Bi-objective Bayesian Optimization of Engineering Problems with Cheap and Expensive Cost Functions

**Nasrulloh Loka · Ivo Couckuyt ·
Federico Garbuglia · Inneke Van
Nieuwenhuyse · Domenico Spina · Tom
Dhaene**

Abstract Multi-objective optimization of complex engineering systems is a
challenging problem. The design goals can exhibit dynamic and nonlinear be-
haviour with respect to the system's parameters. Additionally, modern engi-
neering is driven by simulation-based design which can be computationally
expensive due to the complexity of the system under study. Bayesian op-
timization (BO) is a popular technique to tackle this kind of problem. In
multi-objective BO, a data-driven surrogate model is created for each design
objective. However, not *all* of the objectives may be expensive to compute.
We develop an approach that can deal with a mix of expensive and cheap-to-
evaluate objective functions. As a result, the proposed technique offers lower
complexity than standard multi-objective BO methods and performs signif-
icantly better when the cheap objective function is difficult to approximate.
In particular, we extend the popular hypervolume-based Expected Improve-
ment (EI) and Probability of Improvement (POI) in bi-objective settings. The
proposed methods are validated on multiple benchmark functions and two real-
world engineering design optimization problems, showing that it performs bet-
ter than its *non-cheap* counterparts. Furthermore, it performs competitively
or better compared to other optimization methods.

N. Loka
Department of Information Technology (INTEC), IDLab, Ghent University-imec,
iGent, Technologiepark-Zwijnaarde 126, B-9052 Gent, Belgium
Tel.: +32(0)93314940
Fax: +32(0)93314899
E-mail: nasrulloh.loka@ugent.be

I. Couckuyt · F. Garbuglia · D. Spina · T. Dhaene
Department of Information Technology (INTEC), IDLab, Ghent University-imec

I. Van Nieuwenhuyse
Quantitative Methods, Research Group Logistics
UHasselt, Belgium

1 Introduction

In real-world problems, the optimization goals mostly consist of multiple conflicting objectives. Thus, optimizing all the objectives simultaneously leads to multiple solutions that are mathematically equal. The solution for such a problem can be presented in the form of a Pareto set.

One of the techniques to attain the Pareto set is using weighted sum of the objectives [1–3], which can be optimized by standard single objective optimization algorithms. However, there are many ways to define the weighted sum function and to determine the proper coefficients, relying on experts opinion is fundamental [4]. Another alternative is using approaches based on Multi-objective Evolutionary Algorithms (MOEAs) [5–7], but the number of required function evaluations often is very high. This represents a clear limitation when optimizing engineering systems, whose performance are typically analyzed via computationally expensive and time-consuming simulations.

In this framework, surrogate-based optimization is a popular approach [8]: the idea is to approximate the desired design objective using a data-driven surrogate model. everal models are commonly used, including but not limited to Gaussian processes [9–11], Neural networks [12, 13], Polynomial chaos expansion [14, 15], and Tree-structured Parzen estimator [16] . Such model is built based on a limited number of (expensive) simulations and is cheap-to-evaluate. Examples include optimization of electronic circuits performance [17–19], the shape of airplane components [20–22], and the strength of adhesive joints [23].

A popular technique for surrogate-based optimization is Bayesian Optimization (BO) based on a Gaussian Process (GP) as surrogate model. The technique [24] proposes fast and efficient hypervolume-based BO for multi-objective problems. However, the technique [24] implicitly assumes that all of the objectives can be evaluated with the same computational cost.

This condition does not always hold in multi-objective optimization problems. Typically, some objectives are cheap to compute. For example, the footprint of electrical devices [25] and time of gluing in the adhesive bonding case [26]. Many multi-objective BO techniques [27–29] do not take into account the objectives with very cheap computational cost. In practice, cheap objectives are often modeled with the same surrogate model cost as the expensive ones. This can be an unnecessary burden to the optimization process.

In this work, we extend the standard hypervolume-based acquisition functions to deal with cheap-to-evaluate functions. More specifically, we focus on the bi-objective case, which can be easily extended later on. Instead of modeling the cheap function with a GP, we directly integrate it in the hypervolume-based acquisition functions. We derive the formula analytically resulting in two hypervolume-based acquisition functions: Cheap Hypervolume Expected

Improvement (CHVEI) and Cheap Hypervolume Probability of Improvement (CHVPOI).

For evaluating its performance, we consider four analytical benchmark functions [30], and two realistic design problems in microwave engineering. We show that the proposed method performs better than state-of-the-art approaches. Since the cheap objective is computed directly, the inaccuracies introduced by modeling are eliminated.

This paper is organized as follows: Section 2 introduces the GP probabilistic model and BO. Section 3 presents the hypervolume-based bi-objective BO. Our extension to the hypervolume-based acquisition function is described in Section 4. Then, Section 5 presents relevant experimental results on benchmark functions and realistic design problems. Finally, conclusions are drawn in Section 6.

2 Bayesian Optimization

2.1 Optimization procedure

In global optimization, the goal is to find an optimizer $\mathbf{x}^*$ of an unknown objective function $f(\mathbf{x})$, which can be mathematically described as:

$$\mathbf{x}^\star = \arg\max_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \tag{1}$$

where $\mathcal{X} \in \mathbb{R}^d$ is the design space. The unknown objective function $f(\mathbf{x})$ typically does not have gradient information and is very expensive to evaluate, for example in terms of time or economic cost. Thus, a data-efficient algorithm to find $\mathbf{x}^*$ is desired.

BO is a global optimization method that aims to minimize the number of function evaluations needed to estimate the global optimum of a function. It relies on two elements: a model of the objective function and a sequential sampling strategy. The idea is to iteratively refine the model until the solution to the optimization problem can be found. The sampling strategy relies on the model to estimate which data point should be acquired next. In order to do so, the sampling strategy relies on a function called acquisition function. The acquisition function balances the trade-off between exploration and exploitation, based on the surrogate model. Usually, the acquisition function has an analytical form that can be computed easily [10].

Before running the BO routine, we first need to generate initial points to train the model. In this paper, the Latin Hypercube Design (LHD) [31] is used. Then, the acquisition function is optimized based on the trained model. After a new point is selected, it is evaluated on the true objective function and the result is used to update the surrogate model. These steps are repeated until a suitable stopping criterion is met. The flowchart of BO is shown in Fig. 1.
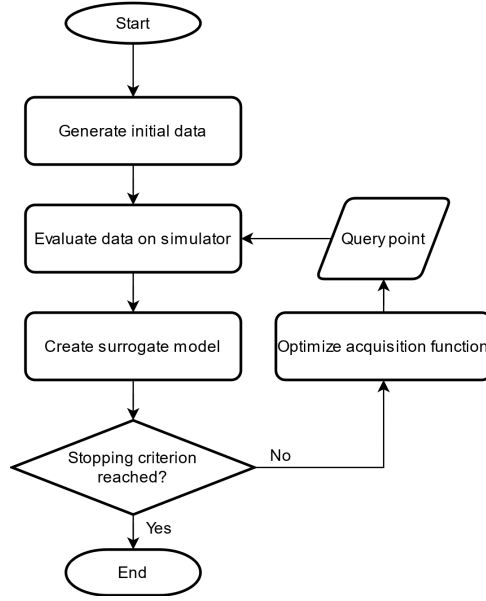
Fig. 1: General flowchart of Bayesian optimization. The two key components are the surrogate model and the acquisition function. The query point from the previous iteration is added to the surrogate model. Thus, the samples in the dataset are increased sequentially.

