Preference-Aware Constrained Multi-Objective Bayesian Optimization For Analog Circuit Design

Alaleh Ahmadianshalchi, Syrine Belakaria, Janardhan Rao Doppa
School of EECS, Washington State University
{a.ahmadianshalchi, syrine.belakaria, jana.doppa}@wsu.edu

Abstract

Many analog circuit design optimization problems involve performing expensive simulations to evaluate circuit configurations in terms of multiple objectives and constraints; Oftentimes, practitioners have preferences over objectives. We aim to approximate the optimal Pareto set over feasible circuit configurations by minimizing the number of simulations. We propose a novel and efficient preference-aware constrained multi-objective Bayesian optimization (PAC-MOO) approach that learns surrogate models for objectives and constraints and sequentially selects candidate circuits for simulation that maximize the information gained about the optimal constrained Pareto-front while factoring in the objective preferences. Our experiments on real-world problems demonstrate PAC-MOO's efficacy over prior methods.

1 Introduction

Although there is a powerful set of design automation (DA) tools for digital circuits, the research and development of DA tools for automating analog circuit design is still in its infancy. Traditionally, the input parameters of analog circuits (e.g., lengths and widths of transistors) are tuned manually using the vast experience of human designers for each design specification. However, this manual approach is not practical to meet the time-to-market requirements for analog circuits due to the huge demand for high-performance and low-power analog circuits for rapidly emerging application scenarios, as well as the circuits, becoming more complex due to scaled technology.

The design automation of analog circuits can be naturally formulated as a Multi-Objective Optimization (MOO) problem. The goal is to find optimal *Pareto set* of design parameters to achieve the best trade-offs among multiple conflicting design objectives. There are many challenges in solving such optimization problems. First, the size of the design space is *large* and evaluating each candidate circuit requires performing a *computationally expensive* circuit simulation. Second, there are many constraints to determine the circuit feasibility and practicality, which we cannot evaluate without performing circuit simulations. In many circuit design problems, the fraction of feasible circuit candidates that satisfy all constraints is very small. Third, the designer prefers some objectives more than others (e.g., efficiency over settling time). In such cases, it suffices to find a subset of the optimal Pareto set that meets the specified preferences. The overall goal is to approximate the solution of a given MOO problem specification for analog circuit design by minimizing the number of circuit simulations, thereby accelerating the DA process.

Bayesian optimization (BO) is an efficient framework to solve black-box optimization problems with expensive objective function evaluations [13, 10]. There are some BO algorithms for handling the large design space challenge. However, none of them can handle the constraint and feasibility challenges. To fill this important gap, we propose a novel and efficient information-theoretic approach referred to as **P***reference-***A***ware* **C***onstrained* **M***ulti-***O***bjective Bayesian* **O***ptimization (PAC-MOO)*. PAC-MOO builds surrogate models for both output objectives and constraints based on the training

36th Conference on Neural Information Processing Systems (NeurIPS 2022) Workshop on ML for Systems.

data from past circuit simulations. PAC-MOO employs an acquisition function in each iteration to select a candidate circuit configuration for performing simulation. The selected circuit configuration maximizes the information gain about the constrained optimal Pareto front while factoring in the designer preferences. The experimental results demonstrate that PAC-MOO without any preferences, outperforms baseline methods by finding a high-quality Pareto set of circuit designs in fewer number of simulations. When preferences over objectives are specified, PAC-MOO was able to find circuit configurations with higher preferred objective values as intended by sacrificing the overall hypervolume indicator.

Contributions. Our key contribution is the development and evaluation of the generic information-theoretic algorithm PAC-MOO to efficiently solve a broad class of analog circuit design optimization problems. Specific contributions include: **1.** A tractable acquisition function based on information gain to select candidate circuits for performing simulations, **2.** Approaches to increase the chances of finding feasible circuits and to incorporate preferences over objectives. **3.** Evaluation of PAC-MOO on two challenging analog circuit design problems and comparison with prior methods.

2 Preference-Aware Constrained Multi-Objective Optimization

Circuit design problem as MOO. We are given a design space $\mathfrak{X}$ with d design variables, where each candidate design $\mathbf{x} \in \mathfrak{X}$ is a d-dimensional vector of parameters corresponding to a circuit configuration. We want to optimize $\mathbf{K} \geq 2$ objective functions $\{f_1(x), \cdots, f_{\mathbf{K}}(x)\}$ over the given circuit design space $\mathfrak{X}$. We assume $\mathbf{L}$ black-box constraints of the form $c_1(x) \geq 0, \cdots, c_{\mathbf{L}}(x) \geq 0$ over the design space $\mathfrak{X}$. We can evaluate these constraints for a given design $\mathbf{x}$ by performing a circuit simulation to get $(y_{c_1}, \cdots, y_{c_{\mathbf{L}}})$, where $y_{c_i} = C_i(x)$ for all $i \in \{1, 2, \cdots, L\}$.

Preferences over black-box functions. The circuit designer can input preferences over multiple black-box functions through the notion of preference specification, which is defined as a vector of values $\mathbf{p} = \{p_{f_1}, \cdots, p_{f_K}, p_{c_1}, \cdots, p_{c_L}\}$ with $0 \leq p_i \leq 1$ and $\sum_{i \in \mathcal{I}} p_i = 1$ such that $\mathcal{I} = \{f_1, \cdots, f_K, c_1, \cdots, c_L\}$. Higher values of p_i mean that the corresponding objective function f_i is highly preferred. In such cases, the solution to the MOO problem should prioritize producing circuit configurations that optimize the preferred objective functions.

