\hat{l}^j}{\hat{l}^i} \right)^{\frac{1}{p^i - p^j}} \implies \frac{\hat{l}^i \varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}})}{\hat{l}^j \varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})} > 1 \quad (59)$$

Hence, the claim  $\hat{l}^i \varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}}) > \hat{l}^j \varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})$  holds. □

#### 10.8. Lemma 10.6

**Lemma 10.6.**

$$\forall \mathbf{x} \in \mathcal{U} \text{ such that } \|\mathbf{x} - \bar{\mathbf{x}}\| > \|\tilde{\mathbf{x}}^j - \bar{\mathbf{x}}^j\| \implies \rho^{(p^i)}(\mathbf{x}) > \rho^{(p^j)}(\mathbf{x}) \quad (60)$$

*Proof of Lemma 10.6.* From Lemma 10.5, it remains to be proven that  $\|\tilde{\mathbf{x}}^j - \bar{\mathbf{x}}^j\| \geq \left( \frac{\hat{l}^j}{\hat{l}^i} \right)^{\frac{1}{p^i - p^j}}$ . From Equations (47), and (48) we have the following:

$$\left( \frac{\hat{l}^j}{\hat{l}^i} \right)^{\frac{1}{p^i - p^j}} = \left( \frac{\hat{f}_{\bar{\mathcal{L}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \bar{\mathbf{x}}^j)} \times \frac{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \bar{\mathbf{x}}^i)}{\hat{f}_{\bar{\mathcal{L}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)} \right)^{\frac{1}{p^i - p^j}} \quad (61)$$



From Equation (47) we conclude:

$$\begin{aligned} & \left( \frac{\hat{f}_{\tilde{\mathbf{x}}^j}(\tilde{\mathbf{x}}^j) - f(\tilde{\mathbf{x}}^j)}{\hat{f}_{\tilde{\mathbf{x}}^i}(\tilde{\mathbf{x}}^i) - f(\tilde{\mathbf{x}}^i)} \right)^{\frac{1}{p^i - p^j}} \times \left( \frac{\varphi^{(p^i)}(\tilde{\mathbf{x}}^i, \tilde{\mathbf{x}}^i)}{\varphi^{(p^j)}(\tilde{\mathbf{x}}^j, \tilde{\mathbf{x}}^j)} \right)^{\frac{1}{p^i - p^j}} \leq \\ & \longrightarrow \left( \frac{\|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\|^{p^i}}{\|\tilde{\mathbf{x}}^i - \tilde{\mathbf{x}}^i\|^{p^i}} \cdot \frac{\|\tilde{\mathbf{x}}^i - \tilde{\mathbf{x}}^i\|^{p^i}}{\|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\|^{p^j}} \right)^{\frac{1}{p^i - p^j}} = \|\tilde{\mathbf{x}}^j - \tilde{\mathbf{x}}^j\| \end{aligned} \quad (62)$$

□

With the necessary lemmas established, we now proceed to the proof of Theorem (5.2).

### 10.9. Theorem 5.2

*Proof of Theorem 5.2.* From Equations (20) and (21) we construct lower bound estimates for any point  $\mathbf{x} \in \mathcal{U}$  as follows:

$$\hat{f}_{lb}^{(p^i)}(\mathbf{x}) = \hat{f}(\mathbf{x}) - \rho^{(p^i)}(\mathbf{x}) = \hat{f}(\mathbf{x}) - (\hat{l}^i \cdot \varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}})) \quad (63)$$

$$\hat{f}_{lb}^{(p^j)}(\mathbf{x}) = \hat{f}(\mathbf{x}) - \rho^{(p^j)}(\mathbf{x}) = \hat{f}(\mathbf{x}) - (\hat{l}^j \cdot \varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}})) \quad (64)$$

Recalling Lemmas 10.5 and 10.6, we have:

$$\hat{l}^i \cdot \varphi^{(p^i)}(\mathbf{x}, \bar{\mathbf{x}}) > \hat{l}^j \cdot \varphi^{(p^j)}(\mathbf{x}, \bar{\mathbf{x}}) \implies \rho^{(p^i)}(\mathbf{x}) > \rho^{(p^j)}(\mathbf{x}) \quad (65)$$

