

# Bayesian Optimization For Multi-Objective Mixed-Variable Problems

Haris Moazam Sheikh<sup>1</sup> Philip S. Marcus<sup>1</sup>

## Abstract

Optimizing multiple, non-preferential objectives for mixed-variable, expensive black-box problems is important in many areas of engineering and science. The expensive, noisy black-box nature of these problems makes them ideal candidates for Bayesian optimization (BO). Mixed-variable and multi-objective problems, however, are a challenge due to the BO's underlying smooth Gaussian process surrogate model. Current multi-objective BO algorithms cannot deal with mixed-variable problems. We present MixMOBO, the first mixed variable multi-objective Bayesian optimization framework for such problems. Using a genetic algorithm to sample the surrogate surface, optimal Pareto-fronts for multi-objective, mixed-variable design spaces can be found efficiently while ensuring diverse solutions. The method is sufficiently flexible to incorporate many different kernels and acquisition functions, including those that were developed for mixed-variable or multi-objective problems by other authors. We also present HedgeMO, a modified Hedge strategy that uses a portfolio of acquisition functions in multi-objective problems. We present a new acquisition function SMC. We show that MixMOBO performs well against other mixed-variable algorithms on synthetic problems. We apply MixMOBO to the real-world design of an architected material and show that our optimal design, which was experimentally fabricated and validated, has a normalized strain energy density  $10^4$  times greater than existing structures.

## 1. Introduction

Optimization is an inherent part of design for complex physical systems. Often optimization problems are posed as

noisy black-box problems subject to constraints, where each function call requires an extremely expensive computation or a physical experiment. A large range of these problems requires optimizing a mixed variables design space (combinatorial, discrete, and continuous) for multiple objectives. Architected material design (Frazier & Wang, 2015; Chen et al., 2018a; 2019; Shaw et al., 2019; Song et al., 2019a; Vangelatos et al., 2021), hyper-parameter tuning for machine learning algorithms (Snoek et al., 2012; Chen et al., 2018b; Oh et al., 2018), drug design (Pyzer-Knapp, 2018; Korovina et al., 2020), controller sensor placement (Krause et al., 2008) pose such problems and Bayesian optimization is a natural candidate for their optimization.

Much research has gone into Bayesian optimization for continuous design spaces using Gaussian processes (GP) as a surrogate model and efficiently optimizing over this design space with minimum number of expensive function calls (Mockus, 1994; Rasmussen & Williams, 2006; Brochu et al., 2010). Despite the success of continuous Bayesian optimization strategies, multi-objective, mixed-variable problems remain an area of open research. The inherent continuous nature of GP makes dealing with mixed-variable problems challenging. Finding a Pareto-front for multi-objective problems, and parallelizing function calls for batch updates also remain as challenges in the sequential setting of the BO algorithm. ‘Hedge’ strategies for acquisition functions have proven to be very efficient for BO for continuous design spaces, however, Hedge strategies have not been formulated for multi-objective and mixed-variable problems.

In this paper, we present a Mixed-variable, Multi-Objective Bayesian Optimization (MixMOBO) algorithm that is a generalized framework that can deal with these types of problems in the small data setting and that can optimize a noisy black-box function with a small number of function calls. Using a GA on a mixed variable surrogate model in a multi-objective setting allows us to work with most of the published modified kernels that were developed for mixed-variable problems. We present a modified Hedge strategy for acquisition functions, Hedge Multi-Objective (HedgeMO). The strategy has batch updates that work in a multi-objective setting for which the regret bounds presented by Brochu et al. (2011) hold. We also propose a new acquisition function (AF), Stochastic Monte-Carlo (SMC), which performs well for categorical variables (Vangelatos

<sup>1</sup>Computational Fluids Dynamics Lab, Department of Mechanical Engineering, University of California, Berkeley, CA, USA. Correspondence to: Haris Moazam Sheikh <haris-sheikh@berkeley.edu>.

et al., 2021). In summary, the main contributions of our work are as follows:

- We present Mixed-variable, Multi-Objective Bayesian Optimization (MixMOBO), the first algorithm that can deal with mixed-variable, multi-objective problems. The framework using GA is flexible, so it can use the modified kernels or surrogate surfaces developed to deal with mixed-variable problems in previous studies. This extends their capabilities to handle mixed-variable and multi-objective problems as well since our constrained GA optimization method is agnostic to the underlying GP kernel over mixed-variables.
- GA is used to optimize surrogate models, which allows the optimization of multi-objective problems. ‘Q-batch’ samples can be extracted in parallel from within the GA generation without sacrificing diversification.
- We present a Hedge Multi-Objective (HedgeMO) strategy for mixed variables and multiple objectives for which regret bounds hold. We also present a new acquisition function, Stochastic Monte-Carlo (SMC), which performs well for combinatorial spaces and use it as part of our HedgeMO portfolio.
- We benchmark our algorithm against other mixed-variable algorithms and prove that MixMOBO performs well on test functions. We applied MixMOBO to a practical engineering problem: the design of a new architected meta-material that was optimized to have the maximum possible strain-energy density within the constraints of a design space. The fabrication and testing of this new material showed that it has a normalized strain energy density that is  $10^4$  times greater than existing unblemished microlattice structures in literature.

## 2. Related Work

### 2.1. Mixed-Variable BO Algorithms:

We provide a brief description of the current approaches in recent studies for dealing with mixed variables.

**One Hot Encoding Approach:** Most BO schemes use Gaussian processes as surrogate models. When dealing with categorical variables, a common method is ‘one-hot encoding’ (Golovin et al., 2017). Popular BO packages, such as GPyOpt and Spearmint (Snoek et al., 2012), use this strategy. However, this can result in inefficiency when searching the parameter space because the surrogate model is continuous. For categorical variables, this approach also leads to a quick explosion in dimensional space (Ru et al., 2020).

**Multi-Armed Bandit (MAB) Approach:** Some studies use the MAB approach when dealing with categorical variables where a surrogate surface for continuous variables is defined for each bandit arm. These strategies can be expensive in terms of the number of samples required (Gopakumar et al., 2018; Nguyen et al., 2019), and they do not share information across categories. An interesting approach, where coupling is introduced between continuous and categorical variables, is presented in the CoCaBO algorithm (Ru et al., 2020), and it is one of the baselines that we test MixMOBO against.

**Latent Space Approach:** A latent variable approach has also been proposed to model categorical variables (Qian et al., 2008; Zhou et al., 2011; Zhang et al., 2020; Deshwal & Doppa, 2021). This approach embeds each categorical variable in a  $\mathcal{Z}$  latent variable space. However, the embedding is dependent on the kernel chosen, and for small-data settings can be inefficient.

**Modified Kernel Approach:** There is a rich collection of studies in which the underlying kernel is modified to work with ordinal or categorical variables. For example, Ru et al. (2020) considers the sum + product kernel; Deshwal et al. (2021) propose hybrid diffusion kernels, HyBO; and Oh et al. (2021) propose frequency modulated kernels. The BOCS algorithm (Baptista & Poloczek, 2018) for categorical variables uses a scalable modified acquisition function. Pelamatti et al. (2018); Oh et al. (2019); Nguyen et al. (2019); Garrido-Merchán & Hernández-Lobato (2020) all use modified kernels to adapt the underlying surrogate surface. Our approach is unique that any modified kernel can be incorporated into our framework for the solution of multi-objective mixed-variable problems. Currently we use the modified RBF kernel for modelling the surrogate surface, with our future research focused on different kernels in our framework.

**Other Surrogate Models:** Other surrogate models can be used in place of the GP to model mixed-variable problems such as random forests, an approach used by SMAC3 (Lindauer et al., 2021) or tree based estimators, used in the Tree-Parzen Estimator (TPE) (Bergstra et al., 2013). Daxberger et al. (2020) considers a linear model with cross-product features. BORE (Tiao et al., 2021) leverages the connection to density ratio estimation.

### 2.2. Multi-Objective BO Algorithms:

Multi-objective Bayesian optimization (MOBO) has been the subject of some recent studies. BoTorch (Balandat et al., 2020), the popular BO framework, uses the EHVI and ParEGO based on the works of Fonseca et al. (2006) and Daulton et al. (2020; 2021). Hyper-volume improvement is the main mechanism used to ensure diversity in generations. ‘Q-batch’ parallel settings of the above two acquisition func-

tions use hyper-volume improvement and the previously selected point in the same batch to choose the next set of points. For most single-objective BO algorithms with parallel batch selection, the next batch of test points is selected by adding the ‘pretend’ cost-function evaluation to the previously selected test point within that batch. However, this commonly used method often leads to overly confident test point selection, and the surrogate surface then needs to be optimized, and often refitted  $Q$  times. Using GA, we can select a ‘ $Q$ -batch’ of points with a single optimization of the surrogate surface from the GA generation.

