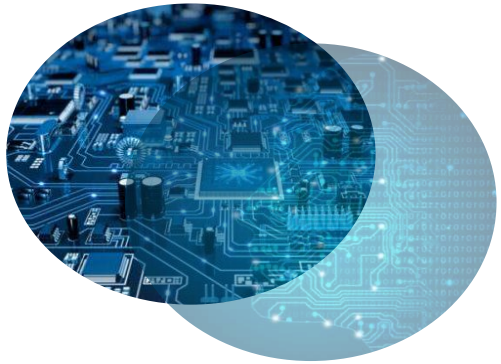
A Study on Exploring and Exploiting the High-dimensional Design Space for Analog Circuit Design Automation

Ruiyu Lyu, Yuan Meng, Aidong Zhao,Zhaori Bi*,
Keren Zhu, Fan Yang, Changhao Yan, Dian Zhou, Xuan Zeng*



2024.1.23

Problem Formulation

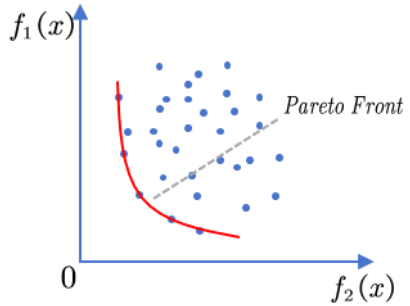
Analog Circuit Sizing Problem:

A constrained optimization problem：

$$\underset{x \in \boldsymbol{\chi}^{\mathcal{D}}}{\text{minimize}}\ \boldsymbol{f}_m(\boldsymbol{x})$$
$$\text{s.t. } c_i(x) < 0$$
$$\forall i \in 1 \ \ldots \ N_c,$$

The objective function is the Figure of Merit(FOM), which is usually the combination of the circuit's performance.

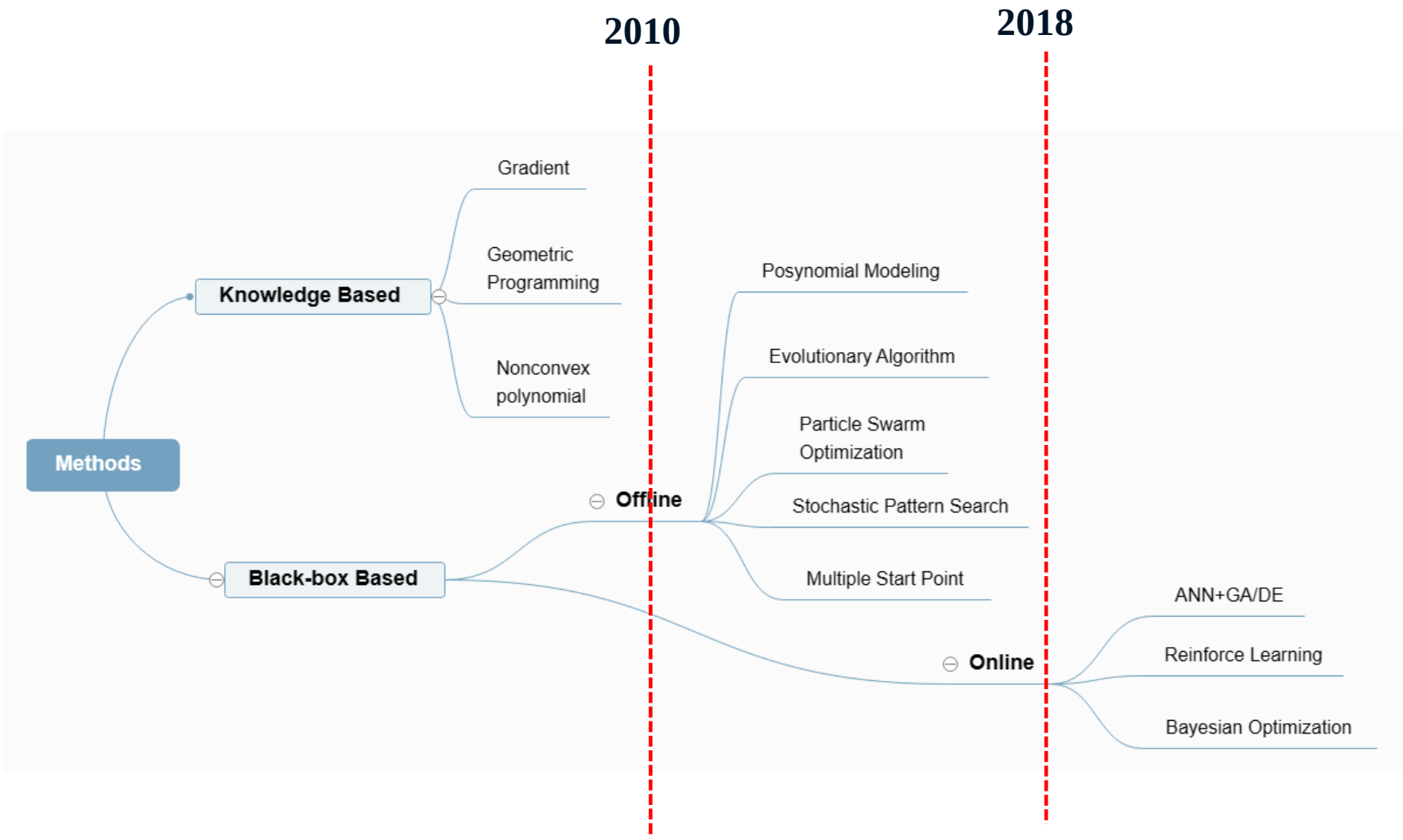
Sometimes be multi-objective:



Find the Pareto Front, the points on the Pareto Front need to meet the conditions：

$$\forall i \ f_i(x_1) \le f_i(x_2), i \in \{1, \ldots, m\}$$
$$\exists j \ f_j(x_1) < f_j(x_2), j \in \{1, \ldots, m\}$$

Methods


2010
2018
Methods
Knowledge Based
Gradient
Geometric Programming
Nonconvex polynomial
Black-box Based
Offline
Posynomial Modeling
Evolutionary Algorithm
Particle Swarm Optimization
Stochastic Pattern Search
Multiple Start Point
Online
ANN+GA/DE
Reinforce Learning
Bayesian Optimization

Knowledge Based

Gradient：[1][2]

Utilizing the partial derivatives of circuit performance metrics for gradient descent optimization of design parameters.

- **Direct method：**

$$\mathbf{A}\cdot\mathbf{v}=\mathbf{J} \Longrightarrow \frac{\partial \boldsymbol{A}}{\partial p_i}\cdot \boldsymbol{v}+\frac{\partial \boldsymbol{v}}{\partial p_i}\cdot \boldsymbol{A}=\frac{\partial \boldsymbol{J}}{\partial p_i}=0 \xrightarrow[\text{decomposition}]{\text{LU}} \boldsymbol{LU}\cdot\frac{\partial \boldsymbol{v}}{\partial p_i}=-\frac{\partial \boldsymbol{A}}{\partial p_i}\cdot \boldsymbol{v}$$

  - **Solve the circuit, get *LU and v.***
  - **Solve** $\frac{\partial \boldsymbol{v}}{\partial p_i}$

Not suitable for scenarios with a large number of parameters.

- **Adjoint Method：**

Tellegen's theorem:

$$\sum_B v_B(t)\phi_B(\tau)\equiv 0 \quad \sum_B i_B(t)\psi_B(\tau)\equiv 0 \Longrightarrow \sum_B [v_B(t)+\Delta v_B(t)]\phi_B(\tau)\equiv 0 \quad \sum_B [i_B(t)+\Delta i_B(t)]\psi_B(\tau)\equiv 0. \Longrightarrow \sum_B \Delta v_B(t)\phi_B(\tau)\equiv 0 \quad \sum_B \Delta i_B(t)\psi_B(\tau)\equiv 0 \Longrightarrow \sum_B [\Delta v_B(t)\phi_B(\tau)-\Delta i_B(t)\psi_B(\tau)]\equiv 0.$$

Introducing an adjoint network, utilizing Taylor's theorem, to solve partial derivatives for all parameters.

Time-domain solution is complex.

Gradient-based methods fully leverage circuit characteristics, but for large-scale circuits, solving circuit gradients is challenging and prone to falling into local optima.

Knowledge Based

Geometric Programming：[3][4]

Posynomial function： $f(x_1, \ldots, x_n) = \sum_{k=1}^{t} c_k x_1^{\alpha_{1k}} x_2^{\alpha_{2k}} \cdots x_n^{\alpha_{nk}}$

$$h(y) = \log(f(e^{y_1}, \ldots, e^{y_n})) = \log\left(\sum_{k=1}^{t} e^{a_k^T y + b_k}\right)$$

$$\begin{array}{lll} \text{minimize} & f_0(x) & \\ \text{subject to} & f_i(x) \le 1, & i = 1, \ldots, m \\ & g_i(x) = 1, & i = 1, \ldots, p \\ & x_i > 0, & i = 1, \ldots, n \end{array} \quad \Longrightarrow \quad \begin{array}{lll} \text{minimize} & \log f_0(e^{y_1}, \ldots, e^{y_n}) & \\ \text{subject to} & \log f_i(e^{y_1}, \ldots, e^{y_n}) \le 0, & i = 1, \ldots, m \\ & \log g_i(e^{y_1}, \ldots, e^{y_n}) = 0, & i = 1, \ldots, p. \end{array}$$

Transforming it into a convex optimization problem.

Theoretically, methods based on geometric programming can achieve global optimality, but they face challenges in model establishment. On one hand, there is poor transferability, requiring entirely different models for different circuits. On the other hand, models become complex under advanced processes, rendering the conventional square-law model no longer applicable.

Black box: Offline Model

Algorithms based on **black-box optimization** do not consider the physical model of the circuit. Instead, they establish **surrogate models** through simulation or **directly perform a search**.

1）Offline Model: At each iteration, the surrogate model is not updated, the algorithm searches for the optimal solution following a fixed pattern.

2）Online Model: Update the surrogate model at each iteration, seeking the optimal solution by optimizing the surrogate model.