2.2 Gaussian Processes

The most common choice for the surrogate model for BO is a GP. It is analytically tractable and provides a predictive distribution given new input data. In a more formal definition, GP defines a prior over functions $f(\mathbf{x}) \sim GP(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$.

GP is fully specified by its mean function $m(\mathbf{x})$ and positive semi-definite covariance function $k(\mathbf{x}, \mathbf{x}')$. Following previous work [24], we assume a zero mean function and train the GP on a set of data by maximizing the likelihood using the L-BFGS algorithm. The predictive distribution with a zero mean function of new data $X_\star = [\mathbf{x}_{\star 1}, \ldots, \mathbf{x}_{\star N}]$ can be calculated using:

$$\mu\left(X_\star\right) = \mathbb{E}\left(\mathbf{f}_\star \mid X_\star, \mathcal{D}_n\right) = K_{\star \mathrm{x}} K_{\mathrm{xx}}^{-1} \mathbf{y} \tag{2}$$

$$\sigma^2\left(X_\star\right) = \mathrm{Var}\left(\mathbf{f}_\star \mid X_\star, \mathcal{D}_n\right) = K_{\star\star} - K_{\star \mathrm{x}} K_{\mathrm{xx}}^{-1} K_{\star \mathrm{x}}^T \tag{3}$$

where $\mathcal{D}_n$ is the observed data, $\mu\left(X_\star\right)$ is the predictive mean, and $\sigma^2\left(X_\star\right)$ is the predictive variance, and $K_{\mathrm{xx}} = k(\mathbf{x}_i, \mathbf{x}_j)$, $K_{\star \mathrm{x}} = k(\mathbf{x}_{\star i}, \mathbf{x}_j)$, $K_{\star\star} = k(\mathbf{x}_{\star i}, \mathbf{x}_{\star j})$. For the covariance function, the Matérn 5/2 kernel [32] is used and defined as:

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \alpha \left(1 + \sqrt{5}r + \frac{5}{3}r^2\right) \exp(-\sqrt{5}r), \ r = \left\|\sum_{m=1}^{d} \frac{(x_m - x'_m)^2}{l_m^2}\right\|_2 \quad (4)$$

This kernel is chosen as it does not put strong smoothness assumptions on the target function compared to the other kernels [33]. Thus, it is more suitable for real-world cases such as the engineering design problems.

3 Multi-objective Bayesian Optimization

3.1 Pareto Optimality

In real life optimization problem, typically there are multiple conflicting objectives. This leads to solutions that cannot be improved in any of the objectives without sacrificing at least one of other objectives. These solutions are called Pareto optimal solutions, represented as a Pareto set [34].

For minimization problems with m objectives, the notation $\mathbf{x}_b \succ \mathbf{x}_a$ means that $\mathbf{x}_b$ dominates $\mathbf{x}_a$ if, and only if $f_j\left(\mathbf{x}_b\right) \leq f_j\left(\mathbf{x}_a\right), \forall j \in \{1, .., m\}$ and $\exists j \in \{1, .., m\}$ such that $f_j\left(\mathbf{x}_b\right) < f_j\left(\mathbf{x}_a\right)$. In other words, $\mathbf{x}_b$ is not worse than $\mathbf{x}_a$ in all objectives and better in at least one objective. The Pareto set can then be defined by:

$$\boldsymbol{P} = \left\{\mathbf{x} \in \mathbb{R}^d \mid \nexists \mathbf{x}' \in \mathbb{R}^d : \mathbf{x}' \succ \mathbf{x}\right\} \quad (5)$$

where m is the number of the objectives. Mathematically, the points inside the resulting Pareto set are equal. Also we denotes the Pareto front, the Pareto optimal solutions in output space as $\mathcal{P}$. In practice, after the optimal Pareto set is obtained, the decision makers can choose which point to use based on their preference.

3.2 Multi-objective hypervolume-based acquisition function

In multi-objective optimization problems, instead of calculating the improvement towards a single maximum, we want to get the improvement over the Pareto set $\boldsymbol{P}$. For the hypervolume-based acquisition function, this improvement can be calculated using the hypervolume indicator $\mathcal{H}(\mathcal{P})$. This indicator denotes the volume of the dominated region, bounded by a reference point r which needs to be dominated by all points in $\mathcal{P}$ [35]. The contribution of new points $\mathbf{y}$ to $\mathcal{P}$ can be estimated by using the exclusive hypervolume (also called hypervolume contribution) $\mathcal{H}_{exc}$ as:

$$\mathcal{H}_{exc}(\mathbf{y}, \mathcal{P}) = \mathcal{H}(\mathcal{P} \cup \{\mathbf{y}\}) - \mathcal{H}(\mathcal{P}) \quad (6)$$

Using $\mathcal{H}_{exc}$, we can define the improvement function for the hypervolume-based multi-objective case as:

$$I(\mathbf{y}, \mathcal{P}) = \begin{cases} \mathcal{H}_{\text{exc}}(\mathbf{y}, \mathcal{P}) & \text{if } \mathbf{y} \text{ is not dominated by } \mathcal{P} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Next, we will build the acquisition function for multi-objective settings upon this improvement function. In order to make the formula simpler, given the predictive distribution defined in equations (2) and (3), the probability density function ϕ_j and cumulative density function Φ_j are compactly defined as:

$$\phi_j [y_j] := \phi_j [y_j; \mu_j(\mathbf{x}), \sigma_j(\mathbf{x})] \tag{8}$$
$$\Phi_j [y_j] := \Phi_j [y_j; \mu_j(\mathbf{x}), \sigma_j(\mathbf{x})] \tag{9}$$

Then, the hypervolume-based multi-objective POI (HVPOI) [24] is defined as follows:

$$HVPOI[I] = I(\mu(\mathbf{x}), \mathcal{P}) \int_{\mathbf{y} \in A} \prod_{j=1}^{m} \phi_j [y_j] \, dy_j \tag{10}$$

where $\mu(\mathbf{x})$ is a GP prediction at $\mathbf{x}$ and A is the non-dominated region, see Fig. 2. m is the number of objectives, $\mathbf{y}$ is the objective vector inside region A.

Furthermore, we can define the hypervolume-based EI (HVEI) [24] as:

$$HVEI[I] = \int_{\mathbf{y} \in A} I(\mathbf{y}, \mathcal{P}) \prod_{j=1}^{m} \phi_j [y_j] \, dy_j \tag{11}$$

Note that these acquisition functions are intractable. To mitigate this problem, Couckuyt et al. [24] suggests calculating the hypervolume from the set of disjoint cells built from the lower and upper bound of the Pareto front. This approach is more computationally efficient compared to the uniform grid search [36].