Suzuki et al. (2020) provide an interesting Pareto-frontier entropy method as an acquisition function, and Shu et al. (2020) use Pareto-frontier heuristics to formulate new acquisition functions. Their approaches were not extended to mixed-variable problems because hyper-volume is difficult to define for combinatorial spaces. Hedge algorithms have proven to be efficient in dealing with a diverse set of problems. They use a portfolio of acquisition functions (Brochu et al., 2011). However, these algorithms have not been considered for MOBO, and, to the authors’ knowledge, there is no existing BO implementation that solves mixed-variable multi-objective problems.

Genetic algorithms (GA), such as NSGA-II (Deb et al., 2002), are well known for dealing with mixed-variable spaces and finding an optimal Pareto-frontier. However, these algorithms require a large number of black-box function calls and are not well-suited to expensive small-data problems. Our approach is to use a GA to optimize the surrogate model itself and find a Pareto-optimal. Diversification is ensured by the distance metrics used while optimizing the surrogate model. This method allows cheap ‘ $Q$ -batch’ samples from within the GA generation, and also allows the use of the commonly used acquisition functions such as Expected Improvement (EI), Probability of Improvement (PI) and Upper Confidence Bound (UCB) (Brochu et al., 2011), which work well for single objective problems. We note here that hyper-volume improvement can easily be incorporated instead of a distance metric within the GA setting in future work. We also present a new acquisition function, ‘Stochastic Monte-Carlo’ (SMC), which preforms well for categorical variable problems (Vangelatos et al., 2021).

Hedge strategies for Bayesian optimization, where a portfolio of acquisition functions is used instead of a single acquisition function, are efficient for single objective algorithms. We present here Hedge Multi-Objective (HedgeMO) algorithm, which uses a portfolio of acquisition functions for multi-objective problems. Hedge algorithms for single-objective problems have regret bounds proven by Brochu et al. (2011), and the same bounds hold for HedgeMO.

### 3. Problem Statement

We pose the multi-objective and mixed-variable problem as:

$$\vec{w}_{opt} = \operatorname{argmax}_{\vec{w} \in \mathcal{W}} (\vec{f}(\vec{w})) \quad (1)$$

for maximizing the objective. Here  $\vec{f}(\vec{w}) = [f_1(\vec{w}), f_2(\vec{w}), \dots, f_K(\vec{w})]$  are the  $K$  non-preferential objectives to be maximized, and  $\vec{w}$  is a mixed-variable vector, defined as  $\{\vec{w} \in \mathcal{W}\} = \{\vec{x} \in \mathcal{X}, \vec{y} \in \mathcal{Y}, \vec{z} \in \mathcal{Z}\}$ .  $\vec{x}$  is an  $m$ -dimensional vector defined over a bounded set  $\mathcal{X} \subset \mathbb{R}^m$  representing  $m$  continuous variables. Ordinal and categorical variables are defined as  $\vec{y} = [y_1, \dots, y_n]$  and  $\vec{z} = [z_1, \dots, z_o]$ , respectively. Each variable  $y_j \in \{O_1, \dots, O_j\}$  takes one of  $O_j$  ordinal ‘levels’ (discrete numbers on the real-number line) and each categorical variable takes a value  $z_j \in \{C_1, \dots, C_j\}$  from  $C_j$  unordered categories (that cannot, by definition, be ordered on the real-number line).  $\mathcal{Y}$  and  $\mathcal{Z}$  are the ordinal and combinatorial spaces respectively.

Generally,  $\{\vec{w}_{opt}\}$  is a set of Pareto-optimal solution vectors i.e., vectors that are not Pareto-dominated by any other vector. A vector  $\vec{w}$  is Pareto-dominated by  $\vec{w}'$ , iff  $f_k(\vec{w}) \leq f_k(\vec{w}') \forall k = 1, \dots, K$ .

### 4. Methodology

#### Preliminaries

Single-objective Bayesian optimization is a sequential optimization technique, aimed at finding the global optimum of a single objective noisy black-box function  $f$  with minimum number of evaluations of  $f$ . For every  $i^{th}$  iteration, a surrogate model,  $g$ , is fit over the existing data set  $\mathcal{D} = \{(w_1, f(w_1)), \dots, (w_i, f(w_i))\}$ . An acquisition function then determines the next point  $\vec{w}_{i+1}$  for evaluation with  $f$ , balancing exploration and exploitation. Data is appended for the next iteration,  $\mathcal{D} = \mathcal{D} \cup (w_{i+1}, f(w_{i+1}))$ , and the process is repeated until the evaluation budget for  $f$  or the global optimum is reached.

Gaussian processes are often used as surrogate models for BO (Rasmussen & Williams, 2006; Murphy, 2012). A GP is defined as a stochastic process such that a linear combination of a finite set of the random variables is a multivariate Gaussian distribution. A GP is uniquely specified by its mean  $\mu(\vec{w})$  and covariance function  $\ker(\vec{w}, \vec{w}')$ . The GP is a distribution over functions, and  $g(\vec{w})$  is a function sampled from this GP:

$$\vec{g}(\vec{w}) \sim GP(\mu(\vec{w}), \ker(\vec{w}, \vec{w}')). \quad (2)$$

Here,  $\ker(\vec{w}, \vec{w}')$  is the covariance between input variables  $\vec{w}$  and  $\vec{w}'$ . Once a GP has been defined, at any  $\vec{w}$  the GP returns the mean  $\mu(\vec{w})$  and variance  $\sigma(\vec{w})$ . The acquisition function  $\mathcal{A}(\mu(\vec{w}), \sigma(\vec{w}))$ , balances exploration and exploitation, and

is optimized to find the next optimal point  $\vec{w}_{i+1}$ . The success of BO comes from the fact that evaluating  $\vec{g}$  is much cheaper than evaluating  $\vec{f}$ .

#### 4.1. MixMOBO Proposed Approach

**Algorithm 1** Mixed-variable Multi-Objective Bayesian Optimization (MixMOBO) Algorithm

- 1: **Input:** Black-box function  $\vec{f}(\vec{w}) : \vec{w} \in \mathcal{W}$ , initial data set size  $N$ , batch points per epoch  $Q$ , total epochs  $N$ , mutation rate  $\beta \in [0, 1]$
- 2: **Initialize:** Sample  $\vec{f}$  for  $\mathcal{D} = \{(\vec{w}_j, \vec{f}(\vec{w}_j))\}_{j=1:N}$
- 3: **for**  $n = 1$  **to**  $N$  **do**
- 4: Fit a noisy Gaussian process surrogate function  $\vec{g}(\vec{w}) \sim GP(\vec{\mu}(\vec{w}), \text{ker}(\vec{w}, \vec{w}'))$
- 5: For  $L$  total acquisition functions, from each  $\mathcal{A}^l$  acquisition function, propose  $Q$ -batch test-points,  $\{(\vec{u}_n^l)\}_{1:Q} = \left\{ \argmax_{\vec{u} \in \mathcal{W}} \mathcal{A}^l(\vec{g}) \right\}_{1:Q}$  within the constrained search space  $\mathcal{W}$  using multi-objective GA
- 6: Mutate point  $\{(\vec{u}_n^l)\}_q$  within the search space  $\mathcal{W}$  with probability rate  $\beta$  if  $L_2$ -norm of its difference with any other member in set  $\{(\vec{u}_n^l)\}_{1:Q}$  is below tolerance
- 7: Select batch of  $Q$  points using HedgeMO (Algorithm 2),  $\{\vec{w}_n\}_{1:Q} = \text{HedgeMO}(\vec{g}, \{(\vec{u}_n^l)\}_{1:Q}, \mathcal{D})$
- 8: Evaluate the selected points from the black-box function,  $\{\vec{f}(\vec{w}_n)\}_{1:Q}$
- 9: Update data set  $\mathcal{D} = \mathcal{D} \cup \{(\vec{w}_n, \vec{f}(\vec{w}_n))\}_{1:Q}$
- 10: **end for**
- 11: **return** Pareto-optimal solution set  $\{(\vec{w}_{opt}, \vec{f}(\vec{w}_{opt}))\}$

Our Mixed-variable Multi-Objective Bayesian Optimization (MixMOBO) algorithm extends the single-objective, continuous variable BO approach presented in the preceding section, to more generalized optimization problems and is detailed in Algorithm 1.

A single noisy GP surrogate surface is fit for multiple objectives,  $\vec{g}(\vec{w}) \sim GP(\vec{\mu}(\vec{w}), \text{ker}(\vec{w}, \vec{w}'))$ . This is equivalent to fitting  $K$  GP surfaces with the same kernel for all of the surfaces, where  $K$  is the total number of objectives. Only one set of hyper-parameters needs to be fit over this single surface, rather than fitting  $K$  sets of hyper-parameters for  $K$  different surfaces; thus, when  $K$  is large, the overall computational cost for the algorithm is reduced. Note that we could fit  $K$  different GP surfaces with different hyper-parameters to the data to add further flexibility to the fitted surfaces, and this idea will be investigated in future work. We use LOOCV (Murphy, 2012) for estimating hyper-parameters since we are dealing with small-data problems.

Gaussian processes are defined for continuous variables.