Problem Formulation - Constrained multi-objective optimization w/ preferences. Constrained MOO is the problem of optimizing $\mathbf{K} \geq 2$ real-valued objective functions $\{f_1(x), \cdots, f_{\mathbf{K}}(x)\}$, while satisfying $\mathbf{L}$ black-box constraints of the form $c_1 \geq 0, \cdots, c_{\mathbf{L}}(x) \geq 0$ over the given circuit design space $\mathfrak{X}$. A simulation experiment with a candidate circuit design parameters $\mathbf{x} \in \mathfrak{X}$ generates two vectors, one consisting of objective values and one consisting of constraint values $\mathbf{y} = (y_{f_1}, \cdots, y_{f_{\mathbf{K}}}, y_{c_1}, \cdots, y_{c_{\mathbf{L}}})$ where $y_{f_j} = f_j(x)$ for all $j \in \{1, \cdots, K\}$ and $y_{c_i} = C_i(x)$ for all $i \in \{1, \cdots, L\}$. We define a circuit configuration $\mathbf{x}$ as feasible if and only if it satisfies all constraints. Circuit configuration $\mathbf{x}$ *Pareto-dominates* another configuration $\mathbf{x}'$ if $f_j(\mathbf{x}) \leq f_j(\mathbf{x}')\ \forall j$ and there exists some $j \in \{1, \cdots, K\}$ such that $f_j(\mathbf{x}) < f_j(\mathbf{x}')$.

The optimal solution of the MOO problem with constraints is a set of circuit configurations $\mathcal{X}^* \subset \mathfrak{X}$ such that no configuration $\mathbf{x}' \in \mathfrak{X} \setminus \mathcal{X}^*$ Pareto-dominates a configuration $\mathbf{x} \in \mathcal{X}^*$ and all configurations in $\mathcal{X}^*$ are feasible. The solution set $\mathcal{X}^*$ is called the optimal constrained *Pareto set* and the corresponding set of function values $\mathcal{Y}^*$ is called the optimal constrained *Pareto front*. The most commonly used measure to evaluate the quality of a given Pareto set is by calculating the Pareto hypervolume (PHV) indicator [1] of the corresponding Pareto front of $(\mathbf{y_{f_1}}, \mathbf{y_{f_2}}, \cdots, \mathbf{y_{f_K}})$ with respect to a reference point $\mathbf{r}$. Our overall goal is to approximate the constrained Pareto set $\mathcal{X}^*$ by minimizing the total number of expensive circuit simulations. If a preference specification p over the objectives is provided, the MOO algorithm should prioritize producing a Pareto set of circuit design parameters that optimize the preferred objective functions.

Surrogate Modeling. Gaussian Processes (GPs) [15] are suitable for solving black-box optimization problems with expensive function evaluations since they are rich and flexible models which can mimic any complex objective function. Intuitively, two candidate circuit design parameters that are close to each other will potentially exhibit approximately similar circuit performance in terms of output objectives. We model the objective functions and black-box constraints by independent GP models $\mathcal{GP}_{f_1}, \cdots, \mathcal{GP}_{f_K}, \mathcal{GP}_{c_1}, \cdots, \mathcal{GP}_{c_K}$ with zero mean and i.i.d. observation noise. Let $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{t-1}$ be the training data from past $t-1$ function evaluations, where $\mathbf{x}_i \in \mathfrak{X}$ is a candidate design parameters of circuit and $\mathbf{y}_i = \{y^i_{f_1}, \cdots, y^i_{f_K}, y^i_{c_1}, \cdots, y^i_{c_L}\}$ is the output vector resulting from evaluating the objective functions and constraints at $\mathbf{x}_i$ using the circuit simulator.

Overview of PAC-MOO. PAC-MOO algorithm is an instance of the BO framework, which takes as input the analog circuit design space $\mathfrak{X}$, preferences over objectives p, circuit simulator, and produces a Pareto set of candidate circuits as per the preferences after T iterations of PAC-MOO as shown in Algorithm 1. In each iteration t, PAC-MOO selects a candidate circuit design $\mathbf{x}_t \in \mathfrak{X}$ (i.e., design parameters) to perform circuit simulation. Consequently, the surrogate models for both objective functions and constraints are updated based on training data from the simulated circuit design.

Preference-Aware Acquisition Function. The state-of-the-art MESMOC approach for solving MOO problems [3] proposed to select the input that maximizes the information gain about the optimal **Pareto front** for evaluation. However, this approach did not address the challenge of handling preferences over objectives. To overcome this challenge, we propose an extension of MESMOC's acquisition function to maximize the information gain between the next candidate input for evaluation $\mathbf{x}$ and the optimal constrained Pareto front $\mathcal{Y}^*$ while factoring in preferences over objectives and constraints:

$$\alpha(\mathbf{x}) = I(\{\mathbf{x}, \mathbf{y}\}, \mathcal{Y}^* \mid D) = H(\mathbf{y} \mid D, \mathbf{x}) - \mathbb{E}_{\mathcal{Y}^*}[H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}^*)] \tag{1}$$

