Physics-Guided Geometric Diffusion for Macro Placement Generation

Jongho Yoon[1] , **Jinsung Jeon**[2,3] , **Seokhyeong Kang**[4]
[1]POSTECH Institute of Artificial Intelligence
[2]KAIST InnoCORE LLM
[3]Seoul National University
[4]Pohang University of Science and Technology
{jh.yoon, shkang}@postech.ac.kr, jjsjjs0902@gmail.com

Abstract

Macro placement is a pivotal stage in VLSI physical design, fundamentally determining the overall chip performance. Recent data-driven placement methods have demonstrated significant potential, yet they often struggle to handle sequential dependencies and to balance topological connectivity with physical constraints. To bridge this gap, we propose **MacroDiff+**, a physics-guided geometric diffusion framework. Specifically, we design a dual-domain denoising architecture that couples topological connectivity encoded by heterogeneous GNNs with global geometric context modeled by a Transformer. Furthermore, we introduce Physics-Guided Sampling, an inference strategy that actively steers the generation using explicit gradients to ensure both statistical plausibility and physical validity. On the ISPD2005 MMS benchmarks, MacroDiff+ outperforms state-of-the-art baselines with a 6.1–6.2% reduction in wirelength. Notably, it exhibits superior stability and scalability on large-scale designs where prior methods fail to converge. The source code is provided at https://github.com/jhy00n/MacroDiff-plus.

1 Introduction

Macro placement is the stage of physical design that determines the locations of large, pre-designed circuit blocks, such as memory arrays, IP cores, and analog blocks, on the chip canvas. As modern system-on-chip (SoC) designs continue to scale in complexity, incorporating a growing number of macros alongside numerous standard cells, the quality of macro placement has become a pivotal determinant of chip success [Agnesina *et al.*, 2023]. Decisions made at this stage create cascading effects throughout subsequent design phases, ultimately determining the power, performance, and area (PPA) metrics [Tseng, 2024].

However, macro placement is an inherently challenging combinatorial optimization problem. The search space grows exponentially with the number of macros, as each macro must be assigned both a location and an orientation on the chip canvas [Mirhoseini *et al.*, 2021]. This complexity makes exhaustive exploration computationally intractable, and manual placement by experienced engineers requires weeks or even months of iterative effort, creating a critical bottleneck in the design cycle [Gao *et al.*, 2022]. These challenges have driven the development of automated macro placement methods to efficiently explore the design space and deliver consistent, high-quality solutions.

To address these complexities, automated macro placement has primarily followed two paradigms: optimization-based and reinforcement learning (RL)-based approaches. Optimization-based methods [Agnesina *et al.*, 2023; Shi *et al.*, 2023] formulate placement as a continuous or black-box optimization problem, leveraging powerful solvers to navigate the design space. Similarly, RL-based methods [Mirhoseini *et al.*, 2021; Lai *et al.*, 2022; Lai *et al.*, 2023] frame the task as a sequential Markov Decision Process, placing macros one at a time. However, despite their advancements, these approaches face fundamental limitations from an AI methodology perspective. Optimization solvers often struggle with the high dimensionality and become trapped in local optima, while the sequential formulation of RL introduces inherent order dependency, preventing the simultaneous optimization of global interdependencies.

Recently, diffusion models have emerged as a promising generative paradigm that addresses some of these limitations [Ho *et al.*, 2020; Trippe *et al.*, 2023; Sun and Yang, 2023]. By generating all macro positions simultaneously through iterative denoising, diffusion models inherently capture global context without the order dependency of RL, while their stochastic sampling naturally explores diverse solutions beyond the local optima that trap optimization-based solvers. However, existing diffusion-based approaches for placement [Lee *et al.*, 2025] treat the problem purely as geometric pattern generation, failing to explicitly model the circuit netlist topology that fundamentally governs wirelength optimization. This disconnect between geometric generation and topological objectives leads to unstable convergence and suboptimal wirelength, as evidenced by the high variance in the prior work.

To reconcile this disconnect, we propose **MacroDiff+**, a physics-guided geometric diffusion framework that explicitly synergizes topological connectivity with spatial constraints. The main contributions are as follows:

- Dual-Branch Denoising Architecture: We propose a novel architecture that integrates Heterogeneous Graph

Neural Networks (Hetero GNN) for topological encoding and Transformers for global spatial context.

- Physics-Guided Sampling: We introduce an inference-time mechanism that steers the generative trajectory using explicit gradients of physical objectives, ensuring generated solutions satisfy hard constraints beyond what the learned distribution captures.
- Empirical Validation: Experiments on ISPD2005 benchmarks demonstrate 6.1–6.2% HPWL reduction over RL-based and optimization-based baselines, significantly improved stability, and successful scaling to large designs where prior methods fail to converge.