Offline Model:

Posynomial Modeling [5] : The geometric programming-based method faces challenges in establishing accurate models. To address this, the method utilizes extensive data obtained through simulation, constructs polynomial surrogate models, and seeks the optimal solution by optimizing these surrogate models. However, it still encounters drawbacks such as poor model generalization and the need for a large number of policies. When tackling large-scale circuits, its applicability is limited.

Evolutionary Algorithm[6] : Initially, a set of initial points is randomly sampled and simulated to form the original population. Candidate points are generated through operations such as crossover and mutation, followed by simulation. After ranking, a group of optimal points is selected to form a new population. This process is iterated until convergence conditions are met. The convergence speed is slow, requiring a substantial number of simulations.

Multiple Start Points[8] : The optimization process is decomposed into two parts: **Global** and **Local**. Initially, the global design space is sampled using appropriate strategies, selecting promising points for local exploration using methods such as Sequential Quadratic Programming (SQP). This approach requires a significant number of simulation points and is prone to local optima.

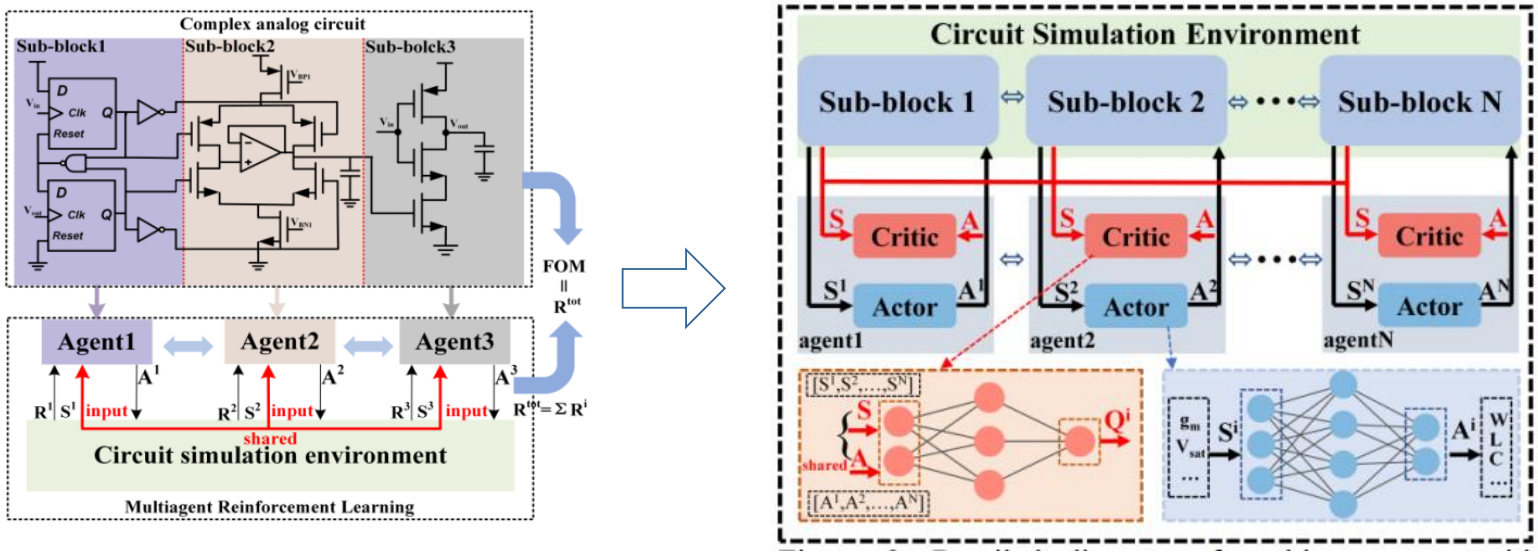
Particle Swarm Optimization[7] : Initially, random sampling is performed, and simulations yield the position and velocity for each particle, with the fitness of a particle represented by the Figure of Merit (FOM) value. Each particle updates its position based on its current fitness to obtain a new solution. PSO is susceptible to local optima and struggles with solving high-dimensional problems.

Black box: Online Model

Online Model:

ANN/DNN+[9][10]: Utilizing artificial neural networks or deep neural networks in conjunction with evolutionary or particle swarm algorithms. The selection of new points relies on the mapping provided by the neural network, which is also updated during the process.

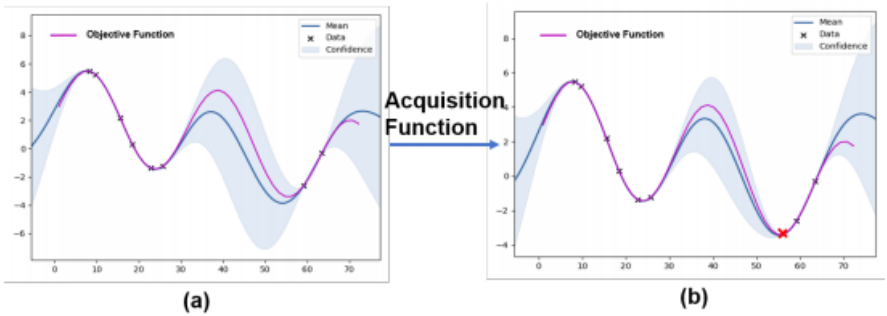
Reinforce Learning[11]: Treating subcircuits as intelligent agents, the interaction between these agents represents tradeoffs between circuits.



