IEEE/ACM
2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN
ICCAD
43rd Edition

Is Vanilla Bayesian Optimization Enough for High-Dimensional Architecture Design Optimization?

Yuanhang Gao*, Donger Luo*, Chen Bai, Bei Yu,
Hao Geng, Qi Sun, Cheng Zhuo

2024.10.29

Outline

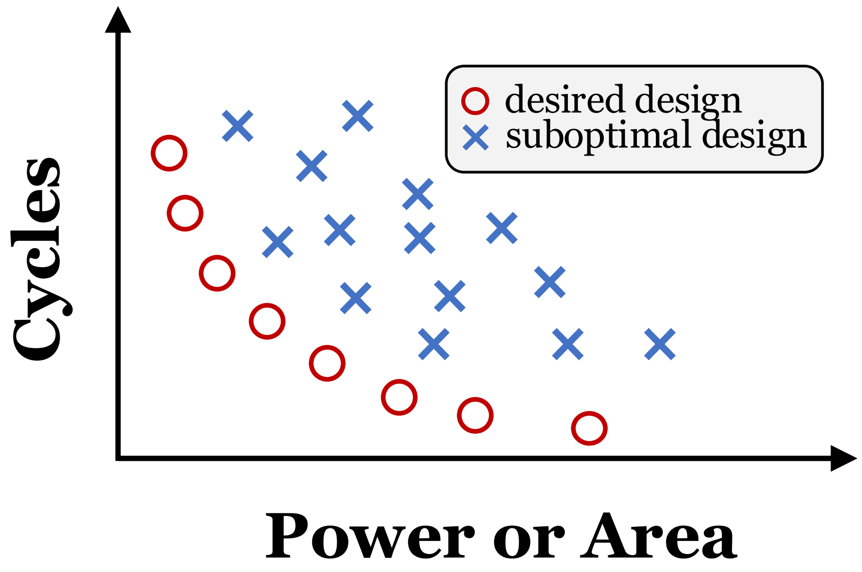
01

Background

Introduction

Microarchitecture Design Space exploration:

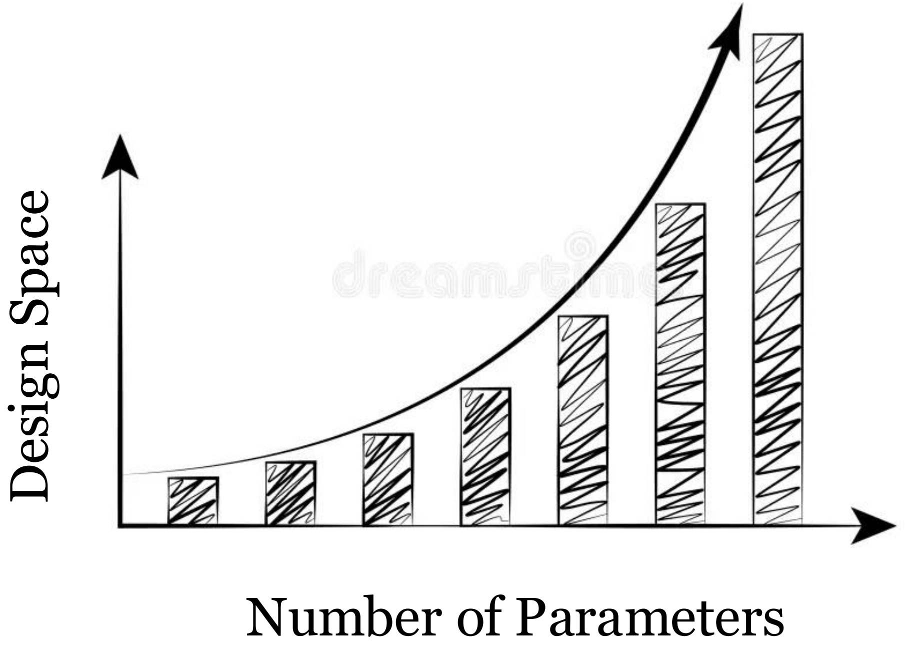
- The design of Microarchitecture could be considered as **setting the configurable parameters**
- Find configurations that **meet the desired performance, power, and area(PPA)**



Problem

Two major challenges:

- Design space **exponentially explode**.
- VLSI integration flow is **time-consuming**.



Previous Solutions

- Design parameters are **manually** configured by computer architects
- **Limitation:** extensive domain expertise and significant amount of human effort

- **Machine learning**-based process simulation models or surrogate models
- **Limitation:** black-box process and lacks effective guidance

Limitations

High-dimensional Problems

The **high-dimensional design parameters** and huge design space, which normally occur in the complicated SoCs for Large Language Model (LLM) tasks, pose a great challenge to existing techniques.

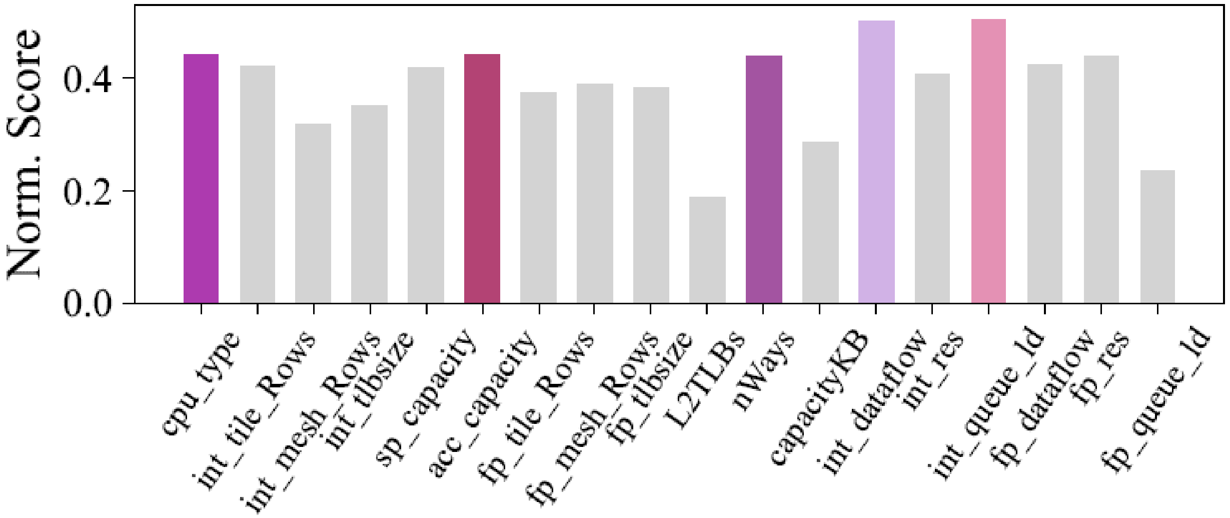
For example, an SoC could have 65 parameters and **$O(10^{30})$ design space**

Limitations of Previous Methods

Poor fitting effect for high-dimensional problem

Motivation

- **Some important parameters** affect the PPA values more while others contribute less to PPA values
- Those important parameters could be explored by **Monte-Carlo Tree Search(MCTS)**



