

T-MSA: Transformer-Driven Multi-Strategy Adaptive Microarchitecture Design Space Exploration

Jingjing Wang¹, Zihan Lin², Fan Yang², Xiaochuan Li¹, Runze Zhang¹, Cong Xu¹,

Rengang Li^{4,1,*}, Baoyu Fan^{3,1,*}

¹ IEIT SYSTEMS Co., Ltd. ² Fudan University ³ Nankai University ⁴ Tsinghua University

* Corresponding author: lrg@ieisystem.com, fanbaoyu@ieisystem.com

Abstract—The design of modern processors ignores the topological relationships among all design parameters, leading to significant simulation costs wasted on invalid designs. Therefore, we propose the T-MSA to address this issue. It is a Transformer-driven multi-strategy adaptive design space exploration scheme. A customized lightweight Transformer (LiteFormer) is devised to model topological relationships among arbitrary design parameters, constructing an implicit interaction graph in the latent space. Secondly, we design a dynamic active learning (DynamicAL) strategy to extract sparse and high-quality initial points via sparse centroid initialization and hybrid sampling. Finally, a triple Pareto frontier acquisition function (TriPFAF) is devised to guide optimization direction based on gains from three types of Pareto frontiers, dynamically balancing exploration and exploitation. We conducted rigorous experiments on two BOOM evaluation platforms, demonstrating that T-MSA efficiently and comprehensively optimizes the performance-power-area (PPA) objective. The designs it identifies achieve significant improvements over state-of-the-art DSE algorithms on Pareto hypervolume (HV). When attaining the same HV value, T-MSA outperforms BOOM-Explorer by 188.24% and 133.33% on two platforms.

Index Terms—design space exploration, Transformer, dynamic active learning, TriPFAF function, Bayesian optimization

I. INTRODUCTION

The complexity of modern microprocessor design, packed with features like branch predictors, caches, and specialized buffers, leads to a vast design space. Consequently, microarchitecture design space exploration (DSE) faces two fundamental challenges—combinatorial explosion and slow, costly performance evaluations. Spaces like that of RISC-V Berkeley Out-of-Order Machine (BOOM) [1] or high-performance multi-core architectures are too immense to traverse comprehensively. Furthermore, simulating each design point individually, given the time required per simulation, is entirely impractical.

Manual design space exploration relies on architects' expertise to navigate processor configuration trade-offs. While invaluable for early-stage optimization, this approach struggles with the scalability demands of brand-new processors. To accelerate architectural evaluation, GEM5 [2] are widely used at the cost of reduced modeling precision compared to register-transfer level (RTL) circuit evaluation [3]. Some techniques are also proposed to enhance simulation efficiency, such as clustering algorithm [4], sensitivity analysis [5] and so on. Though the simulation speed is accelerated, the inaccuracy poses potential threat to effective evaluation, which may lead to suboptimal results of microarchitecture.

Recent advancements in machine learning (ML) have opened new possibilities for processor microarchitecture design space

exploration. Some methods [6], [7] predict power-performance-area (PPA) metrics by training surrogate models on simulation-derived datasets. However, extensive training requirements demand massive simulation costs (e.g., 2.5K samples for multiprocessor modeling). Considering the potential of injecting prior knowledge to reduce simulation cost, a previous study [8] takes full advantage of heuristic microarchitecture knowledge and realizes exploration based on *Markov decision process*, effectively avoiding unnecessary simulations of certain design points. BOOM-Explorer [9] takes a further research on sampling algorithm to segment the design space by key microarchitecture parameters. Prior-boosted GRL [10] introduces graph representation to improve the modeling of variables dependencies and designs a prior knowledge-based initial sampling method, providing better focused targeting of the optimal point. However, it relies on a prebuilt microarchitecture interdependence graph [11] and lacks the ability to focus on complicated distant cross variable dependencies. Despite these limitations, these studies still demonstrate the immense potential of Bayesian optimization (BO) [12] in design space exploration.

Specifically, these challenges of microarchitecture design space exploration remain unresolved:

Complexity of variables dependencies. The numerous configurable variables of modern cores exhibit nonlinear, cross-dimensional dependencies, where localized tuning may trigger unforeseen bottlenecks elsewhere. Traditional encoding methods and even graph-based representations [11] struggle to capture these full cross-variable dependencies, leading to partial modeling and information loss. The mutual constraints critically complicates the problem, posing great challenges to appropriate modeling.

Inefficiency of initial sampling. High-quality initial sampling is foundational for efficient microarchitecture design space exploration, serving as the critical first step in accelerating optimization. Common random and k-means sampling [13] ignores Pareto-critical regions, leading to incomplete frontier discovery. Moreover, existing static strategies fail to adapt sampling density to evolving optimization needs, giving rise to inefficiency and ineffectiveness of microarchitecture design space exploration.

Inadaptability of single acquisition function. A critical challenge in DSE is the inflexibility of standard acquisition functions. Common approaches like expected improvement (EI) [14] and expected hypervolume improvement (EHVI) [15] employ fixed criteria that fail to adapt to evolving requirements across different stages. An outstanding acquisition function

should enable automatic and dynamic balancing between exploration and exploitation.