Black box：Online Model

Online Model：

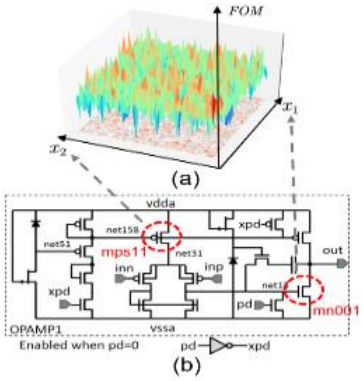
Bayesian Optimization[12]： Start by randomly sampling to establish a prior Gaussian process model. Optimize to obtain the next sampling point based on the function, update model parameters after sampling, and obtain the posterior model.



Bayesian methods exhibit **high sampling efficiency**, reducing the number of sampling points from several thousand to around a hundred for low-dimensional circuits. However, the training time cost for Gaussian Processes (GP) scales cubically with the number of sample points. Therefore, directly applying Bayesian methods to solve high-dimensional circuit optimization problems may not achieve satisfactory results within finite computational resources.

High-dimensional Challenges

Strong nonlinearity:



The response functions of circuit performance are often discontinuous. Therefore, optimization algorithms need to be capable of automatically exploring the feasible design space in high dimensions.

Massive free variables:

With the development of Moore's Law, the scale of analog integrated circuit systems has grown from a few dozen transistors in the early days to several thousand or even tens of thousands of transistors today. The number of free design variables available for tuning has grown exponentially.

High-dimensional Challenges

Time-consuming simulation:

Taking commonly used circuits such as Analog-to-Digital Converters (ADCs) and Phase-Locked Loops (PLLs) as examples, a single simulation for noise, multi-process corners, and other design metrics validation can take several hours.

Therefore, optimization algorithms for analog integrated circuit automation need to obtain feasible circuit designs with a minimal number of iterations.

The key lies in balancing the **exploration-exploitation** trade-off.

High-dimensional Methods

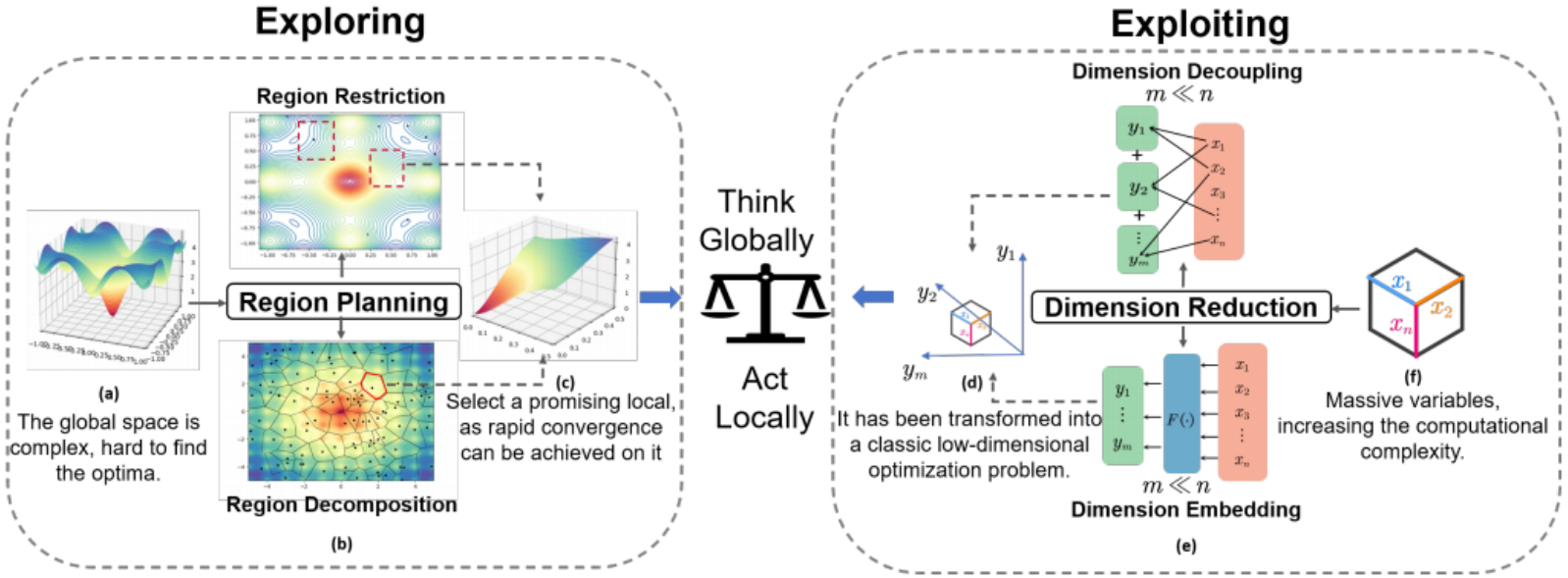
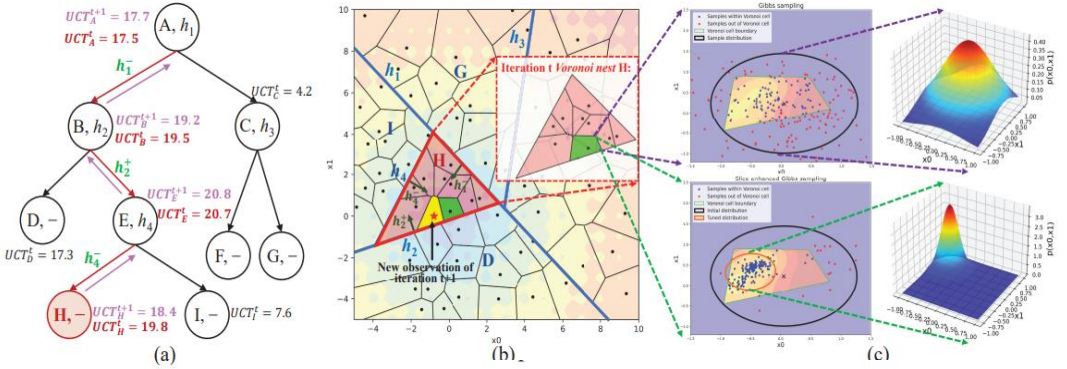