02

Preliminary

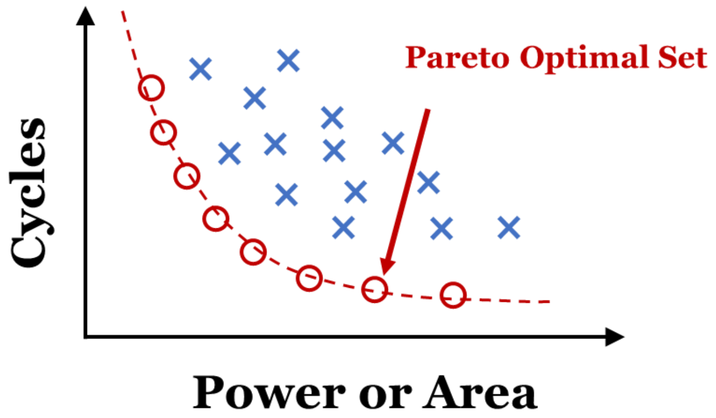
Problem Definition

Pareto Optimality:

Let functions $\{\mathbf{f}_i(\mathbf{x})\}_{i=1}^{m}$, *i.e.*, $\mathbf{f}_1$ = power, $\mathbf{f}_2$ = area, $\mathbf{f}_3$ = cycles, denote the m-dimension metrics to be minimized and $\mathbb{X}$ denotes the parameter space. $\mathbf{x}_1$ is said to (Pareto) dominate $\mathbf{x}_2$ ($\mathbf{x}_1 \succeq \mathbf{x}_2$) if

$$\mathbf{f}_i(\mathbf{x}_1) \leq \mathbf{f}_i(\mathbf{x}_2), \forall i \in \{1, ..., m\},$$
$$\mathbf{f}_i(\mathbf{x}_1) < \mathbf{f}_i(\mathbf{x}_2), \exists i \in \{1, ..., m\}.$$

Pareto-optimal set $\mathbb{X}^*$: The collection of parameter vectors that are not dominated by others.



Design Space Exploration:

Given a search space $\mathbb{X}$, each microarchitecture design inside $\mathbb{X}$ is regarded as a feature vector $\mathbf{x}$. Metric space is $\mathbb{Y} = \{\mathbf{y} | \mathbf{y} = \mathbf{f}(\mathbf{x}), \mathbf{x} \in \mathbb{X}\}$. The objective is to find the subset $\mathbb{X}^* \subset \mathbb{X}$ forming the Pareto-optimal set.

03

MCT-Explorer

Framework

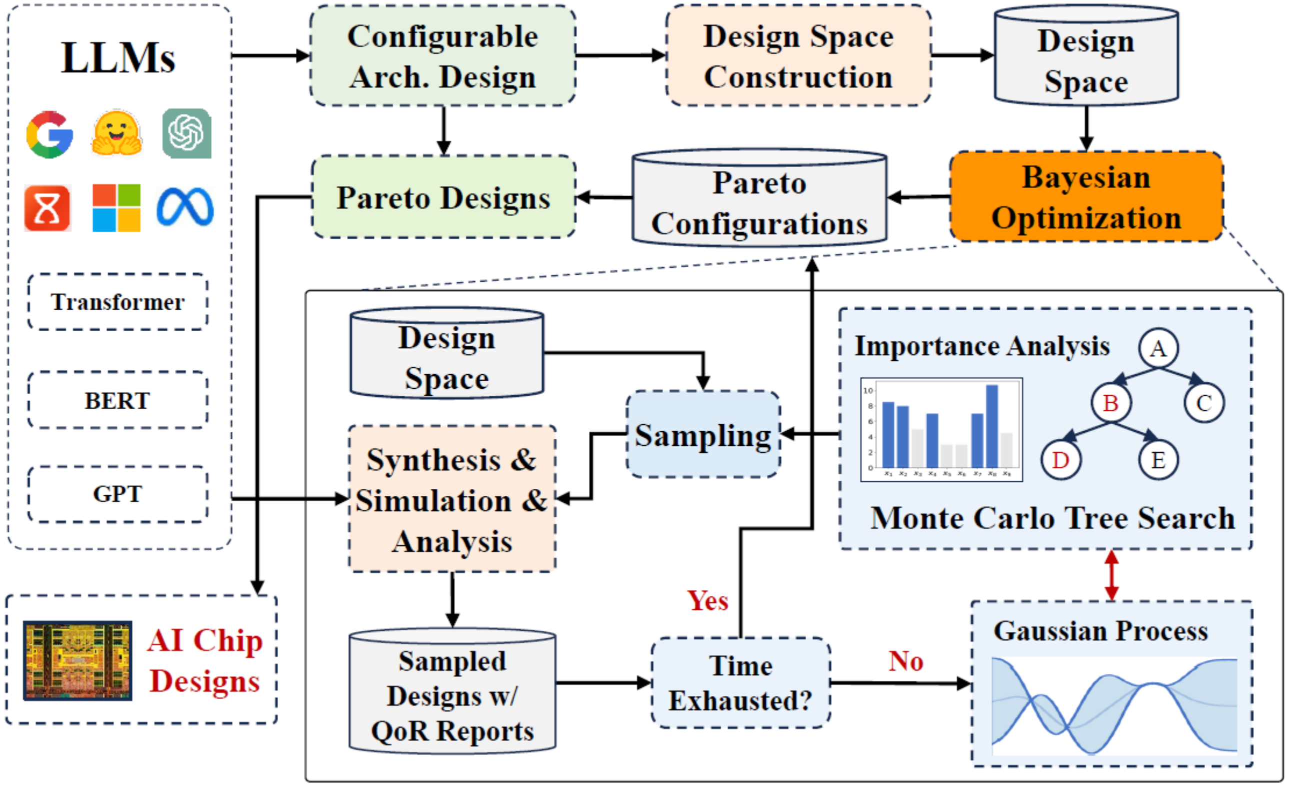


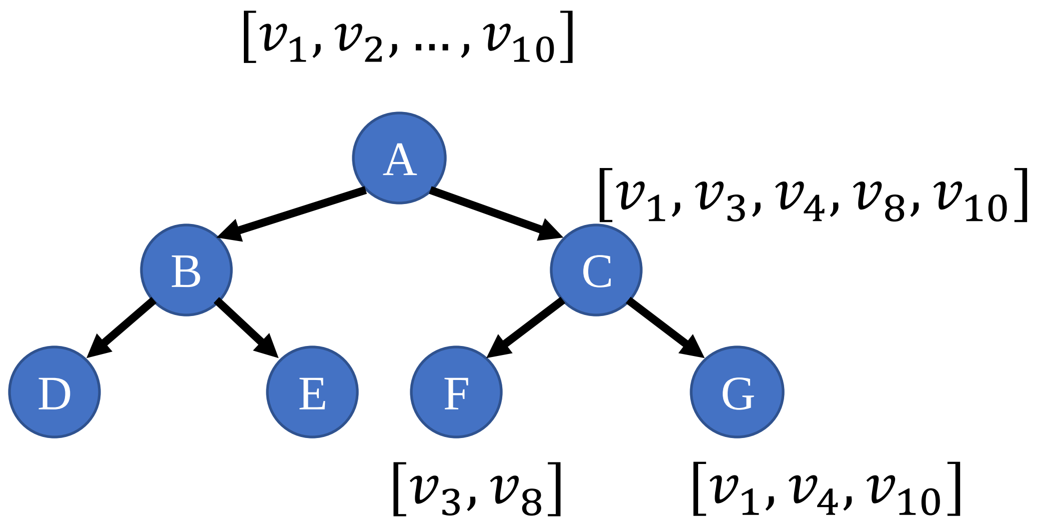
Figure 1: Overview of our MCT-Explorer.