Due to space constraints, we provide the full derivation of the acquisition function in sections 5.1 and 5.2 in the appendix. The final form of PAC-MOO's acquisition function is shown below:

$$AF(i,x) = \sum_{s=1}^{S} \frac{\gamma_s^i(\mathbf{x})\phi(\gamma_s^i(\mathbf{x}))}{2\Phi(\gamma_s^i(\mathbf{x}))} - \ln \Phi(\gamma_s^i(\mathbf{x})) \quad \text{with} \quad i \in \mathcal{I} \quad \text{and} \quad \mathcal{I} = \{c_1, ..., c_L, f_1, ..., f_K\} \tag{2}$$

$$\alpha_{pref}(\mathbf{x}) \simeq \sum_{i \in \mathcal{I}} p_i \times AF(i,x) \text{ with } i \in \mathcal{I} \ s.t \sum_{i \in \mathcal{I}} p_i = 1 \tag{3}$$

where S is the number of samples, $\gamma_s^{c_i}(x) = \frac{y_s^{c_i *} - \mu_{c_i}(\mathbf{x})}{\sigma_{c_i}(\mathbf{x})}$, $\gamma_s^{f_j}(x) = \frac{y_s^{f_j *} - \mu_{f_j}(\mathbf{x})}{\sigma_{f_j}(\mathbf{x})}$, ϕ and Φ are the p.d.f and c.d.f of a standard normal distribution respectively.

Algorithm 1 PAC-MOO Algorithm

Inputs: Input space $\mathcal{X}$, black-box functions $\{f_1, ..., f_\mathbf{K}\}$, constraint functions $\{c_1, ..., c_L\}$, preferences $\mathbf{p} = \{p_{f_1}, \cdots, p_{f_K}, p_{c_1}, \cdots, p_{c_L}\}$, number of initial points $\mathcal{N}_0$, number of iterations T

1: Initialize Gaussian processes for functions $\mathcal{M}_{f_1}, \cdots, \mathcal{M}_{f_\mathcal{K}}$ and constraints $\mathcal{M}_{c_1}, \cdots, \mathcal{M}_{c_\mathcal{L}}$ by evaluating them on $\mathcal{N}_0$ initial circuit design parameters
2: **for** each iteration t = $\mathcal{N}_0$ to T+$\mathcal{N}_0$ **do**
3: **if** feasible circuit design parameters $\mathbf{x}_{feasible} \notin \mathcal{D}$ **then**
4: Select design parameters $\mathbf{x}_t \leftarrow \arg max_{\mathbf{x} \in \mathcal{X}} \ \alpha_{prob}(\mathbf{x})$ *# eq. 4*
5: **else**
6: Select design parameters $\mathbf{x}_t \leftarrow \arg max_{\mathbf{x} \in \mathcal{X}} \ \alpha_{pref}(\mathbf{x}, \mathbf{p})$ in Algorithm 2 **s.t** $(\mu_{c_1} \geq 0, \cdots, \mu_{c_L} \geq 0)$
7: **end if**
8: Perform circuit simulation with the selected design parameters $\mathbf{x}_t$: $\mathbf{y}_t \leftarrow (f_1(\mathbf{x}_t), \cdots, f_K(\mathbf{x}_t), C_1(\mathbf{x}_t), \cdots, C_L(\mathbf{x}_t))$
9: Aggregate data: $\mathcal{D} \leftarrow \mathcal{D} \cup \{(\mathbf{x}_t, \mathbf{y}_t)\}$
10: Update models $\mathcal{M}_{f_1}, \cdots, \mathcal{M}_{f_K}$ and $\mathcal{M}_{c_1}, \cdots, \mathcal{M}_{c_L}$ using $\mathcal{D}$
11: **end for**
12: **return** the Pareto set of feasible circuit design parameters from $\mathcal{D}$

Algorithm 2 Preference based Acquisition function (α_{pref})

$\alpha_{pref}(\mathbf{x}, \mathbf{p})$

1: **for** Each sample $s \in \{1, \cdots, S\}$ **do**
2: Sample functions $\tilde{f}_j \sim \mathcal{M}_{f_j}, \quad \forall j \in \{1, \cdots, K\}$
3: Sample constraints $\tilde{C}_i \sim \mathcal{M}_{c_i}, \quad \forall i \in \{1, \cdots, L\}$
4: Solve *cheap* MOO over $(\tilde{f}_1, \cdots, \tilde{f}_K)$ constrained by $(\tilde{C}_1, \cdots, \tilde{C}_L)$ $\mathcal{Y}_s^* \leftarrow \arg max_{x \in \mathcal{X}}(\tilde{f}_1, \cdots, \tilde{f}_K)$ **s.t** $(\tilde{C}_1 \geq 0, \cdots, \tilde{C}_L \geq 0)$
5: **end for**
6: **for** $i \in \mathcal{I}$ and $\mathcal{I} = \{c_1 \cdots c_L, f_1 \cdots f_K\}$ **do**
7: Compute $AF(i,x)$ based on S samples of $\mathcal{Y}_s^*$ via Equation 13
8: **end for**
9: **Return** $\sum_{i \in \mathcal{I}} p_i \times AF(i,x)$