For mixed variables, we need to adapt the kernel so that a GP can be fit over these variables. Cited works in § 2 dealt with modified kernels that were designed to model mixed variables. Those kernels can be used in the MixMOBO algorithm. For the current study, we use a simple modified squared exponential kernel:

$$\text{ker}(\vec{w}, \vec{w}') \equiv e_f^2 \exp \left[ -\frac{1}{2} |\vec{w}, \vec{w}'|_C^T \underline{M} |\vec{w}, \vec{w}'|_C \right], \quad (3)$$

where  $\underline{M}$  is the covariance hyper-parameter matrix, and  $(\underline{M})_{pq} = \delta_{pq} h_p^{-2}$  with  $p$  total number of variables. The distance metric,  $|\vec{w}, \vec{w}'|_C$ , is an concatenated vector, with the distance between categorical variables defined to be the Hamming distance, and the distance between continuous variables and the distance between ordinal variables defined to be their Euclidean distances. We emphasize that *any* modified kernel discussed in the citations of § 2 can be used within our framework.

Once the GP is fit over multi-objective data, we use a multi-objective genetic algorithm (GA) to optimize the acquisition functions,  $\mathcal{A}^l$ . Acquisition functions explore the surrogate model to maximize reward by balancing exploration and exploitation. Using a standard acquisition function is problematic when dealing with multi-objective mixed-variable problems due to non-smooth surrogate surface and conflicting objectives. We propose using a constrained, multi-objective GA to optimize the acquisition functions, which, although expensive to use on an actual black-box function, is an ideal candidate for optimizing the acquisition function working on the surrogate surface:

- For multi-objective problems, multi-objective GA algorithms, such as (Deb et al., 2002), are ideal candidates for obtaining a Pareto-front of optimal solutions. Within a GA generation, the members of a non-dominated Pareto-front are ranked by a ‘distance crowding function’. This can be computed in decision-variable space, in function space or a hybrid of the two. The ranking takes place when choosing the test points from an acquisition function for a multi-objective problem because the choice must take into account the diversity of the solution and propagate the Pareto-front.
- Because the members of the population are ranked by the GA, we can easily extract a ‘Q-batch’ of points from each of the acquisition functions without needing to add any ‘pretend’ cost function evaluations or optimizing the acquisition functions again.
- Genetic algorithms (GA) can be constrained to work in mixed variable spaces. These variables can be dealt with by using probabilistic mutation rates. The genes



are allowed to mutate within their prescribed categories, thereby constraining the proposed test points to the  $\mathcal{W}$  space.

We use constrained, multi-objective GAs to optimize the acquisition functions  $\mathcal{A}^l$  for mixed-variable, multi-objective problems. Common acquisition functions, such as EI, PI, and UCB, can be used within this framework and can be used to nominate a ‘Q-batch’ of points. If a candidate in a ‘Q-batch’ is within the tolerance limit of another candidate in the same batch or a previous data point (for convex functions), we mutate the proposed point within  $\mathcal{W}$  to avoid sampling the same data point again.

Test points are selected from  $\mathcal{W}$  to evaluate their  $\vec{f}$  using HedgeMO algorithm which is detailed in the next section. HedgeMO selects a ‘Q-batch’ of test-points from the candidates proposed by each of the acquisition functions. These points are then, along with their function evaluations  $\vec{f}$ s, are appended to the data set.

## 4.2. HedgeMO Algorithm

Hedge strategies use a portfolio of acquisition functions, rather than a single acquisition function (Brochu et al., 2011). HedgeMO is part of our MixMOBO algorithm that not only extends the Hedge strategy to multi-objective problems, but also allows ‘Q-batches’. Our algorithm is shown in Algorithm 2.

Using a methodology similar to the one developed by Brochu et al. (2011), HedgeMO chooses the next ‘Q-batch’ of test points from the history of the candidates nominated by all of the acquisition functions. Rewards are calculated for each acquisition function from the surrogate surface for the entire history of the nominated points by the  $L$  acquisition functions. The rewards are then normalized to scale them to the same range for each objective. This step is fundamentally important because it prevents biasing the probability of any objective. This type of bias, of course, cannot occur in single objective problems. The rewards for different objectives  $k$  are then summed and the probability,  $p^l$ , of choosing a nominee from a specific acquisition function is calculated using step 6 in Algorithm 2. For a ‘Q-batch’ of tests points, the test points are chosen  $Q$  times.

**Regret Bounds:** The regret bounds derived by Brochu et al. (2011) hold for HedgeMO if and only if the Upper Confidence Bound (UCB) acquisition function is a part of the portfolio of acquisition functions. The regret bounds follow from the work of Srinivas et al. (2012) who derived cumulative regret bounds for UCB. In essence, the cumulative regret in our case is bounded by two sublinear terms as for UCB and an additional term which depends on proximity of the chosen point with the test point proposed by UCB. The interested reader is directed to Srinivas et al. (2012) and

## Algorithm 2 HedgeMO Algorithm

- 1: **Input:** Surrogate function  $\vec{g}(\vec{w}) : \vec{w} \in \mathcal{W}$ , proposed test points by AFs  $\left(\{(\vec{u})_{1:n}^{1:L}\}_{1:Q}\right)$ , batch points per epoch  $Q$ , current epoch  $n$ , total objective  $K$ , parameter  $\eta \in \mathbb{R}^+$
- 2: **for**  $l = 1$  **to**  $L$  **do**
- 3: For  $l^{th}$  acquisition function, find rewards for  $Q$ -batch points nominated by that AF from epochs  $1:n-1$ , by sampling from  $\vec{g}, \{\vec{\theta}_{1:n-1}^l\}_{1:Q} = \vec{\mu}(\{(\vec{u})_{1:n-1}^l\}_{1:Q})$ , where  $\vec{\theta} = \{\theta\}^k$  for each objective  $k$
- 4: **end for**
- 5: Normalize rewards for each  $l^{th}$  AF and  $k^{th}$  objective,  $\phi_l^k = \sum_{j=1}^{n-1} \sum_{q=1}^Q \frac{\{\theta_j^l\}_q^k - \min(\Theta)}{\max(\Theta) - \min(\Theta)}$ , where  $\Theta$  is defined as  $\Theta = \{\theta_{1:n-1}^{1:L}\}_{1:Q}^k$
- 6: Calculate probability for selecting nominees from  $l^{th}$  acquisition function,  $p^l = \frac{\exp(\eta \sum_{k=1}^K \phi_l^k)}{\sum_{i=1}^L \exp(\eta \sum_{k=1}^K \phi_i^k)}$
- 7: **for**  $q = 1$  **to**  $Q$  **do**
- 8: Select  $q^{th}$  nominee as test-point  $\vec{w}_n^q$  from  $l^{th}$  AF with probability  $p^l$
- 9: **end for**
- 10: **return** Batch of test points  $\{\vec{w}_n\}_{1:Q}$

Brochu et al. (2011) for a description of the exact regret bounds and their derivation.

## 4.3. SMC Acquisition Function

We introduce a new acquisition function, Stochastic Monte-Carlo (SMC), which for the maximization of an objective, is defined as:

$$SMC \equiv \operatorname{argmax}_{\vec{w} \in \mathcal{W}} [\vec{\mu}(\vec{w}) + r(\vec{w})], \quad (4)$$

where  $r(\vec{w})$  is sampled from  $U(0, 2\sigma(\vec{w}))$ , and  $\vec{\mu}(\vec{w})$  and  $\sigma(\vec{w})$  are the mean and standard deviation returned by the GP at  $\vec{w}$ , respectively. This is equivalent to taking Monte-Carlo samples from a truncated distribution. For categorical and ordinal variable problems, this acquisition function performs well across a range of benchmark tests (Vangelatos et al., 2021). We use this acquisition function as part of our portfolio of HedgeMO in the MixMOBO algorithm.

## 5. Experiments

We benchmarked MixMOBO against a range of existing state-of-the-art optimization strategies that are commonly used for optimizing expensive black-box functions with mixed-variable design spaces. We chose the following single objective optimization algorithms for comparison: **Co-CaBO** (Ru et al., 2020), which combines the multi-armed bandit (MAB) and Bayesian optimization approaches by

using a mixing kernel. CoCaBO has been shown to be more efficient than GPyOpt (one-hot encoding approach (GPyOptAuthors, 2016)) and EXP3BO (multi-armed bandit (MAB approach (Gopakumar et al., 2018))). We used CoCaBO with a mixing parameter of 0.5. We also tested MixMOBO against GBRT, a sequential optimization technique using gradient boosted regression trees (Scikit-learn, 2021). TPE\_Hyperopt (Tree-structured Parzen Estimator) is a sequential method for optimizing expensive black-box functions, introduced by Bergstra et al. (2011). SMAC3 is a popular Bayesian optimization algorithm in combination with an aggressive racing mechanism (Hutter et al., 2011). Both of these algorithms, in addition to Random Sampling, were used as baselines. Publicly available libraries for these algorithms were used.

Six different test functions for mixed variables were chosen as benchmarks. A brief description of these test functions and their properties is given below with further details in Appendix A:

**Contamination Problem:** This problem, introduced by Hu et al. (2011), considers a food supply chain with various stages in the chain where food may be contaminated with pathogens. The objective is to maximize the reward of prevention efforts while making sure the chain does not get contaminated. It is widely used as a benchmark for binary categorical variables. We use the problem as a benchmark with 21 binary categorical variables.