- Utilize the Monte Carol Tree Search to select important parameters
- Performing Bayesian Optimization(BO) according to partial parameters
- Mitigating the issue of inaccurate fitting in high-dimensional Bayesian Optimization

Custom Monte Carlo Tree Search Variants(1)

Node: represent a set of parameter index

Selection: choose the next node
Node split: split the parameter indexes
Analysis: acquire new candidates points and update parameter score
Back-propagation: divide parameter indexes into child nodes



Custom Monte Carlo Tree Search Variants(2)

Upper Confidence Bound(UCB): determine which branch to choose

$$\mathrm{UCB}(X) = v_X + 2C_p\sqrt{2(\log n_p)/n_X},$$

- First term: evaluates the **average importance score** of parameters in the node
- Second term: **encourages exploration** of less visited nodes

The first term is computed according to the global importance score **s**

$$v_X = \mathbf{s} \cdot g(\mathbb{A}_X)/|\mathbb{A}_X|,$$

where $\mathbb{A}_X$ is the set of parameter indexes in node X

Global Importance Score

Importance score s:

a vector with length equal to the total number of variables

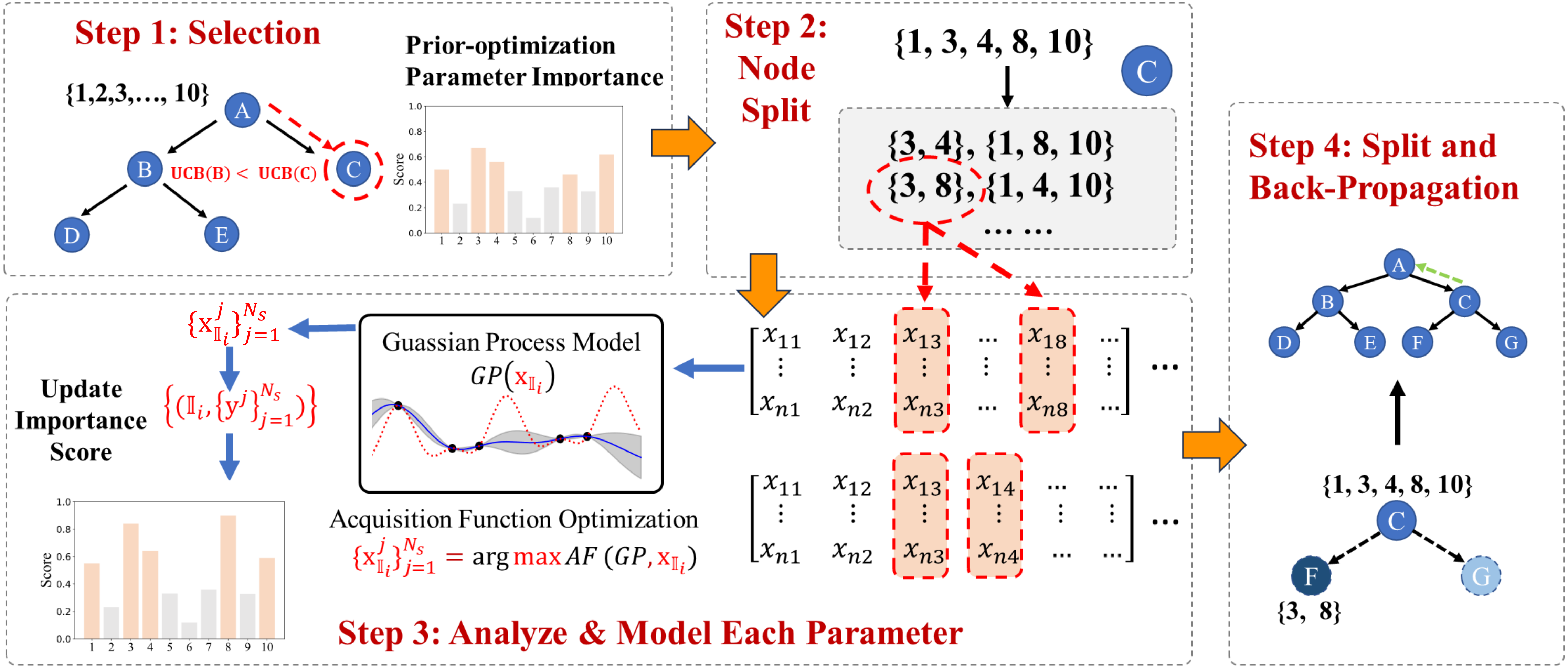
$$\mathbf{s} = \frac{\sum_{(\mathbb{I},\mathbb{M})\in\mathbb{T}} \sum_{\mathbf{y}\in\mathbb{M}} \mathrm{HV}(\mathbf{f}^{\mathrm{ref}}, \mathbf{y}) \cdot g(\mathbb{I})}{\sum_{(\mathbb{I},\mathbb{M})\in\mathbb{T}} |\mathbb{M}| \cdot g(\mathbb{I})}$$

element-wise division

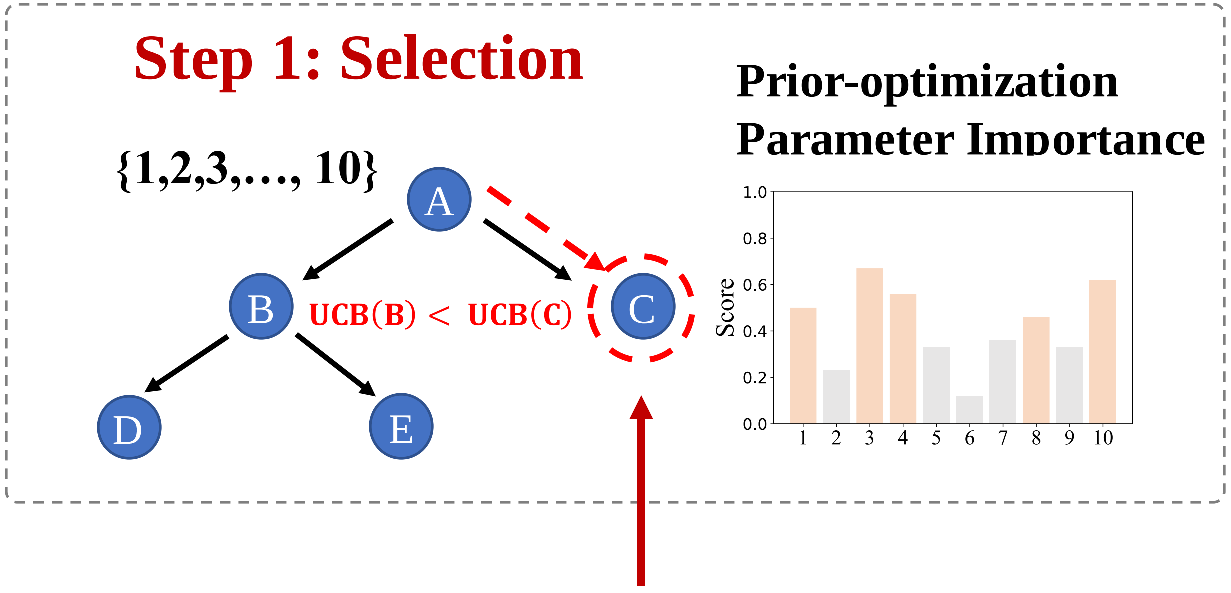
- Numerator : **the sum of contributions** of each parameter
- Denominator : **the frequency** of each parameter participates in obtaining new candidates