Finding Feasible Regions of Design Space

The acquisition function defined in equation 3 will build constrained Pareto front samples $\mathcal{Y}_s^*$ by sampling functions and constraints from the Gaussian process posterior. The posterior of the GP is built based on the current training data $\mathcal{D}$ (circuit simulations performed in the past). The truncated Gaussian approximation defined in Equations 9 and 10 (appendix) requires the upper bound $y_s^{f_j*}$ and $y_s^{c_i*}$ to be defined. However, in the early Bayesian optimization iterations of the algorithm, the circuit configurations evaluated may not include any feasible circuit design parameters. This is especially true for scenarios where the fraction of feasible circuit configurations in the entire design space is very small (e.g., high-conversion ratio converter with inductor where many designs are not stable). In such cases, the sampling process of the constrained Pareto fronts $\mathcal{Y}_s^*$ is susceptible to failure because the surrogate models did not gather any knowledge about feasible regions of the design space *yet*. Consequently, the upper bounds $y_s^{f_j*}$ and $y_s^{c_i*}$ are not well-defined and the acquisition function in 3 is not well-defined. Intuitively, the algorithm should first aim at identifying feasible circuit configurations by maximizing the probability of satisfying all the constraints. We define a special case of our acquisition function for such challenging scenarios as shown below:

$$\alpha_{prob}(x) = \prod_{i=1}^{L} Pr(c_i(x) \geq 0) \tag{4}$$

This acquisition function enables an efficient feasibility search due to its exploitation characteristics [7]. Given that the probability of constraint satisfaction is binary (0 or 1), the algorithm will be able to quickly prune unfeasible regions of the circuit design space and move to other promising regions until it identifies feasible circuit configurations. This approach will enable a more efficient search over feasible regions later and an accurate computation of the acquisition function. The complete pseudo-code of PAC-MOO is given in Algorithm 1.

3 Experimental Setup and Results

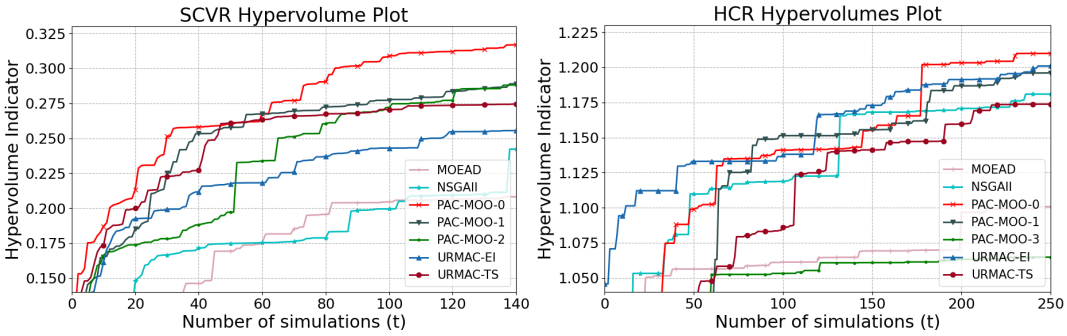
Baselines. We compare PAC-MOO with state-of-the-art constrained MOO evolutionary algorithms, namely, NSGA-II [5] and MOEAD [16]. We also compare to the state-of-the-art BO for analog circuit design algorithm, the Uncertainty Reduction via Multiple Acquisition Constrained (URMAC) [17]. We evaluated two variants of URMAC: URMAC-EI and URMAC-TS, using expected improvement (EI) and Thompson sampling (TS) acquisition functions.

PAC-MOO: We employ a Gaussian process (GP) with squared exponential kernel for all our surrogate models. We evaluated several preference values for the efficiency objective function. PAC-MOO-0 refers to the preference being equal over all objectives and constraints. PAC-MOO-1 refers to giving 80% preference to the efficiency objective while giving equal importance to functions and constraints, resulting in a preference value $p_i = 0.5 * 0.8 = 0.4$ for the efficiency. With PAC-MOO-3, we give more importance to the objective functions by assigning a total of 0.65 preference to them and 0.35 to the constraints. Additionally, we provide 88% importance to the efficiency resulting in a preference value of $p_i = 0.65 * 0.88 = 0.572$. With PAC-MOO-2, we assign a total preference of 85% to the objective functions with 92% importance to the efficiency resulting in a preference value of $p_i = 0.85 * 0.92 = 0.782$. We assign equal preference to all other functions.

Benchmarks: *1.Switched-Capacitor Voltage Regulator (SCVR) design optimization setup.* The constrained MOO problem for SCVR circuit design consists of 33 input design variables, nine objective functions, and 14 constraints. Every method is initialized with 24 randomly sampled circuit configurations. *2. High Conversion Ratio (HCR) design setup.* The constrained MOO problem for HCR circuit design consists of 32 design variables, 5 objective functions, and 6 constraints. Considering that the fraction of feasible circuit configurations in the design space is extremely low (around 4%), every method is initialized with 32 initial feasible designs provided by a domain expert.

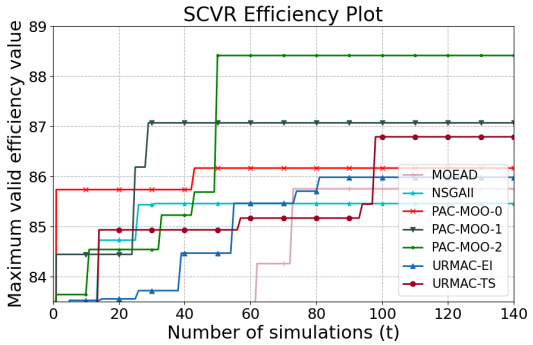
A detailed explanation of the two analog circuits is included in section 5.3 in the appendix.