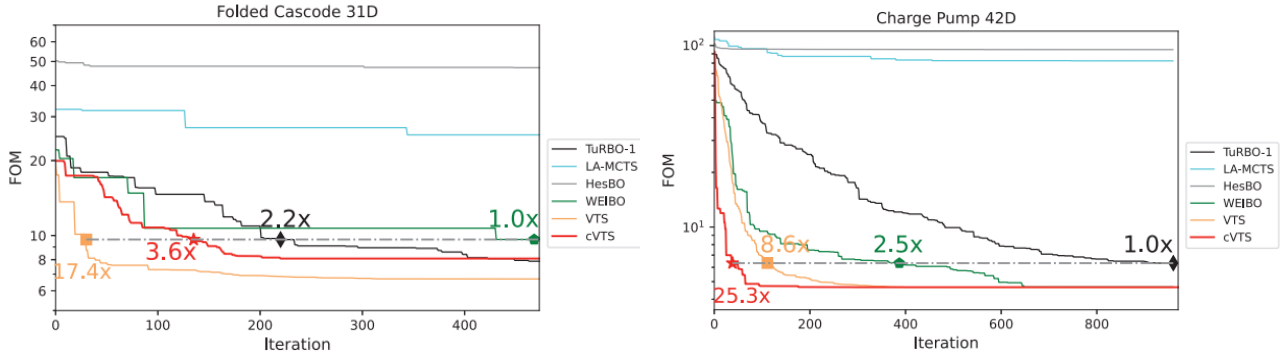
Fig. 3. Conclusion on High-Dimensional Optimization Approaches. On the left, two region planning approaches are introduced: region restriction and region decomposition, as depicted in **(b)**. Region restriction involves selecting promising areas within the entire design space, while region decomposition divides the design space into parts and selects promising ones through methods like tree search. Region planning methods can transform complex design spaces, as shown in **(a)**, into smoother subspaces, as in **(c)**, ensuring algorithm convergence. On the right, two dimension reduction methods are discussed: dimension decoupling and dimension embedding, as seen in **(e)**. Dimension decoupling breaks the high-dimensional variable space into lower-dimensional subspaces, assuming additivity between them. Dimension embedding maps high-dimensional variables to a lower-dimensional feature space using methods like neural networks. After dimension reduction, the space originally spanned by n variables is mapped to a space with m variables, where $m \ll n$, as depicted in **(d)**.

Region Decomposition

High-dimensional Bayesian Optimization[13]： Global+Local. Start by partitioning the design space into Voronoi cells. Use an evaluation function and tree search algorithm to select the current region, and establish a local Gaussian Process (GP) on each cell.



The optimization of high-dimensional acquisition functions becomes complex; hence, the method of optimization using Gibbs sampling is employed.



Optimization results of benchmark circuits.

Region Restriction

Locomobo[14]： Methods based on region restriction often rely on stochastic mechanisms to initialize trust region positions and can perform poorly for complex multi-modal optimization problems, easily getting stuck in local optima. Allocating more computational resources and performing parallel optimization within multiple trust regions can to some extent escape local optima.