One Iteration of MCTS

IEEE/ACM 2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN ICCAD 43rd Edition

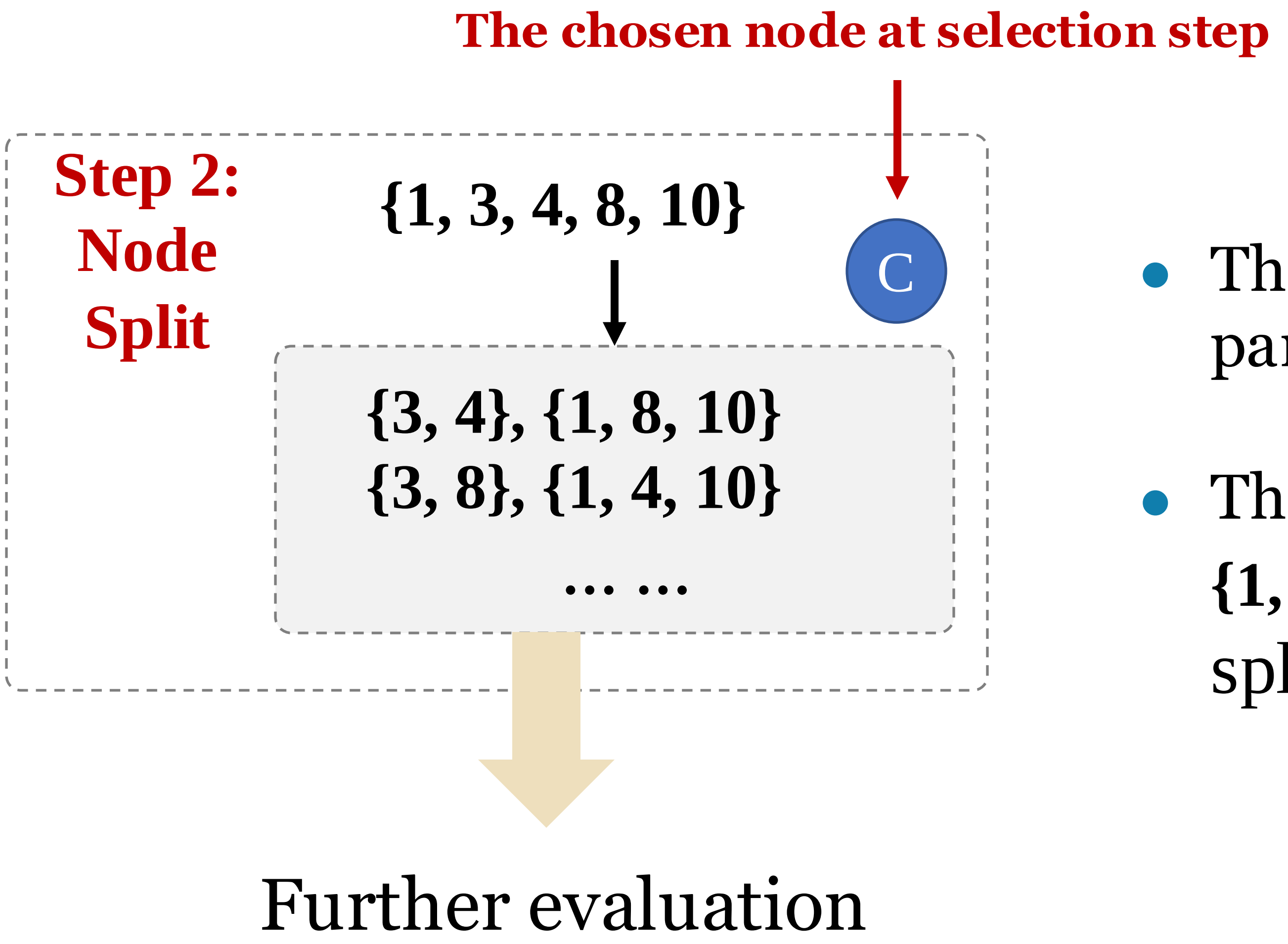


Selection



- Root Node A represents all parameter indexes **{1,2,3,…, 10}**
- Node C is chosen at this iteration

Node Split



- The chosen node C represents parameter indexes **{1, 3, 4, 8, 10}**
- The parameter indexes inside **{1, 3, 4, 8, 10}** are randomly split into several subsets

Analyze & Model Each Parameter

Reduced microarchitecture embedding: $\mathbf{x}_{\mathbb{I}}$

For example, given $\mathbf{x} = \underbrace{(2, 4, 8, \ldots, 3, 4, 4)}_{\text{10-dimension}}$ and $\mathbb{I} = \{3, 8\}$, $\mathbf{x}_{\mathbb{I}}$ is $(8, 3)$

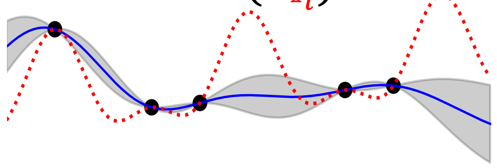
{3, 4}, {1, 8, 10}
{3, 8}, {1, 4, 10}
… …

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{18} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & x_{n3} & \cdots & x_{n8} & \cdots \end{bmatrix} \cdots$$

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & x_{n3} & x_{n4} & \cdots & \cdots \end{bmatrix} \cdots$$

Guassian Process Model

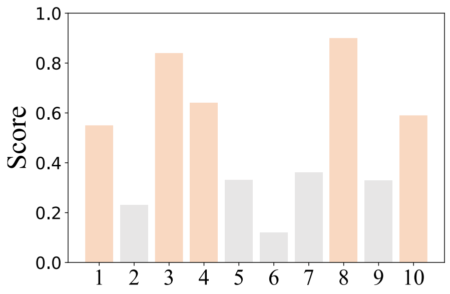
$GP(\mathbf{x}_{\mathbb{I}_i})$



$\{\mathbf{x}_{\mathbb{I}_i}^{j}\}_{j=1}^{N_s}$

Update Importance Score

$\{(\mathbb{I}_i, \{\mathbf{y}^j\}_{j=1}^{N_s})\}$



Acquisition Function Optimization

$$\{\mathbf{x}_{\mathbb{I}_i}^{j}\}_{j=1}^{N_s} = \arg\max AF\left(GP, \mathbf{x}_{\mathbb{I}_i}\right)$$

Step 3: Analyze & Model Each Parameter

Acquire New Candidates

Surrogate model: Gaussian Process

$$\begin{bmatrix} \mathbf{y} \\ f' \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \mu_* \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{\mathbf{X}_\mathbb{I}\mathbf{X}_\mathbb{I}|\theta} + \sigma_e^2 \mathbf{I} & \mathbf{K}_{\mathbf{X}_\mathbb{I}\mathbf{x}'_\mathbb{I}|\theta} \\ \mathbf{K}_{\mathbf{x}'_\mathbb{I}\mathbf{X}_\mathbb{I}|\theta} & k_{\mathbf{x}'_\mathbb{I}\mathbf{x}'_\mathbb{I}|\theta} \end{bmatrix} \right).$$