However, these multi-objective acquisition functions implicitly assume that the models of all objectives must be computed, regardless of the computational cost of evaluating each objective. A problem might arise using this assumption: If we have a cheap but complex objective function, the GP might introduce inaccuracies as well as a waste of computational resources.

4 Cheap-expensive hypervolume-based acquisition function

Without loss of generality, let us consider problems where two objective functions can be defined, where one is expensive f_1 and the other is cheap f_2. Our extended approach models f_1 into a GP and uses it to predict new data $\mathbf{x}^*$

and estimate the corresponding mean μ_1 and variance σ_1^2 in the acquisition function calculation. Next, $\mathbf{y} := (y_1, y_2)$ is used to compute the improvement function $I(\mathbf{y}, \mathcal{P})$, where y_1 and y_2 are potential observation and observation of f_1 and f_2, respectively. Here, $\mathcal{P} = \{(p_1^1, p_2^1), (p_1^2, p_2^2), \ldots, (p_1^M, p_2^M)\}$, where p_1^m denotes the expensive dimension sorted in ascending order. Then it follows that p_2^m, the cheap dimension, is sorted in descending order. Next, the Cheap Hypervolume EI (CHVEI) is defined as follows:

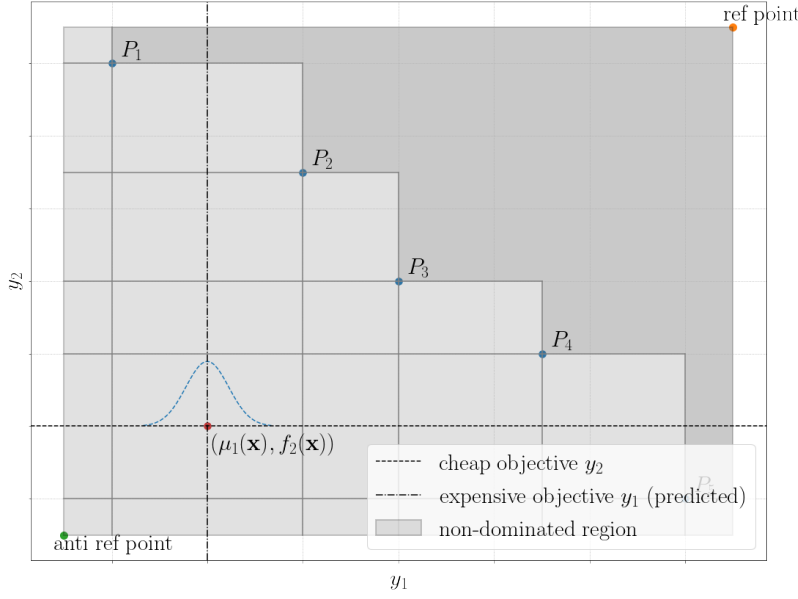$$CHVEI(x) = \int_{(y) \in A} I(\mathbf{y}, \mathcal{P}) \phi(y_1) dy_1 \tag{12}$$

Fig. 2: Illustration of the non-dominated region and the way the cheap objective function evaluation y_2 is incorporated into the hypervolume-based acquisition function. P_i are the points in the Pareto set. The blue dotted curve illustrates the prediction of the expensive objective y_1. The orange dot is the reference point, used as the bound to calculate the hypervolume.

Using $\mathcal{P}$ we define horizontal and vertical cells as shown in Fig. 2. Based on these cells, we can derive the closed form of the CHVEI as follows:

$$CHVEI(x) = \sum_m \sum_k \left(u_2^k - y_2\right)^+ \int_{l_1^m}^{u_1^m} \left(u_1^k - \max\left(l_1^k, l_1^m\right)\right) \phi(y_1) dy_1 \tag{13}$$

where m is the cell of interest, k is the cell improvement relative to m, l is lower bound, u is upper bound and $z^+ = \max(z, 0)$. Then, we can calculate the closed form solution of CHVEI. For cells of interest ($m = k$), the integrals on equation 13 are the definition of the single-objective EI [37] defined as:

$$\left(u_1^k - \mu_1(\mathbf{x})\right)\left(\Phi[u_1^m] - \Phi[l_1^m]\right) + \sigma_1^2(\mathbf{x})\left(\phi[u_1^m] - \phi[l_1^m]\right) \tag{14}$$

while the integral for cells on the right of the cells of interest ($l_1^k \geq u_1^m$) are calculated by:

$$\left(u_1^k - l_1^k\right)\left(\Phi\left[u_1^m\right] - \Phi\left[l_1^m\right]\right) \tag{15}$$

and 0 for cells in the left ($u_1^k < l_1^m$).

It is important to mention that the cheap objective is directly incorporated in the acquisition function, thus avoiding the inaccuracies due to modeling the cheap objective with a surrogate model. This is favorable especially when the cheap objective function has a complex dynamic behaviour in the design space. Additionally, it leads to a reduced computational complexity of the overall algorithm. The proposed BO approach based on CHVEI is summarized in Algorithm 1.

Algorithm 1: Cheap Hypervolume Expected Improvement

Input: Predefined evaluation budget $t_{max} \in \mathbb{Z}$;
Output: X^{nd}, Y^{nd};
Generate k initial points $X^k = [x_1, \ldots, x_k]$;
Evaluate it on expensive function $Y_1^k = [f_1(x_1), \ldots, f_1(x_k)]$;
Set $t = 0$;
while $t < t_{max}$ **do**
    Update GP with X_t, Y_1^t;
    Evaluate X^t on cheap function $Y_2^t = [f_2(x_1), \ldots, f_2(x_t)]$;
    $Y^t \leftarrow Concat(Y_1^t, Y_2^t)$;
    Get the non dominated set X^{nd}, Y^{nd} among X^t, Y^t;
    $t \leftarrow t + 1$;
    $\mathcal{P} \leftarrow Y^{nd}$;
    $x_t = \text{argmax}_{x \subset \mathcal{X}} \text{CHVEI}(x)$;
    $y_{1t} = f_1(x_t)$;
    $X^t = X^{(t-1)} \circ x_t; Y_1^t = Y_1^{(t-1)} \circ y_{1t}$
end
return X^{nd}, Y^{nd} among X^t, Y^t

A similar approach can be used to define the cheap version of bi-objective POI as follows:

$$CHVPOI(x) = \sum_{m=1}^{M-1} (p_2^m - y_2)^+ (p_1^m - y_1)^+ \left(\Phi_1\left[p_1^{(m+1)}\right] - \Phi_1\left[p_1^m\right]\right) \tag{16}$$

Now, a BO routine based on CHVPOI can be defined as for the CHVEI: the only difference with the approach described in Algorithm 1 is that equation (16) is used instead of CHVEI(x).

5 Result and discussion

The proposed BO approach is implemented using the GPFlowOpt library [38] in python. For the initial data, 21 points are arranged using a Latin-Hypercube Design [31]. The acquisition function is optimized using the best solution of Monte Carlo sampling, further fine-tuned by a gradient-based optimizer. We consider four variants of the well known DTLZ benchmark functions [30] and two real-life electrical device design optimization problems to validate the proposed method.