Hence we have:

$$\hat{f}(\mathbf{x}) - \rho^{(p^i)}(\mathbf{x}) < \hat{f}(\mathbf{x}) - \rho^{(p^j)}(\mathbf{x}) \implies \hat{f}_{lb}^{(p^i)}(\mathbf{x}) < \hat{f}_{lb}^{(p^j)}(\mathbf{x}) \quad (66)$$

□

### References

- [1] Ahmed Abdelkader, Sunil Arya, Guilherme D da Fonseca, and David M Mount. Approximate nearest neighbor searching with non-euclidean and weighted distances. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 355–372. SIAM, 2019.
- [2] R.A. Adams and J.J.F. Fournier. *Sobolev Spaces*. ISSN. Elsevier Science, <https://books.google.com/books?id=R5A65Koh-EoC>, 2003. ISBN 9780080541297.
- [3] Alekh Agarwal and Leon Bottou. A lower bound for the optimization of finite sums. In *International conference on machine learning*, pages 78–86. PMLR, 2015.

- [4] Mohamed Osama Ahmed, Sharan Vaswani, and Mark Schmidt. Combining bayesian optimization and lipschitz optimization. *Machine Learning*, 109: 79–102, 2020.
- [5] Hadis Anahideh, Jay Rosenberger, and Victoria Chen. High-dimensional black-box optimization under uncertainty. *Computers & Operations Research*, 137:105444, 2022.
- [6] Sylvain Arlot and Alain Celisse. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4:40–79, 2009.
- [7] Charles Audet, J Denni, Douglas Moore, Andrew Booker, and Paul Frank. A surrogate-model-based method for constrained optimization. In *8th Symposium on Multidisciplinary Analysis and Optimization*, page 4891, 2000.
- [8] Dominique Azé and J N Corvellec. Nonlinear local error bounds via a change of metric. *Journal of Fixed Point Theory and Applications*, 16: 351–372, 2014.
- [9] Lisa Beck and Giuseppe Mingione. Lipschitz bounds and nonuniform ellipticity. *Communications on Pure and Applied Mathematics*, 73(5):944–1034, 2020.
- [10] Chao Bian, Xiaofang Wang, Wenyang Shao, Jianchi Xin, Rui Hu, Yeming Lu, and Haitao Liu. Adaptive confidence bound based bayesian optimization via potentially optimal lipschitz conditions. *Engineering Optimization*, 55(12):2051–2069, 2023.
- [11] Paul Bratley and Bennett L Fox. Algorithm 659: Implementing sobol’s quasirandom sequence generator. *ACM Transactions on Mathematical Software (TOMS)*, 14(1):88–100, 1988.
- [12] Peter Bubenik et al. Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research*, 16(1):77–102, 2015.
- [13] Coralia Cartis, Jaroslav M Fowkes, and Nicholas IM Gould. Branching and bounding improvements for global optimization algorithms with lipschitz continuity properties. *Journal of Global Optimization*, 61:429–457, 2015.
- [14] Leocadio G Casado, José A Martínez, Inmaculada García, and Ya D Sergeyev. New interval analysis support functions using gradient information in a global minimization algorithm. *Journal of Global Optimization*, 25:345–362, 2003.
- [15] Ankush Chakrabarty, Devesh K Jha, and Yebin Wang. Data-driven control policies for partially known systems via kernelized lipschitz learning. In *2019 American Control Conference (ACC)*, pages 4192–4197. IEEE, 2019.
- [16] Mian-tao Chao and Cao-zong Cheng. Linear and nonlinear error bounds for lower semicontinuous functions. *Optimization Letters*, 8:1301–1312, 2014.

