

Accelerated Gamut Discovery via Massive Parallelization

NAVID ANSARI, Max Planck Institute for Informatics, Germany
HANS-PETER SEIDEL, Max Planck Institute for Informatics, Germany
VAHID BABAEI, Max Planck Institute for Informatics, Germany

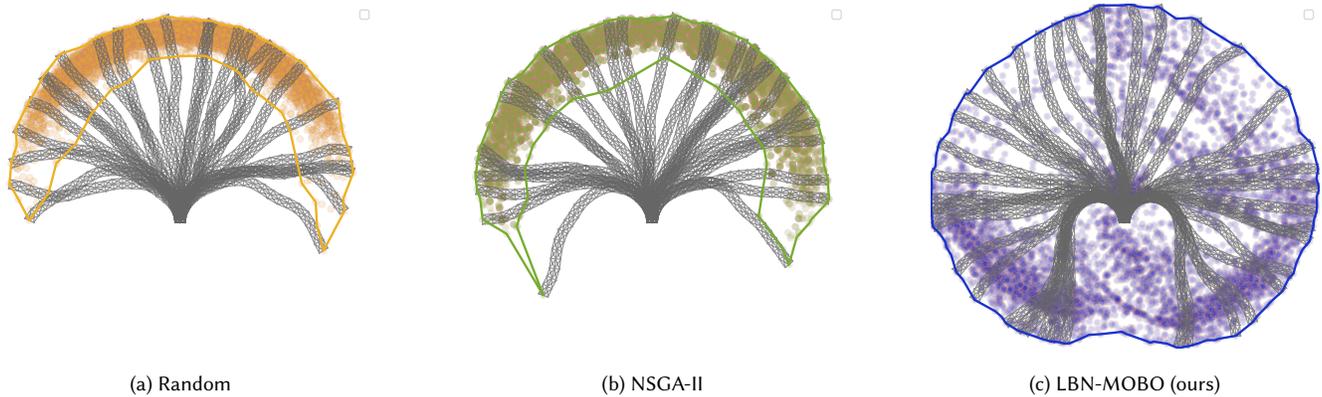


Fig. 1. We propose a highly accelerated approach for discovering the gamut of different design processes. Our method is capable of taking and proposing an extremely large batch of samples at each of its iterations while exploring the design space. In this teaser, we compare the performance of random sampling, NSGA-II (a stochastic multi-objective optimization), and our LBN-MOBO method in identifying the reachable space of a soft robot’s tip. All methods operate under the same computational budget of 4 iterations, each with a batch size of 1,000 samples. Our method significantly outperforms existing approaches in both coverage and efficiency.

This paper presents a scalable framework for efficiently discovering the performance gamut of different processes. Gamut boundaries comprise the set of highest-performing solutions within a design space. While sampling methods are often inefficient or prone to premature convergence, Bayesian optimization struggles with taking advantage of existing large-scale parallel computation or experimentation. To address these challenges, we utilize Bayesian neural networks as scalable surrogates for performance prediction and uncertainty estimation. We further introduce a novel acquisition function that combines the diversity-driven exploration of stochastic optimization with the information-efficient exploitation of Bayesian optimization. This enables generating large, high-quality batches of samples. Our approach leverages large batch sizes to reduce the number of iterations needed for optimization. We demonstrate its effectiveness on real-world engineering and robotic problems, achieving faster and more extensive discovery of the performance gamut. Code and data are available at https://gitlab.mpi-klb.mpg.de/nansari/lbn_mobo.

CCS Concepts: • **Computing methodologies** → **Machine learning; Artificial intelligence**.

Authors’ addresses: Navid Ansari, Max Planck Institute for Informatics, Saarbrücken, Germany, nansari@mpi-inf.mpg.de; Hans-Peter Seidel, Max Planck Institute for Informatics, Saarbrücken, Germany, hkseidel@mpi-sb.mpg.de; Vahid Babaei, Max Planck Institute for Informatics, Saarbrücken, Germany, vbabaei@mpi-inf.mpg.de.

Please use nonacm option or ACM Engage class to enable CC licenses 
This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.
SIGGRAPH Conference Papers ’25, August 10–14, 2025, Vancouver, BC, Canada
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1540-2/2025/08
<https://doi.org/10.1145/3721238.3730712>

Additional Key Words and Phrases: Computational design, fabrication, inverse design, gamut discovery, neural networks, surrogate models, Bayesian optimization

ACM Reference Format:

Navid Ansari, Hans-Peter Seidel, and Vahid Babaei. 2025. Accelerated Gamut Discovery via Massive Parallelization. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers ’25)*, August 10–14, 2025, Vancouver, BC, Canada. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3721238.3730712>

1 INTRODUCTION

In many engineering fields, such as computational design and robotics, we often encounter a key question: What is the performance *gamut* of a particular design process, robot, or fabrication hardware? This gamut determines the limits of achievable performance of the corresponding process. Having access to the performance gamut is extremely valuable. Most importantly, the gamut draws a boundary between feasible and infeasible. For example, the gamut determines whether a particular printer with its set of inks is capable of reproducing a certain appearance. It also has safety implications; for example it reveals whether a robot can potentially reach a particular location. Additionally, having access to the gamut allows us to create representative datasets for different data-driven applications.

A straightforward solution for gamut discovery is to sample the design space of the underlying *native forward process* (NFP) and evaluate the performance. This strategy works for NFPs with a low-dimensional design space but would fail when faced with complex, high-dimensional problems. Bayesian optimization (BO) is a significantly more efficient way of finding the gamut boundaries.

Treating the NFP as a black box, BO finds gamut boundaries through optimizing for multiple performances that define the gamut. BO uses a surrogate for the NFP and guides sample selection via an *acquisition function* (AF). Unfortunately, the BO literature almost exclusively focuses on using the least number of NFP evaluations, favoring small batches (typically only one sample) and many iterations. This is while advances in high-performance computing as well as experimentation and measurement hardware provide unprecedented parallel processing capabilities. In many cases, simulating (or measuring) the performance of thousands of samples in parallel is almost as feasible as evaluating a single sample. Therefore, the bottleneck of the process of optimization shifts from the number of NFP evaluations toward the number of iterations. To our surprise, the literature does not offer a BO method capable of working with large batches (and consequently small iterations).

In this paper, we propose a method of gamut discovery through a novel Bayesian optimization which is capable of working with extremely large batches. Particularly, we address two bottlenecks that hinder the scalability of traditional BO approaches: surrogate scalability and acquisition function scalability. First, standard surrogate models struggle to scale with large batch sizes. We address this by using Bayesian neural networks (BNNs), which offer scalability and built-in uncertainty estimation to enhance exploration. Second, existing acquisition functions for multi-objective batch optimization fail when handling large batches (e.g., >1000 samples). We propose a novel 2MD acquisition function that efficiently generates diverse, high-quality samples by combining the diversity-driven exploration of NSGA-II, a stochastic multi-objective optimization [Deb et al. 2002], with the information-efficient exploitation of BO. Our key contributions are:

- Leveraging modern parallel computing and experimentation to accelerate gamut discovery.
- Introducing a novel, highly scalable acquisition function that balances exploration and exploitation efficiently for extremely large batches of samples.
- A detailed analysis of neural surrogate models and identifying the most scalable options for large-batch optimization.
- Demonstrating the effectiveness of our method in discovering gamut boundaries across a range of real-world design and robotics problems.