In all experiments, the CHVEI and CHVPOI acquisition functions are compared with the standard HVEI and HVPOI, as well as random sampling. Additionally, we also compare against the MOEA algorithms SMS-EMOA [39] and NSGA-II [40]. We used hypervolume indicator metrics to assess the quality of the Pareto set per iteration, where the computational budget is fixed to 100 function evaluations for all methods except the MOEA, that uses a budget of 250 function evaluations instead. This choice guarantees a fair comparison, since MOEA needs more evaluations compared to approaches based on BO. Finally, when optimizing electrical devices we inspect the quality of the Pareto set by checking the design layout and its corresponding responses visually.

5.1 DTLZ benchmark functions

We consider four variants of the DTLZ functions: DTLZ1, DTLZ2, DTLZ5, and DTLZ7, as indicated in Table 1. DTLZ2 and DTLZ5 have a smooth set of Pareto solution, while DTLZ1 and DTLZ7 have a disjoint Pareto set.

Table 1: We configure the DTLZ benchmark functions for 5 dimensional inputs, 2 outputs, and a fixed reference point r.

Function	input dimension	Reference point r
DTLZ1	5	(350, 350)
DTLZ2	5	(2.5, 2.5)
DTLZ5	5	(2.5, 2.5)
DTLZ7	5	(20, 20)

The final hypervolume indicator is shown in Table 2. Overall results show that CHVPOI always performs better than the other methods, even compared to the MOEAs with 250 function evaluations budget.

Figure 3 shows the hypervolume indicator evolution with respect to the function evaluations number: the CHVEI and CHVPOI achieve a better hypervolume indicator in less iterations compared with their standard counterpart HVEI and HVPOI, respectively. Additionally, CHVEI and CHVPOI offer the best performance compared with all other methods for DTLZ1 and DTLZ7, while standard HVPOI is better than CHVEI in iteration 100 for the functions

DTLZ2 and DTLZ5, that have a smooth Pareto solution. This is because EI based methods are less exploitative compared to the POI based methods.

Table 2: Hypervolume with 95% confidence interval of the DTLZ benchmark experiments. Best results for each problem are highlighted in bold.

Test Problem	Budget	Method	Hypervolume	
DTLZ1	100	Random	1.1854e5	± 4.8133e2
	100	HVPOI	1.1972e5	± 4.1228e2
	100	HVEI	1.1921e5	± 5.4444e2
	100	CHVPOI	**1.2239e5**	± **4.1724e1**
	100	CHVEI	1.2208e5	± 1.1495e2
	250	SMSEMOA	1.2155e5	± 6.9861e1
	250	NSGA2b	1.2152e5	± 4.6356e1
DTLZ2	100	Random	5.2191	± 0.0192
	100	HVPOI	5.4211	± 0.0025
	100	HVEI	5.3278	± 0.0094
	100	CHVPOI	**5.4472**	± **0.0014**
	100	CHVEI	5.3912	± 0.0057
	250	SMSEMOA	5.3702	± 0.0126
	250	NSGA2b	5.3652	± 0.0226
DTLZ5	100	Random	5.2178	± 0.0214
	100	HVPOI	5.4188	± 0.0010
	100	HVEI	5.3207	± 0.0172
	100	CHVPOI	**5.4478**	± **0.0007**
	100	CHVEI	5.3725	± 0.0240
	250	SMSEMOA	5.3698	± 0.0140
	250	NSGA2b	5.3831	± 0.0135
DTLZ7	100	Random	2.8146e2	± 6.7193
	100	HVPOI	3.2672e2	± 3.3173
	100	HVEI	3.1370e2	± 8.0552
	100	CHVPOI	**3.5191e2**	± **0.1175**
	100	CHVEI	3.3798e2	± 7.2954
	250	SMSEMOA	3.5087e2	± 2.3901
	250	NSGA2b	3.4867e2	± 4.8165

Furthermore, we randomly sample 1 million points for all the benchmark functions to approximate the true Pareto set of the functions. Next, we calculate the distance of the sampled-based Pareto set and the Pareto set obtained via BO-based approaches and MOEAs. This metric is used to measure the convergence of the results with respect to approximate "true" Pareto set: the lower the value, the more accurate the approximation. It is defined as:

$$CM(P, \hat{P}) = \frac{1}{n} \sum_{j}^{n} \min_{i} ||P_i - \hat{P}_j||_2 \qquad (17)$$

$\hat{P}$ is a vector of Pareto front obtained by the algorithm, P is an approximation of true Pareto front obtained, for instance, by random sampling.

(a) DTLZ1

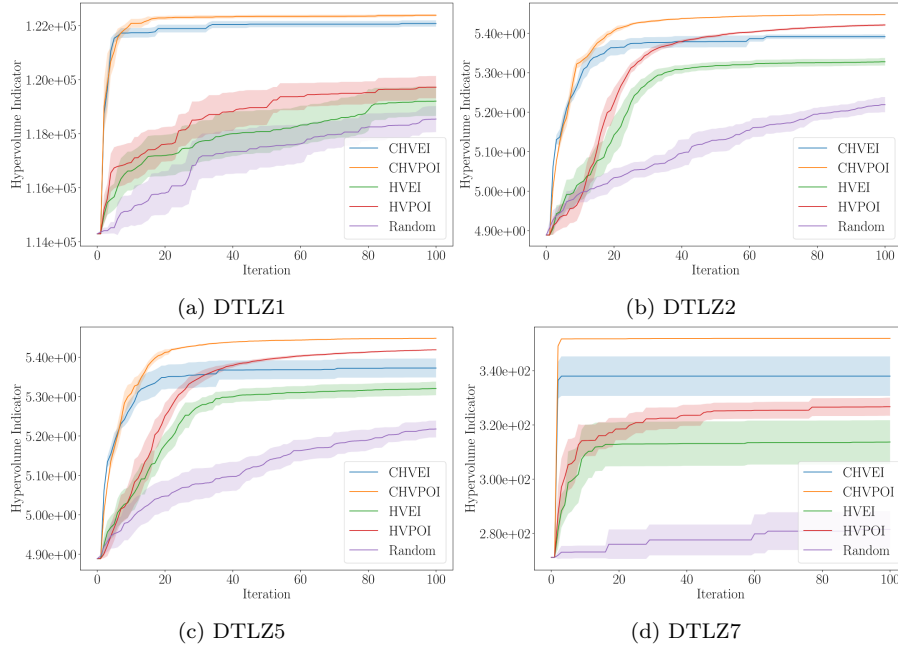(b) DTLZ2

(c) DTLZ5

(d) DTLZ7

Fig. 3: Evolution of hypervolume indicator for the DTLZ benchmark functions

The convergence measure in Table 3 shows that CHVPOI is better in most cases except on DTLZ7. As we see in Fig. 3d, the hypervolume indicator does not improve much after iteration 10, indicating that the method is finding the extrema faster, which results in a less uniform Pareto set. This is prevalent in hypervolume-based improvement functions: since hypervolume is a product, it samples less in regions where at least one objective has a very small improvement.