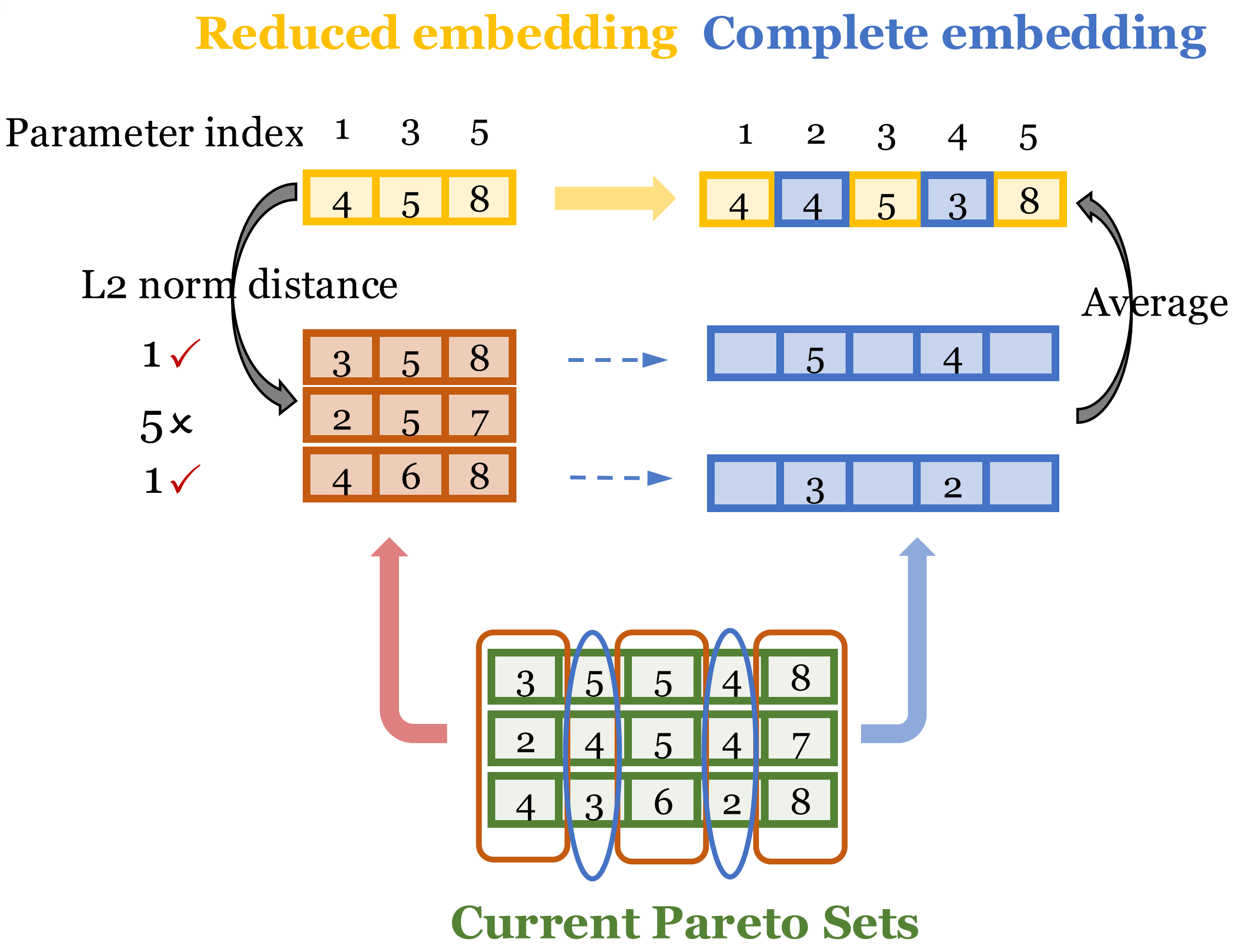
Acquisition function: Joint entropy search

maximize

$$\begin{aligned} I(\mathbf{x}'_\mathbb{I}|\mathbf{X}_\mathbb{I}, \mathbf{Y}) =& MI(\mathbf{y}; (\mathbb{X}^*_\mathbb{I}, \mathbb{Y}^*)|\mathbf{x}'_\mathbb{I}, \mathbf{X}_\mathbb{I}, \mathbf{Y}) \\ =& H[p(\mathbf{y}|\mathbf{x}'_\mathbb{I}, \mathbf{X}_\mathbb{I}, \mathbf{Y})] \\ & - \mathbb{E}_{p((\mathbb{X}^*_\mathbb{I}, \mathbb{Y}^*)|\mathbf{X}_\mathbb{I}, \mathbf{Y})} [H[p(\mathbf{y}|\mathbf{x}'_\mathbb{I}, \mathbf{X}_\mathbb{I}, \mathbf{Y}, \mathbb{X}^*_\mathbb{I}, \mathbb{Y}^*)]] \\ \approx& H[p(\mathbf{y}|\mathbf{x}'_\mathbb{I}, \mathbf{X}_\mathbb{I}, \mathbf{Y})] \\ & - \frac{1}{S} \sum_{s=1}^{S} H[p(\mathbf{y}|\mathbf{x}'_\mathbb{I}, \mathbf{X}_\mathbb{I}, \mathbf{Y}, \mathbb{X}^*_{\mathbb{I},s}, \mathbb{Y}^*_s)], \end{aligned}$$

Monte Carlo Sampling

Fill in Reduced Embedding

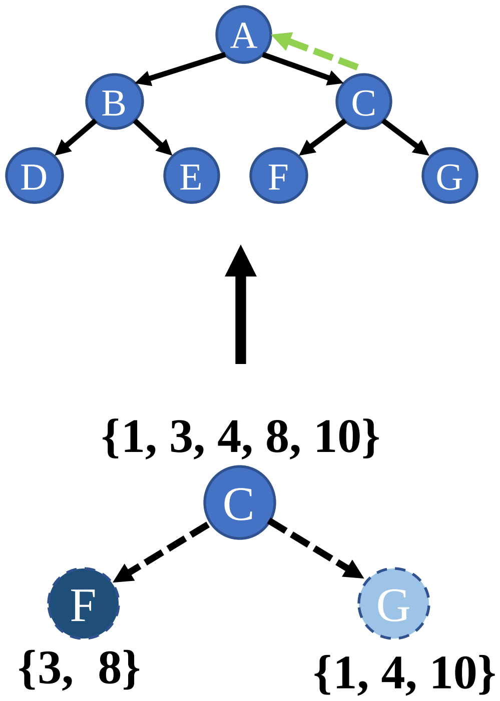


- The candidates acquired from BO is not complete
- We fill in the missing part using data from current Pareto Sets
- Search for the closest possible alternative within design space

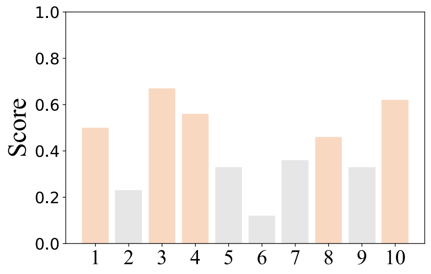
Split and Back-Propagation

IEEE/ACM ICCAD
2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN
43rd Edition