Hypervolume of Pareto set vs. No of circuit simulations. Figures 1a and 1b show the results for PHV of Pareto set as a function of the number of circuit simulations for SCVR and HCR design, respectively. We make the following observations. **1)** PAC-MOO with no preferences (i.e., PAC-MOO-0) outperforms all the baseline methods. This is attributed to the efficient information-theoretic

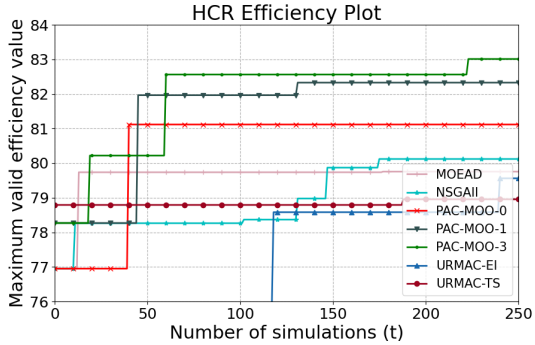


(a) Hypervolume - SCVR

(b) Hypervolume - HCR



(c) Efficiency - SCVR



(d) Efficiency - HCR

Figure 1: Results of different multi-objective algorithms including PAC-MOO. The hypervolume indicator and maximum discovered efficiency are shown as functions of the number of simulations

acquisition function and the exploitation approach to finding feasible regions in the circuit design space. **2)** At least one version of URMAC performs better than all evolutionary baselines: URMAC-EI for both SCVR and HCR designs. These results demonstrate that BO methods have the potential for accelerating analog circuit design optimization over evolutionary algorithms .**3)** The performance of PAC-MOO with preference (i.e., PAC-MOO-1,2,3) is lower in terms of the hypervolume since the metric evaluates the quality of general Pareto front, while our algorithm puts emphasis on specific regions of the Pareto front via preference specification. This behavior is expected, nevertheless, we notice that the PHV with PAC-MOO-1 and PAC-MOO-2 is still competitive and degrades only when a significantly high preference is given to efficiency (PAC-MOO-3).

Efficiency of optimized circuits with preferences. Since efficiency is the most important objective for both SCVR and HCR circuits, we evaluate PAC-MOO by giving higher preference to efficiency over other objectives. Figures 1c and 1d show the results for maximum efficiency of the optimized circuit configurations as a function of the number of circuit simulations for SCVR and HCR design optimization. **1)** As intended by design, PAC-MOO with preferences outperforms all baseline methods, including PAC-MOO without preferences. **2)** The improvement in maximum efficiency of uncovered circuit configurations for PAC-MOO with preferences comes at the expense of loss in hypervolume metric as shown in Figure 1a and Figure 1b.

4 Conclusions

Motivated by challenges in hard analog circuit design optimization problems, this paper proposed a principled and efficient Bayesian optimization algorithm called PAC-MOO. The key innovations behind PAC-MOO include an effective exploitation approach to finding feasible regions of the design space, and incorporating preferences over multiple objectives using a convex combination of the corresponding acquisition functions. Experimental results on two challenging analog circuit design optimization problems demonstrated that PAC-MOO outperforms baseline methods.

References

[1] A. Auger, J. Bader, D. Brockhoff, and E. Zitzler. Theory of the hypervolume indicator: Optimal μ-distributions and the choice of the reference point. In *Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms*, FOGA '09, page 87–102, New York, NY, USA, 2009. Association for Computing Machinery.

[2] S. Belakaria, A. Deshwal, and J. R. Doppa. Max-value entropy search for multi-objective Bayesian optimization. In *Conference on Neural Information Processing Systems*, pages 7823–7833, 2019.

[3] S. Belakaria, A. Deshwal, and J. R. Doppa. Max-value entropy search for multi-objective bayesian optimization with constraints. *arXiv preprint arXiv:2009.01721*, 2020.

[4] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley and Sons, 2012.

[5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[6] P. Feliot, J. Bect, and E. Vazquez. A Bayesian approach to constrained single-and multi-objective optimization. *Journal of Global Optimization*, 67(1-2):97–133, 2017.

[7] J. R. Gardner, M. J. Kusner, Z. E. Xu, K. Q. Weinberger, and J. P. Cunningham. Bayesian optimization with inequality constraints. In *ICML*, volume 2014, pages 937–945, 2014.

[8] J. M. Hernández-Lobato, M. W. Hoffman, and Z. Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Advances in Neural Information Processing Systems*, pages 918–926, 2014.

[9] W. Hong, B. Nguyen, Z. Zhou, N. Tang, J. Kim, P. P. Pande, and D. Heo. A dual-output step-down switched-capacitor voltage regulator with a flying capacitor crossing technique for enhanced power efficiency. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 27(12):2861–2871, 2019.

[10] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[11] J. V. Michalowicz, J. M. Nichols, and F. Bucholtz. *Handbook of differential entropy*. Chapman and Hall/CRC, 2013.

[12] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2008.

[13] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

[14] N. Tang, W. Hong, B. Nguyen, Z. Zhou, J.-H. Kim, and D. Heo. Fully integrated switched-inductor-capacitor voltage regulator with 0.82-a/mm² peak current density and 78*IEEE Journal of Solid-State Circuits*, 56(6):1805–1815, 2021.

[15] C. K. Williams and C. E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT Press, 2006.

[16] Q. Zhang and H. Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

[17] Z. Zhou, S. Belakaria, A. Deshwal, W. Hong, J. R. Doppa, P. Pratim Pande, and D. Heo. Design of multi-output switched-capacitor voltage regulator via machine learning. In *2020 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 502–507, 2020.