2 RELATED WORK

Functional Fabrication and Gamut. A crucial goal of computational design and fabrication is to transform functional objectives or performance requirements into manufacturable designs [Bermano et al. 2017]. We call the complete range of achievable performances within the functional fabrication problem gamut [Smith 1978]. The concept of the gamut has been widely applied in fabrication problems, such as using the spectral reflectance gamut of 3D printers to create illumination-invariant reproductions of paintings [Shi et al. 2018], investigating advanced LED technologies for improved display color gamuts [Ren et al. 2024], and exploring microstructures to expand the gamut of physical properties, including thermal, electrical, and magnetic profiles [Chen et al. 2018]. Makatura [2020] introduce Pareto gamuts, which capture Pareto fronts over a range

of contexts in multi-objective optimization problems. They develop a global-local optimization algorithm to discover the Pareto gamut directly, facilitating the exploration of optimal designs across varying scenarios. In this work, we aim to accelerate gamut discovery in complex systems by leveraging the ever-increasing parallelization. This parallelization is occurring in both computer simulations, thanks to the increased computational power with more CPU and GPU cores, and real-world experimentation, thanks to, e.g., autonomous materials science labs [Szymanski et al. 2023]. Our proposed algorithm fully utilizes these parallelization capabilities, enabling a faster and more efficient gamut discovery procedure.

Pareto Front. Pareto front is closely related to the concept of gamut in multi-objective problems. While the gamut represents all achievable results within a system, the Pareto front defines the set of optimal solutions, where no objective can be improved without compromising another [Van Veldhuizen et al. 1998]. In essence, the Pareto front represents the boundaries of the gamut. Cibulski et al. [2020] developed an interactive Pareto front visualization tool for multi-objective engineering design, aiding engineers in leveraging their expertise during the functional fabrication process. Suresh [2013] solved large-scale 3D topology optimization, efficiently generating Pareto-optimal solutions in complex domains with millions of degrees of freedom. In this work, we discover the gamut by finding the gamut boundaries *in all directions* of the performance space.

Multi-Objective Optimization (MOO). Multi-objective optimization is key to Pareto front and gamut discovery. It involves solving for optimal trade-offs between performance metrics. Various methods have been developed to address these types of problems. The *Non-dominated Sorting Genetic Algorithm II (NSGA-II)* [Deb et al. 2002] is a widely adopted method for MOO, particularly in scenarios where the Native Forward Process (NFP) is treated as a black-box function. NSGA-II’s Pareto-front ranking and diversity-preserving mechanisms effectively balance convergence and exploration, making it applicable across a wide range of problems. However, NSGA-II requires numerous NFP evaluations, leading to high computational costs for problems with high-dimensional spaces or expensive objective functions [Konakovic Lukovic et al. 2020]. This limitation, along with its susceptibility to local optima, underscores the need for more efficient approaches to gamut discovery. Schulz et al. [2018] introduced a first-order approximation method for discovering and navigating the Pareto front, representing trade-offs as smooth patches. This approach enables efficient exploration of design trade-offs, particularly in CAD models.

Bayesian Optimization (BO). Bayesian Optimization [Jones et al. 1998] is a powerful method for optimizing expensive functions with minimal NFP evaluations. Multi-objective extensions, such as Expected Hypervolume Improvement (EHVI) [Emmerich et al. 2005] and batch methods like parallel Expected Hypervolume Improvement (qEHVI) and parallel Noisy Expected Hypervolume Improvement (qNEHVI) [Daulton et al. 2020, 2021], offer effective tools for exploring Pareto fronts. However, these approaches face scalability challenges [Springenberg et al. 2016], particularly when

handling batch sizes exceeding 1000 samples—a critical requirement for modern gamut discovery tasks. To address these limitations, Konakovic Lukovic et al. [2020] introduced a multi-objective Bayesian optimization algorithm designed for parallel evaluation. By a piecewise-continuous approximation of the Pareto set, this method balances hypervolume improvement and diversity in sample selection, greatly enhancing the efficiency of Pareto front discovery and accelerating multi-objective optimization.

To further improve scalability, neural networks have been proposed to replace traditional Gaussian Processes (GPs) in BO. Bayesian neural networks (BNNs) [Arbel et al. 2023] like Deep Kernel Learning (DKL) [Wilson et al. 2016], Infinite Width Bayesian neural networks (IBNNs) [Lee et al. 2017] and techniques like Stochastic Gradient Hamiltonian Monte Carlo (HMC) [Chen et al. 2014] provide scalable surrogation in high-dimensional design spaces. Approximate Bayesian models like Deep Ensembles [Lakshminarayanan et al. 2016], and Monte Carlo Dropout (MC Dropout) [Gal and Ghahramani 2016] further enhance the scalability of surrogates.

Despite improvements in surrogate modeling, designing appropriate acquisition functions, whose role is to propose new samples at each iteration of BO, remains a more critical bottleneck in scaling BO for gamut discovery. Methods such as *qEHVI*, *qParEGO* [Knowles 2006], and *qNEHVI* struggle with computational complexity, particularly for large batch sizes or high-dimensional problems (Section B.2). In this work, we propose a novel acquisition function that, when combined with a scalable surrogate model, enables multi-objective Bayesian optimizations to efficiently handle batch sizes exceeding 10,000 samples. More details and experiments on classical multi-objective batch capable Bayesian optimizations are provided in Sections A and B of the supplementary materials.

3 BACKGROUND

In computational design, the gamut \mathcal{G} represents the full range of achievable performances within a feasible design space. The *gamut boundary* $\partial\mathcal{G}$ defines the outer limits of this space, consisting of the highest-performing solutions that cannot be improved in one objective without sacrificing performance in another. This definition is closely related to Pareto front \mathcal{P} .