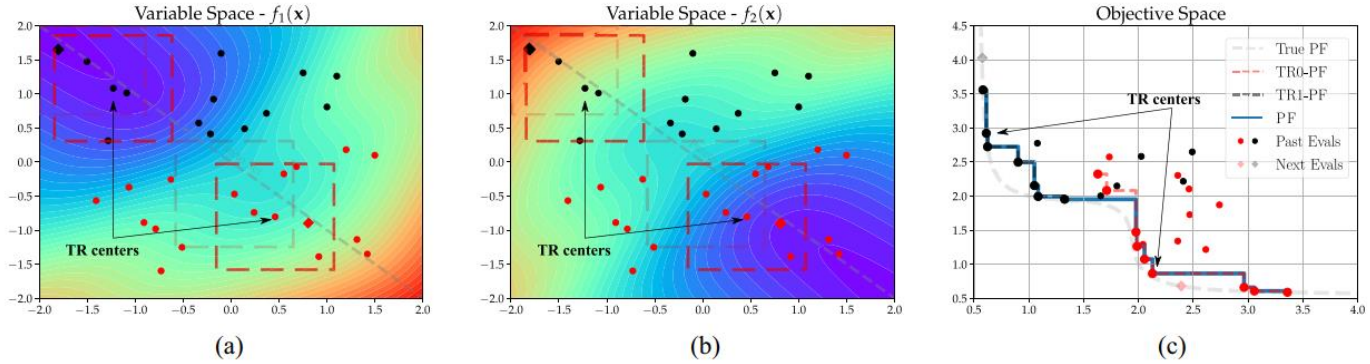
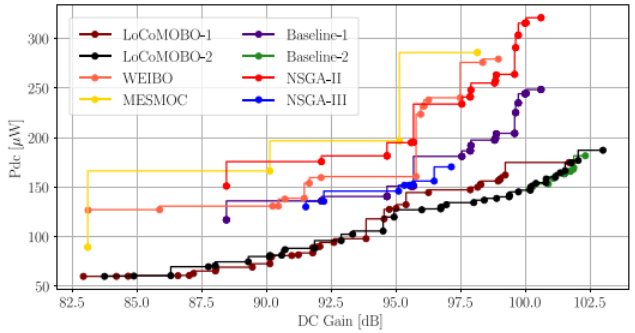
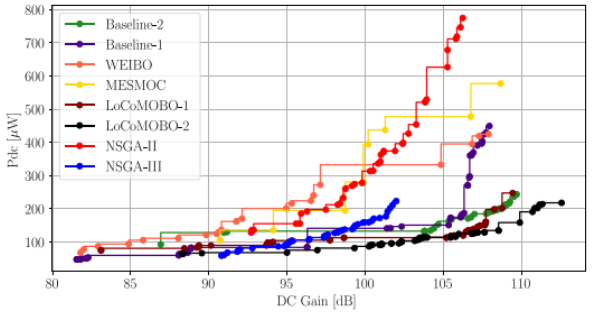


Illustration of the Locomobo algorithm.



PFs for the three stage amplifier. Bottom-right values result in better dc gain versus Pdc tradeoff



PFs for the four stage amplifier. Bottom-right values result in better dc gain versus Pdc tradeoff.

Dimension Embedding

[15]: Adopt a dropout strategy based on mutual information analysis to optimize a subset of selected variables at each iteration.

BBGP-sDFO[16]:integrates a batch Bayesian querying strategy for exploring the global design space and a Gaussian process enhanced subspace derivative free optimization method for exploiting promising regions in an effective low-dimensional subspace. The subspace is spanned by the approximate gradients and the previous iteration path.

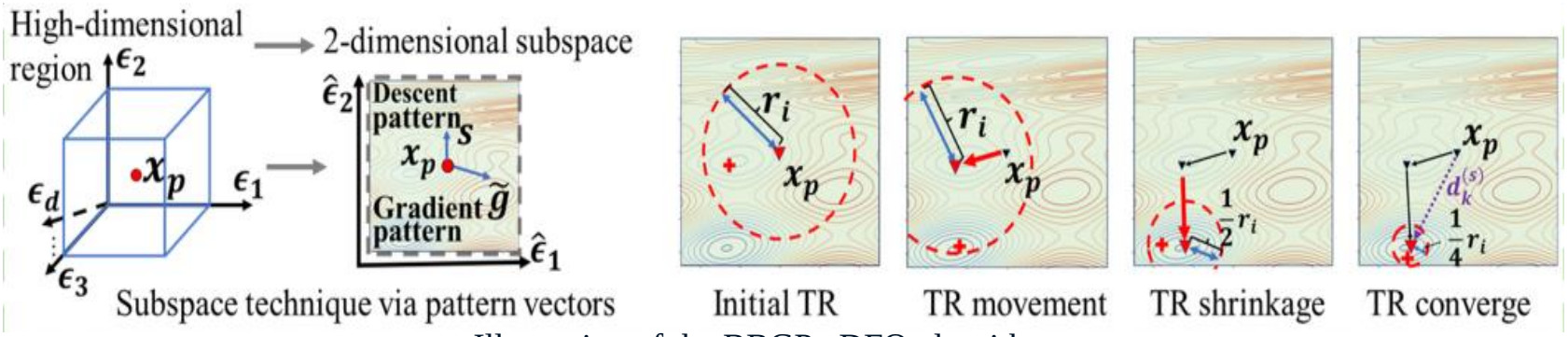
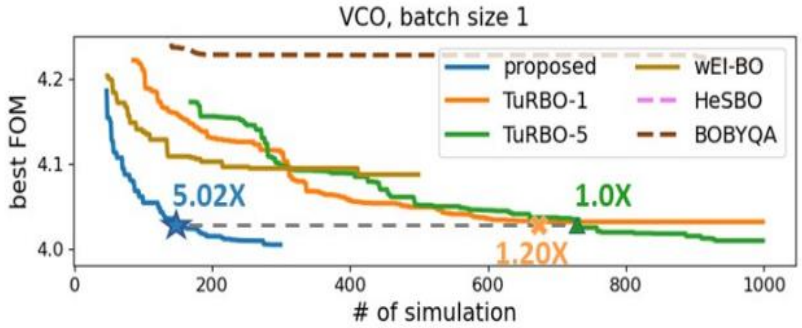
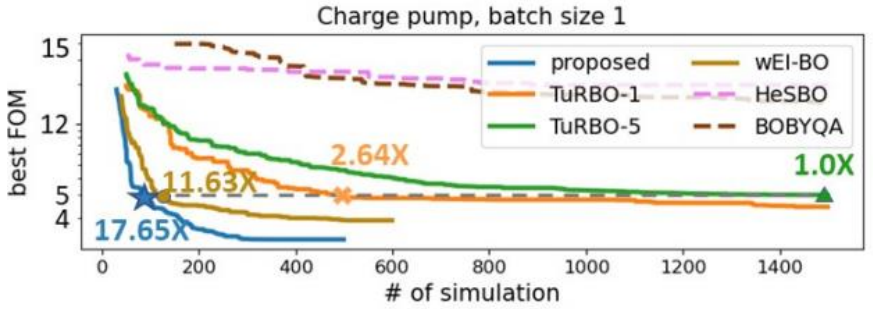