Table 3: The convergence measure with 95% confidence interval of the DTLZ benchmark experiments. The distance between the true Pareto front and the Pareto set as generated by the algorithm. Best results for each problem are highlighted in bold.

Test Problem	Method	Convergence Measure		
DTLZ1	Random	21.3342	±	1.4618
	HVPOI	19.1453	±	1.3509
	HVEI	20.9401	±	1.5363
	CHVPOI	**13.2936**	±	1.1313
	CHVEI	16.9271	±	1.0657
	SMSEMOA	18.7077	±	0.9536
	NSGA2b	18.8002	±	**0.6227**
DTLZ2	Random	0.1163	±	0.0085
	HVPOI	0.0227	±	0.0011
	HVEI	0.1150	±	0.0104
	CHVPOI	**0.0203**	±	**0.0008**
	CHVEI	0.0811	±	0.0058
	SMSEMOA	0.0698	±	0.0183
	NSGA2b	0.0607	±	0.0138
DTLZ5	Random	0.1195	±	0.0108
	HVPOI	0.0252	±	0.0011
	HVEI	0.1264	±	0.0175
	CHVPOI	**0.0202**	±	**0.0005**
	CHVEI	0.0915	±	0.0146
	SMSEMOA	0.0617	±	0.0084
	NSGA2b	0.0551	±	0.0098
DTLZ7	Random	1.7510	±	0.1888
	HVPOI	**0.6931**	±	0.0396
	HVEI	0.9274	±	0.1595
	CHVPOI	1.2566	±	0.0321
	CHVEI	1.3335	±	**0.0045**
	SMSEMOA	1.8680	±	0.0127
	NSGA2b	1.8729	±	0.0297

5.2 Microstrip low-pass filter

Our first engineering problem is the design of a two-port low-pass stepped impedance microstrip filter device [41]. The simulator for the device is implemented in the MATLAB RF Toolbox (Mathworks Inc., Natick, MA, USA). The corresponding layout is presented in Fig. 4.

The filter is a cascade of 6 microstrip lines, each specified by width and length, where by design $w1 = w3 = w5$ and $w2 = w4 = w6$. The cross-section view of each microstrip is depicted in Fig. 5.

Our target design is a filter with a 3 dB cut-off frequency at 2.55 GHz. To achieve this, we define the design goals as follows:

$$|S_{21}| \geq -3\,\mathrm{dB} \quad \mathrm{for} \quad 0\,\mathrm{GHz} \leq freq \leq 2.55\,\mathrm{GHz} \tag{18}$$

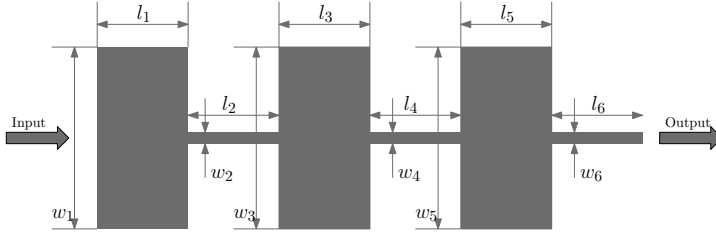$$|S_{21}| \leq -3\,\mathrm{dB} \quad \mathrm{for} \quad 2.55\,\mathrm{GHz} \leq freq \tag{19}$$

Fig. 4: Top-view of microstrip low-pass filter. We use 1 widths $w_1 = w_3 = w_5$ and 3 different lengths l_1, l_3, l_5 as design parameters.
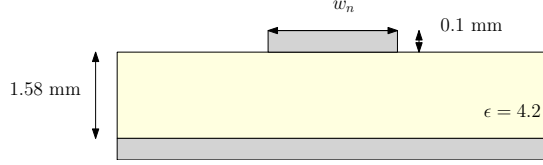

Fig. 5: The cross-section view of each microstrip. For $w_n = w_1 = w_3 = w_5$, the values changes within the optimization iterations, while for $w_n = w_2 = w_4 = w_6$ the values are fixed at 0.428 mm.

where $|S_{21}|$ is the magnitude of the element S_{21} of the scattering matrix, and $freq$ is the frequency. Visually, the target response is illustrated in Fig. 6. We also want to optimize the cost to produce the device, by minimizing the footprint of the filter. Hence, the target design and the area of the filter are used as our expensive and cheap objective, respectively.
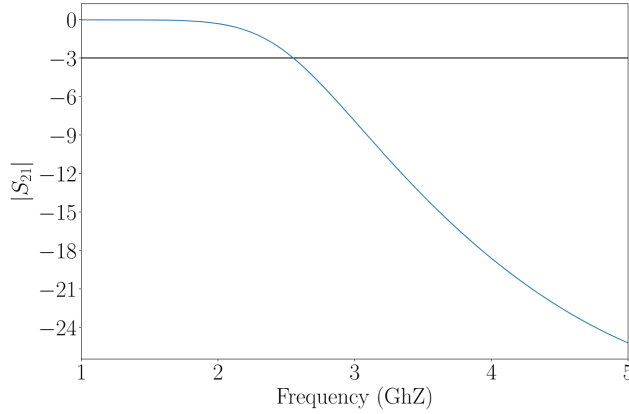

Fig. 6: Example of the desired response for the microstrip low-pass filter.

In the optimization problem formulation, the chosen design parameters are the length and the width of the first, third and fifth microstrips, indicated as l_1, l_3, l_5, w_1, w_3, w_5, (see Fig. 4). Note that, these microstrips have the same width by design [41]: the width of all three microstrips is one single design parameter indicated as $w_{1,3,5}$. The other geometrical and electrical characteristics of the filter are chosen according to Table 4.

Table 4: Microstrip low-pass filter design parameters.

Parameter	Description	Range/Value
$w_{1,3,5}$	Width of microstrip 1, 3, and 5	$[5.6, 16.9]$ mm
$w_{2,4,6}$	Width of microstrip 2, 4, and 6	0.428 mm
l_1	Length of microstrip 1	$[1.05, 3.05]$ mm
l_2	Length of microstrip 2	6.63 mm
l_3	Length of microstrip 3	$[6.69, 8.69]$ mm
l_4	Length of microstrip 4	9.04 mm
l_5	Length of microstrip 5	$[4.63, 6.63]$ mm
l_6	Length of microstrip 6	2.41 mm
ϵ	Relative permittivity	4.2
h	Thickness	1.58 mm

To achieve our optimization goals, we formulate our objectives as follows:

$$f_1 = -\min_{freq \in [1, f_{pass}]} S_{21}(freq) + \max_{freq \in [f_{stop}, 5]} S_{21}(freq) \qquad (20)$$

$$f_2 = \log(\sum_{n=1}^{3} w_{1,3,5} \, l_{2n-1}) \qquad (21)$$

In the first objective defined in equation (20), we want a response that assumes values as high as possible until f_{pass}, and as low as possible for frequencies above f_{stop}. This ensures that our device has a low-pass filter behavior, as shown in Fig. 6. The second cheap objective expressed in equation (21) represents the log of sum of the three microstrips's area, this means that the goal is to minimize the footprint of the filter. The log term is to ensure numerical stability of the second objective. We solve this problem with our proposed methods. The hypervolume per iteration of all methods is shown in Fig. 7.