2 Background and Related Work

2.1 Macro Placement in VLSI

Macro placement significantly impacts the PPA metrics of the final chip layout. Since poor placement decisions can lead to irreversible routing failures and timing violations, achieving an optimal macro configuration is critical [Chen *et al.*, 2023a; Xue *et al.*, 2024]. Classical heuristics such as Simulated Annealing [Sechen and Sangiovanni-Vincentelli, 2003] and Force-Directed methods [Chan *et al.*, 2005] established foundational concepts in this domain. However, these methods struggle with the scalability and heterogeneity of modern SoC designs [Qiu *et al.*, 2023]. Consequently, recent research has shifted towards advanced optimization and machine learning (ML) techniques to navigate this complex combinatorial search space.

Optimization-based Approaches

Optimization-based methods typically formulate placement as a continuous or black-box optimization problem. For instance, AutoDMP [Agnesina *et al.*, 2023] enhances mixed-size placement quality through the automated parameter tuning of the analytical placer DREAMPlace [Chen *et al.*, 2023b]. It utilizes a Multi-Objective Tree-structured Parzen Estimator (Bayesian optimization) to efficiently optimize PPA proxy objectives such as wirelength and congestion. Similarly, WireMask-BBO [Shi *et al.*, 2023] approaches macro placement as a black-box optimization problem. Instead of purely analytical solvers, it represents solutions via continuous coordinates and employs a wire-mask-guided evaluation to greedily improve placement using evolutionary algorithms.

While these approaches leverage powerful solvers to minimize objectives like Half-Perimeter Wirelength (HPWL) and demonstrate effectiveness in local refinement, they face significant challenges. They often suffer from excessive runtime, struggle to navigate the non-convex optimization landscape characterized by discrete physical constraints, and their performance heavily relies on the quality of initialization.

RL-based Approaches

Leveraging advancements in deep learning, RL-based methods typically frame macro placement as a sequential Markov Decision Process (MDP). Chip Placement with Deep RL [Mirhoseini *et al.*, 2021] garnered significant attention by demonstrating that an RL agent could outperform human experts in hours by predicting optimal positions for cells sequentially. Building on this, MaskPlace [Lai *et al.*, 2022] recasts the problem as learning pixel-level visual representations. It utilizes a convolutional neural network to process position, wire, and view masks, and employs a dense reward based on incremental HPWL changes to guarantee non-overlapping placements. Furthermore, ChiPFormer [Lai *et al.*, 2023] advances this domain by utilizing a Decision Transformer for offline RL, enabling transferable policies that generalize to new circuits with minimal fine-tuning.

Despite their potential to learn complex policies, these RL-based approaches face inherent structural limitations. The sequential nature of placement decisions introduces order dependency, where early suboptimal decisions propagate, making it difficult to optimize the global context of all macros simultaneously. Moreover, these methods are often plagued by significant sample inefficiency and incur substantial computational costs or extensive retraining on unseen designs.

2.2 Generative Diffusion Models

Diffusion models have emerged as a powerful generative paradigm, capable of modeling complex continuous distributions by learning to reverse a gradual noise addition process [Ho *et al.*, 2020]. Unlike RL-based methods that rely on sequential decision-making, diffusion models generate holistic solutions for all objects simultaneously, making them well-suited for the combinatorial nature of chip placement.

Fundamentals of Diffusion Models

As a representative framework of diffusion models, Denoising Diffusion Probabilistic Models (DDPMs) [Ho *et al.*, 2020] characterize the generation process through two Markov chains: a fixed *forward process* (q) that gradually adds Gaussian noise, and a learned *reverse process* (p_θ) that iteratively recovers the original data. The core objective is to train a network $\epsilon_\theta(x_t, t)$ to predict the noise ϵ added to the input. This is achieved by minimizing the simplified mean squared error (MSE):

$$\mathcal{L}_{\text{simple}}(\theta) = \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(x_t, t)\|^2\right]. \tag{1}$$

By minimizing this objective, the model implicitly learns the score function $\nabla_x \log p(x)$, which guides the generation from random noise to a structured layout. We provide the detailed mathematical preliminaries in Appendix A.

Generative Diffusion for Macro Placement

Recently, ChipDiffusion [Lee *et al.*, 2025] pioneered the application of this paradigm to VLSI, framing macro placement as an image generation task. By leveraging the continuous spatial modeling capabilities of diffusion, it overcomes the local optima issues of optimization-based methods and the sequential limitations of RL. However, existing diffusion approaches typically rely on image-based representations or synthetic data, frequently overlooking the precise *topological connectivity* inherent in real-world netlists. By treating placement purely as a geometric generation task, they lack explicit mechanisms to balance wirelength minimization with strict physical constraints (e.g., overlaps) during denoising.