5 Appendix

5.1 Acquisition Function

In this case, the output vector $\mathbf{y}$ is $K + L$ dimensional: $\mathbf{y} = (y_{f_1}, y_{f_2}, \cdots, y_{f_K}, y_{c_1} \cdots y_{c_L})$ where $y_{f_j} = f_j(x) \forall j \in \{1, 2, \cdots, K\}$ and $y_{c_i} = C_i(x) \forall i \in \{1, 2, \cdots, L\}$. Consequently, the first term in Equation (1), entropy of a factorizable $(K + L)$-dimensional Gaussian distribution $P(\mathbf{y} \mid D, \mathbf{x})$, can be computed in closed form as shown below:

$$H(\mathbf{y} \mid D, \mathbf{x}) = \frac{(K + C)(1 + \ln(2\pi))}{2} + \sum_{j=1}^{K} \ln(\sigma_{f_j}(\mathbf{x})) + \sum_{i=1}^{L} \ln(\sigma_{c_i}(\mathbf{x})) \tag{5}$$

where $\sigma^2_{f_j}(\mathbf{x})$ and $\sigma^2_{c_i}(\mathbf{x})$ are the predictive variances of j^{th} function and i^{th} constraint GPs respectively at input $\mathbf{x}$. The second term in Equation (1) is an expectation over the Pareto front $\mathcal{Y}^*$. We can approximately compute this term via Monte-Carlo sampling as shown below:

$$\mathbb{E}_{\mathcal{Y}^*}[H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}^*)] \simeq \frac{1}{S} \sum_{s=1}^{S} [H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}^*_s)] \tag{6}$$

where S is the number of samples and $\mathcal{Y}^*_s$ denote a sample Pareto front. There are two key algorithmic steps to compute this part of the equation: 1) How to compute constrained Pareto front samples $\mathcal{Y}^*_s$?; and 2) How to compute the entropy with respect to a given constrained Pareto front sample $\mathcal{Y}^*_s$? We provide solutions for these two questions below.

1) Computing constrained Pareto front samples via cheap multi-objective optimization. To compute a constrained Pareto front sample $\mathcal{Y}^*_s$, we first sample functions and constraints from the posterior GP models via random Fourier features [8, 12] and then solve a cheap constrained multi-objective optimization over the K sampled functions and L sampled constraints.

Cheap MO solver. We sample $\tilde{f}_i$ from GP model $\mathcal{GP}_{f_j}$ for each of the K functions and $\tilde{C}_j$ from GP model $\mathcal{GP}_{c_j}$ for each of the L constraints. A *cheap* constrained multi-objective optimization problem over the K sampled functions $\tilde{f}_1, \tilde{f}_2, \cdots, \tilde{f}_k$ and the L sampled constraints $\tilde{C}_1, \tilde{C}_2, \cdots, \tilde{C}_L$ is solved to compute the sample Pareto front $\mathcal{Y}^*_s$. Note that we refer to this optimization problem as cheap because it is performed over sampled functions and constraints, which are cheaper to evaluate than performing expensive circuit simulations. We employ the popular constrained NSGA-II algorithm [6, 5] to solve the constrained MO problem with cheap sampled objective functions and constraints.

2) Entropy computation with a sample constrained Pareto front. Let $\mathcal{Y}^*_s = \{\mathbf{v}^1, \cdots, \mathbf{v}^l\}$ be the sample constrained Pareto front, where l is the size of the Pareto front and each $\mathbf{v}^i$ is a $(K + L)$-vector evaluated at the K sampled functions and L sampled constraints $\mathbf{v}^i = \{v^i_{f_1}, \cdots, v^i_{f_K}, v^i_{c_1}, \cdots, v^i_{c_L}\}$. The following inequality holds for each component y_j of the $(K + L)$-vector $\mathbf{y} = \{y_{f_1}, \cdots, y_{f_K}, y_{c_1}, \cdots y_{c_L}\}$ in the entropy term $H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}^*_s)$:

$$y_j \leq \max\{v^1_j, \cdots v^l_j\} \quad \forall j \in \{f_1, \cdots, f_K, c_1, \cdots, c_L\} \tag{7}$$

The inequality essentially says that the j^{th} component of $\mathbf{y}$ (i.e., y_j) is upper-bounded by a value obtained by taking the maximum of j^{th} components of all l $(K + L)$-vectors in the Pareto front $\mathcal{Y}^*_s$. This inequality had been proven by a contradiction for MESMO [2] for all objective functions $j \in \{f_1, \cdots, f_K\}$. We assume the same for all constraints $j \in \{c_1, \cdots, c_L\}$.

By combining the inequality (7) and the fact that each function is modeled as an independent GP, we can approximate each component y_j as a truncated Gaussian distribution since the distribution of y_j needs to satisfy $y_j \leq \max\{v^1_j, \cdots v^l_j\}$. Let $y^{c_i*}_s = \max\{v^1_{c_i}, \cdots v^l_{c_i}\}$ and $y^{f_j*}_s = \max\{v^1_{f_j}, \cdots v^l_{f_j}\}$. Furthermore, a common property of entropy measure allows us to decompose the entropy of a set of independent variables into a sum over entropies of individual variables [4]:

$$H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}^*_s) = \sum_{j=1}^{K} H(y_{f_j} | D, \mathbf{x}, y^{f_j*}_s) + \sum_{i=1}^{L} H(y_{c_i} | D, \mathbf{x}, y^{c_i*}_s) \tag{8}$$