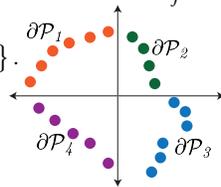
Formally, let $\mathcal{X} \in \mathbb{R}^N$ be the design space and $\mathcal{Y} \in \mathbb{R}^M$ be the performance space with M objectives. Let $f : \mathcal{X} \rightarrow \mathcal{Y}$ be the underlying NFP mapping from a design $x \in \mathcal{X}$ to its performance $y = f(x)$. A solution $y^1 = f(x^1)$ is said to dominate another solution $y^2 = f(x^2)$ in a minimization problem, $y^1 \prec y^2$, if:

$$\forall i \in \{1, \dots, M\}, \quad y_i^1 \leq y_i^2 \quad \text{and} \quad \exists j \in \{1, \dots, M\}, \quad y_j^1 < y_j^2.$$

This means for every component i , the value of y_i^1 is smaller than or equal to the corresponding value of y_i^2 . Additionally, there exists at least one component j where y_j^1 is strictly smaller than y_j^2 . The Pareto front \mathcal{P} is then defined as:

$$\mathcal{P} = \{y \in \mathcal{Y} \mid \nexists y' \in \mathcal{Y}, y' \prec y\}.$$

This means the Pareto front \mathcal{P} consists of all solutions y in the performance space \mathcal{Y} that are not strictly dominated by any other solution y' in



\mathcal{Y} . In practice, we compute the Pareto front for each of the 2^M orthants in an M -dimensional performance space and aggregate them to obtain the gamut boundary $\partial\mathcal{G}$:

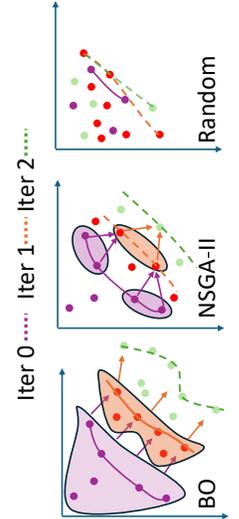
$$\partial\mathcal{G} = \sum_{i=1}^{2^M} \mathcal{P}_i.$$

Identifying the gamut boundary is critical for understanding the true limits of achievable performance. However, due to the high dimensionality of the design space and the significant computational cost of evaluating solutions using Native Forward Processes (NFPs), discovering the gamut boundary remains a challenging task [Skouras et al. 2012; Bächer et al. 2012].

A straightforward yet inefficient method for discovering the gamut boundary is random sampling, where we exploit all available resources to generate as many candidate solutions as possible in the hope of finding the high-performing ones (the top plot in the inset figure). However, due to the large size of the design space, random sampling is computationally expensive and slow to converge.

To address the inefficiencies of random sampling, Genetic Algorithm (GA), especially its well-known multi-objective extension NSGA-II, provides a guided, evolutionary approach for discovering gamut boundaries. Inspired by natural selection, NSGA-II [Deb et al. 2002] begins with a population of random solutions, evaluates their performance, and evolves the population through selection, recombination, and mutation over multiple iterations (middle plot in the inset figure). Two key mechanisms make NSGA-II effective:

- **Non-dominated Sorting:** Solutions are ranked into gamut boundaries based on dominance. The first front contains solutions that are not dominated by any others, while subsequent fronts contain solutions dominated only by those in earlier fronts.
- **Crowding Distance:** To maintain diversity along the gamut boundary, NSGA-II favors solutions that are well spread out, preventing clustering and ensuring uniform coverage of the gamut boundary.



While NSGA-II efficiently balances convergence and diversity, it can still suffer from premature convergence if the initial population lacks variety or diversity diminishes over generations.

To mitigate the risk of local minima and leverage gathered data more intelligently, Bayesian optimization (BO) has emerged as an effective alternative. BO replaces the computationally expensive Native Forward Process (NFP) with a surrogate model $\hat{f}(x)$ that approximates the objective function based on previously evaluated data points. This surrogate model enables more efficient exploration of the design space by reducing the need for direct NFP evaluations (bottom plot in the inset figure).

A key component of BO is the acquisition function (AF), which intelligently selects the next candidate solutions by balancing two goals:

- **Exploration:** Sampling regions where the surrogate model has high uncertainty to investigate under-explored areas of the design space.
- **Exploitation:** Prioritizing regions where the surrogate model predicts high performance to refine promising areas.

Traditional BO commonly uses Gaussian Processes (GPs) as surrogate models due to their ability to quantify uncertainty via the predictive mean $\hat{f}(x)$ and standard deviation $\sigma(x)$.

One widely used acquisition function in traditional BO is the Upper Confidence Bound (UCB) [Auer 2002]. UCB formalizes the trade-off between exploration and exploitation by using a weighted combination of the surrogate model’s predictive mean and uncertainty. The UCB acquisition function is defined as:

$$\text{UCB}(x) = \hat{f}(x) + \kappa\sigma(x), \quad (1)$$

where $\hat{f}(x)$ is the predicted mean, $\sigma(x)$ is the predicted standard deviation, and κ is a parameter that controls the trade-off between exploration and exploitation. The next candidate sample x_{next} is determined by maximizing the acquisition function:

$$x_{\text{next}} = \arg \max_{x \in \mathcal{X}} \text{UCB}(x).$$

The goal is to identify the most promising samples in each iteration, minimizing the number of expensive NFP queries needed to reach the optimal solution. The strength of UCB lies in its simplicity and flexibility, allowing the user to adjust κ to emphasize exploration or exploitation based on the problem requirements.

While BO is highly effective for optimizing functions with limited evaluations, it struggles in modern setups where large-scale parallelization allows tens of thousands of candidates to be evaluated simultaneously. Two major limitations arise:

- **Surrogate scalability:** Traditional surrogate models are not designed to scale efficiently with large batches of data.
- **Acquisition function scalability:** Existing acquisition functions fail to propose large, diverse batches of candidates effectively.

To address surrogate scalability, Bayesian neural networks offer a viable solution. Unlike standard neural networks, BNNs provide uncertainty estimation, which is critical for guiding the exploration. Among the various BNN approaches, Deep Ensembles [Lakshminarayanan et al. 2016] have demonstrated scalability and reliability, as they combine predictions from multiple independently trained neural networks to estimate both performance and uncertainty.

To overcome acquisition function limitations, we introduce a scalable acquisition function that integrates the diversity-driven exploration of NSGA-II with the information-efficient exploitation of BO. This hybrid approach leverages BNN surrogates to propose diverse, high-quality candidate batches, making full use of modern parallel computing infrastructure.

Combining NSGA-II’s diversity-preserving mechanisms with the intelligent exploration-exploitation trade-off of Bayesian optimization provides a powerful framework for discovering gamut boundaries. In Section 5, we demonstrate how our approach accelerates

convergence in real-world gamut discovery problems by exploiting highly parallelized environments.

4 METHOD: LARGE-BATCH NEURAL MULTI-OBJECTIVE BAYESIAN OPTIMIZATION

Our method, abbreviated as LBN-MOBO, works based on a similar principle to traditional BO but is devised to achieve scalability. LBN-MOBO begins with a random sampling of the design space $\mathcal{U}_S(\mathcal{X})$ of the given NFP (Φ). Subsequently, it fits an approximation of a Bayesian neural network surrogate f_{BNN} to the randomly sampled dataset \mathbf{X}^0 . The Bayesian neural network f_{BNN} is capable of fitting to large data batches. Additionally, f_{BNN} , and particularly its approximation through Deep Ensembles (DE) [Lakshminarayanan et al. 2016], enables computing predictive uncertainties ($\mathbb{F}_\sigma(\mathbf{x})$) in a fully parallelized manner (Section 4.1). Upon training f_{BNN} , we utilize our novel acquisition function (A_F) to compute the next batch of candidates. This function strategically balances *exploitation* of promising regions and *exploration* of under-represented areas in the design space (Section 4.2).

We evaluate the candidate batch using the NFP, append the results to our dataset, and utilize the updated dataset to retrain the BNN for the next generation.