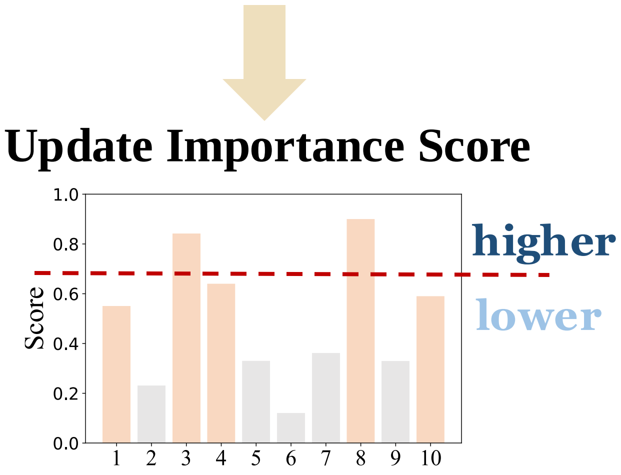
Step 4: Split and Back-Propagation



Prior-optimization Parameter Importance



Update Importance Score



- Parameter importance scores are updated according to newly acquired points
- Parameter indexes inside node C are split into two child nodes

The Complete Algorithm

IEEE/ACM 2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN ICCAD 43rd Edition

Algorithm 3 **MCT-Explorer**$(N_v, N_s, k, T, \mathbb{X}, \mathbb{D})$

Input: N_v is the batch size of parameter index subsets, N_s is the sample batch size, k is the number of Pareto configurations to fill in the absent part, T is the total time budget for EDA_flow, $\mathbb{X}$ is the design space, $\mathbb{D}$ is the index set of SoC parameters.

Output: $\mathbb{X}^*$: Pareto-optimal designs.

1: **for** $i = 1 : N_v$ **do**
2: $\quad \mathbb{I}_i$ = sampled index subset from $\mathbb{D}$;
3: $\quad \bar{\mathbb{I}}_i = \mathbb{D} \setminus \mathbb{I}_i$;
4: $\quad \mathbb{X}_i = \{\mathbf{x}_j\}_{j=1}^{N_s}$ sampled from $\mathbb{X}$; $\bar{\mathbb{X}}_i = \{\mathbf{x}_j\}_{j=1}^{N_s}$ sampled from $\mathbb{X}$;
5: $\quad \mathbb{M}_i$ = EDA_flow($\mathbb{X}_i$); $\bar{\mathbb{M}}_i$ = EDA_flow($\bar{\mathbb{X}}_i$);
6: $\quad T = T -$ runtime_overhead of EDA_flow ;
7: **end for**
8: $\mathbb{T} = \{(\mathbb{I}_i, \mathbb{M}_i), (\bar{\mathbb{I}}_i, \bar{\mathbb{M}}_i)\}_{i=1}^{N_v}$;
9: $\mathbb{X}^*$ = Current Pareto Set;
10: Calculate the parameter score $\mathbf{s}$ using $\mathbb{T}$ by Equation (5);
11: Initialize the Monte Carlo Tree;
12: **while** $T > 0$ **do**
13: $\quad X$ = the leaf node selected by UCB;
14: $\quad T_X, \mathbb{T}$ = Node-Analysis$(X, \mathbb{T}, \mathbb{X}^*, N_v, N_s, k)$; ▷ Algorithm 2
15: $\quad T = T - T_X$;
16: $\quad \mathbb{X}^*$ = Updated Pareto Set;
17: $\quad$ Back-propagate to update the UCB value of ancestor node;
18: **end while**
19: **return** Pareto-optimal designs $\mathbb{X}^*$;

Initialization (lines 1-11)

- Initialize global score **s**
- Initialize the Monte Carlo Tree

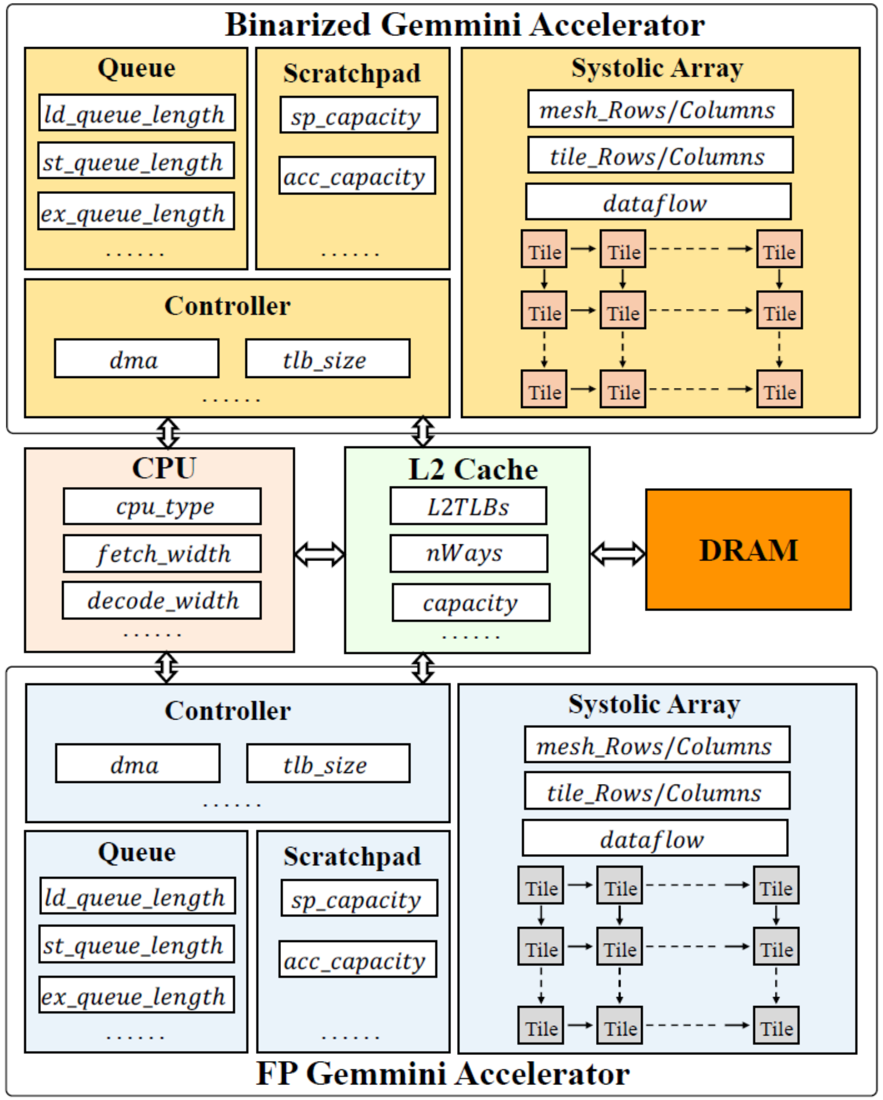
Iteration of MCTS (lines 13-17)

- Selection
- Node-Analysis
- Update Pareto Set
- Back-propagation

04

Experiments

Experiment Setting



Three dataset

- Single-Gemmini SoC dataset 1124 designs (**19 parameters**)
- Dual-Gemmini SoC dataset 1035 designs (**65 parameters**)
- In-house dataset contains 1300 designs (**270 parameters**)

Each dataset consumed 3000 to 5000 CPU hours to run syntheses and simulations to obtain power, performance, and area (PPA) values.