To address the above issues, this paper proposes T-MSA, a Transformer-driven multi-strategy adaptive DSE framework. Detailed contributions are as follows:

- To address the challenge in capturing complete cross-variable dependencies, a custom lightweight Transformer (LiteFormer) framework is designed. It constructs implicit interaction graphs in the latent space through multi-head attention, and enables direct quantification of coupling strengths among arbitrary variables.
- To ensure comprehensive coverage of sampling points, a dynamic active learning strategy (DynamicAL) is proposed. Sparse centroid initialization combined with hybrid sampling based on transductive experimental design (TED) and density features ensures comprehensive coverage of sampling points while avoiding local optima.
- To deal with the inflexibility of single acquisition functions, a stage-aware triple Pareto frontier acquisition function (TriPFAF) is devised. It dynamically considers three types of gains related to the Pareto frontier to balance exploration and exploitation, intelligently guiding the decision-making process in multi-objective Bayesian optimization.
- We comprehensively evaluate T-MSA on two BOOM evaluation platforms. Experimental results demonstrate that T-MSA significantly outperforms existing DSE algorithms in terms of both design quality and runtime. At the same Pareto hypervolume, it outperforms BOOM-Explorer by 188.24% and 133.33% on two platforms, respectively.

The remainder of this paper is organized as follows. Section II shows prior knowledge and defines the microarchitecture design space exploration problem. Section III proposes our algorithms. Section IV presents the experimental results. Section V concludes this paper.

II. PRELIMINARIES

A. RISC-V BOOM Microarchitecture

The BOOM [1] represents an open-source, superscalar RISC-V microprocessor implementation renowned within computer architecture research for its empirically validated PPA efficiency, particularly in low-power embedded contexts. Its microarchitecture organizes instruction processing across ten pipeline stages, functionally partitioned into three primary modules: the frontend, execution units (EU), and load-store unit (LSU). A defining characteristic of BOOM is its extensive parameterization across these modules. Key microarchitectural parameters, such as pipeline width, reorder buffer (ROB) entries, cache configurations (I-Cache/D-Cache associativity, block size, TLB structure), and load-store queue depth—collectively define a vast design space. However, this space is constrained by microarchitectural dependencies; for instance, fetch width must equal or exceed decode width, and ROB entries must be divisible by decode width. The ICCAD 2022 Contest Problem C [16] quantified this space with 29 interdependent parameters, yielding approximately 1.5×10^4 valid configurations.

B. Multi-Objective Bayesian Optimization

Multi-objective Bayesian optimization (MOBO) [17] extends BO principles to efficiently navigate trade-offs between competing objectives. This methodology is essential for high-dimensional design spaces where evaluating all configurations is computationally infeasible. A probabilistic surrogate model is used to approximate the expensive-to-evaluate objective functions, jointly characterizing prediction uncertainty across the entire Pareto front. Multi-criteria acquisition functions will quantify the expected improvement in Pareto hypervolume or dominance likelihood, transforming multi-objective trade-offs into analytically tractable scalarizations, strategically determining subsequent evaluation points to accelerate convergence. The decision-theoretic mechanism maximizes information gain per evaluation while progressively refining the estimated Pareto frontier.

C. DSE Problem Formulation

Microarchitectural design space exploration targets designs maximizing PPA trade-off efficiency within constrained parameter spaces, which is a complex and negatively correlated multi-objective optimization. The problem can be formulated as follows:

$$\min f(x) = \{-Performance(x), Power(x), Area(x)\} \quad (1)$$

where $f(x)$ denotes the objective function of DSE, x represents parameters in the whole design space. Pareto-optimal solution sets constitute the fundamental objective of multi-objective optimization, where no individual solution can be improved in any objective without degrading another. The collection of non-dominated points intrinsically defines the multi-objective problem resolution, with each point representing a distinct trade-off configuration. By extension, microarchitecture design space exploration seeks to identify corresponding Pareto-optimal configurations across competing design parameters.

III. ALGORITHM

A. Overview

In this paper, we propose T-MSA to rapidly and efficiently conduct design space exploration. It incorporates three strategies, a lightweight Transformer (LiteFormer) framework, a dynamic active learning (DynamicAL) approach, and a triple Pareto frontier acquisition function (TriPFAF). The overall framework of T-MSA is illustrated in Fig. 1.

The Fig. 1(a) LiteFormer converts the original discrete design space into a continuous one to facilitate BO. More importantly, it introduces interdependent relationships among design variables in the coding space, fully considering the mutual influences among variables (such as DecodeWidth, ROBEntries, and ICacheSets). Fig. 1(b) illustrates the DynamicAL strategy, which applies sparse centroid initialization to guide the dynamic k-means clustering process and designs hybrid sampling to ensure the representativeness of sampling points through transductive experimental design (TED) [18] and density features. Subsequently, a BO process is conducted, employing an ensemble surrogate model to describe the relationship between the LiteFormer encoding space and PPA. Fig. 1 (c) TriPFAF acquisition function considers three dimensions:

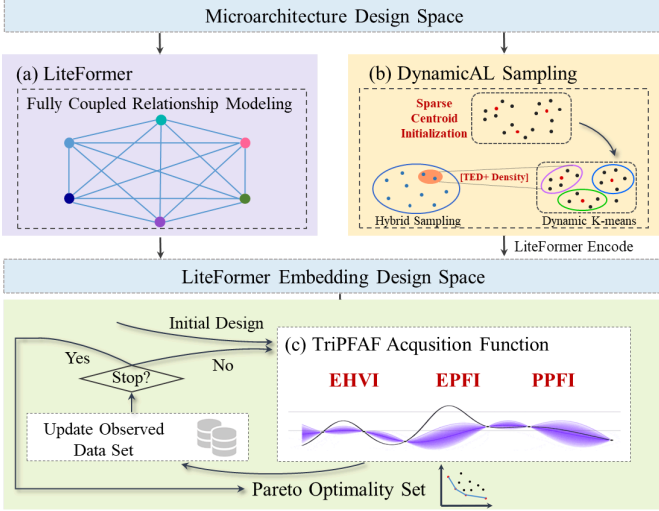


Fig. 1: The overall framework of T-MSA.

expected hypervolume improvement (EHVI), expected Pareto frontier improvement (EPFI), and probability of Pareto frontier improvement (PPFI), dynamically guiding the selection of the next sample point. The PPA metrics are obtained via RTL-level simulation tools. After completing the predefined N iterations, T-MSA extracts the Pareto optimal design as the final result.

B. LiteWeight Transformer Modeling

There are significant interactions among the design parameters of the BOOM core, and these coupling relationships can lead to nonlinear changes in PPA. For instance, increasing the FetchWidth requires a larger DecodeWidth to ensure the successful compilation of Verilog code, and a larger DecodeWidth must be paired with a larger IssueWidth and RobEntry, otherwise the decoded instructions will be blocked. Due to these factors, LiteFormer constructs an implicit interaction graph in the latent space, which can directly quantify the coupling strength among arbitrary variable pairs, as shown in Fig. 2.

Specifically, the LiteFormer framework consists of two components: the encoder and the decoder. Firstly, the design combinations of the BOOM processor are subjected to embedding and positional encoding. Subsequently, after being processed by N encoders, the resulting matrix contains fully coupled information among all design parameters. Each encoder consists of two main modules, the multi-head self-attention and the feed-forward network. The multi-head self-attention layer calculates the correlation between different positions in the input sequence through the self-attention mechanism, which is the core of LiteFormer’s implementation.

All features of each design undergo normalization before entering the attention module to generate query (Q), key (K), and value (V) matrices. The multi-head self-attention mechanism then produces the corresponding weighted output matrix, resulting in the improvement of the LiteFormer’s ability to represent global design features. The calculation of multi-head attention can be formulated as:

$$Attention(Q_i, K_i, V_i) = softmax\left(\frac{Q_i \cdot K_i^T}{\sqrt{d_{head}}}\right) \cdot V_i \quad (2)$$

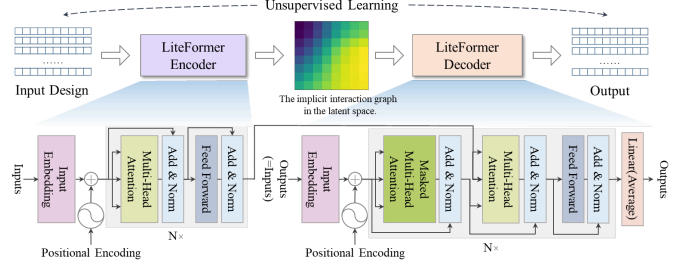


Fig. 2: Microarchitecture interdependence modeling based on LiteFormer.

where i is the index of the attention head, and Q_i , K_i , and V_i stand for the i -th head of Q , K , and V , respectively. d_{head} denotes the dimension of the attention head, and $\sqrt{d_{head}}$ is used to mitigate gradient vanishing.

A standard Transformer decoder [19] consists of N identical layers, each containing three submodules, a masked self-attention mechanism, a cross-attention module, and a feed-forward network. For the decoded information matrix, we perform mean processing along each design parameter dimension, which aligns the decoder’s target sequence with the encoder’s input encoded sequence.

C. Dynamic Active Learning

In DSE, a satisfactory set of initial data can enable the surrogate model to establish a relatively accurate global approximation at minimal cost. Therefore, the initial sampling points are required to provide as much global information about the exploration space as possible. A simple solution is random selection [20]. In classical k-means clustering [21], random initialization of centroids leads to local optima issues. Our proposed DynamicAL addresses this issue by adequately exploring the entire design space through sparse centroid initialization and hybrid sampling, as detailed in Algorithm 1.

In sparse centroid initialization, the first cluster center is randomly selected. Then, the minimum distance between each sample point and the existing cluster centers is calculated, and the distance is normalized into a probability distribution. The newly obtained centroid tends to fill the sparse region in the exploration data. In the dynamic k-means clustering process, DecodeWidth affects the execution bandwidth of EU and LSU (i.e., instructions executed per clock cycle). Bai et al. [9] demonstrated that microarchitectures with the same DecodeWidth can achieve similar power-performance ratios. It inspires us to group according to DecodeWidth in DynamicAL. Setting larger candidate values for DecodeWidth and allocating a balanced amount of hardware resources can separate the power-performance space along a potential Pareto optimal path.