- [17] Erh-Chung Chen, Pin-Yu Chen, I Chung, Che-Rung Lee, et al. Data-driven lipschitz continuity: A cost-effective approach to improve adversarial robustness. *arXiv preprint arXiv:2406.19622*, 2024.
- [18] Alison Cozad, Nikolaos V Sahinidis, and David C Miller. Learning surrogate models for simulation-based optimization. *AIChE Journal*, 60(6): 2211–2227, 2014.
- [19] George B Dantzig. Linear programming. *Operations research*, 50(1):42–47, 2002.
- [20] Roy de Winter, Bas van Stein, and Thomas Bäck. Multi-point acquisition function for constraint parallel efficient multi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 511–519, 2022.
- [21] Santanu S Dey, Yatharth Dubey, and Marco Molinaro. Lower bounds on the size of general branch-and-bound trees. *Mathematical Programming*, 198(1):539–559, 2023.
- [22] William R Esposito and Christodoulos A Floudas. Deterministic global optimization in nonlinear optimal control problems. *Journal of global optimization*, 17:97–126, 2000.
- [23] Qi Fan and Jiaqiao Hu. Surrogate-based promising area search for lipschitz continuous simulation optimization. *INFORMS Journal on Computing*, 30(4):677–693, 2018.
- [24] Christodoulos A Floudas. *Deterministic global optimization: theory, methods and applications*, volume 37. Springer Science & Business Media, <https://doi.org/10.1007/978-1-4757-4949-6>, 1999.
- [25] Christodoulos A Floudas and Panos M Pardalos. *State of the art in global optimization: computational methods and applications*. Springer Science & Business Media, <https://doi.org/10.1007/978-1-4613-3437-8>, 1996.
- [26] Alexander IJ Forrester and Andy J Keane. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45(1-3):50–79, 2009.
- [27] Alexander IJ Forrester, András Sóbester, and Andy J Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the Royal Society Mathematical, Physical and Engineering Sciences*, 463(2088):3251–3269, 2007.
- [28] B G.-Tóth, Leocadio G Casado, Eligius MT Hendrix, and Frédéric Messine. On new methods to construct lower bounds in simplicial branch and bound based on interval arithmetic. *Journal of Global Optimization*, 80(4):779–804, 2021.

- [29] Dirk Gorissen, Karel Crombecq, Ivo Couckuyt, and Tom Dhaene. Automatic approximation of expensive functions with active learning. *Foundations of Computational, Intelligence Volume 1: Learning and Approximation*, 1:35–62, 2009.
- [30] Jean-Bastien Grill, Michal Valko, and Rémi Munos. Black-box optimization of noisy functions with unknown smoothness. *Advances in Neural Information Processing Systems*, 28:1–9, 2015.
- [31] DJ Hand. Branch and bound in statistical data analysis. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 30(1):1–13, 1981.
- [32] Pierre Hansen and Brigitte Jaumard. *Lipschitz optimization*. Springer, [https://doi.org/10.1007/978-1-4615-2025-2\\_9](https://doi.org/10.1007/978-1-4615-2025-2_9), 1995.
- [33] Reiner Horst and Panos M Pardalos. *Handbook of global optimization*. Springer Science & Business Media, <https://doi.org/10.1007/978-1-4615-2025-2>, 2013.
- [34] Hao Huang, Pariyakorn Maneekul, Danielle F Morey, Zelda B Zabinsky, and Giulia Pedrielli. A computational study of probabilistic branch and bound with multilevel importance sampling. In *2022 Winter Simulation Conference (WSC)*, pages 3251–3262, <https://doi.org/10.1109/WSC57314.2022.10015267>, 2022. IEEE.
- [35] Julien Walden Huang, Stephen J Roberts, and Jan-Peter Calliess. On the sample complexity of lipschitz constant estimation. *Transactions on Machine Learning Research*, 2023.
- [36] Zeyuan Jin, Mohammad Khajenejad, and Sze Zheng Yong. Data-driven model invalidation for unknown lipschitz continuous systems via abstraction. In *2020 American Control Conference (ACC)*, pages 2975–2980, <https://doi.org/10.23919/acc45564.2020.9147725>, 2020. IEEE.
- [37] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13:455–492, 1998.
- [38] Rohit Kannan and Paul I Barton. Convergence-order analysis of branch-and-bound algorithms for constrained problems. *Journal of Global Optimization*, 71:753–813, 2018.
- [39] SD Karhbet and Ralph Baker Kearfott. Range bounds of functions over simplices, for branch and bound algorithms. *Reliable Comput*, 25(7):53–73, 2017.
- [40] Ramkumar Karuppiah and Ignacio E Grossmann. Global optimization for the synthesis of integrated water systems in chemical processes. *Computers & Chemical Engineering*, 30(4):650–673, 2006.