Figure 2 illustrates the stages of the LBN-MOBO algorithm using an NFP with two objectives as an example. Note that some of the candidates may not lie on the Pareto front of the NFP (indicated by the red regions). This highlights the exploratory property of LBN-MOBO, as these candidates are still retained in the dataset. Their inclusion contributes to enhancing the information captured by f_{BNN} and reduces its uncertainty level ($\mathbb{F}_\sigma(\mathbf{x})$). Algorithm 1 provides a concise summary of all the steps of LBN-MOBO.

ALGORITHM 1: Large-batch, neural multi-objective Bayesian optimization (LBN-MOBO).

```

Input
S // Batch size
Q // Number of iterations of the main algorithm
X //  $\mathcal{X} \in \mathcal{R}^n$ , n dimensional design space
Φ // Native Forward Process, e.g., a simulation
Output  $P_S, P_F$  // Pareto set(designs) and Pareto front(performances) of NFP
begin
 $\mathbf{X}^0 \leftarrow \mathcal{U}_S(\mathcal{X})$  // Draw S random samples from the design space.
 $\mathbf{Y}^0 \leftarrow \Phi(\mathbf{X}^0)$  // Query Φ and form the data set.
dataset  $\leftarrow (\mathbf{X}^0, \mathbf{Y}^0)$ 
 $f_{BNN}^0 \xleftarrow{\text{train}} \text{dataset}$  // Train the BNN surrogate.
for i  $\leftarrow 1$  to Q do
   $P_S^i \leftarrow A_F(f_{BNN}^{i-1}, S)$ 
   $P_F^i \leftarrow \Phi(P_S^i)$  // Calculate the performance on the NFP.
  dataset  $\leftarrow (P_F^i, P_S^i)$  // Append new data to the old.
   $f_{BNN}^i \xleftarrow{\text{train}} \text{dataset}$  // Train the BNN surrogate.
end
end

```

4.1 Bayesian neural network surrogate

Given that Deep Ensembles (DE) [Lakshminarayanan et al. 2016] presents the most balanced trade-off between performance and scalability (Section B.2), it will serve as the primary surrogate function in our pipeline. Here, we delve into its implementation and make

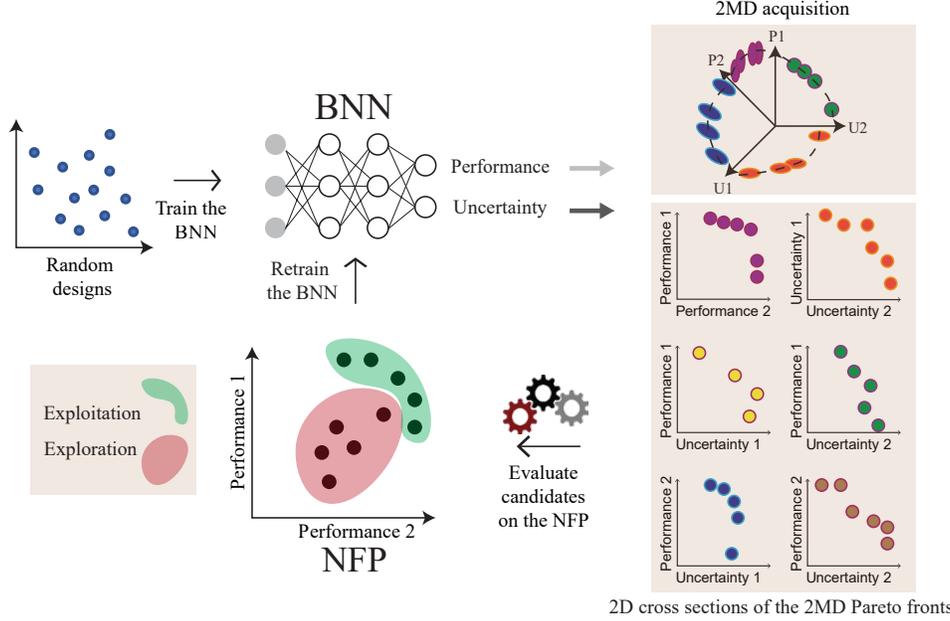


Fig. 2. LBN-MOBO starts with training a Bayesian neural network (f_{BNN}) on random designs. We then run our acquisition function (A_f) and compute a 2MD Pareto front to explore promising (green) and under-represented regions (red) of the NFP. We then append the acquired candidates to the data set and retrain f_{BNN} . By incorporating uncertainty information alongside the Pareto front of the best performances, we identify promising candidates in areas of high uncertainty, where there is potential for additional information.

slight modifications to enhance its performance further. In this work, we employ a modified version of DE as an approximation of a BNN [Snoek et al. 2015]. DE consists of an ensemble of K neural networks, \hat{f}_k , each capable of providing a prediction $\mu_k(\mathbf{x})$ and its associated *aleatoric* uncertainty $\sigma_k(\mathbf{x})$ in the form of a Gaussian distribution $\mathcal{N}(\mu_k(\mathbf{x}), \sigma_k(\mathbf{x}))$. Lakshminarayanan et al. [2016] proposed to ensemble these K sub-networks as following:

$$\mathbb{F}_\mu(\mathbf{x}) := \frac{1}{K} \sum_{k=1}^K \mu_k(\mathbf{x}), \quad (2a)$$

$$\mathbb{F}_{\sigma E}^2(\mathbf{x}) = \frac{1}{K} \sum_{k=1}^K (\mu_k^2(\mathbf{x}) - \mathbb{F}_\mu^2(\mathbf{x})), \quad (2b)$$

$$\mathbb{F}_{\sigma A}^2(\mathbf{x}) = \frac{1}{K} \sum_{k=1}^K \sigma_k^2(\mathbf{x}) \quad (2c)$$

$$\mathbb{F}_\sigma^2(\mathbf{x}) = \mathbb{F}_{\sigma E}^2(\mathbf{x}) + \mathbb{F}_{\sigma A}^2(\mathbf{x}), \quad (2d)$$

where $\mathbb{F}_\mu(\mathbf{x})$ is the ensemble prediction, and $\mathbb{F}_{\sigma A}(\mathbf{x})$ and $\mathbb{F}_{\sigma E}(\mathbf{x})$ are epistemic and aleatoric uncertainty, respectively. As evident from Equation 2d, DE has the unique advantage of separation between the epistemic and aleatoric uncertainty [Valdenegro-Toro and Mori 2022; Egele et al. 2022] allowing us to use these information selectively [Kendall and Gal 2017] in LBN-MOBO. Epistemic uncertainty measures the variation in predictions among the subnetworks of the ensemble compared to the average prediction (Equation 2b). Agreement among subnetworks indicates regions with abundant data, whereas divergence highlights underexplored areas. Epistemic

uncertainty serves as a critical signal for guiding exploration. In conventional BO with UCB acquisition (Equation 1) and in our novel scalable 2MD acquisition (Equation 5), we leverage epistemic uncertainty to ensure that no region is overlooked during the search for the optimum.