Illustration of the BBGP-sDFO algorithm.



The convergence curve of VCO circuit



The convergence curve of charge pump circuit

Dimension Decoupling

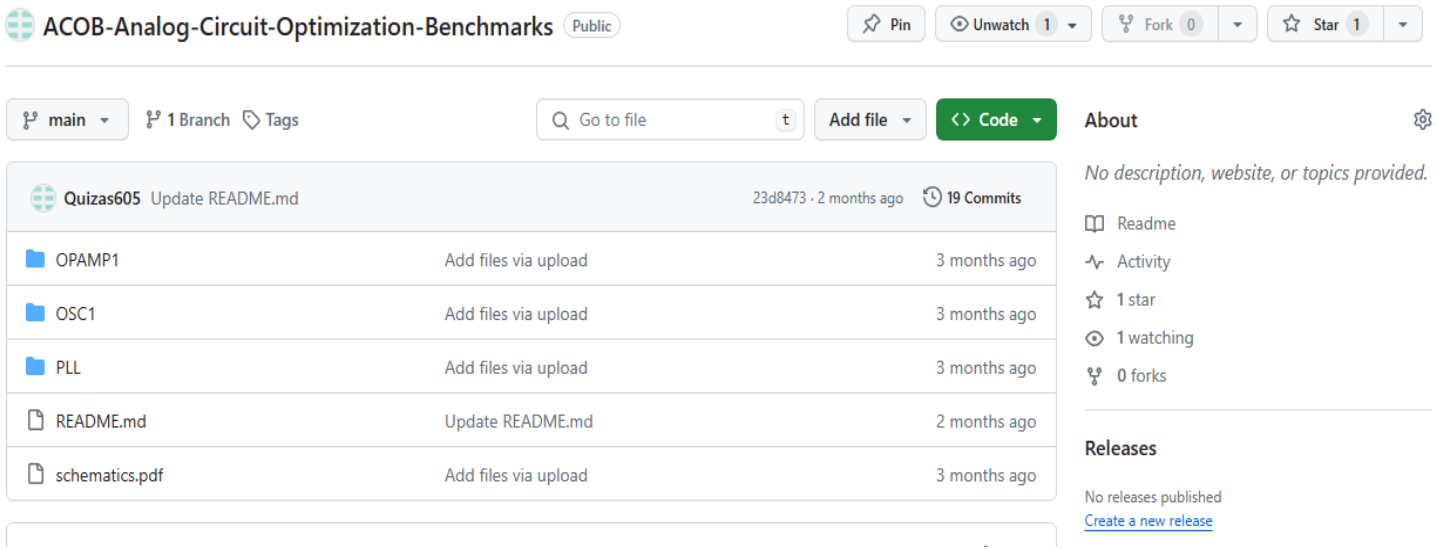
HMBO[17]: Assumes the objective function has an additive structure, i.e. it is a sum of lower dimensional component. An additive Gaussian process model is used to handle the high dimensionality from extra LDE variables.

Algorithm 2 High-Dimensional Many-Objective Gaussian-Process-Based Bayesian Optimization (HMBO)