Chipyard, Cadence Genus and PrimeTime were used to get PPA reports

[A. Amid, Micro, 2021], [H. Genc, DAC, 2021]

Evaluation Metrics

IEEE/ACM 2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN ICCAD 43rd Edition

Hypervolume (HV):

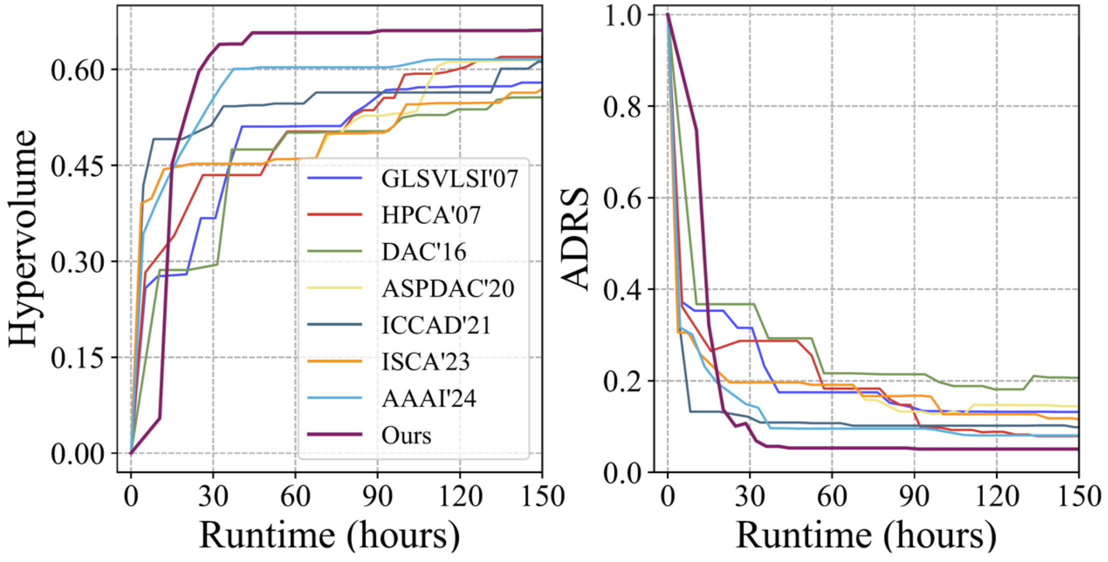
$$\mathrm{HV}\left(\mathbf{f}^{\mathrm{ref}}, \mathbf{X}\right) = \Lambda\left(\bigcup_{\mathbf{x}\in\mathbf{X}} \left[\mathbf{f}_1\left(\mathbf{x}\right), \mathbf{f}_1^{\mathrm{ref}}\right] \times \cdots \times \left[\mathbf{f}_m\left(\mathbf{x}\right), \mathbf{f}_m^{\mathrm{ref}}\right]\right),$$

Average distance to reference set (ADRS):

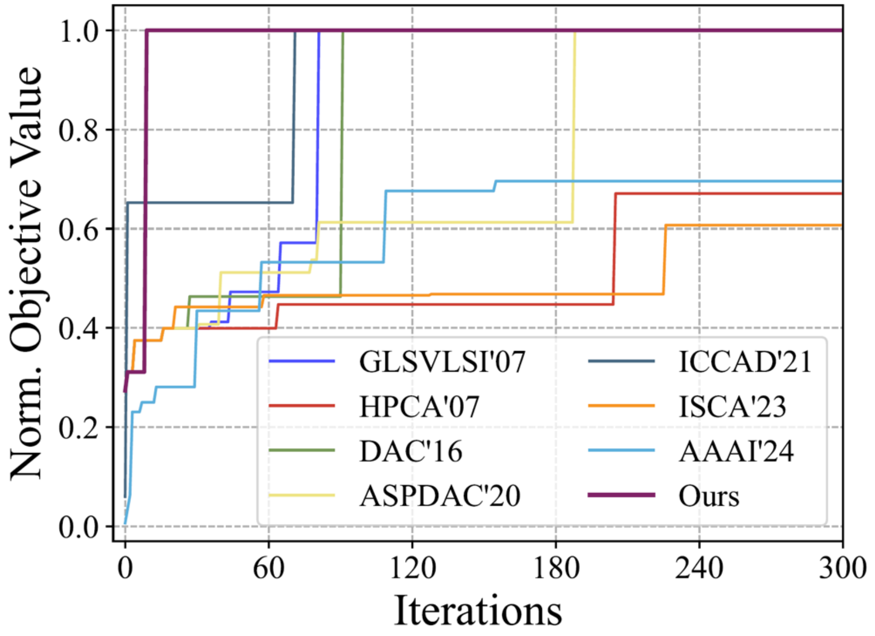
$$\mathrm{ADRS}(\Gamma, \Omega) = \frac{1}{|\Gamma|} \sum_{\gamma\in\Gamma} \min_{\omega\in\Omega} dist(\gamma, \omega),$$