On the other hand, aleatoric uncertainty quantifies irreducible noise. In our application, where noise is minimal, we leverage the separability property of Deep Ensembles to exclude aleatoric uncertainty from our equations without impacting epistemic uncertainty calculations. Consequently, we train K neural networks using the traditional mean squared error (MSE) loss¹:

$$\mathcal{L}_k^{MSE} := (\mathbf{y}^* - \mu_k(\mathbf{x}))^2. \quad (3)$$

Next, we extract the epistemic uncertainty $\mathbb{F}_{\sigma E}(\mathbf{x})$ from the networks in the ensemble:

$$\mathbb{F}_\mu(\mathbf{x}) := \frac{1}{K} \sum_k \mu_k(\mathbf{x}), \quad (4a)$$

$$\mathbb{F}_{\sigma E}(\mathbf{x}) = \frac{1}{K} \sum_k (\mu_k^2(\mathbf{x}) - \mathbb{F}_\mu^2(\mathbf{x})). \quad (4b)$$

Apart from this modification, we find that providing a diverse set of activation functions across K members of the ensemble significantly helps with obtaining higher quality epistemic uncertainty. More details are provided in Section C.1 of the supplementary materials.

¹Sometimes this is referred to as the frequentist approach [Tagasovska and Lopez-Paz 2018; Hüllermeier and Waegeman 2021].

4.2 2MD acquisition function

An acquisition function should predict the worthiest candidates for the next iteration of the Bayesian optimization [Shahriari et al. 2015]. This translates to not only selecting designs with high performance on the surrogate model but also considering the uncertainty of the surrogate model. Candidates in uncertain regions of the surrogate model may contain optima and a powerful acquisition function should be able to explore these regions effectively. Several popular acquisition functions such as Expected Improvement [Jones et al. 1998] and Upper Confidence Bound [Brochu et al. 2010] operate on this principle.

Without the loss of generality, we assume a problem that seeks to *maximize* performance objectives. Our acquisition function employs the widely-used NSGA-II [Deb et al. 2002] for its multi-objective non-dominated sorting property. Applying NSGA-II on the surrogate rather than the NFP itself is the key for scalability of the acquisition. The main insight of our acquisition method is that instead of finding an M dimensional Pareto front corresponding to M objectives (each given by $\mathbb{F}_\mu^m(\mathbf{x})$, $m \in [1, M]$), it finds a $2M$ dimensional Pareto front where M dimensions correspond to performance objectives and the other M dimensions correspond to the uncertainty of those objectives (each given by $\mathbb{F}_{\sigma E}^m(\mathbf{x})$, $m \in [1, M]$). In other words, our acquisition function A_F *simultaneously* maximizes the predicted objectives (exploitation) and their associated epistemic uncertainties (exploration):

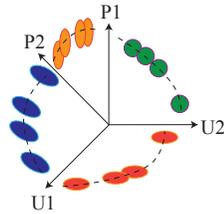
$$\mathbb{F}(\mathbf{x}) = \mathbb{F}_\mu^m(\mathbf{x}) \oplus \mathbb{F}_{\sigma E}^m(\mathbf{x}), \quad m \in [1, M], \quad (5a)$$

$$A_F(\mathbb{F}(\mathbf{x}), S) := \text{ParetoFront}(\mathbb{F}(\mathbf{x}), S), \quad (5b)$$

where \oplus represents the concatenation of M prediction vectors and M epistemic uncertainty estimation vectors and $\text{ParetoFront}(\mathbb{F}(\mathbf{x}), S)$ returns the set of S Pareto-optimal solutions that maximize $\mathbb{F}(\mathbf{x}) \in \mathcal{R}^{2M}$.

In practice, NSGA-II experiences a sample size bottleneck and struggles to scale effectively as the population expands. To overcome this limitation, we propose to compute in parallel independent acquisitions (using different NSGA-II seeds) with smaller batch sizes, and combine the results. Ultimately, similar to our surrogate model, our acquisition function (A_F) is fully parallelizable, and its performance remains mostly unchanged when the batch size increases. Therefore, the sole limiting factor for executing LBN-MOBO is our parallel computation or experimentation capability when querying the NFP.

The inset figure provides an intuitive explanation of our 2MD acquisition function by showing a schematic four-dimensional acquisition function. Clearly, we are interested in evaluating the orange samples (currently measured only using the surrogate) on the NFP as they are suggested by A_F to be dominant in at least one *performance* dimension (P1 or P2). On the other hand, the blue, red, and green samples are chosen partially or entirely due to their high uncertainty in at least one dimension (U1 or U2). These samples correspond to unexplored regions in the design space. They are beneficial in two ways: either they prove to be part of the Pareto



front once being evaluated on the NFP, or they contribute to filling the gap between the surrogate and the NFP [Ansari et al. 2022], leading to a more informative surrogate model. This enhances the quality of the surrogate model, making it as similar as possible to the NFP, thereby improving its predictive power for subsequent iterations. An ablation is provided in Section 5.4 to emphasize the importance of epistemic uncertainty for exploration.

5 EVALUATION AND DISCUSSION

In this section, we demonstrate the advantage of LBN-MOBO in making the most out of the already evaluated data and discovering a significantly larger gamut than competing methods in a variety of real-world scenarios. Since the computational overhead of LBN-MOBO (supplementary materials Section B.8) is negligible compared to the simulation time of the batch of samples using the NFP, we allocate an identical batch size for each iteration across random sampling, LBN-MOBO, and NSGA-II. The batch size in each experiment is the largest possible batch allowed by the parallel evaluation limit of our infrastructure. We keep only competing methods that can keep up with the huge batch sizes in each iteration (up to 20,000), namely random sampling and NSGA-II. The output dimension in all these real-world problems is two. In Section B.5, we extend the analysis to performance spaces with up to 10 dimensions, showing LBN-MOBO's effectiveness in handling larger dimensions.

5.1 Soft robot

Soft robotics focuses on designing flexible robots with applications such as minimally invasive surgeries [Majidi 2014] and prosthetics [Polygerinos et al. 2015]. In this section, we evaluate the outreach *gamut* of a snake-like soft robotic arm introduced by Sun et al. [2021], using our LBN-MOBO approach. The robotic arm consists of 103 vertices connected by flexible edges, with the robot's base fixed. Among the edges, 40 side edges are controllable (highlighted in Figure 4a with color coded contraction/expansion), defining a 40-dimensional design space. Stretches and contractions of these edges determine the final configuration of the robot.

This problem's NFP is a PDE-constrained optimization, where the design parameters serve as boundary conditions, and the solution provides the positions of all vertices [Xue et al. 2020]. Solving these optimizations is computationally expensive. Neural surrogate models offer a more efficient alternative. We train a Deep Ensembles model comprising 10 subnetworks, each with three hidden layers containing 128, 256, and 128 neurons and varying activations (see Section C.1 of supplementary materials). The input to the model is the expansions and contractions of the 40 controllable edges, while the output predicts the (x, y) coordinates of the robot's tip position.