Within each cluster, perform hybrid sampling based on TED and density uncertainty. TED measures sample similarity via a Gaussian kernel and dynamically updates sample importance weights based on matrix dimension reduction. Density-based uncertainty sampling calculates the average distance from each point to its k nearest neighbors and selects the point with the largest average distance as the sampling point. It compensates for TED’s tendency to get stuck in local optima, for example,

Algorithm 1 DynamicAL Sampling Algorithm

Require: \mathcal{U} is the unsampled microarchitecture design space, β is the hybrid sampling coefficient, and b is the number of selected samples.

Ensure: \mathcal{X} , the sampled set with $|\mathcal{X}| = b$.

- 1: $\mathcal{X} \leftarrow \emptyset$;
 - 2: $c_1 \leftarrow$ randomly select x_i from \mathcal{U} , centroid number is k , and centroid cluster $C = \{c_1\}$;
 - 3: **while** $|C| < k$ **do**
 - 4: $d_i = \min_{j \in \{1, 2, \dots, k\}} L(x_i - c_j), \forall x_i \in \mathcal{U}$, where L is the distance function;
 - 5: $p_i = \frac{d_i}{\sum_{i=1}^k d_i}$;
 - 6: $\hat{C} \leftarrow$ randomly select x_i from \mathcal{U} based on p ;
 - 7: $C = C \cup \hat{C}$;
 - 8: **end while**
 - 9: **while** not converged **do**
 - 10: $c^i = \operatorname{argmin}_{j \in \{1, 2, \dots, k\}} \Phi(x_i - c_j), \forall x_i \in \mathcal{U}$; where Φ is the distance function considering DecodeWidth;
 - 11: $c_j = \frac{\sum_{i=1}^{|\mathcal{U}|} \mathbb{1}\{c^i=j\}x_i}{\sum_{i=1}^{|\mathcal{U}|} \mathbb{1}\{c_i=j\}}, \forall j \in \{1, 2, \dots, k\}$;
 - 12: **end while**
 - 13: $\zeta \leftarrow$ neighborhood of $c_i \in C, \forall i \in \{1, 2, \dots, k\}$;
 - 14: **for** \mathcal{K} in ζ **do**
 - 15: $\hat{\mathcal{X}} = \operatorname{TED}(\mathcal{K}, \beta, \lfloor \frac{b}{k} \rfloor) \cup \operatorname{Density Sample}(\mathcal{K}, \beta, \lfloor \frac{b}{k} \rfloor)$;
 - 16: $\mathcal{X} = \mathcal{X} \cup \hat{\mathcal{X}}$;
 - 17: **end for**
 - 18: **return** The sampled set \mathcal{X} ;
-

when a cluster centroid is biased toward low-power regions, the uncertainty sampling can actively select points closer to the high-performance boundary.

D. Triple Pareto Frontier Acquisition Function

The role of the acquisition function is to guide the T-MSA algorithm in selecting the next most promising design. An excellent acquisition function would balance exploration (tending to select areas with the highest model uncertainty) and exploitation (tending to select areas with the best performance), dynamically and intelligently balancing the two based on the current state of the model.

The proposed TriPPFAF acquisition function considers three factors, EHVI, EPFI, and PPFi. In the early stages of design space exploration, EHVI is given a high weight due to its strong global optimization capability, while PPFi is designed to assist in exploring unknown regions. In the later stages, EPFI measures the expected improvement of the new design sample on the Pareto frontier based on dominance relations, and its weight gradually increases. EHVI is then utilized to maintain limited global search capabilities. The TriPPFAF function can be formulated as:

$$\operatorname{TriPPFAF}(x) = \mathbb{T}\{\operatorname{EHVI}(x), \operatorname{EPFI}(x), \operatorname{PPFi}(x)\} \quad (3)$$

where \mathbb{T} denotes the dynamic acquisition strategy.

The hypervolume (HV) metric calculates the volume of the objective space that is dominated by the Pareto frontier set

and the reference point. EHVI function measures the expected incremental HV of a new design solution relative to the current Pareto frontier, which can be expressed as:

$$\operatorname{HV}_{y_{ref}}(P(Y)) = \int_Y \mathbb{1}[y \succeq y_{ref}][1 - \prod_{y_* \in P(Y)} \mathbb{1}[y_* \not\prec y]] dy, \quad (4)$$

$$\operatorname{EHVI}(x) = \mathbb{E}_{p(f(x))}[\operatorname{HV}_{y_{ref}}(P(y) \cup f(x)) - \operatorname{HV}_{y_{ref}}(P(y))], \quad (5)$$

where $P(Y)$ is the Pareto-optimal set and y_{ref} denotes the reference point. $\mathbb{1}(\cdot)$ is the indicator function. $f(x)$ means the prediction of x from ensemble surrogate models. $\mathbb{E}_{p(f(x))}$ outputs the expectation of the argument.