Comparison of the hypervolume indicator of all methods for a predefined computational budget is presented in Table 5. CHVPOI performs better compared to the other benchmark methods, while CHVEI performs worse in terms of hypervolume indicator compared to HVEI. To check this unexpected behavior, we evaluate the quality of the pareto set by using the same convergence measure adopted for the DTLZ functions. In particular, the Pareto set is approximated via 50000 evaluations on random points in the design space and the distance metric defined in equation (17) is computed for CHVEI and HVEI.
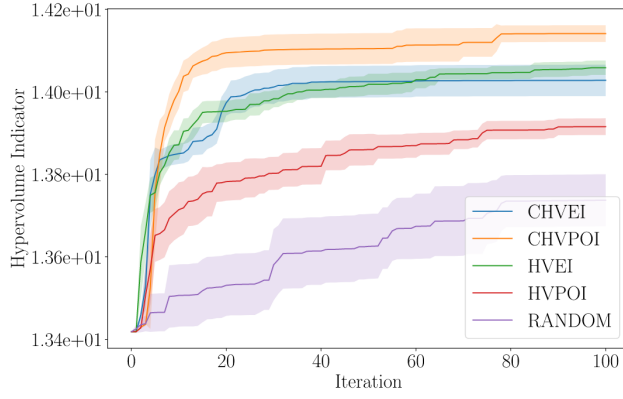
Fig. 7: Hypervolume indicator evolution for low-pass filter case. The hypervolume is calculated using $(1, 0)$ as the reference point.

The result show that CHVEI gives a better convergence measure (0.0127 ± 0.0009) than the HVEI (0.0468 ± 0.0027). This means that the CHVEI yields a Pareto front that is spread more evenly along the approximated true front than the HVEI. The full convergence measure results are described in Table 5.

Table 5: Hypervolume and Convergence Measure with 95% confidence interval of the low-pass filter example experiments.

Method	Budget	Hypervolume			Convergence Measure		
Random	100	13.7372	$\pm$	0.0629	0.0341	$\pm$	0.0023
HVPOI	100	13.9156	$\pm$	0.0203	0.0105	$\pm$	0.0011
HVEI	100	14.0585	$\pm$	**0.0173**	0.0468	$\pm$	0.0027
CHVPOI	100	**14.1413**	$\pm$	0.0203	**0.0095**	$\pm$	**0.0003**
CHVEI	100	14.0279	$\pm$	0.0381	0.0127	$\pm$	0.0009
SMSEMOA	250	14.0988	$\pm$	0.0257	0.0561	$\pm$	0.0014
NSGA2b	250	14.0889	$\pm$	0.0188	0.0550	$\pm$	0.0013

5.3 Tapped-line filter

The second engineering example is a tapped-line filter [42, 43], implemented in the Advanced Design System simulator (Keysight EEsof EDA). The full layout of this device is shown in Fig. 8.

The design requirements for this filter are described in equation (22) and (23), as:
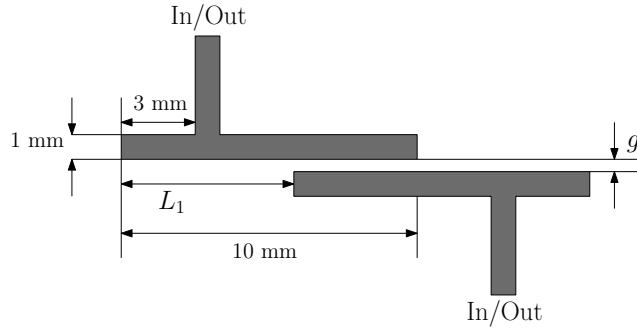
Fig. 8: Top-view of Tapped-line Filter. The two conductors (in gray) are placed on a dielectric substrate [19, 20].

$$|S_{21}| \geq -3\,\text{dB} \qquad \text{for } 4.75\,\text{GHz} \leq freq \leq 5.25\,\text{GHz} \qquad (22)$$
$$|S_{21}| \leq -20\,\text{dB} \qquad \text{for } 3.0\,\text{GHz} \leq freq \leq 4.0\,\text{GHz}$$
$$\text{and } 6.0\,\text{GHz} \leq freq \leq 7.0\,\text{GHz} \qquad (23)$$

where $|S_{21}|$ is the magnitude of the element S_{21} of the scattering matrix, and $freq$ is the frequency. The requirements in equations (22) and (23) lead to the to response depicted in Fig. 9: the desired filter response is lower than $-20\,\text{dB}$ in the low and high frequency parts (called stopbands) and higher than $-3\,\text{dB}$ in the mid part (passband of the filter).
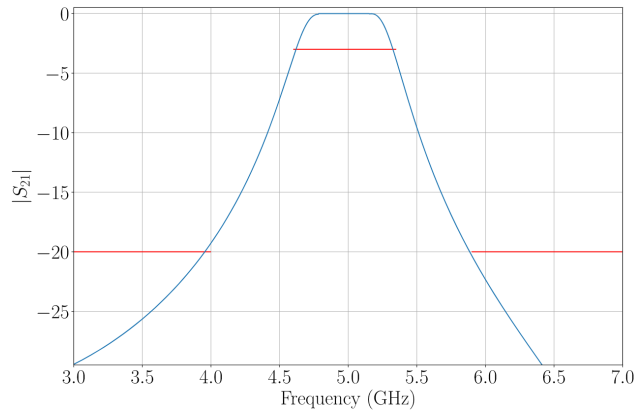

Fig. 9: The target response of the tapped-line filter. The red horizontal lines indicate the values -3dB and -20dB in equations (22) and (23).

The design requirements shown above are used as the first optimization goal. For the second cheap objective, the footprint of the device is used. We formulate these objectives as follows:

$$f_1 = \max_{freq \in [3, fs_1]} S_{21}(freq) - \min_{freq \in [fp_1, fp_2]} S_{21}(freq) \times 10 \qquad (24)$$
$$+ \max_{freq \in [fs_2, 7]} S_{21}(freq)$$
$$f_2 = (L_1 + 10) \times (4 \times 2 + g) \qquad (25)$$

where $fs_1 = 4\,\text{GHz}$, $fp_1 = 4.75\,\text{GHz}$, $fp_2 = 2.25\,\text{GHz}$ and $fs_2 = 6\,\text{GHz}$. Equation (24) ensures that the filter's response follows the design requirements. To balance the value, we put a higher weight for the response in the passband. Equation (25) represents the footprint of the device.

The variables for the optimization are the displacement L_1 (mm) and the spacing g (mm) between the two conductors. Additionally, the dielectric constant ϵ (mil), and height h (mm) of the substrate are also considered as design parameters. The corresponding design space is shown in Table 6.

Table 6: Tapped line filter design parameters.