Comparison on Hypervolume and ADRS

Dual-Gemmini SoC dataset



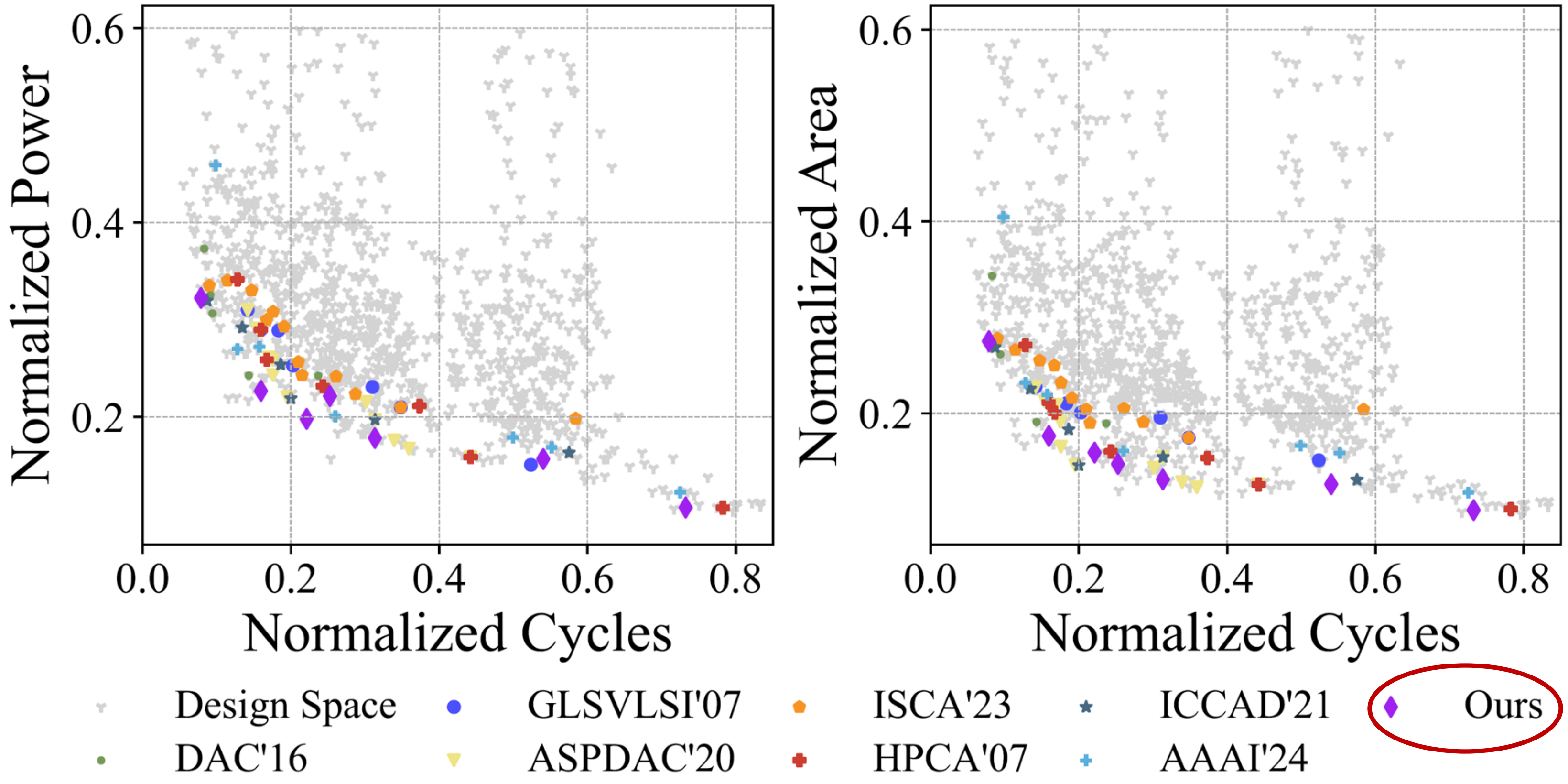
In-house dataset



Outperform **30.9%** with only **33.3% runtime** overhead in ADRS metric

[M. Barros, GLSVLSI, 2007] , [B. Lee, HPCA, 2007], [D. Li, DAC, 2016], [Z. Xie, ASPDAC, 2020], [C. Bai, ICCAD, 2021], [S. Krishnan, ISCA, 2023], [C. Bai, AAAI, 2024]

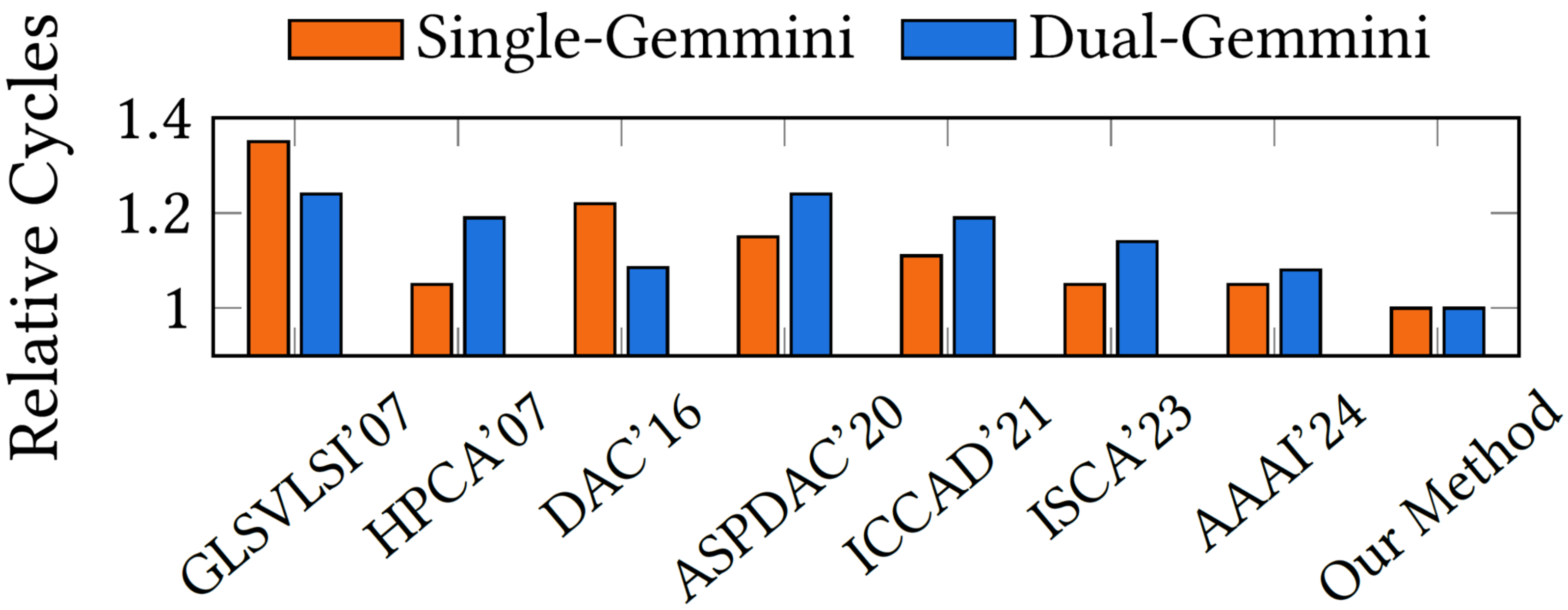
Learned-Pareto Sets



Nearly **outmost** Pareto Sets

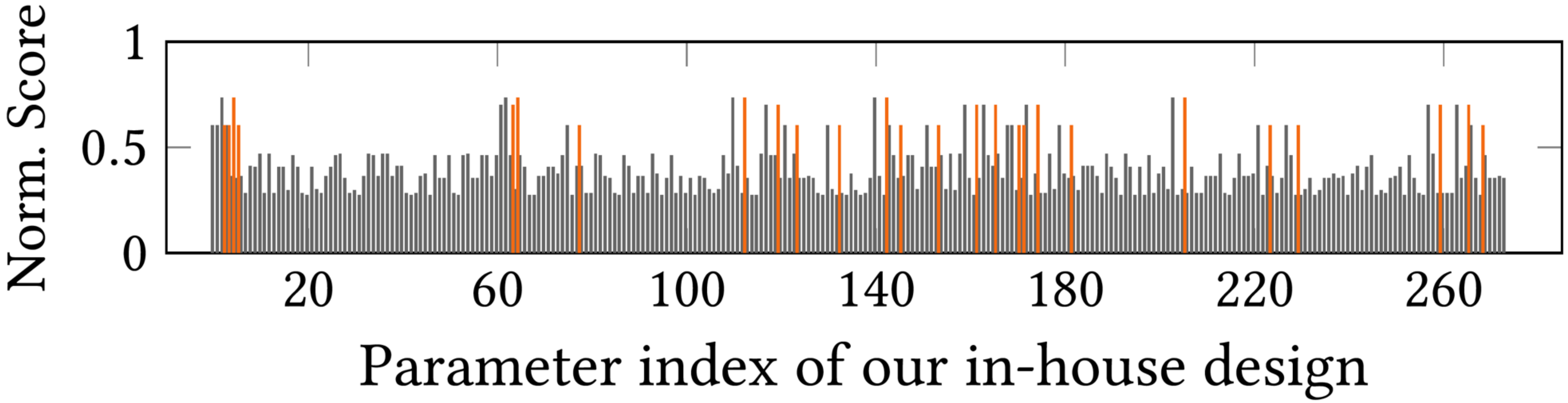
Relative Cycles of LLM Tasks on the Found Pareto-optimal Sets

IEEE/ACM
2024 INTERNATIONAL CONFERENCE ON COMPUTER-AIDED DESIGN
ICCAD
43rd Edition



Achieve **less cycles** on LLM tasks.

The Importance Score of Parameters



Among **270** parameters, **32** score higher than others

Thanks!