In the TriPPFAF function, we define EPFI as the expected improvement of the new design sample on the Pareto frontier. EPFI can be formulated as:

$$\operatorname{EPFI}_{\mathcal{L}}(x) = \mathbb{E}_{p(f(x))}[\mathcal{L}(P(y) \cup f(x)) - \mathcal{L}(P(y))], \quad (6)$$

where \mathcal{L} is a generalized distance function.

PPFi focuses on possibility, considering the probability that a new design can be joined to the current Pareto frontier in a non-dominated manner. PPFi can be calculated as:

$$\operatorname{PPFi}(x) = \mathbb{E}_{p(f(x))}[\mathbb{1}(f(x) \not\prec P(Y))] = \mathbb{P}(f(x) \not\prec P(Y)), \quad (7)$$

where $f(x) \not\prec P(Y)$ indicates that $f(x)$ is not dominated by any point in the current Pareto frontier $P(Y)$.

IV. EXPERIMENTAL RESULTS

A. Experiment Setting

The proposed T-MSA algorithm is implemented in the Python programming language with the HEBO framework [22]. In the experiments, LiteFormer consists of an encoder and a decoder, with 4 heads for multi-head attention, and is trained in an unsupervised manner. The optimizer employs the adaptive moment estimation (Adam) [23], with an initial learning rate of 0.001. LiteFormer is trained by mean squared error loss (MSELoss) [24]. T-MSA is initialized based on 10 microarchitecture designs sampled by DynamicAL, and then Bayesian exploration is performed. The experimental results presented are the average of ten independent experiments, ensuring the robustness and reliability of the algorithm evaluation.

In Bayesian optimization, we adopt random forest (RF) [25], Gaussian processes (GP) [26], and sparse variational Gaussian processes (SVGP) [27] as the base surrogate models and conduct optimization experiments using different weight combinations.

We conducted experiments on two datasets, ICCAD [16] and ICCAD_GRL [10], both of which are related to the RISC-V BOOM processor. ICCAD dataset is proposed in Problem C of the 2022 ICCAD CAD Contest. The design space contains 1.5×10^4 different microarchitectures and corresponding PPA metrics. The PPA metric values for each microarchitecture are obtained through electronic design automation tools, a specific technology node, and a set of benchmarks. The VLSI flow includes RTL generation, logic synthesis, netlist simulation, power analysis, and more. The design space in the ICCAD_GRL dataset aligns with ICCAD, and PPA values are

obtained using RTL-level simulation tools and various RISC-V processor benchmarks. Evaluation tools include the Chipyard framework [28], Synopsys VCS simulator, and Design Compiler. Simulations are executed on a cluster equipped with an Intel Xeon Platinum 8354H CPU @ 3.10 GHz and utilize multiple CPU benchmarks.

B. LiteFormer Modeling Evaluation

We compared LiteFormer with methods utilizing Graph Neural Network (GNN) modeling [10], [11] and direct searching in the original design space. As shown in Fig. 3 and Fig. 4, our framework exhibits significant advantages from the start. Compared to original space modeling, T-MSA using LiteFormer achieved a comparable HV value in just 22 iterations on ICCAD, which represents a 50% reduction in iteration count, significantly improving design space exploration efficiency.

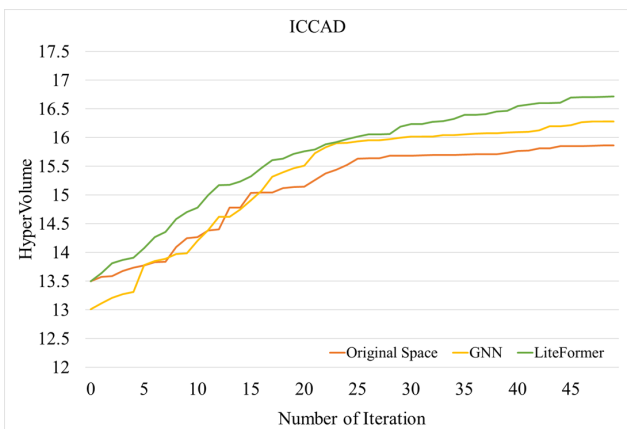


Fig. 3: Comparison of iterative curves in several embedding space on ICCAD.

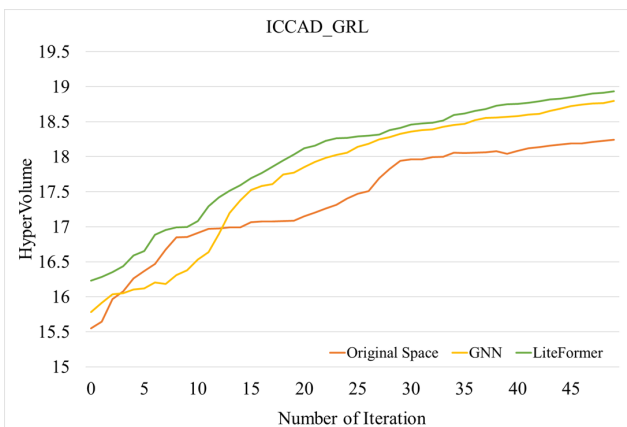


Fig. 4: Comparison of iterative curves in several embedding space on ICCAD_GRL.