**Encrypted Amalgamated:** An anisotropic, mixed-variable function created using a combination of other commonly used test functions (Tušar et al., 2019). We modify the com-

bined function so that it can be used with mixed variables. In particular, it is adapted for categorical variables by encrypting the input space with a random vector, which produces a random landscape mimicking categorical variables (Vangelatos et al., 2021). Our Encrypted Amalgamated function has 13 inputs: 8 categorical, 3 ordinal variables (with 5 categories or states each) and 2 continuous.

**NK Landscapes:** This is a popular benchmark for simulating categorical variable problems using randomly rugged, interconnected landscapes (Kauffman & Levin, 1987; Li et al., 2006). The fitness landscape can be produced with random connectivity and number of optima. The problem is widely used in evolutionary biology and control optimization and is *NP*-complete. The probability of connectivity between *NK* is controlled by a ‘ruggedness’ parameter, which we set at 20%. We test the Li et al. (2006) variant with 8 categorical variables with 4 categories each.

**Rastrigin:** This is an isotropic test function, commonly used for continuous design spaces (Tušar et al., 2019). We use a 9-D Rastrigin function for testing a design space of 3 continuous and 6 ordinal variables with 5 discrete states.

**Encrypted Syblinski-Tang:** This function is isotropic (Tušar et al., 2019), and we have modified it as we did with the Encrypted Amalgamated test function so that it can work with categorical variables and was used as a representative benchmark for *N*-categorical variable problems. The 10-D variant tested here consists only of categorical variables with 5 categories each.

**Encrypted ZDT6:** This is a multi-objective test function introduced by Zitzler et al. (2000) that we modified with

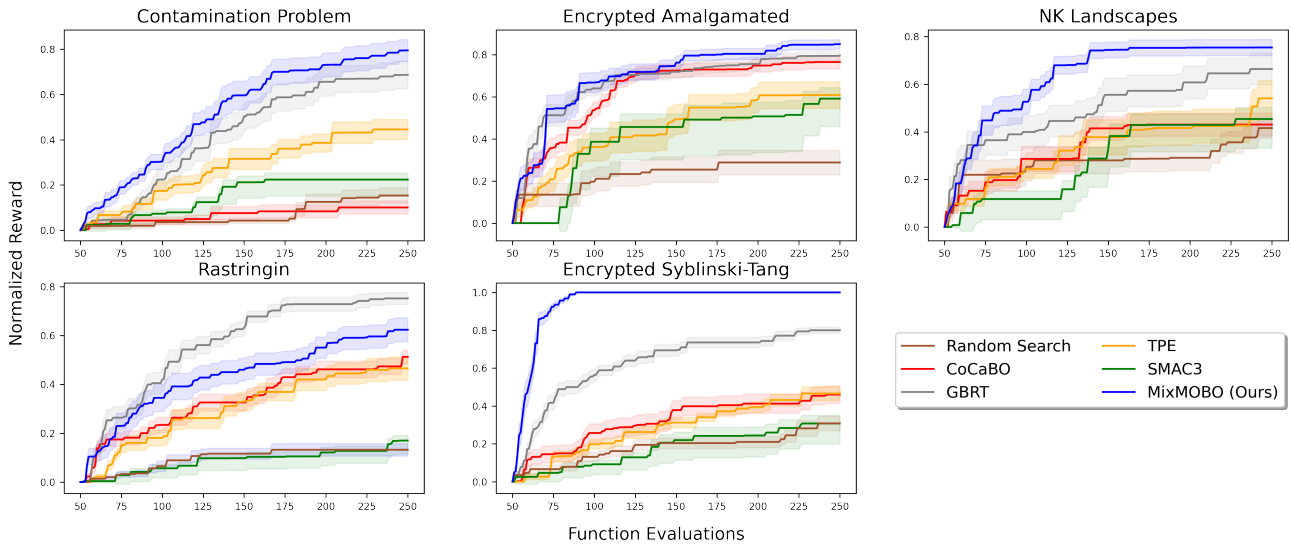


Figure 1. Performance comparison of MixMOBO against other mixed-variable algorithms

encryption so that it can deal with mixed variables. The test function is non-convex and non-uniform in the parameter space. We test ZDT6 with 10 categorical variables with 5 states each. ZDT6 was only used for testing HedgeMO.

To the extent of our knowledge, no other optimization algorithm is capable of handling mixed-variable, multi-objective problems in small-data settings. Thus, we have no direct comparisons between MixMOBO and other published algorithms. Therefore, we tested MixMOBO against a variant of NSGA-II (Deb et al., 2002) with the ZDT4 and ZDT6 test functions with mixed variables. However, we found that using a GA required more than  $10^2$  more function calls to find the Pareto front to a similar tolerance. For visualization purposes, we do not plot the GA results.

All of the optimization algorithms were run as maximizers, with a 0.005 noise variance built into all the benchmarks. The budget for each benchmark test was fixed at 250 function calls including the evaluations of 50 initial randomly sampled data points for all algorithms, except for SMAC3 which determines its own initial sample size. The algorithms were run in single output setting (GBRT, CoCaBO and MixMOBO’s batch mode was not used for fair comparison). Each algorithm was run 10 times for every benchmark. Our metric for optimization is the ‘Normalized Reward’, defined as  $(\text{current optimum} - \text{random sampling optimum}) / (\text{global optimum} - \text{random sampling optimum})$ . Figure 1 shows the Normalized Rewards versus the number of black-box function evaluations for MixMOBO and five other algorithms. The mean and standard deviation of the Normalized Rewards of the 10 runs for each algorithm, along with their standard deviations, are plotted. The width of each of the translucent colored bands around the mean line depicts equal to 1/5 of the standard deviation for visualization purposes.

MixMOBO outperforms all of the other baselines and is significantly better in dealing with mixed-variable problems. GBRT is the next best algorithm and performs better than MixMOBO on the Rastrigin function; however, note that the Rastrigin function does not include any categorical vari-

ables. For problems involving categorical variables, MixMOBO clearly outperforms the others. TPE and CoCaBO have similar performances, and SMAC3 has the poorest performance. All three are outdistanced by MixMOBO.

We also tested the efficacy of our HedgeMO algorithm by comparing it to different acquisition functions, namely, EI, UCB, and SMC, on three different test functions: the Encrypted Amalgamated, Encrypted Syblinski-Tang, and Encrypted ZDT6. The latter is used as the multi-objective test function. The Normalized Reward for the multi-objective Encrypted ZDT6 is defined as  $(\text{current P-optimum} - \text{random sampling P-optimum}) / (\text{global P-optimum} - \text{random sampling P-optimum})$ . Here,  $\text{P-optimum} = \frac{1}{N} \sum_{i=1}^N \exp(-\min. \text{distance in parameter space between } i^{\text{th}} \text{ global Pareto-optimal point and any point in the current Pareto-optimal set})$ , where  $N$  is the number of global Pareto-optimal points.

The results of our acquisition function comparisons are shown in Figure 2, which shows that HedgeMO performs well across all three test functions. For single-objective test functions, EI performs on par with HedgeMO. However, for the multi-objective Encrypted ZDT6 test function, EI performs significantly worse and is outperformed by both SMC and UCB. HedgeMO, on the other hand, consistently performs well in all scenarios. For unknown black-box functions, HedgeMO should be the acquisition function of choice for mixed-variable and multi-objective problems.

## 6. Real-World Applications

We applied our MixMOBO framework to the optimization of the design of architected, microlattice structures. Advances in modeling, fabrication, and testing of architected materials have promulgated their utility in engineering applications, such as ultralight (Zheng et al., 2014; Pham et al., 2019; Zhang et al., 2019), reconfigurable (Xia et al., 2019), and high-energy-absorption materials (Song et al., 2019b), and in bio-implants (Song et al., 2020). The optimization of architected materials (Bauer et al., 2016; Pham et al., 2019;

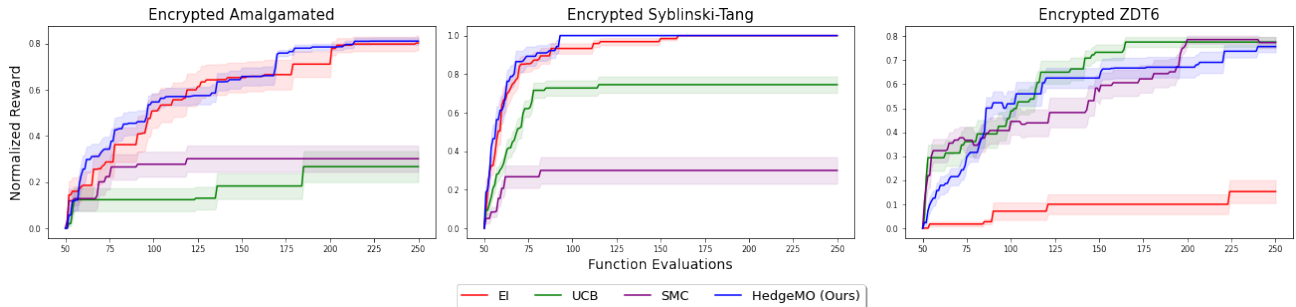


Figure 2. Performance comparison of HedgeMO against other acquisition functions

Table 1. Experimental determined values of micro-lattice structures. critical buckling  $P_c$  which was minimized for MixMOBO, strain energy density at buckling and fracture,  $u_b$  and  $u_f$  respectively, elastic stiffness  $S$ , and ratio of normalized strain energy density of each structure compared to the Unblemished structure.