- [41] Jungtaek Kim and Seungjin Choi. On local optimizers of acquisition functions in bayesian optimization. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part II*, pages 675–690. Springer, 2021.
- [42] Samory Kpotufe. Lipschitz density-ratios, structured data, and data-driven tuning. In *Artificial Intelligence and Statistics*, pages 1320–1328. PMLR, 2017.
- [43] Jakub Kudela and Radomil Matousek. Combining lipschitz and rbf surrogate models for high-dimensional computationally expensive problems. *arXiv preprint arXiv:2204.14236*, 2022.
- [44] Jakub Kudela and Radomil Matoušek. Combining lipschitz and rbf surrogate models for high-dimensional computationally expensive problems. *Information Sciences*, 619:457–477, 2023.
- [45] Dmitri E Kvasov and Ya D Sergeyev. Lipschitz global optimization methods in control problems. *Automation and Remote Control*, 74:1435–1448, 2013.
- [46] Dmitri E Kvasov and Yaroslav D Sergeyev. A univariate global search working with a set of lipschitz constants for the first derivative. *Optimization Letters*, 3(2):303–318, 2009.
- [47] Guanghui Lan. An optimal method for stochastic composite optimization. *Mathematical Programming*, 133(1-2):365–397, 2012.
- [48] Leo Liberti. Reformulation and convex relaxation techniques for global optimization. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 2(3):255–258, 2004.
- [49] Leo Liberti. Introduction to global optimization. *Ecole Polytechnique*, 1: 1–43, 2008.
- [50] Youdong Lin and Mark A Stadtherr. Deterministic global optimization of nonlinear dynamic systems. *AIChE Journal*, 53(4):866–875, 2007.
- [51] Haitao Liu, Shengli Xu, Ying Ma, and Xiaofang Wang. Global optimization of expensive black box functions using potential lipschitz constants and response surfaces. *Journal of Global Optimization*, 63(2):229–251, 2015.
- [52] Yiping Liu, Dunwei Gong, Xiaoyan Sun, and Yong Zhang. Many-objective evolutionary optimization based on reference points. *Applied Soft Computing*, 50:344–355, 2017.
- [53] Marco Locatelli and Fabio Schoen. *Global optimization: theory, algorithms, and applications*. SIAM, <https://doi.org/10.1137/1.9781611972672>, 2013.

- [54] AV Lyamin and SW Sloan. Lower bound limit analysis using non-linear programming. *International Journal for Numerical Methods in Engineering*, 55(5):573–611, 2002.
- [55] Kaiwen Ma, Luis Miguel Rios, Atharv Bhosekar, Nikolaos V Sahinidis, and Sreekanth Rajagopalan. Branch-and-model: a derivative-free global optimization algorithm. *Computational Optimization and Applications*, 85(2):337–367, 2023.
- [56] Cédric Malherbe and Nicolas Vayatis. Global optimization of lipschitz functions. In *International Conference on Machine Learning*, pages 2314–2323, <https://doi.org/10.48550/arXiv.1703.02628> , 2017. PMLR.
- [57] Nadia Martinez, Hadis Anahideh, Jay M Rosenberger, Diana Martinez, Victoria CP Chen, and Bo Ping Wang. Global optimization of non-convex piecewise linear regression splines. *Journal of Global Optimization*, 68:563–586, 2017.
- [58] Rupert G. Miller. *Simultaneous statistical inference*. Springer Science & Business Media, <https://doi.org/10.1007/978-1-4613-8122-8>, 2012.
- [59] Ruth Misener and Christodoulos A Floudas. Antigone: algorithms for continuous/integer global optimization of nonlinear equations. *Journal of Global Optimization*, 59(2-3):503–526, 2014.
- [60] Juliane Müller, Christine A Shoemaker, and Robert Piché. So-mi: A surrogate model algorithm for computationally expensive nonlinear mixed-integer black-box global optimization problems. *Computers & Operations Research*, 40(5):1383–1400, 2013.
- [61] János D Pintér. Global optimization: software, test problems, and applications. *Handbook of Global Optimization*, 2(2):515–569, 2002.
- [62] SA Piyavskii. An algorithm for finding the absolute extremum of a function. *USSR Computational Mathematics and Mathematical Physics*, 12(4):57–67, 1972.
- [63] David M Rosen, Charles DuHadway, and John J Leonard. A convex relaxation for approximate global optimization in simultaneous localization and mapping. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5822–5829, <https://doi.org/10.1109/icra.2015.7140014>, 2015. IEEE.
- [64] Yaroslav D Sergeyev and Dmitri E Kvasov. Global search based on efficient diagonal partitions and a set of lipschitz constants. *arXiv preprint arXiv:1103.2056*, 2011.
- [65] Yaroslav D Sergeyev, Antonio Candelieri, Dmitri E Kvasov, and Riccardo Perego. Safe global optimization of expensive noisy black-box functions in the  $\delta$ -lipschitz framework. *Soft Computing*, 24(23):17715–17735, 2020.