Figure 3a compares the reachable gamut discovered using random sampling, NSGA-II and LBN-MOBO. In many scenarios, where domain knowledge is limited, random sampling is often used to generate training data for surrogates. However, as shown in the figure, random sampling (yellow dots) and NSGA-II over 4 iterations with a batch size of 2000 miss significant portions of the gamut. In contrast, LBN-MOBO efficiently explores the design space, uncovering a much larger gamut of reachability. Figure 4a illustrates how reaching the bottom region requires challenging soft robot

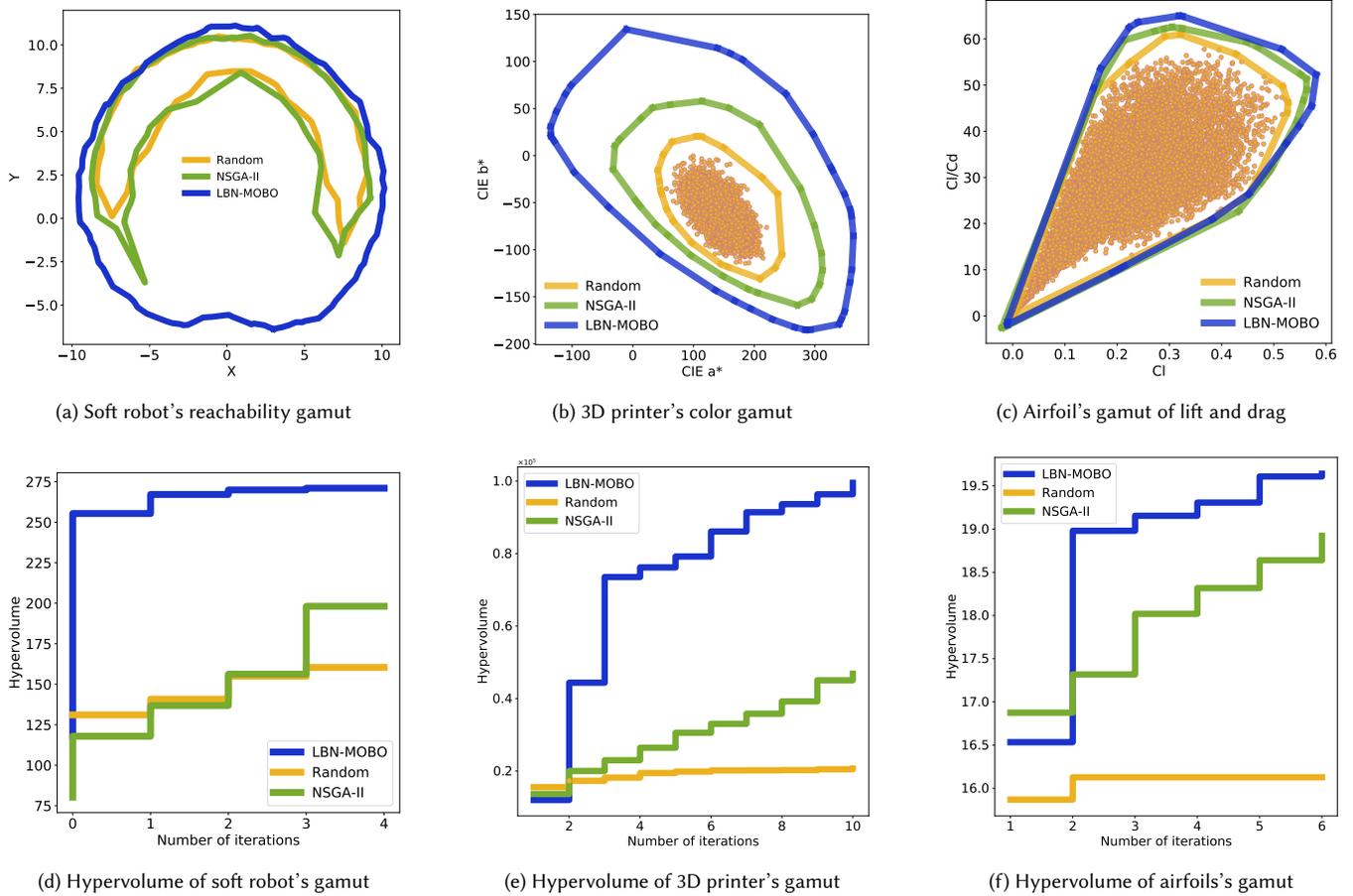


Fig. 3. This figure demonstrates LBN-MOBO’s performance on three challenging real-world problems, compared with random sampling and NSGA-II, the only competing methods capable of handling large batch sizes (up to 20,000 samples per iteration). For the soft robot, Figure 3a shows the total reachable area after 4 iterations, with the hypervolume progression depicted in Figure 3d. Figure 3b presents the final color gamut of a 44-ink printer after 10 iterations, alongside the iterative gamut expansion in Figure 3e. Finally, Figure 3c highlights airfoil shapes with optimal lift-to-drag ratios discovered by LBN-MOBO after 6 iterations, outperforming the competing methods.

conformations. These shapes demand significant asymmetry, with most edges on one side contracting while the other side expands. Such configurations are highly unlikely to be discovered through sampling. While LBN-MOBO find these configurations in less than 4 generation, NSGA-II would require significantly more generations to achieve them.

This has critical implications. A surrogate model trained on incomplete random data would struggle to accurately predict stretches and contractions for desired tip positions in underexplored regions [Xue et al. 2020]. By using LBN-MOBO, not only is the gamut expanded, but the dataset becomes more comprehensive, leading to a more reliable surrogate model. Figure 3d illustrates the area progression of the gamut discovered through all competing methods, highlighting LBN-MOBO’s superiority in exploring the reachability space effectively.

5.2 Printer’s color gamut

LBN-MOBO significantly speeds up experimental workflows by enabling the evaluation of large batches of samples in parallel, greatly reducing the burden of iterative laboratory work. A prime example is the exploration of a printer’s color gamut, where the goal is to maximize the hue diversity and saturation of achievable colors.

Here, we quantify the color gamut in the CIE a^*b^* color space [CIE 2004], where the range of colors is represented as the *area* within the contour of the CIE a^*b^* plot. In this space, CIE a^* corresponds to the red-green axis (negative values represent green and positive values represent red), while CIE b^* corresponds to the blue-yellow axis (negative values represent blue and positive values represent yellow). Expanding the gamut requires discovering more saturated colors, which directly increases the area enclosed by the contour².

²For this problem, we solve four LBN-MOBO for four quadrants in order to advance the Pareto front in four different segments.

For this experiment, the printer’s Native Forward Process (NFP) simulates how varying ink amounts produce specific colors. Following [Ansari et al. 2022], we modeled the NFP using an ensemble of 10 neural networks trained on 344,000 simulated color patches with varying combinations of 44 inks [Ansari et al. 2021]. This results in a 44-dimensional design space, with the a*b* color space serving as the 2-dimensional performance space.

LBN-MOBO was initialized with 10,000 random samples, and subsequent iterations processed batch sizes of 20,000 samples. The ability to handle such large batches makes LBN-MOBO particularly effective for this high-dimensional optimization task. In contrast, traditional optimization algorithms often struggle with the scale and dimensionality of this problem. As shown in Figure 3e, LBN-MOBO achieved a rapid and substantial increase in the hypervolume of the color gamut, outperforming random sampling with the same budget and NSGA-II. Furthermore, Figure 3b highlights the significantly larger estimated gamut achieved by LBN-MOBO compared to NSGA-II after 10 iterations.