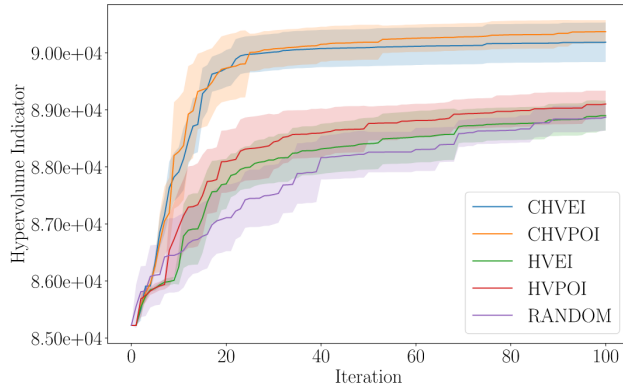| Parameter	Description	Range/Value
L_1	Displacement between two geometries	$[4, 10]$ mm
g	Space between two geometries	$[0.02, 0.1]$ mm
ϵ	Dielectric constant	$[8, 11]$ mil
h	Height of the device	$[0.2, 0.4]$ mm

Fig. 10: Evolution of hypervolume for tapped-line filter case. (20, 20) is used as the reference point to calculate the hypervolume.

Using these settings, we run the optimization with our proposed methods and other benchmark methods. The results in Fig. 10 show that CHVEI and CHVPOI get a higher hypervolume indicator faster than the other methods. Additionally, in Table 7 we can see that our methods have higher hypervolume indicator compared to the MOEAs: SMSEMOA and NSGA2b, even with a lower function evaluation budget.

Table 7: Hypervolume with 95% confidence interval of the tapped-line filter example experiments.

Method	Budget	Hypervolume		
Random	100	8.8870e4	±	2.3087e2
HVPOI	100	8.9102e4	±	2.3402e2
HVEI	100	8.8898e4	±	2.6408e2
CHVPOI	100	**9.0369e4**	±	**2.0433e2**
CHVEI	100	9.0184e4	±	3.4530e2
SMSEMOA	250	8.9854e4	±	2.8950e2
NSGA2b	250	8.9928e4	±	3.1912e2

6 Conclusion

We proposed the Cheap Hypervolume Expected Improvement (CHVEI) and Cheap Hypervolume Probability of Improvement (CHVPOI) which can directly exploit cheap-to-evaluate objective functions. The direct evaluation can speed up the optimization process and removes possible inaccuracies introduced by surrogate modeling. To evaluate the performance of the proposed method, we apply our algorithm to multiple benchmark functions and two engineering design problems. We evaluate the performance of the CHVEI and CHVPOI by measuring the hypervolume indicator and convergence measure at each iteration. It is shown that in the engineering design problems our proposed methods outperform the standard hypervolume-based methods, random sampling, and Genetic algorithm-based methods. In future works we will extend the algorithm for $n > 2$ objectives, and consider the case of noisy objective functions.

References

1. Joshua Knowles. ParEGO: A hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE*

Transactions on Evolutionary Computation, 10(1):50–66, 2006. ISSN 1089778X. doi: 10.1109/TEVC.2005.851274.

2. Joshua Knowles, David Corne, and Alan Reynolds. Noisy multiobjective optimization on a budget of 250 evaluations. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5467 LNCS:36–50, 2010. ISSN 03029743. doi: 10.1007/978-3-642-01020-0_8.

3. Joan Davins-Valldaura, Saïd Moussaoui, Guillermo Pita-Gil, and Franck Plestan. ParEGO extensions for multi-objective optimization of expensive evaluation functions. *Journal of Global Optimization*, 67(1-2):79–96, 2017. ISSN 15732916. doi: 10.1007/s10898-016-0419-3.

4. Raul Astudillo and Peter I Frazier. Multi-Attribute Bayesian Optimization under Utility Uncertainty. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, page 5, 2017.

5. Carlos A. Coello Coello, Silvia González Brambila, Josué Figueroa Gamboa, Ma Guadalupe Castillo Tapia, and Raquel Hernández Gómez. Evolutionary multiobjective optimization: open research areas and some challenges lying ahead. *Complex & Intelligent Systems*, 6(2):221–236, 2020. ISSN 2199-4536. doi: 10.1007/s40747-019-0113-4. URL https://doi.org/10.1007/s40747-019-0113-4.

6. Tapabrata Ray, Kang Tai, and Kian Chye Seow. Multiobjective design optimization by an evolutionary algorithm. *Engineering Optimization*, 33 (4):399–424, 2001. ISSN 0305215X. doi: 10.1080/03052150108940926.

7. Qi Zhou, Jinhong Wu, Tao Xue, and Peng Jin. A two-stage adaptive multi-fidelity surrogate model-assisted multi-objective genetic algorithm for computationally expensive problems. *Engineering with Computers*, 37 (1):623–639, 2021. ISSN 14355663. doi: 10.1007/s00366-019-00844-8. URL https://doi.org/10.1007/s00366-019-00844-8.

8. Wenze Shao, Haisong Deng, Yizhong Ma, and Zhuihui Wei. Extended Gaussian Kriging for computer experiments in engineering design. *Engineering with Computers*, 28(2):161–178, 2012. ISSN 01770667. doi: 10.1007/s00366-011-0229-7.

9. Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2018. doi: 10.7551/mitpress/3206.001.0001.

10. Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando De Freitas. Taking the human out of the loop: A review of Bayesian optimization, 2016. ISSN 15582256.

11. Youwei He, Jinju Sun, Peng Song, and Xuesong Wang. Variable-fidelity hypervolume-based expected improvement criteria for multi-objective efficient global optimization of expensive functions. *Engineering with Computers*, 2021. ISSN 14355663. doi: 10.1007/s00366-021-01404-9. URL https://doi.org/10.1007/s00366-021-01404-9.

12. Gang Sun and Shuyue Wang. A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233

(16):5863–5872, 2019. ISSN 20413025. doi: 10.1177/0954410019864485.

13. M. Shahriari, D. Pardo, B. Moser, and F. Sobieczky. A deep neural network as surrogate model for forward simulation of borehole resistivity measurements. *Procedia Manufacturing*, 42:235–238, 2020. ISSN 23519789. doi: 10.1016/j.promfg.2020.02.075. URL `https://doi.org/10.1016/j.promfg.2020.02.075`.

14. Biswarup Bhattacharyya. Uncertainty quantification and reliability analysis by an adaptive sparse Bayesian inference based PCE model. *Engineering with Computers*, (0123456789), 2021. ISSN 14355663. doi: 10.1007/s00366-021-01291-0. URL `https://doi.org/10.1007/s00366-021-01291-0`.

15. Yicheng Zhou, Zhenzhou Lu, Kai Cheng, and Yan Shi. An expanded sparse Bayesian learning method for polynomial chaos expansion. *Mechanical Systems and Signal Processing*, 128:153–171, 2019. ISSN 10961216. doi: 10.1016/j.ymssp.2019.03.032. URL `https://doi.org/10.1016/j.ymssp.2019.03.032`.

16. James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, pages 1–9, 2011.

17. Mach František. Bayesian approach to design optimization of electromagnetic systems under uncertainty. *IEEE CEFC 2016 - 17th Biennial Conference on Electromagnetic Field Computation*, 15(c):5090, 2017. doi: 10.1109/CEFC.2016.7816315.