Structure	$P_c[\mu N]$	$u_b[MJm^{-3}]$	$u_f[MJm^{-3}]$	$S[MPa]$	$(u_{fi}/u_{bi})/(u_{f1}/u_{b1})$
Unblemished	3814.5	1.08	0.071	388.21	1
Random Sampling Optimal	996.2	0.08	2.85	347.19	526
MixMOBO Optimal	<b>545.1</b>	<b>0.02</b>	<b>14.71</b>	<b>460.35</b>	<b>12030</b>

Xia et al., 2019; Zhang et al., 2019) often requires searching huge combinatorial design spaces, where the evaluation of each design is expensive. (Meza et al., 2014; Berger et al., 2017; Tancogne-Dejean et al., 2018). The design

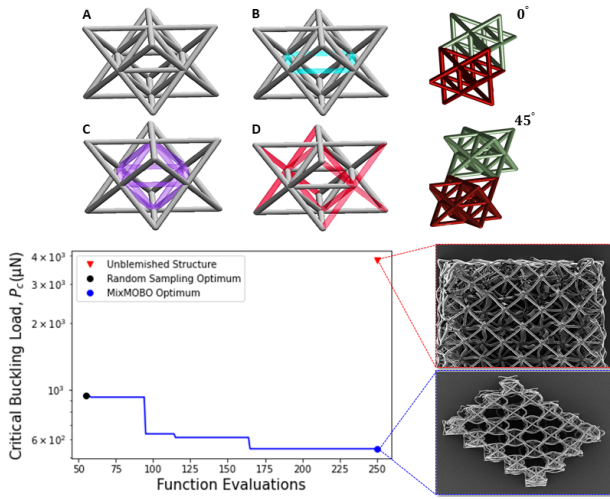


Figure 3. Top Left: The 4 unit cells, labelled A – D. Top Right: The 2 orientations in which they can be joined. Bottom Left: Optimization results using MixMOBO. Bottom Right: SEM images of Unblemished and Optimum structures.

space for the architected material we optimize here has  $\sim 8.5$  billion possible combinations of its 17 categorical inputs (one with 2 possible states, and the other 16 with 4 possible states). Our goal is to maximize the strain energy density of a microlattice structure. We maximize the strain energy density (which is extremely expensive to compute, even for one design) by minimizing the buckling load  $P_c$ , while maintaining the lattice’s structural integrity and stiffness before fracturing. Minimizing  $P_c$  (a proxy for maximizing the strain energy density by instigating buckling which leads to the densification of the deformed lattice members) is a more computationally tractable cost function to evaluate (but, it is still expensive and involves solving a numerical finite element code for each evaluation of the cost function.)

The design space consists of choosing one of four possible unit cells (shown in the upper left of Fig. 3, each with one

or more defects (shown in color) in them, at each of the 16 independent lattice sites) creating 16 of the categorical inputs with 4 possible values; and the choice of whether the cells are connected along their faces or along their edges on  $45^\circ$ -diagonals (shown in the upper right panel of Fig. 3) creating the 17<sup>th</sup> categorical input with 2 possible values.

The minimization of  $P_c$  using MixMOBO was initialized with 50 random structures and the evaluation budget, including initial samples, was set at 250. The algorithm achieved a 42% improvement in the  $P_c$  of the lattice structure over the best structure obtained with the first 50 random samples (Figure 3). The optimal microlattice obtained using  $P_c$  as a proxy with MixMOBO has an experimentally measured normalized strain energy density that is 12,030 times greater than that of the unblemished microlattice structure with no defects that is cited in the literature to have the best strain energy density (Vangelatos et al., 2020), a 4 orders of magnitude increase. Table 1 shows the properties of the fabricated and experimentally measured design created with MixMOBO. The choices of the units cells in the optimally designed lattice that were determined by MixMOBO are not intuitive and have no obvious pattern or structure (Vangelatos et al., 2021). The manufacturing and testing details of our methodology are included in Appendix B.

## 7. Conclusions

The existing optimization literature does not offer an algorithm for optimizing multi-objective, mixed-variable problems with expensive black-box functions. We have introduced Mixed-variable Multi-Objective Bayesian Optimization (MixMOBO), the first BO based algorithm for optimizing such problems. MixMOBO is agnostic to the underlying kernel and extends the GP-based BO algorithms to handle multi-objective, mixed-variable problems. Our formulation is compatible with modified kernels and other surrogate methods developed in previous studies for mixed-variable problems and also allows for parallel batch updates without repeated evaluations of the surrogate surface, while maintaining diversification within the solution set. We presented the Hedge Multi-Objective (HedgeMO) algorithm, a novel Hedge strategy for which regret bounds hold for multi-objective problems. A new acquisition function,



Stochastic Monte-Carlo (SMC) was also proposed as part of the HedgeMO portfolio. MixMOBO and HedgeMO were benchmarked and shown to be significantly better on a variety of test problems compared to existing mixed-variable optimization algorithms. MixMOBO was then applied to the real-world optimization of an architected micro-lattice, and we increased the structure’s strain-energy density by  $10^4$  compared to existing Unblemished structures in the literature reported to have highest strain energy density. Our future work entails further testing multi-objective and ‘Q-batch’ settings. Currently, we are applying MixMOBO to real-world expensive engineering problems, including wind turbine farms, hydrokinetic turbines, and aircraft wings.

## Acknowledgements

The authors would like to thank like to thank Chiyu ‘Max’ Jiang, research scientist at Waymo Research, and Professor Uros Seljak, Department of Physics, University of California at Berkeley (UCB) for insightful discussions regarding Bayesian optimization. We would also like to thank Zacharias Vangelatos and Professor Costas P. Grigoropoulos, Department of Mechanical Engineering, University of California at Berkeley (UCB) for the collaboration for designing and manufacturing architected materials, and nanoindentation, SEM, and HIM experiments. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1548562 through allocation TG-CTS190047.

## Conflicts and Funding Disclosure

The authors declare no funding sources or conflicts that need to be disclosed.

## Data Availability

Complete data sets to reproduce any and all experiments are included in the Supplementary Materials. The algorithm will be publicly accessible after publication.

## References

- Balandat, M., Karrer, B., Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., and Bakshy, E. Botorch: A framework for efficient monte-carlo bayesian optimization, 2020.
- Baptista, R. and Poloczek, M. Bayesian optimization of combinatorial structures, 2018.
- Bauer, J., Schroer, A., Schwaiger, R., and Kraft, O. Approaching theoretical strength in glassy carbon nanolattices. *Nature Materials*, 15(4):438–443, 2016.
- Berger, J., Wadley, H., and McMeeking, R. Mechanical metamaterials at the theoretical limit of isotropic elastic stiffness. *Nature*, 543(7646):533–537, 2017.
- Bergstra, J., Bardenet, R., Bengio, Y., and Kégl, B. Algorithms for hyper-parameter optimization. In *Proceedings of the 24th International Conference on Neural Information Processing Systems, NIPS’11*, pp. 2546–2554, Red Hook, NY, USA, 2011. Curran Associates Inc. ISBN 9781618395993.
- Bergstra, J., Yamins, D., and Cox, D. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pp. 115–123, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.
- Brochu, E., Cora, V. M., and de Freitas, N. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, 2010.
- Brochu, E., Hoffman, M. W., and de Freitas, N. Portfolio allocation for bayesian optimization, 2011.
- Chen, D., Skouras, M., Zhu, B., and Matusik, W. Computational discovery of extremal microstructure families. *Science Advances*, 4(1):eaao7005, 2018a.
- Chen, W., Watts, S., Jackson, J. A., Smith, W. L., Tortorelli, D. A., and Spadaccini, C. M. Stiff isotropic lattices beyond the maxwell criterion. *Science Advances*, 5(9): eaaw1937, 2019.
- Chen, Y., Huang, A., Wang, Z., Antonoglou, I., Schrittwieser, J., Silver, D., and de Freitas, N. Bayesian optimization in alphago. *CoRR*, abs/1812.06855, 2018b.
- Daulton, S., Balandat, M., and Bakshy, E. Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization, 2020.
- Daulton, S., Eriksson, D., Balandat, M., and Bakshy, E. Multi-objective bayesian optimization over high-dimensional search spaces, 2021.
- Daxberger, E., Makarova, A., Turchetta, M., and Krause, A. Mixed-variable bayesian optimization. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, Jul 2020. doi: 10.24963/ijcai.2020/365.
- Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2): 182–197, 2002. doi: 10.1109/4235.996017.