5.3 Airfoil

The optimization of an airfoil’s lift coefficient (C_L) and lift-to-drag ratio (C_L/C_D) exemplifies a challenging multi-objective problem with high-dimensional design constraints and a computationally expensive NFP. The goal is to identify the Pareto front of C_L and C_L/C_D by exploring various airfoil shapes. These performance metrics are critical for designing efficient and effective aerodynamic structures such as airplane wings, with C_L representing the upward force counteracting weight and C_D quantifying the resistive drag force.

In this experiment, we leverage OpenFOAM, a high-fidelity computational fluid dynamics (CFD) solver [OpenFOAM Foundation 2021], to evaluate airfoil designs. The CFD simulations solve the Navier-Stokes equations to compute C_L and C_L/C_D [Thuerey et al. 2020]. To reduce the complexity of the 192-dimensional airfoil design space, we use a generative adversarial network (GAN) to encode the complex shapes into a five-dimensional latent space, enabling efficient optimization [Chen and Ahmed 2021]. As depicted by the inset figure GAN and OpenFOAM together form the NFP in this problem.

Random sampling, LBN-MOBO, and NSGA-II begin with 15,000 samples, and each iteration processes a batch size of 15,000 designs, all simulated using OpenFOAM. This large-batch setup allows LBN-MOBO to efficiently explore the design space while finding the largest possible gamut with the least number of iterations. Figure 3f highlights the hypervolume progression of the gamut over iterations. LBN-MOBO outperforms NSGA-II and random by discovering a larger gamut with similar number of iterations. The enhanced exploration and exploitation capabilities of LBN-MOBO, driven by its large-batch processing and high-quality epistemic uncertainty

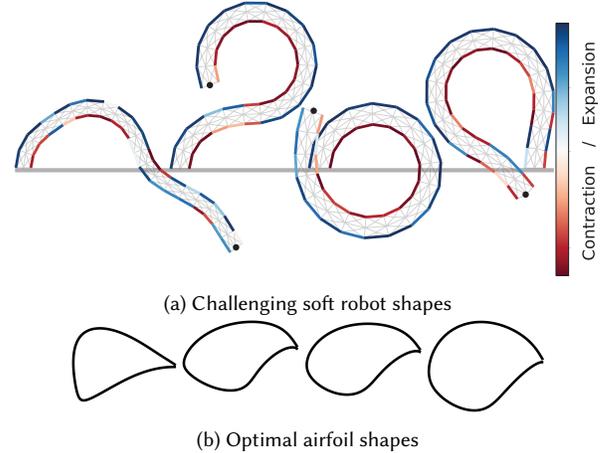


Fig. 4. Part (a) showcases some of the challenging conformations of soft robots discovered by LBN-MOBO. Achieving these shapes requires significant asymmetry, with most edges on one side contracting while the other side expands. Such configurations are highly improbable with random sampling, explaining why random sampling and NSGA-II fail to capture the bottom region in Figures 3a, 1a, and 1b. Similarly, part (b) highlights airfoil shapes whose performances lie on the gamut boundary, as calculated by LBN-MOBO.

estimation, enable rapid convergence to high-performing designs. Figure 4b showcases examples of airfoil shapes whose performances lie on the gamut boundary discovered by LBN-MOBO.

5.4 The impact of epistemic uncertainty on the performance of LBN-MOBO

One of the key factors enhancing the performance of LBN-MOBO is its use of uncertainty to effectively explore under-represented parts of the design space. We investigate the impact of uncertainty on the computation of the Pareto front for both airfoil design and color gamut exploration. Both experimental setups mirror the conditions described in Sections 5.2 and 5.3, except that they exclude uncertainty information. The candidate distribution from iteration 4 to 8 is illustrated in Figures 5a and 5b. For a clearer depiction of the samples’ spatial distribution, we have illustrated their convex hull. Note that in the absence of uncertainty, the candidates have a tendency to cluster within particular areas. This clustering leads to diminished diversity and, as a consequence, a reduction in the capacity for exploration (as represented by the yellow samples). Conversely, when uncertainty is incorporated into the process, we observe an increase in the diversity of the candidates and consequently, a broader Pareto front is discovered (represented by blue samples). Furthermore, uncertainty guides the candidates to progressively bridge the information gap in the surrogate models, making them increasingly similar to the NFP. This factor further enhances the quality of the Pareto front retrieved through the LBN-MOBO process.

We also observe that when uncertainty is excluded from the process, the budget for surrogate Pareto front optimization is concentrated solely on performance dimensions. This concentration

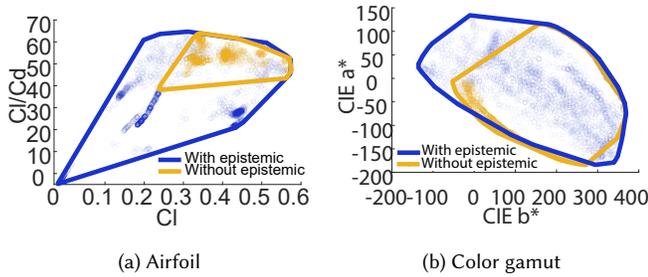


Fig. 5. Ablation studies on the effect of epistemic uncertainty in our $2MD$ acquisition function, using our real-world problems.

may occasionally lead to a slight local enhancement in optimization, as illustrated in the bottom-left part of the gamut in Figure 5b.

6 CONCLUSION

We presented a highly scalable framework for the rapid discovery of the performance gamut. LBN-MOBO proves to be a potent black-box optimizer for problems where an increase in the batch size does not significantly inflate simulation or experimentation costs, but iterations are expensive. Notably, LBN-MOBO not only retrieves a superior Pareto front but also enhances the surrogate model throughout the optimization process, making it closely resemble the NFP. This has important implication for active learning where one could start with a random dataset and incrementally train the network with missing data until it converges to the NFP. Looking forward, there are a few key aspects of this method that warrant further exploration. First, the potential of LBN-MOBO in managing design constraints needs to be assessed. Second, we can undertake an analysis of the method’s performance in the presence of highly noisy data, and possibly, enhance its robustness against noise. Finally, while our current acquisition function is tuning-free, it is intriguing to explore explicit methods that manipulate the balance between exploration and exploitation (by differently weighing the uncertainty) and observe how this balance affects the overall performance.