The r.h.s is a summation over entropies of $(K+L)$-variables $\mathbf{y} = \{y_{f_1}, \cdots, y_{f_K}, y_{c_1}, \cdots y_{c_L}\}$. The differential entropy for each y_j is the entropy of a truncated Gaussian distribution [11] and is given by the following equations:

$$H(y_{f_j} \mid D, \mathbf{x}, y_s^{f_j*}) \simeq [\frac{(1+\ln(2\pi))}{2} + \ln(\sigma_{f_j}(\mathbf{x})) + \ln\Phi(\gamma_s^{f_j}(\mathbf{x})) - \frac{\gamma_s^{f_j}(\mathbf{x})\phi(\gamma_s^{f_j}(\mathbf{x}))}{2\Phi(\gamma_s^{f_j}(\mathbf{x}))}] \tag{9}$$

$$H(y_{c_i}|D, \mathbf{x}, y_s^{c_i*}) \simeq [\frac{(1+\ln(2\pi))}{2} + \ln(\sigma_{c_i}(\mathbf{x})) + \ln\Phi(\gamma_s^{c_i}(\mathbf{x})) - \frac{\gamma_s^{c_i}(\mathbf{x})\phi(\gamma_s^{c_i}(\mathbf{x}))}{2\Phi(\gamma_s^{c_i}(\mathbf{x}))}] \tag{10}$$

Consequently, we have:

$$\begin{aligned} H(\mathbf{y} \mid D, \mathbf{x}, \mathcal{Y}_s^*) \simeq & \sum_{j=1}^{K} \left[\frac{(1+\ln(2\pi))}{2} + \ln(\sigma_{f_j}(\mathbf{x})) + \ln\Phi(\gamma_s^{f_j}(\mathbf{x})) - \frac{\gamma_s^{f_j}(\mathbf{x})\phi(\gamma_s^{f_j}(\mathbf{x}))}{2\Phi(\gamma_s^{f_j}(\mathbf{x}))}\right] \\ & + \sum_{i=1}^{L} \left[\frac{(1+\ln(2\pi))}{2} + \ln(\sigma_{c_i}(\mathbf{x})) + \ln\Phi(\gamma_s^{c_i}(\mathbf{x})) - \frac{\gamma_s^{c_i}(\mathbf{x})\phi(\gamma_s^{c_i}(\mathbf{x}))}{2\Phi(\gamma_s^{c_i}(\mathbf{x}))}\right] \end{aligned} \tag{11}$$

where $\gamma_s^{c_i}(x) = \frac{y_s^{c_i*} - \mu_{c_i}(\mathbf{x})}{\sigma_{c_i}(\mathbf{x})}$, $\gamma_s^{f_j}(x) = \frac{y_s^{f_j*} - \mu_{f_j}(\mathbf{x})}{\sigma_{f_j}(\mathbf{x})}$, ϕ and Φ are the p.d.f and c.d.f of a standard normal distribution respectively. By combining equations (5) and (11) with equation (1), we get the final form of our acquisition function as shown below:

$$\alpha(\mathbf{x}) \simeq \sum_{i \in \mathcal{I}} AF(i, x) \text{ with } i \in \mathcal{I} \text{ and } \mathcal{I} = \{c_1 \cdots c_L, f_1 \cdots f_K\} \tag{12}$$

And

$$AF(i, x) = \sum_{s=1}^{S} \frac{\gamma_s^i(\mathbf{x})\phi(\gamma_s^i(\mathbf{x}))}{2\Phi(\gamma_s^i(\mathbf{x}))} - \ln\Phi(\gamma_s^i(\mathbf{x})) \tag{13}$$

5.2 Convex Combination for Preferences

We now describe how to incorporate preference specification (when available) into the acquisition function. The derivation of the acquisition function proposed in Equation 12 resulted in a function in the form of a summation of an entropy term defined for each of the objective functions and constraints as $AF(i, x)$. Given this expression, the algorithm will select an input while giving the same importance to each of the functions and constraints. However, in circuit design optimization, efficiency is typically the most important objective function. The designer would like to find a trade-off between the objectives. Nevertheless, candidate circuits with high voltage and very low efficiency might be useless in practice. Therefore, we propose to inject preferences from the designer into our algorithm by associating different weights to each of the objectives. A principled approach would be to assign appropriate preference weights resulting in a convex combination of the individual components of the summation $AF(i, x)$. Let p_i be the preference weight associated with each individual component. The preference-based acquisition function is defined as follows (see Algorithm 2):

$$\alpha_{pref}(\mathbf{x}) \simeq \sum_{i \in \mathcal{I}} p_i \times AF(i, x) \text{ with } i \in \mathcal{I} \qquad s.t \sum_{i \in \mathcal{I}} p_i = 1 \tag{14}$$

It is important to note that in practice if a candidate circuit does not satisfy the constraints, the optimization will fail regardless of the preferences over objectives. Therefore, the cumulative weights assigned to the constraints have to be at least equal to the total weight assigned to the objective functions:

$$\sum_{i \in \{c_1 \cdots c_L\}} p_i = \sum_{i \in \{f_1, \cdots, f_K\}} p_i = \frac{1}{2} \tag{15}$$

Given that satisfying all the constraints is equally important, the weights over the constraints would be equal. Finally, only the weights over the functions will need to be explicitly specified.