Taken overall, LiteFormer achieved higher HV values on both the ICCAD and ICCAD_GRL under the same number of iterations compared to other approaches. When the number of iterations exceeded 20, a consistent trend emerged where LiteFormer exhibited superior search quality, followed by GNN, and then the original space method. This indicates

that while graph representation learning can effectively model certain variable relationships, it relies on prior knowledge and captures only partial dependencies among variables. However, LiteFormer demonstrates a stronger capability in modeling complex cross-variable coupling relationships.

C. The Efficiency of DynamicAL

The DynamicAL sampling algorithm can effectively select representative high-quality initial points. Table I presents the Pareto hypervolume (HV) achieved by DynamicAL sampling and compares it with other DSE sampling algorithms. DynamicAL sampling demonstrates superior performance over MicroAL [9], as well as the commonly used random sampling method. Our sampling method also get higher HV compared to K-Means algorithm. Even when compared to the results of random sampling, MicroAL and K-Means after ten rounds of optimization, DynamicAL sampling still maintains the highest HV value.

TABLE I: Comparison of Initial Sampling Results.

| Sampling Points | Algorithm | ICCAD [16] | ICCAD_GRL [10] |
|---|------------------|----------------------|----------------------|
| | | HV(Initial Sampling) | HV(Initial Sampling) |
| 10 random sampling points | Random Sampling | 12.5437 | 14.4054 |
| | MicroAL [9] | 13.2939 | 14.6394 |
| | K-Means [21] | 13.4742 | 15.4040 |
| | DynamicAL | 14.3241 | 16.1828 |
| 10 random sampling points + 10 optimization | Random Sampling | 13.6551 | 15.4155 |
| | MicroAL [9] | 14.4314 | 15.6732 |
| | K-Means [21] | 14.4773 | 16.3942 |
| | DynamicAL | 15.3184 | 17.0056 |

D. The Efficiency of TriPFAF

Table II presents the Pareto HV results obtained using different acquisition functions on the ICCAD and ICCAD_GRL. We evaluated the PPFi, EPFI, and EHVI. Additionally, we tested an averaged combination of these three factors, denoted as Avg. The results indicate that the Avg. method yields higher HV than individual EHVI, EPFI, or PPFi on the ICCAD_GRL dataset, but lower HV compared to EHVI alone on the ICCAD dataset. This suggests that fixed weighting limits sufficient exploration in certain contexts. By introducing stage-aware weight adaptation, TriPFAF achieves higher HV than all individual acquisition functions and the Avg. strategy, demonstrating the effectiveness of our proposed acquisition functions.

TABLE II: Results for Different Acquisition Functions.

| Algorithm | ICCAD [16]/ HV | ICCAD_GRL [10]/ HV |
|----------------|----------------|--------------------|
| PPFI | 14.1684 | 15.8217 |
| EPFI | 14.4914 | 16.4236 |
| EHVI | 16.1916 | 18.4697 |
| Avg. | 15.3067 | 18.4997 |
| TriPFAF | 16.3556 | 18.5488 |

E. Ablation Analysis

To evaluate the effectiveness of the three key components of our proposed method, we conducted a series of ablation experiments. Comparative results with BOOM-Explorer [9] are summarized in Table III. On the ICCAD dataset, Liteformer, DynamicAL sampling, and TriPFAF contribute 57.85%, 47.20%, and 64.81%, respectively, to the overall performance. Similarly,

TABLE III: Abalation Experiment Result Compared to BOOM-Explorer [9].

| Algorithm | (a) LiteFormer | (b) DynamicAL | (c) TriPFAF | ICCAD [16] | | ICCAD_GRL [10] | |
|---------------------|----------------|---------------|-------------|----------------|----------------------|----------------|----------------------|
| | | | | HV | HV Improvement Ratio | HV | HV Improvement Ratio |
| BOOM-Explorer [9] | | | | 15.6815 | 0.00% | 17.9437 | 0.00% |
| with (a) | ✓ | | | 16.0715 | 57.85% | 18.4132 | 77.59% |
| with (b) | | ✓ | | 15.9997 | 47.20% | 18.3282 | 63.54% |
| with (c) | | | ✓ | 16.1184 | 64.81% | 18.3933 | 74.30% |
| with (a) & (b) | ✓ | ✓ | | 16.1916 | 75.67% | 18.4697 | 86.93% |
| T-MSA (ours) | ✓ | ✓ | ✓ | 16.3556 | 100.00% | 18.5488 | 100.00% |

on the ICCAD_GRL dataset, their contributions are 77.59%, 63.54%, and 74.30%, respectively. The LiteFormer framework captures comprehensive cross-variable dependencies via self-attention mechanisms, establishing a flexible latent foundation for downstream optimization and constructing a high-quality search space. DynamicAL sampling further improves initial sampling quality by maximizing the use of global information, allowing the surrogate model to develop a more accurate global approximation. Meanwhile, TriPFAF enhances both exploration and exploitation across the design space through its stage-aware acquisition function adjustment.