REFERENCES

- Navid Ansari, Hans-Peter Seidel, and Vahid Babaei. 2021. Mixed Integer Neural Inverse Design. *arXiv preprint arXiv:2109.12888* (2021).
- Navid Ansari, Hans-Peter Seidel, Nima Vahidi Ferdowsi, and Vahid Babaei. 2022. Autoinverse: Uncertainty Aware Inversion of Neural Networks. *Advances in Neural Information Processing Systems* 35 (2022), 8675–8686.
- Julyan Arbel, Konstantinos Pitas, Mariia Vladimirova, and Vincent Fortuin. 2023. A primer on Bayesian neural networks: review and debates. *arXiv preprint arXiv:2309.16314* (2023).
- Peter Auer. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* 3, Nov (2002), 397–422.
- Moritz Bächer, Bernd Bickel, Doug L James, and Hanspeter Pfister. 2012. Fabricating articulated characters from skinned meshes. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–9.
- Amit H Bermanno, Thomas Funkhouser, and Szymon Rusinkiewicz. 2017. State of the art in methods and representations for fabrication-aware design. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 509–535.
- Eric Brochu, Vlad M Cora, and Nando De Freitas. 2010. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599* (2010).
- Desai Chen, Mélina Skouras, Bo Zhu, and Wojciech Matusik. 2018. Computational discovery of extremal microstructure families. *Science advances* 4, 1 (2018), ea07005.
- Tianqi Chen, Emily Fox, and Carlos Guestrin. 2014. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*. PMLR, 1683–1691.
- Wei Chen and Faez Ahmed. 2021. Mo-padgan: Reparameterizing engineering designs for augmented multi-objective optimization. *Applied Soft Computing* 113 (2021), 107909.
- Lena Cibulski, Hubert Mitterhofer, Thorsten May, and Jörn Kohlhammer. 2020. Paved: Pareto front visualization for engineering design. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 405–416.
- CIE. 2004. *CIE 15:2004 Colorimetry* (3rd ed.). CIE Central Bureau.
- Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. 2020. Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. *Advances in Neural Information Processing Systems* 33 (2020), 9851–9864.
- Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. 2021. Parallel bayesian optimization of multiple noisy objectives with expected hypervolume improvement. *Advances in Neural Information Processing Systems* 34 (2021), 2187–2200.
- Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.
- Romain Egele, Romit Maulik, Krishnan Raghavan, Bethany Lusch, Isabelle Guyon, and Prasanna Balaprakash. 2022. Autodeuq: Automated deep ensemble with uncertainty quantification. In *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 1908–1914.
- Michael Emmerich, Nicola Beume, and Boris Naujoks. 2005. An EMO algorithm using the hypervolume measure as selection criterion. In *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 62–76.
- Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*. PMLR, 1050–1059.
- Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine learning* 110, 3 (2021), 457–506.
- Donald R Jones, Matthias Schonlau, and William J Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global optimization* 13, 4 (1998), 455.
- Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems* 30 (2017).
- Joshua Knowles. 2006. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation* 10, 1 (2006), 50–66.
- Mina Konakovik Lukovic, Yunsheng Tian, and Wojciech Matusik. 2020. Diversity-guided multi-objective bayesian optimization with batch evaluations. *Advances in Neural Information Processing Systems* 33 (2020), 17708–17720.
- Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2016. Simple and scalable predictive uncertainty estimation using deep ensembles. *arXiv preprint arXiv:1612.01474* (2016).
- Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. 2017. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165* (2017).
- Carmel Majidi. 2014. Soft robotics: a perspective—current trends and prospects for the future. *Soft robotics* 1, 1 (2014), 5–11.
- Liane Makatura. 2020. *Pareto gamuts: Exploring optimal designs across varying contexts*. Ph. D. Dissertation. Massachusetts Institute of Technology.
- OpenFOAM Foundation. 2021. OpenFOAM. <https://openfoam.org/>.

- Panagiotis Polygerinos, Zheng Wang, Kevin C Galloway, Robert J Wood, and Conor J Walsh. 2015. Soft robotic glove for combined assistance and at-home rehabilitation. *Robotics and Autonomous Systems* 73 (2015), 135–143.
- Tianyang Ren, Yuandong Ruan, Lintao Yan, Xinyi Shan, Daqi Shen, Cuili Tan, Xugao Cui, and Pengfei Tian. 2024. Performance improvement of red, green and blue InGaN micro-LEDs with distributed Bragg reflector. *Semiconductor Science and Technology* 39, 11 (2024), 115006.
- Adriana Schulz, Harrison Wang, Eitan Grinspun, Justin Solomon, and Wojciech Matusik. 2018. Interactive exploration of design trade-offs. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–14.
- Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. 2015. Taking the human out of the loop: A review of Bayesian optimization. *Proc. IEEE* 104, 1 (2015), 148–175.
- Liang Shi, Vahid Babaei, Changil Kim, Michael Foshey, Yuanming Hu, Pitchaya Sitthi-Amorn, Szymon Rusinkiewicz, and Wojciech Matusik. 2018. Deep multispectral painting reproduction via multi-layer, custom-ink printing. *ACM Transactions on Graphics* (2018).
- Mélina Skouras, Bernhard Thomaszewski, Bernd Bickel, and Markus Gross. 2012. Computational design of rubber balloons. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 835–844.
- Alvy Ray Smith. 1978. Color gamut transform pairs. *ACM Siggraph Computer Graphics* 12, 3 (1978), 12–19.
- Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. 2015. Scalable bayesian optimization using deep neural networks. In *International conference on machine learning*. PMLR, 2171–2180.
- Jost Tobias Springenberg, Aaron Klein, Stefan Falkner, and Frank Hutter. 2016. Bayesian optimization with robust Bayesian neural networks. *Advances in neural information processing systems* 29 (2016).
- Xingyuan Sun, Tianju Xue, Szymon Rusinkiewicz, and Ryan P Adams. 2021. Amortized Synthesis of Constrained Configurations Using a Differentiable Surrogate. In *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (Eds.). <https://openreview.net/forum?id=wdIDt--oLmV>
- Krishnan Suresh. 2013. Efficient generation of large-scale pareto-optimal topologies. *Structural and Multidisciplinary Optimization* 47 (2013), 49–61.
- Nathan J Szymanski, Bernardus Rendy, Yuxing Fei, Rishi E Kumar, Tanjin He, David Milsted, Matthew J McDermott, Max Gallant, Ekin Dogus Cubuk, Amil Merchant, et al. 2023. An autonomous laboratory for the accelerated synthesis of novel materials. *Nature* 624, 7990 (2023), 86–91.
- Natasa Tagasovska and David Lopez-Paz. 2018. Frequentist uncertainty estimates for deep learning. *arXiv preprint arXiv:1811.00908* (2018).
- Nils Thuerey, Konstantin Weissenow, Lukas Prantl, and Xiangyu Hu. 2020. Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows. *AIAA journal* 58, 1 (2020), 25–36.
- Matias Valdenegro-Toro and Daniel Saromo Mori. 2022. A deeper look into aleatoric and epistemic uncertainty disentanglement. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 1508–1516.
- David A Van Veldhuizen, Gary B Lamont, et al. 1998. Evolutionary computation and convergence to a pareto front. In *Late breaking papers at the genetic programming 1998 conference*. Citeseer, 221–228.
- Andrew Gordon Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. 2016. Deep kernel learning. In *Artificial intelligence and statistics*. PMLR, 370–378.
- Tianju Xue, Alex Beatson, Sigrid Adriaenssens, and Ryan Adams. 2020. Amortized finite element analysis for fast pde-constrained optimization. In *International Conference on Machine Learning*. PMLR, 10638–10647.