18. Nicolas Knudde, Ivo Couckuyt, Domenico Spina, Konstanty Lukasik, Pawel Barmuta, Dominique Schreurs, and Tom Dhaene. Data-Efficient Bayesian Optimization with Constraints for Power Amplifier Design. *2018 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization, NEMO 2018*, pages 1–3, 2018. doi: 10.1109/NEMO.2018.8503107.

19. Passos Fabio, Ye Yinghao, Spina Domenico, Roca Elisenda, Castro-Lopez Rafael, Dhaene Tom, and Vidal Fernández Francisco. Parametric macromodeling of integrated inductors for RF circuit design. *Microwave and Optical Technology Letters*, 59(5):1207–1212, 2017. doi: https://doi.org/10.1002/mop.30498.

20. Cian Conlan-Smith, Néstor Ramos-García, Ole Sigmund, and Casper Schousboe Andreasen. Aerodynamic shape optimization of aircraft wings using panel methods. *AIAA Journal*, 58(9):3765–3776, 2020. ISSN 00011452. doi: 10.2514/1.J058979.

21. Lavi Rizki Zuhal, Pramudita Satria Palar, and Koji Shimoyama. A comparative study of multi-objective expected improvement for aerodynamic design. *Aerospace Science and Technology*, 91(May):548–560, 2019. ISSN 12709638. doi: 10.1016/j.ast.2019.05.044. URL `https://doi.org/10.1016/j.ast.2019.05.044`.

22. Timothy Man Shui Jim, Ghifari Adam Faza, Pramudita Satria Palar, and Koji Shimoyama. Bayesian methods for multi-objective optimiza-

tion of a supersonic wing planform. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, GECCO '20, page 1641–1643, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371278. doi: 10.1145/3377929.3398122. URL https://doi.org/10.1145/3377929.3398122.

23. Lucas F.M. da Silva and Maria João C.Q. Lopes. Joint strength optimization by the mixed-adhesive technique. *International Journal of Adhesion and Adhesives*, 29(5):509–514, 2009. ISSN 01437496. doi: 10.1016/j.ijadhadh.2008.09.009.

24. Ivo Couckuyt, Dirk Deschrijver, and Tom Dhaene. Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization. *Journal of Global Optimization*, 60(3): 575–594, 2014. ISSN 1573-2916. doi: 10.1007/s10898-013-0118-2. URL https://doi.org/10.1007/s10898-013-0118-2.

25. Angada B. Sachid, P. Paliwal, S. Joshi, M. Shojaei, D. Sharma, and V. Rao. Circuit optimization at 22nm technology node. *Proceedings of the IEEE International Conference on VLSI Design*, 25(February 2016): 322–327, 2012. ISSN 10639667. doi: 10.1109/VLSID.2012.91.

26. A. Sánchez Cebrián, R. Basler, M. Zogg, and Paolo Ermanni. Multistep heating to optimizethe curing process of a paste adhesive. In *ECCM 2012 - Composites at Venice, Proceedings of the 15th European Conference on Composite Materials*, 2012. ISBN 9788888785332.

27. Ivo Couckuyt, Dirk Deschrijver, and Tom Dhaene. Towards efficient multiobjective optimization: Multiobjective statistical criterions. *2012 IEEE Congress on Evolutionary Computation, CEC 2012*, pages 10–15, 2012. doi: 10.1109/CEC.2012.6256586.

28. Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Max-value entropy search for multi-objective Bayesian optimization. *Advances in Neural Information Processing Systems*, 32(NeurIPS), 2019. ISSN 10495258.

29. Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. Differentiable expected hypervolume improvement for parallel multi-objective bayesian optimization. *arXiv*, pages 1–30, 2020. ISSN 23318422.

30. Kalyanmoy Deb, Lothar Thiele, Marco Laumanns, and Eckart Zitzler. *Scalable Test Problems for Evolutionary Multiobjective Optimization*. Wiley, 2005. doi: 10.1007/1-84628-137-7_6.

31. Felipe A.C. Viana, Gerhard Venter, and Vladimir Balabanov. An algorithm for fast optimal latin hypercube design of experiments. *International Journal for Numerical Methods in Engineering*, 82(2), 2010. ISSN 00295981. doi: 10.1002/nme.2750.

32. Budiman Minasny and Alex B. McBratney. The Matérn function as a general model for soil variograms. *Geoderma*, 128(3-4 SPEC. ISS.):192–207, 2005. ISSN 00167061. doi: 10.1016/j.geoderma.2005.04.003.

33. Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 4:2951–2959, 2012. ISSN 10495258.

34. Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M. Fonseca, and Viviane Grunert Da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review, apr 2003. ISSN 1089778X.
35. Qingfu Zhang, Wudong Liu, Edward Tsang, and Botond Virginas. Expensive multiobjective optimization by MOEA/D with gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3):456–474, 2010. ISSN 1089778X. doi: 10.1109/TEVC.2009.2033671.
36. Michael T.M. Emmerich, André H. Deutz, and Jan Willem Klinkenberg. Hypervolume-based expected improvement: Monotonicity properties and exact computation. *2011 IEEE Congress of Evolutionary Computation, CEC 2011*, pages 2147–2154, 2011. doi: 10.1109/CEC.2011.5949880.
37. Donald R Jones, Matthias Schonlau, and W. J. Welch. Efficient Global Optimization of Expensive Black-Box Functions,", vol. 13, no. 4, pp. 455-492, 1998. *Journal of Global Optimization*, 13:455–492, 1998.
38. Nicolas Knudde, Joachim Van Der Herten, Tom Dhaene, and Ivo Couckuyt. GPflowOpt: A Bayesian optimization library using TensorFlow. *arXiv*, pages 0–1, 2017. ISSN 23318422.
39. Nicola Beume, Boris Naujoks, and Michael Emmerich. SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007. ISSN 03772217. doi: 10.1016/j.ejor.2006.08.008.
40. Juan J. Durillo, Antonio J. Nebro, Francisco Luna, and Enrique Alba. On the effect of the steady-state selection scheme in multi-objective genetic algorithms. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5467 LNCS:183–197, 2010. ISSN 03029743. doi: 10.1007/978-3-642-01020-0_18.
41. Jixiang Qing, Nicolas Knudde, Ivo Couckuyt, Domenico Spina, and Tom Dhaene. Bayesian active learning for electromagnetic structure design. *14th European Conference on Antennas and Propagation, EuCAP 2020*, pages 1–5, 2020. doi: 10.23919/EuCAP48036.2020.9136051.
42. Alexandre Manchec, Cédric Quendo, Jean Francois Favennec, Eric Rius, and Christian Person. Synthesis of Capacitive-Coupled Dual-Behavior Resonator (CCDBR) Filters. *IEEE Transactions on Microwave Theory and Techniques*, 54(6):2346–2355, 2006. ISSN 15579670. doi: 10.1109/TMTT.2006.875271.
43. Slawomir Koziel, Qingsha S. Cheng, and John W. Bandler. Space mapping. *IEEE Microwave Magazine*, 9(6):105–122, 2008. ISSN 15273342. doi: 10.1109/MMM.2008.929554.

Conflict of interest

The authors declare that they have no conflict of interest.