F. Comparison with Prior Work

The performance of our proposed DSE framework was compared against the current state-of-the-art BOOM DSE algorithms on both ICCAD and ICCAD_GRL datasets. Their experimental results are presented in Table IV. On two evaluation platforms, the designs T-MSA identifies achieve significant improvements over BOOM-Explorer [9], increasing the Pareto hypervolume by 4.30% and 3.37%, respectively. Compared to the Prior-boosted GRL [10] algorithm, T-MSA also demonstrates superior optimization performance on two datasets.

TABLE IV: Comparison of Exploration Results of Different DSE Frameworks.

| Dataset | Algorithm / HV | | |
|----------------|-------------------|------------------------|---------------------|
| | BOOM-Explorer [9] | Prior-boosted GRL [10] | T-MSA (ours) |
| ICCAD [16] | 15.6815 | 15.8340 | 16.3556 |
| ICCAD_GRL [10] | 17.9437 | 18.2601 | 18.5488 |

Additionally, to provide a more intuitive visualization of the exploration process, we compared the iteration curves of several state-of-the-art DSE algorithms. Results on two datasets are illustrated in Fig. 5 and Fig. 6. It is evident that T-MSA achieves the fastest optimization. When attaining the same HV value, T-MSA outperformed BOOM-Explorer by 188.24% on ICCAD and by 133.33% on ICCAD_GRL.

These results highlight the advantages of embedding microarchitectural dependencies and performing multi-strategy adaptation during the search process. Ultimately, T-MSA significantly outperforms existing algorithms in both the quality of the Pareto solution and runtime.

V. CONCLUSION

This paper introduces T-MSA, a Transformer-driven multi-strategy DSE scheme. It utilizes LiteFormer to convert discrete and isolated microarchitecture designs into a fully coupled continuous space, directly quantifying the topological relationships among arbitrary design variable pairs. It then employs sparse centroid initialization and designs a hybrid sampling

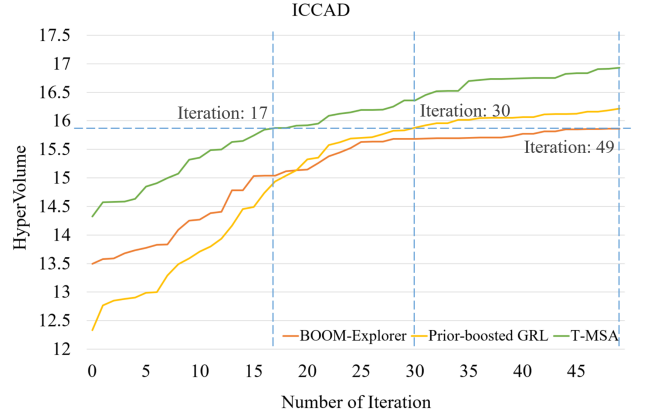


Fig. 5: Comparison of iterative curves of several outstanding DSE algorithms on ICCAD.

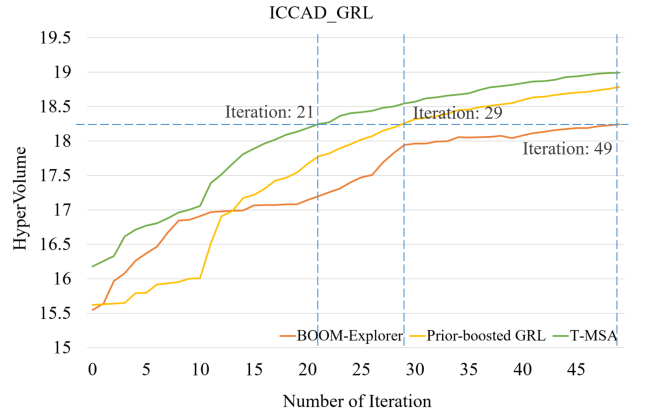


Fig. 6: Comparison of iterative curves of several outstanding DSE algorithms on ICCAD_GRL.

method based on TED and density features, thereby devising the DynamicAL strategy to ensure the significance of sampled points. During Bayesian optimization, the TriPFAF function dynamically allocates EHVI, EPFI, and PPFI to intelligently recommend Pareto optimal designs. It is the first application of Transformer in the DSE domain. Experiments demonstrate that T-MSA achieves over 2x faster optimization than existing DSE algorithms and significantly enhances Pareto optimal quality.

VI. ACKNOWLEDGEMENT

This work was supported by the Innovative Development Joint Fund Key Projects of Shandong NSF (ZR2023LZH003) and Taishan Industrial Experts Program.