- Deshwal, A. and Doppa, J. R. Combining latent space and structured kernels for bayesian optimization over combinatorial spaces. *CoRR*, abs/2111.01186, 2021.
- Deshwal, A., Belakaria, S., and Doppa, J. R. Bayesian optimization over hybrid spaces, 2021.
- Flamourakis, G., Spanos, I., Vangelatos, Z., Manganas, P., Papadimitriou, L., Grigoropoulos, C., Ranella, A., and Farsari, M. Laser-made 3d auxetic metamaterial scaffolds for tissue engineering applications. *Macromolecular Materials and Engineering*, 305(7):2000238, 2020.
- Fonseca, C., Paquete, L., and Lopez-Ibanez, M. An improved dimension-sweep algorithm for the hypervolume indicator. In *2006 IEEE International Conference on Evolutionary Computation*, pp. 1157–1163, 2006. doi: 10.1109/CEC.2006.1688440.
- Frazier, P. I. and Wang, J. Bayesian optimization for materials design. *Springer Series in Materials Science*, pp. 45–75, Dec 2015. ISSN 2196-2812. doi: 10.1007/978-3-319-23871-5\_3.
- Garrido-Merchán, E. C. and Hernández-Lobato, D. Dealing with categorical and integer-valued variables in bayesian optimization with gaussian processes. *Neurocomputing*, 380:20–35, Mar 2020. ISSN 0925-2312. doi: 10.1016/j.neucom.2019.11.004.
- Golovin, D., Solnik, B., Moitra, S., Kochanski, G., Karro, J., and Sculley, D. Google vizier: A service for black-box optimization. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’17, pp. 1487–1495, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450348874. doi: 10.1145/3097983.3098043.
- Gopakumar, S., Gupta, S., Rana, S., Nguyen, V., and Venkatesh, S. Algorithmic assurance: An active approach to algorithmic testing using bayesian optimisation. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL <https://proceedings.neurips.cc/paper/2018/file/cc70903297fe1e25537ae50aea186306-Paper.pdf>.
- GPyOptAuthors. GPyOpt: A bayesian optimization framework in python. <http://github.com/SheffieldML/GPyOpt>, 2016.
- Hu, Y., Hu, J., Xu, Y., Wang, F., and Cao, R. Contamination control in food supply chain. pp. 2678 – 2681, 01 2011. doi: 10.1109/WSC.2010.5678963.
- Hutter, F., Hoos, H. H., and Leyton-Brown, K. Sequential model-based optimization for general algorithm configuration. In Coello, C. A. C. (ed.), *Learning and Intelligent Optimization*, pp. 507–523, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-25566-3.
- Kauffman, S. and Levin, S. Towards a general theory of adaptive walks on rugged landscapes. *Journal of Theoretical Biology*, 128(1):11–45, 1987. ISSN 0022-5193. doi: [https://doi.org/10.1016/S0022-5193\(87\)80029-2](https://doi.org/10.1016/S0022-5193(87)80029-2). URL <https://www.sciencedirect.com/science/article/pii/S0022519387800292>.
- Korovina, K., Xu, S., Kandasamy, K., Neiswanger, W., Poczos, B., Schneider, J., and Xing, E. Chembo: Bayesian optimization of small organic molecules with synthesizable recommendations. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3393–3403. PMLR, 26–28 Aug 2020.
- Krause, A., Singh, A., and Guestrin, C. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(8):235–284, 2008.
- Li, R., Emmerich, M., Eggermont, J., Bovenkamp, E., Bäck, T., Dijkstra, J., and Reiber, J. Mixed-integer nk landscapes. volume 4193, pp. 42–51, 01 2006. ISBN 978-3-540-38990-3. doi: 10.1007/11844297\_5.
- Lindauer, M., Eggensperger, K., Feurer, M., Biedenkapp, A., Deng, D., Benjamins, C., Sass, R., and Hutter, F. Smac3: A versatile bayesian optimization package for hyperparameter optimization, 2021.
- Meza, L. R., Das, S., and Greer, J. R. Strong, lightweight, and recoverable three-dimensional ceramic nanolattices. *Science*, 345(6202):1322–1326, 2014.
- Mockus, J. Application of bayesian approach to numerical methods of global and stochastic optimization. *Journal of Global Optimization*, 4:347–365, 1994.
- Murphy, K. P. *Machine learning: a probabilistic perspective*. MIT Press, 2012.
- Nguyen, D., Gupta, S., Rana, S., Shilton, A., and Venkatesh, S. Bayesian optimization for categorical and category-specific continuous inputs, 2019.
- Oh, C., Gavves, E., and Welling, M. BOCK : Bayesian optimization with cylindrical kernels. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3868–3877. PMLR, 10–15 Jul 2018.

- Oh, C., Tomczak, J. M., Gavves, E., and Welling, M. Combinatorial bayesian optimization using the graph cartesian product, 2019.
- Oh, C., Gavves, E., and Welling, M. Mixed variable bayesian optimization with frequency modulated kernels, 2021.
- Pelamatti, J., Brevault, L., Balesdent, M., Talbi, E.-G., and Guerin, Y. Efficient global optimization of constrained mixed variable problems, 2018.
- Pham, M.-S., Liu, C., Todd, I., and Lertthanasarn, J. Damage-tolerant architected materials inspired by crystal microstructure. *Nature*, 565(7739):305–311, 2019.
- Pyzer-Knapp, E. Bayesian optimization for accelerated drug discovery. *IBM Journal of Research and Development*, PP:1–1, 11 2018. doi: 10.1147/JRD.2018.2881731.
- Qian, P. Z. G., Wu, H., and Wu, C. J. Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics*, 50(3):383–396, 2008.
- Rasmussen, C. E. and Williams, C. K. I. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006. ISBN 026218253X.
- Ru, B., Alvi, A. S., Nguyen, V., Osborne, M. A., and Roberts, S. J. Bayesian optimisation over multiple continuous and categorical inputs, 2020.
- Scikit-learn. scikit-optimize. <https://scikit-optimize.github.io/stable/>, 2021.
- Shaw, L. A., Sun, F., Portela, C. M., Barranco, R. I., Greer, J. R., and Hopkins, J. B. Computationally efficient design of directionally compliant metamaterials. *Nature Communications*, 10(1):1–13, 2019.
- Shu, L., Jiang, P., Shao, X., and Wang, Y. A New Multi-Objective Bayesian Optimization Formulation With the Acquisition Function for Convergence and Diversity. *Journal of Mechanical Design*, 142(9), 03 2020. ISSN 1050-0472. doi: 10.1115/1.4046508. 091703.
- Snoek, J., Larochelle, H., and Adams, R. P. Practical bayesian optimization of machine learning algorithms, 2012.
- Song, J., Wang, Y., Zhou, W., Fan, R., Yu, B., Lu, Y., and Li, L. Topology optimization-guided lattice composites and their mechanical characterizations. *Composites Part B: Engineering*, 160:402–411, 2019a.
- Song, J., Zhou, W., Wang, Y., Fan, R., Wang, Y., Chen, J., Lu, Y., and Li, L. Octet-truss cellular materials for improved mechanical properties and specific energy absorption. *Materials & Design*, 173:107773, 2019b.
- Song, J., Michas, C., Chen, C. S., White, A. E., and Grinstaff, M. W. From simple to architecturally complex hydrogel scaffolds for cell and tissue engineering applications: Opportunities presented by two-photon polymerization. *Advanced Healthcare Materials*, 9(1):1901217, 2020.
- Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. W. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, May 2012. ISSN 1557-9654. doi: 10.1109/tit.2011.2182033. URL <http://dx.doi.org/10.1109/TIT.2011.2182033>.
- Suzuki, S., Takeno, S., Tamura, T., Shitara, K., and Karasuyama, M. Multi-objective bayesian optimization using pareto-frontier entropy, 2020.
- Tancogne-Dejean, T., Diamantopoulou, M., Gorji, M. B., Bonatti, C., and Mohr, D. 3d plate-lattices: An emerging class of low-density metamaterial exhibiting optimal isotropic stiffness. *Advanced Materials*, 30(45):1803334, 2018.
- Terzaki, K., Vasilantonakis, N., Gaidukeviciute, A., Reinhardt, C., Fotakis, C., Vamvakaki, M., and Farsari, M. 3d conducting nanostructures fabricated using direct laser writing. *Optical Materials Express*, 1(4):586–597, 2011.
- Tiao, L. C., Klein, A., Seeger, M., Bonilla, E. V., Archambeau, C., and Ramos, F. Bore: Bayesian optimization by density-ratio estimation, 2021.
- Tušar, T., Brockhoff, D., and Hansen, N. Mixed-integer benchmark problems for single- and bi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '19*, pp. 718–726, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361118. doi: 10.1145/3321707.3321868. URL <https://doi.org/10.1145/3321707.3321868>.
- Vangelatos, Z., Komvopoulos, K., and Grigoropoulos, C. Regulating the mechanical behavior of metamaterial microlattices by tactical structure modification. *Journal of the Mechanics and Physics of Solids*, pp. 104112, 2020.
- Vangelatos, Z., Sheikh, H. M., Marcus, P. S., Grigoropoulos, C. P., Lopez, V. Z., Flamourakis, G., and Farsari, M. Strength through defects: A novel bayesian approach for the optimization of architected materials. *Science Advances*, 7(41), 2021. doi: 10.1126/sciadv.abk2218.
- Vasilantonakis, N., Terzaki, K., Sakellari, I., Purlys, V., Gray, D., Soukoulis, C. M., Vamvakaki, M., Kafesaki, M., and Farsari, M. Three-dimensional metallic photonic crystals with optical bandgaps. *Advanced Materials*, 24(8):1101–1105, 2012.