5.3 Case Studies for Analog Circuit Design

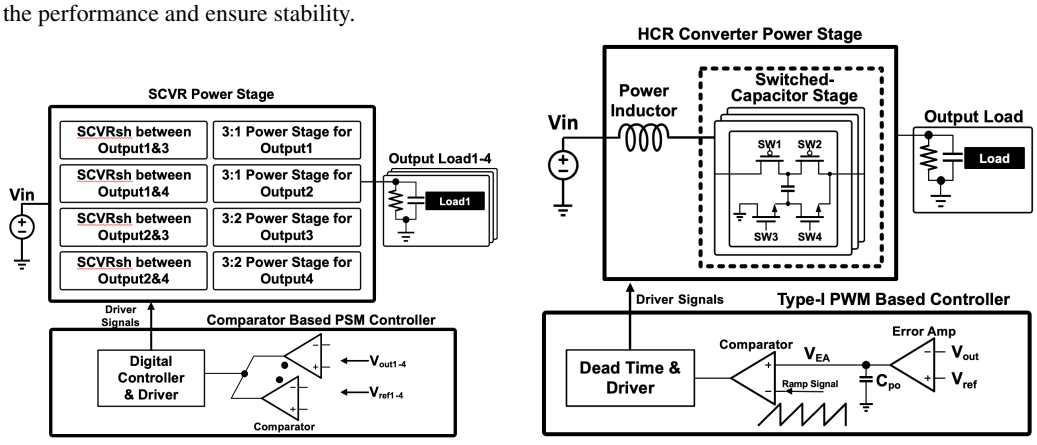
This section briefly describes the two analog circuit design problems: switched-capacitor voltage regular (SCVR) and high-conversion ratio (HCR) converter. Both SCVR and HCR circuits are extremely useful in providing

power supply for data centers, large manycore chips, and portable devices. They can directly convert high voltages (3.3V-48V) to low voltages (0.4V-1.8V). We will employ them in our experimental evaluation.

SCVR design optimization. A flying-capacitor crossing technique (FCCT) is used in the multi-output SCVR to achieve dynamic capacitor optimization, as discussed in [17]. Figure 2a illustrates the operating principle of an SCVR power stage with FCCT generating two outputs with $1/3$x, and $2/3$x conversion ratios. The FCCT technique is realized by the additional $SCVR_{sh}$ stage, which is comprised of four power switches and one flying capacitor, C_s.

This $SCVR_{sh}$ topology regulates C_s at $1/3$x of the input voltage: in phase1, C_s is charged and generates an additional current path to the $2/3$x output; in phase2, C_s is discharged, generating an additional current path to the 1/3x output. Thus, the additional $SCVR_{sh}$ power stage provides a parallel current path to both outputs. This path helps in reducing the portion of charges that each flying capacitor carries to the output. The reduction in the charge flowing through the flying capacitors, in turn, decreases the power loss governed by the charging and discharging of the capacitors, i.e., charge redistribution loss. Moreover, the charge passing through the resistive elements also decreases. Hence, resistive conduction loss is reduced. We employ a four-output SCVR for design optimization. The SCVR simultaneously generates four individual output voltages (V_{out1}, V_{out2}, V_{out3}, and V_{out4}): V_{out1}, V_{out2} are regulated at $1/3$x conversion ratio, and V_{out3}, V_{out4} are regulated at $2/3$x conversion ratio. The four outputs are regulated individually by a pulse-skipping modulation (PSM) [9] with a fixed frequency clock for the 3:1 power stages and 3:2 power stages and adjust frequency for controlling the $SCVR_{sh}$ power stages. There are eight sub-power stages in the SCVR: two 3:1 power stages for V_{out1} and V_{out2}, two 3:2 power stages for V_{out3} and V_{out4}, and four $SCVR_{shs}$ (connected between V_{out1} and V_{out3}, V_{out1} and V_{out4}, V_{out2} and V_{out3}, V_{out2} and V_{out4}, respectively). The controller feedback loop is self-stable due to the comparator-based PSM methods. We want to optimize the flying capacitor and switching frequency to enhance the performance under different load conditions.

HCR converter design optimization. For the HCR converter experiments, we use an inductor-first hybrid power stage, which has been previously introduced in [14] with extending the conversion ratio from the original value of 0.5 to 1 to a high conversion ratio of from 9 to 10. This converter can handle an input voltage of 5.5V-6V with a nominal 1.8V CMOS process. Figure 2b presents the operational principle of an HCR power stage. The inductor-first hybrid power stage consists of a power inductor directly connected to the input source and a switched-capacitor stage, including flying capacitors and power switches. It can provide fine-grained voltage regulation by using pulse-width-modulation (PWM) control. A fully integrated inductor-first hybrid converter with 0.73-W/mm^2 peak power density has been proposed by [14], which uses one flying capacitor and one inductor to produce a voltage conversion ratio (VCR) between 1/2 and 1 by changing the duty cycle of the PWM controller. In this proposed HCR inductor-first hybrid power stage, 4 flying capacitors and 12 power switches are involved in achieving the 9 to 10 step-down conversion ratio and releasing the voltage stress for the power switches. With the inductor-first structure, the power inductor current is reduced by more than $9x$ under the high conversion ratio scenario, which results in a dramatic improvement in the converter's efficiency. In order to regulate the output voltage with fine line/load resolution and high power efficiency, a type-I compensation circuit with the compensation capacitor C_{po} is implemented in the PWM controller. We want to optimize the flying capacitors, power inductor, power switches, switching frequency, and compensation capacitor to enhance the performance and ensure stability.



(a) Power stage of the multi-output SCVR.

(b) Power stage of the multi-output HCR.

Figure 2: Power stages of the multi-output HCR and SCVR analog circuits