1: Generate Gaussian process priors on $\boldsymbol{f}|_i$; //$i = 1,\ldots,N_{obj}$
2: Initialize splitting parameter c, *pattern information*, and $t = 1$;
3: **while** ($t \leq T_{max}$ and performance margin has not been achieved)
4: $P = min(\frac{|\Omega_{t-1}|}{A_{min}}, P_{max})$
5: Conduct a Mondrian process to slice the input space into P partitions: $\mathcal{X} = \cup_{p=1}^{P} \mathcal{X}_p$, and distribute observations accordingly among sliced partitions: $\Omega_{t-1} = \cup_{p=1}^{P} \Omega_{t-1}^{p}$;
6: **for** $p = 1,\ldots,P$
7: Conduct Tile Coding to discretize Ω_{t-1}^{p} into feature vectors;
8: Derive c^p via Gibbs-UCB(c, *pattern information*); // Alg. 3
9: Construct $\text{TileGP}_p|_i$ using Ω_{t-1}^{p}, c^p, and feature vectors;
10: Use c^p to split D-dim. space to $\cup_{g=1}^{G} \mathcal{X}_p^{(\mathcal{S}_g)}$ sub-spaces;
11: Construct *Acq. Func.* $\Psi_{t-1}^{p}|_i$ by using Ω_{t-1}^{p}, c^p, and $\text{TileGP}_p|_i$;
12: **for** $g = 1,\ldots,G$
13: $\tilde{\boldsymbol{x}}_t^{p,(\mathcal{S}_g)} \leftarrow$ many-objective $\max_{\boldsymbol{x} \in \mathcal{X}_p^{(\mathcal{S}_g)}} \{\Psi_{t-1}^{p,(\mathcal{S}_g)}(\boldsymbol{x})|_i\}_{i=1}^{N_{obj}}$;
14: **end for**
15: **end for**
16: Apply correlation clustering on c^p to update c; //Merge c^p to c
17: **if** ($B > B_{max}$) // B: number of solutions from all P partitions
18: $\{\boldsymbol{x}_t^b\}_{b=1}^{B_{max}} \leftarrow NondominateSort\,\Psi(\boldsymbol{x})|_i^{N_{obj}}, \forall \boldsymbol{x} \in \{\tilde{\boldsymbol{x}}_t^p\}_{p=1}^{P}$;
19: **else**
20: Select all B solutions $\{\boldsymbol{x}_t^b\}_{b=1}^{B}$ as query points;
21: **end**
22: Perform multiple function evaluations to get $\{\boldsymbol{y}_t^b\} = f(\{\boldsymbol{x}_t^b\})$;
23: Update *pattern information* based on $\{\boldsymbol{x}_t^b, \boldsymbol{y}_t^b\}$ & related $\{c^p\}_{p=1}^{P}$;
24: Update the observation set $\Omega_t = \Omega_{t-1} \cup \{(\boldsymbol{x}_t^b, \boldsymbol{y}_t^b)\}_{b=1}^{N_{query}^t}$;
25: $t = t + 1$;
26: **end while**

ACOB:ANALOG CIRCUIT OPTIMIZATION BENCHMARKS

Another significant constraint is the absence of a unified metric. To address this, we have designed an open-source benchmark: ACOB, using analog circuits provided by the IEEE Standard Association. ACOB covers circuit optimization problems ranging from low-dimensional to high-dimensional, enabling researchers to evaluate their algorithms. We will continue to maintain and expand this benchmark with more state-of-the-art analog circuits in the future.



Reference

[1]G. D. Hachtel and R. Rohrer, “Techniques for the optimal design and synthesis of switching circuits,” *Proceedings of the IEEE*, vol. 55, no. 11,pp. 1864–1877, 1967.

[2]S. Director and R. Rohrer, “Automated network design-the frequency domain case,” IEEE Transactions on Circuit Theory, vol. 16, no. 3, pp. 330–337, 1969.

[3] S. P. Boyd, T. H. Lee et al., “Optimal design of a cmos op-amp via geometric programming,” IEEE Transactions on Computer-aided design of integrated circuits and systems, vol. 20, no. 1, pp. 1–21, 2001.

[4] M. del Mar Hershenson, “Design of pipeline analog-to-digital converters via geometric programming,” in Proceedings of the 2002 IEEE/ACM international conference on Computer-aided design, 2002, pp.

[5] W. Daems, “Simulation-based generation of posynomial performance models for the sizing of analog integrated circuits,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,

[6] B. Liu, “An efficient evolutionary algorithm for chance-constrained bi-objective stochastic optimization,” IEEE Transactions on Evolutionary Computation,

[7]P. P. Prajapati “Two stage cmos operational amplifier design using particle swarm optimization algorithm,” in 2015 IEEE UP Section Conference on Electrical Computer and Electronics (UPCON).

[8] Y. Yang “Smartmsp: A self-adaptive multiple starting point optimization approach for analog circuit synthesis,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[9] Y. Li,“An artificial neural network assisted optimization system for analog design space exploration,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

[10] A. F. Budak, “An efficient analog circuit sizing method based on machine learning assisted global optimization,” IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems

[11] Zhang J, Automated Design of Complex Analog Circuits with Multiagent based Reinforcement Learning[C]//2023 60th ACM/IEEE Design Automation Conference .

[12] W. Lyu, “An efficient bayesian optimization approach for automated optimization of analog circuits,” IEEE Transactions on Circuits and Systems I: Regular Papers

[13] A. Zhao, “cvts: A constrained voronoi tree search method for high dimensional analog circuit synthesis,” in 2023 60th ACM/IEEE Design Automation Conference (DAC).

[14]Touloupas K, Sotiriadis P P. LoCoMOBO: A Local Constrained Multiobjective Bayesian Optimization for Analog Circuit Sizing[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2021, 41(9): 2780-2793.

[15] C. Chen, H. Wang, X. Song, F. Liang, K. Wu and T. Tao, "High-Dimensional Bayesian Optimization for Analog Integrated Circuit Sizing Based on Dropout and gm/ID Methodology," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[16] T. Gu et al., "BBGP-sDFO: Batch Bayesian and Gaussian Process Enhanced Subspace Derivative Free Optimization for High-Dimensional Analog Circuit Synthesis," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

[17] T. Liao and L. Zhang, "High-Dimensional Many-Objective Bayesian Optimization for LDE-Aware Analog IC Sizing," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems,.

Thanks for Listening!