- Xia, X., Afshar, A., Yang, H., Portela, C. M., Kochmann, D. M., Di Leo, C. V., and Greer, J. R. Electrochemically reconfigurable architected materials. *Nature*, 573(7773): 205–213, 2019.
- Zhang, X., Vyatskikh, A., Gao, H., Greer, J. R., and Li, X. Lightweight, flaw-tolerant, and ultrastrong nanoarchitected carbon. *Proceedings of the National Academy of Sciences*, 116(14):6665–6672, 2019.
- Zhang, Y., Apley, D. W., and Chen, W. Bayesian optimization for materials design with mixed quantitative and qualitative variables. *Scientific Reports*, 10(1):1–13, 2020.
- Zheng, X., Lee, H., Weisgraber, T. H., Shusteff, M., DeOtte, J., Duoss, E. B., Kuntz, J. D., Biener, M. M., Ge, Q., Jackson, J. A., et al. Ultralight, ultrastiff mechanical metamaterials. *Science*, 344(6190):1373–1377, 2014.
- Zhou, Q., Qian, P. Z., and Zhou, S. A simple approach to emulation for computer models with qualitative and quantitative factors. *Technometrics*, 53(3):266–273, 2011.
- Zitzler, E., Deb, K., and Thiele, L. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evol. Comput.*, 8(2):173–195, jun 2000. ISSN 1063-6560. doi: 10.1162/106365600568202. URL <https://doi.org/10.1162/106365600568202>.



## A. Benchmark Test Functions

In this section, we define the benchmark test functions, all of which are set to be maximized during our optimizations.

### Contamination Problem

The contamination problem was introduced by [Hu et al. \(2011\)](#) and is used to test categorical variables with binary categories. The problem aims to maximize the reward function for applying a preventative measure to stop contamination in a food supply chain with  $D$  stages. At each  $i^{th}$  stage, where  $i \in [1, D]$ , decontamination efforts can be applied. However, this effort comes at a cost  $c$  and will decrease the contamination by a random rate  $\Gamma_i$ . If no prevention effort is taken, the contamination spreads with a rate of  $\Omega_i$ . At each stage  $i$ , the fraction of contaminated food is given by the recursive relation:

$$Z_i = \Omega_i(1 - w_i)(1 - Z_{i-1}) + (1 - \Sigma_i w_i)Z_{i-1} \quad (5)$$

here  $w_i \in 0, 1$  and is the decision variable to determine if preventative measures are taken at  $i^{th}$  stage or not. The goal is to decide which stages  $i$  action should be taken to make sure  $Z_i$  does not exceed an upper limit  $U_i$ .  $\Omega_i$  and  $\Sigma_i$  are determined by a uniform distribution. We consider the problem setup with Lagrangian relaxation ([Baptista & Poloczek, 2018](#)):

$$f(\vec{w}) = - \sum_{i=1}^D \left( cw_i + \frac{\rho}{T} \sum_{k=1}^T 1_{\{Z_k > U_i\}} \right) - \lambda \|\vec{w}\|_1 \quad (6)$$

Here violation of  $Z_k < U_i$  is penalized by  $\rho = 1$  and summing the contaminated stages if the limit is violated and our total stages or dimensions are  $D = 21$ . The cost  $c$  is set to be 0.2 and  $Z_1 = 0.01$ . As in the setup for ([Baptista & Poloczek, 2018](#)), we use  $T = 100$  stages,  $U_i = 0.1$ ,  $\lambda = 0.01$  and  $\epsilon = 0.05$ .

### Encrypted Amalgamated

Analytic test functions generally cannot mimic mixed variables. To map the continuous output of a function into  $N$  discrete ordinal or categorical variables, the continuous range of the test function's output is first discretized into  $N$  discrete subranges by selecting  $(N - 1)$  break points, often equally spaced, within the bounds of the range. Then, the continuous output variable is assigned the integer round-off value of the subrange defined by its surrounding pair of break points. If necessary, the domain of the test function's output is first mapped into a larger domain so that each subrange has a unique integer value. To mimic ordinal variables, we are done, but for categorical variables, a random vector for each categorical variable is then generated which scrambles or 'encrypts' the indices of these values, thus creating random landscapes as is the case for categorical variables with a latent space. The optimization algorithm only sees the encrypted space and the random vector is only used when evaluating the black-box function.

We also define a new test function that we call the *Amalgamated function*, a piece-wise function formed from commonly used analytical test functions with different features (for more details on these functions we refer to [Tušar et al. \(2019\)](#)). Amalgamated function is non-convex and anisotropic, unlike conventional test functions where isotropy can be exploited.

For  $i = 1 \dots n$ ,  $k = \text{mod}(i - 1, 7)$ :

$$f(\vec{w}) = \sum_{i=1}^D g(w_i) \quad (7)$$

where

$$g(w_i) = \begin{cases} \sin(w_i) & \text{if } k = 0, w_i \in (0, \pi) \\ -\frac{w_i^4 - 16w_i^2 + 5w_i}{2} & \text{if } k = 1, w_i \in (-5, 5) \\ -(w_i^2) & \text{if } k = 2, w_i \in (-10, 10) \\ -[10 + w_i^2 - 10\cos(2\pi w_i)] & \text{if } k = 3, w_i \in (-5, 5) \\ -[100(w_i - w_{i-1}^2)^2 + (1 - w_i)^2] & \text{if } k = 4, w_i \in (-2, 2) \\ |\cos(w_i)| & \text{if } k = 5, w_i \in (-\pi/2, \pi/2) \\ -w_i & \text{if } k = 6, w_i \in (-30, 30) \end{cases} \quad (8)$$

To create the Encrypted Amalgamated function, for categorical and ordinal variables, equally spaced points are taken within the bounds defined above. For our current work, we use a  $D = 13$  with 8 categorical and 3 ordinal variables with 5 states each, and 2 continuous variables.

### NK Landscapes

NK Landscapes were introduced by [Kauffman & Levin \(1987\)](#) as a way of creating optimization problems with categorical variables.  $N$  describes the number of genes or number of dimensions  $D$  and  $K$  is the number of epistatic links of each gene to other genes, which describes the ‘ruggedness’ of the landscape. A large number of random landscapes can be created for given  $N$  and  $K$  values. The global optimum of a generated landscape for experimentation can only be computed through complete enumeration. The landscape cost for any vector is calculated as an average of each component cost. Each component cost is based on the random values generated for the categories, not only by its own alleles, but also by the alleles in the other genes connected through the random epistasis matrix, with  $K$  probability or ruggedness. A  $K = 1$  ruggedness translates to a fully connected genome.

The  $NK$  Landscapes from [Kauffman & Levin \(1987\)](#) were formulated only for binary variables. They were extended by [Li et al. \(2006\)](#) for multi-categorical problems, which is the formulation we use. Details of the  $NK$  Landscape test-functions we use can be found in [Li et al. \(2006\)](#). For the current study, we use  $N = 8$  with 4 categories each and ruggedness  $K = 0.2$ .

### Rastrigin

Rastrigin function is a commonly used non-convex optimization function ([Tušar et al., 2019](#)) with a large number of local optima. It is defined as:

$$f(\vec{w}) = -[10 + w_i^2 - 10\cos(2\pi w_i)], w_i \in (-5, 5) \quad (9)$$

We use  $D = 9$  for testing with 6 ordinal with 5 discrete states and 3 continuous variables. The ordinal variables are equally spaced within the bounds.

### Encrypted Syblinski-Tang

We use the *Syblinski-Tang function* ([Tušar et al., 2019](#)), an isotropic non-convex function. The function is considered difficult to optimize because many search algorithms get ‘stuck’ at a local optimum. For use with categorical variables, we encrypt it as described previously. The Syblinski-Tang function, in terms of input vector  $\vec{w}$ , is defined as:

$$f(\vec{w}) = -\frac{\sum_{i=1}^D w_i^4 - 16w_i^2 + 5w_i}{2}, w_i \in (-5, 2.5) \quad (10)$$

For the current study, this function was tested with  $D = 10$  categorical variables and 5 categories for each variable.