REFERENCES

- [1] K. Asanovic, D. A. Patterson, and C. Celio, "The berkeley out-of-order machine (boom): An industry-competitive, synthesizable, parameterized risc-v processor," Tech. Rep., 2015.
- [2] J. Umeike, N. Patel, A. Manley, A. Mamandipoor, H. Yun, and M. Alian, "Profiling gem5 simulator," in *2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 2023, pp. 103–113.
- [3] A. S. Priya, S. Kamatchi, and E. L. Prasad, "Early register transfer level (rtl) power estimation in real-time system-on-chips (socs)," *Journal of Integrated Science and Technology*, vol. 11, no. 1, pp. 454–454, 2023.
- [4] L. Ferretti, G. Ansaloni, and L. Pozzi, "Cluster-based heuristic for high level synthesis design space exploration," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 35–43, 2018.
- [5] Z. Wu, X. Yi, L. Shang, and F. Yang, "Sensedse: Sensitivity-based performance evaluation for design space exploration of microarchitecture," in *2024 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2024, pp. 1–6.
- [6] Q. Gautier, A. Althoff, C. L. Crutchfield, and R. Kastner, "Sherlock: A multi-objective design space exploration framework," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 27, no. 4, pp. 1–20, 2022.
- [7] N. Shi, J. Xu, H. Li, H. Guo, J. Woodring, and H.-W. Shen, "Vdl-surrogate: A view-dependent latent-based model for parameter space exploration of ensemble simulations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 820–830, 2022.
- [8] G. Beltrame, L. Fossati, and D. Sciuto, "Decision-theoretic design space exploration of multiprocessor platforms," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 7, pp. 1083–1095, 2010.
- [9] C. Bai, Q. Sun, J. Zhai, Y. Ma, B. Yu, and M. D. Wong, "Boom-explorer: Rise-v boom microarchitecture design space exploration framework," in *2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*. IEEE, 2021, pp. 1–9.
- [10] Z. Wu, J. Shen, X. Yi, L. Shang, F. Yang, and X. Zeng, "Prior-boosted grl: Microarchitecture design space exploration via graph representation learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2024.
- [11] X. Yi, J. Lu, X. Xiong, D. Xu, L. Shang, and F. Yang, "Graph representation learning for microarchitecture design space exploration," in *2023 60th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2023, pp. 1–6.
- [12] M. L. Santoni, E. Raponi, R. D. Leone, and C. Doerr, "Comparison of high-dimensional bayesian optimization algorithms on bbob," *ACM Transactions on Evolutionary Learning*, vol. 4, no. 3, pp. 1–33, 2024.
- [13] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," *Electronics*, vol. 9, no. 8, p. 1295, 2020.
- [14] S. Ament, S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy, "Unexpected improvements to expected improvement for bayesian optimization," *Advances in Neural Information Processing Systems*, vol. 36, pp. 20 577–20 612, 2023.
- [15] J. Deng, J. Sun, Q. Zhang, and H. Li, "Expected hypervolume improvement is a particular hypervolume improvement," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 15, 2025, pp. 16 217–16 225.
- [16] S. Li, C. Bai, X. Wei, B. Shi, Y.-K. Chen, and Y. Xie, "2022 iccad cad contest problem c: Microarchitecture design space exploration," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–7.
- [17] S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy, "Multi-objective bayesian optimization over high-dimensional search spaces," in *Uncertainty in Artificial Intelligence*. PMLR, 2022, pp. 507–517.
- [18] T. Liu, Q. Wang, L. Liu, F. Wang, and E. F. Young, "On advanced methodologies for microarchitecture design space exploration," in *Proceedings of the Great Lakes Symposium on VLSI 2024*, 2024, pp. 376–382.
- [19] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu *et al.*, "A survey on vision transformer," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 1, pp. 87–110, 2022.
- [20] L. Nardi, D. Koeplinger, and K. Olukotun, "Practical design space exploration," in *2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE, 2019, pp. 347–358.
- [21] M. Fan, X. Han, J. Fan, C. Chai, N. Tang, G. Li, and X. Du, "Cost-effective in-context learning for entity resolution: A design space exploration," in *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 3696–3709.
- [22] A. I. Cowen-Rivers, W. Lyu, R. Tutunov, Z. Wang, A. Grosnit, R. R. Griffiths, H. Jianye, J. Wang, and H. B. Ammar, "An empirical study of assumptions in bayesian optimisation," *arXiv preprint arXiv:2012.03826*, vol. 445, 2020.
- [23] J. Yang and Q. Long, "A modification of adaptive moment estimation (adam) for machine learning," *Journal of Industrial and Management Optimization*, vol. 20, no. 7, pp. 2516–2540, 2024.
- [24] X. Qiao, Q. Yan, and W. Huang, "Hybrid cnn-transformer network with a weighted mse loss for global sea surface wind speed retrieval from gnss-r data," *IEEE Transactions on Geoscience and Remote Sensing*, 2025.
- [25] H. A. Salman, A. Kalakech, and A. Steiti, "Random forest algorithm overview," *Babylonian Journal of Machine Learning*, vol. 2024, pp. 69–79, 2024.
- [26] J. Donnelly, A. Daneshkhan, and S. Abolfathi, "Forecasting global climate drivers using gaussian processes and convolutional autoencoders," *Engineering Applications of Artificial Intelligence*, vol. 128, p. 107536, 2024.
- [27] H. J. Cunningham, D. A. de Souza, S. Takao, M. van der Wilk, and M. P. Deisenroth, "Actually sparse variational gaussian processes," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 10 395–10 408.
- [28] A. Amid, D. Biancolin, A. Gonzalez, D. Grubb, S. Karandikar, H. Liew, A. Magyar, H. Mao, A. Ou, N. Pemberton *et al.*, "Chipyard: Integrated design, simulation, and implementation framework for custom socs," *Ieee Micro*, vol. 40, no. 4, pp. 10–21, 2020.