- [66] Alexander Shapiro. Statistical inference of multistage stochastic programming problems. *Mathematical Methods of Operations Research*, 58:57–68, 2003.
- [67] Nicola Soave, Hugo Tavares, Susanna Terracini, and Alessandro Zilio. Hölder bounds and regularity of emerging free boundaries for strongly competing schrödinger equations with nontrivial grouping. *Nonlinear Analysis*, 138:388–427, 2016.
- [68] Xueguan Song, Liye Lv, Wei Sun, and Jie Zhang. A radial basis function-based multi-fidelity surrogate model: exploring correlation between high-fidelity and low-fidelity models. *Structural and Multidisciplinary Optimization*, 60:965–981, 2019.
- [69] Sonja Surjanovic and Derek Bingham. Virtual library of simulation experiments, Test Functions and Datasets. <https://www.sfu.ca/ssurjano/optimization.html>, 2013. [Last accessed on 18-April-2023].
- [70] Mohit Tawarmalani and Nikolaos V Sahinidis. Global optimization of mixed-integer nonlinear programs: A theoretical and computational study. *Mathematical Programming*, 99(3):563–591, 2004.
- [71] Mohit Tawarmalani and Nikolaos V Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103(2): 225–249, 2005.
- [72] Joachim Van Der Herten, Tom Van Steenkiste, Ivo Couckuyt, and Tom Dhaene. Surrogate modelling with sequential design for expensive simulation applications. *Computer Simulation*, 173:173–192, 2017.
- [73] Abraham P Vinod, Arie Israel, and Ufuk Topcu. Constrained, global optimization of unknown functions with lipschitz continuous gradients. *SIAM Journal on Optimization*, 32:1239–1264, 2022.
- [74] Hao Wang, Bas van Stein, Michael Emmerich, and Thomas Back. A new acquisition function for bayesian optimization based on the moment-generating function. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 507–512. IEEE, 2017.
- [75] James Wilson, Frank Hutter, and Marc Deisenroth. Maximizing acquisition functions for bayesian optimization. *Advances in neural information processing systems*, 31, 2018.
- [76] Joschka Winz and Sebastian Engell. Optimization based sampling for gray-box modeling using a modified upper confidence bound acquisition function. In *Computer Aided Chemical Engineering*, volume 50, pages 953–958. Elsevier, 2021.

- [77] GR Wood and BP Zhang. Estimation of the lipschitz constant of a function. *Journal of Global Optimization*, 8:91–103, 1996.
- [78] Junru Wu. On a linearity between fractal dimension and order of fractional calculus in hölder space. *Applied Math and Computation*, 385:125433, 2020.
- [79] Tao Wu, Tie Luo, and Donald C Wunsch II. Lrs: Enhancing adversarial transferability through lipschitz regularized surrogate. In *Proceedings of the AAAI Conference on Artificial Intelligence*, number 6 in 38, pages 6135–6143, 2024.
- [80] Wenjie Xu, Yuning Jiang, Emilio T Maddalena, and Colin N Jones. Lower bounds on the noiseless worst-case complexity of efficient global optimization. *Journal of Optimization Theory and Applications*, 201(2):583–608, 2024.
- [81] Jie Yu, Jaume Amores, Nicu Sebe, and Qi Tian. A new study on distance metrics as similarity measurement. In *2006 IEEE international conference on multimedia and expo*, pages 533–536. IEEE, 2006.
- [82] Zelda B Zabinsky, Robert L Smith, and Birna P Kristinsdottir. Optimal estimation of univariate black-box lipschitz functions with upper and lower error bounds. *Computers & Operations Research*, 30(10):1539–1553, 2003.
- [83] Jianyuan Zhai and Fani Boukouvala. Data-driven spatial branch-and-bound algorithms for box-constrained simulation-based optimization. *Journal of Global Optimization*, 82(2):21–50, 2022.
- [84] Dawei Zhan and Huanlai Xing. Expected improvement for expensive optimization: a review. *Journal of Global Optimization*, 78(3):507–544, 2020.
- [85] Jialing Zhou, Yuezun Lv, Changyun Wen, and Guanghui Wen. Solving specified-time distributed optimization problem via sampled-data-based algorithm. *IEEE Transactions on Network Science and Engineering*, 9(4): 2747–2758, 2022.