### Encrypted ZDT6

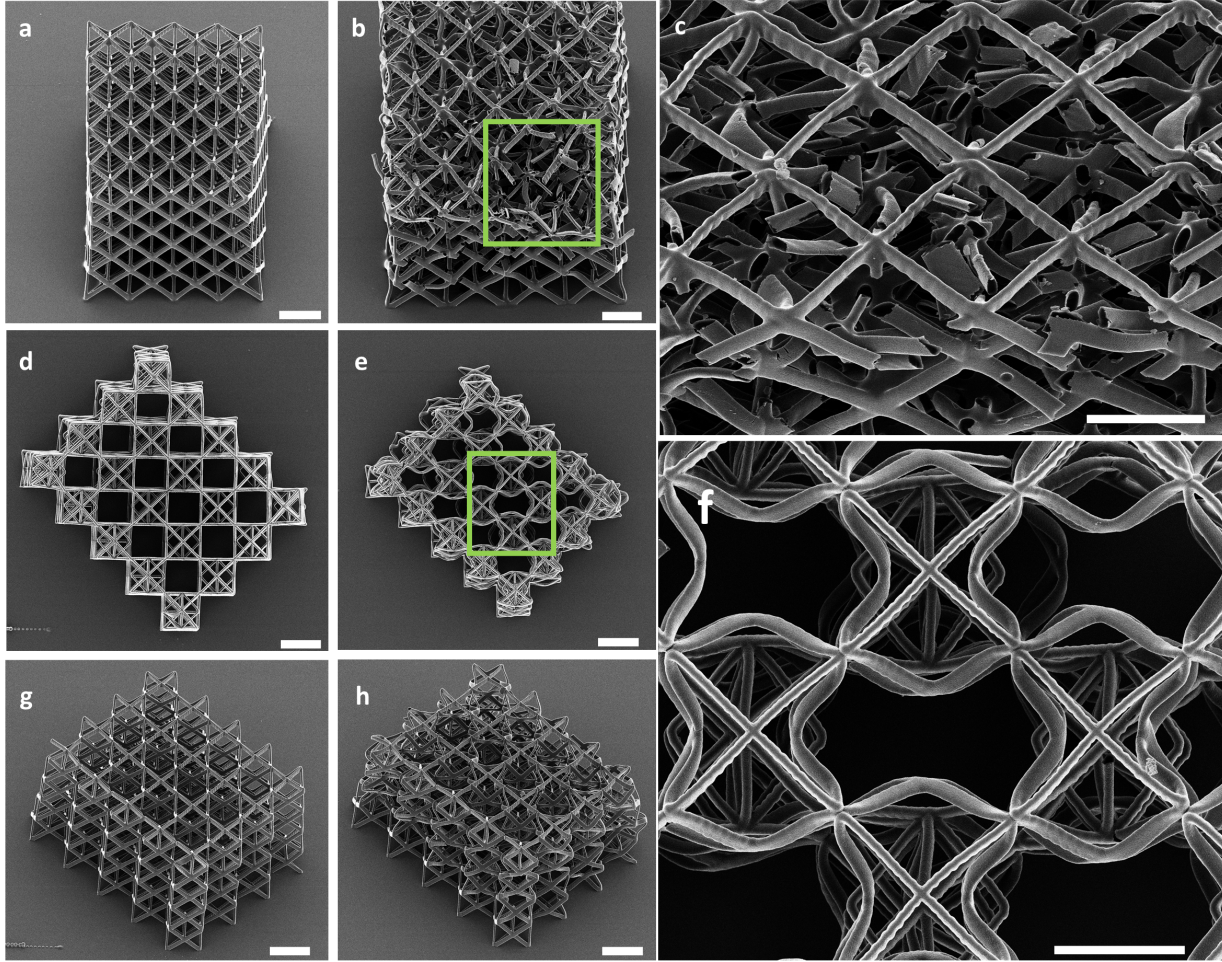
*ZDT* benchmarks are a suite of multi-objective problems, suggested by [Zitzler et al. \(2000\)](#), and most commonly used for testing such problems. We use *ZDT6*, which is non-convex and non-uniform in its parameter space. We again modify the function by encrypting it to work with categorical problems. *ZDT6* is defined as:

$$\begin{aligned} f_1(\vec{w}) &= \exp(-4w_1)\sin^6(6\pi w_1) - 1 \\ f_2(\vec{w}) &= -g(\vec{w}) \left[ 1 - (f_1(\vec{w})/g(\vec{w}))^2 \right] \\ g(\vec{w}) &= 1 + 9 \left[ \left( \sum_{i=2}^D w_i \right) / (n-1) \right]^{1/4} \end{aligned} \quad (11)$$

Here  $w_1 \in [0, 1]$  and  $w_i = 0$  for  $i = 2, \dots, D$ . The function was tested for  $D = 10$  with 5 categories each. We note that to evaluate the performance of MixMOBO, we compared it against the NSGA-II variant ([Deb et al., 2002](#)) that can deal with mixed variables (by running *ZDT4* in a mixed variable setting and *ZDT6* with categorical variables). No encryption is necessary for GAs. GAs required, on average,  $10^2$  more function calls compared to MixMOBO.

## B. Architected Materials - Manufacturing and Testing

The microlattice structures were fabricated with a hybrid organic-inorganic material Zr-DMAEMA (30 wt%). The composition of this material is 70 wt% zirconium propoxide and 10 wt% (2-dimethylaminoethyl) methacrylate (DMAEMA) (Sigma-Aldrich). Before the fabrication of the structures, the material was placed onto glass substrates and remained in vacuum for 24 h. Further information about the material preparation can be found elsewhere (Terzaki et al., 2011; Vasilantonakis et al., 2012).



**Figure 4.** HIM images of the loaded and unloaded unblemished and optimum structures. (a) Image of the unblemished structure consisting only of units cells of type A as in Fig. 3. (b) Same as (a) but after loading, showing severe fracture and collapse of many beam members. (c) High-depth-of-focus image of the region inside the square box shown in (b) revealing several fractured beams and the internal collapse of the upper layer that subsequently instigated the accumulation of damage in the underlying layers. (d) Same as in (a) but for the unloaded optimum structure. (e) Same as in (b) but after the structure was subjected to the same maximum compressive load as the structure shown in (b). Unloading of the optimum structure showed only excessive plastic deformation without catastrophic collapse and the manifestation of the buckling mode. (f) High-depth-of-focus image of the region inside the square box shown in (e) revealing the effect of buckling that led to deformation but no fracture due to the occurrence of densification. (g) Side view of the unloaded optimum structure shown from an isometric view. (h) Side view of the unloaded optimum structure shown from an isometric view revealing that fracture was inhibited throughout the structure due to the densification precipitated by the low critical buckling load. Each scale bar is equal to 10  $\mu\text{m}$ .

The test structures were fabricated by diffusion-assisted high-resolution direct femtosecond laser writing, which employs MPL and the aforementioned photoresist for high-resolution fabrication. The system consists of a FemtoFiber pro NIR laser with a wavelength of 780nm, pulse width of 100fs, and repetition rate of 80 MHz. Local photopolymerization of the photosensitive material was accomplished by tight focusing the laser beam onto the material with a 100X microscope

objective lense (Plan-ApoChromat 100x/1.40 Oil M27, Zeiss).

Our experimental laser setup utilizes precise piezoelectric stages (Physik Instrumente M-110.1DG), where their movements are controlled by a program (3dpoli, Femtika). The IGES file of each unit cell was created in Solidworks 2019 and used to create the coordinates needed to translate the 3D model into movements of the stages. The laser output energy for the fabrication was measured right before the objective lens opening at 10 *mW* and the scanning speed used was 80  $\mu\text{m/s}$ . The fabrication of each scaffold was conducted by scanning the top layer of unit cells and then moving downwards to the interface on the glass substrate in a unit cell-by-unit cell manner. The stage movements were directed along the vectors of each unit cell from top to bottom avoiding slicing the scaffolds and printing it in a layer-by-layer manner, thus ensuring the best mechanical stability and reproducibility of the experiment. Only a single scan was made for each vector as it proved to be sufficient for the rigidity of the structures. A detailed description of the setup can be found in (Flamourakis et al., 2020).

Uniaxial compression tests were performed in situ with a nanoindentation apparatus (PI 85 SEM PicoIndenter, Hysitron) mounted inside the chamber of a scanning electron microscope (FEI Quanta 3D FEG) to enable high-precision nanomechanical testing and real-time recording of the deformation of the test structures. We show the results of the compression tests in Figure 4. A flat molybdenum tip (model # 72SC-D3/035 (407A-M)) of 90  $\mu\text{m}$  diameter was used in all the compression tests. To prevent the displacement of the substrate during testing, the glass substrates on which the specimens were mounted were fixed onto an SEM pin stub mount (TED PELLA) with PELCO® Pro C100 Cyanoacrylate Glue, TED PELLA). All of the structures were aligned such that the front face to be visible during testing to facilitate the detection of the location and instant of buckling and fracture. To capture the structure collapse and track the proliferation of damage, each structure was deformed at a rate of 500 *nm/s* to a maximum compressive strain of 55%. To ensure repeatability, at least 5 tests were performed with structures of a given design. To capture the instigation of the first buckling instability and, consequently, compute the critical buckling load, the measured force-displacement curves were frame-by-frame juxtaposed with the recordings of the deformation to identify whether excessive deformation produced a singularity in the force-displacement response. HIM that provides high depth of focus and high resolution was used to track the evolution of damage in the interior of the tested structures (Vangelatos et al., 2021).