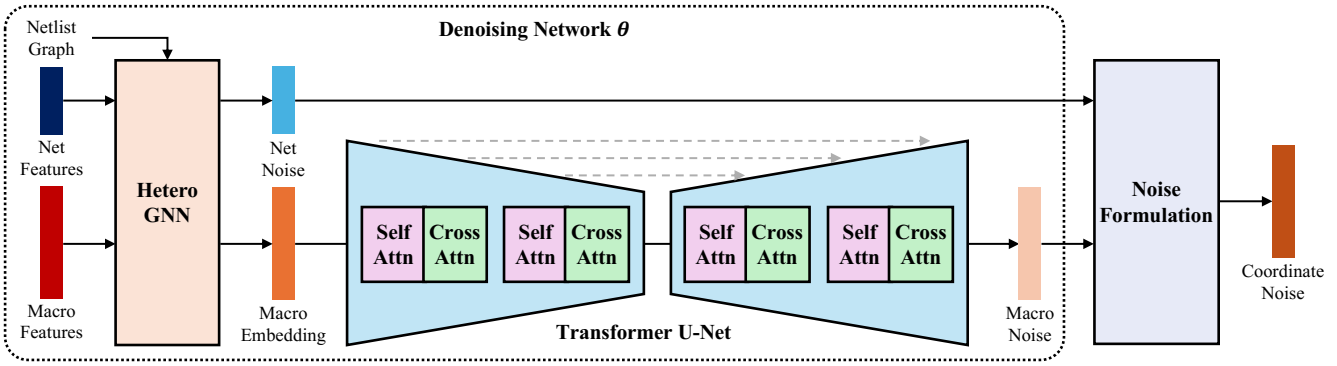


Figure 1: Overall framework of the dual-branch noise prediction network in **MacroDiff+**. The architecture integrates topological insights from the Hetero GNN, yielding Macro Embeddings (H_{macro}) and Net-Centric Noise (ϵ_{net}), with global geometric context from the Transformer producing Macro-Centric Noise (ϵ_{cell}). The Noise Formulation module fuses these outputs to generate the final Coordinate Noise, $\epsilon_\theta(x_t, t)$.

3 Proposed Method

To overcome these limitations, we propose **MacroDiff+**, a physics-guided graph diffusion framework. Unlike prior geometric-only approaches, our method explicitly models the circuit topology using a heterogeneous graph and steers the generation process based on physical gradients.

3.1 Problem Formulation and Training Objective

We formulate macro placement as a generative task on a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. The node set $\mathcal{V}$ is partitioned into macros $\mathcal{V}_m$ and nets $\mathcal{V}_n$, where edges $\mathcal{E}$ represent pin-level connectivity. Each macro node $v_i \in \mathcal{V}_m$ is characterized by its fixed dimensions (w_i, h_i), while the placement solution is defined by the coordinates $x \in \mathbb{R}^{|\mathcal{V}_m|\times 2}$ representing the bottom-left corners. Our objective is to generate the optimal positions x that minimize the Half-Perimeter Wirelength (HPWL) while strictly satisfying non-overlap constraints. Our heterogeneous graph encodes both static circuit properties (macro dimensions, chip boundaries, net degrees, pin offsets) and dynamic placement states (macro positions x_t and net-level HPWL) that evolve during denoising. The pin-offset feature on edges enables wirelength computation at the pin level, which is critical for large macros. Detailed specifications are provided in Appendix B.

To train our denoising network θ, we follow the standard diffusion training paradigm. Specifically, the model is optimized to predict the noise ϵ added to the clean placement x_0 at a given timestep t. The Training objective is defined by minimizing the mean squared error between the added noise and the noise predicted by our dual-branch architecture, as formulated in Eq. 1. By optimizing this objective, the model learns to capture the joint distribution of circuit connectivity and spatial constraints, enabling the generation of high-quality macro placements from random noise.

3.2 Dual-Branch Denoising Architecture

To generate high-quality placements, the model must simultaneously capture local connectivity (topology) and global spatial relationships (geometry). We design a dual-branch architecture (cf. Fig. 1) that handles these two modalities via a Hetero GNN and a Transformer, respectively.

Heterogeneous Graph Neural Network (Hetero GNN)

The Hetero GNN serves as the topological backbone of our framework, specifically designed to handle the non-Euclidean structure of the netlist. Since macro placement is fundamentally driven by connectivity, where macros linked by critical nets must be placed in proximity, capturing these dependencies is crucial. We employ a relational graph attention to explicitly model these interactions between macros and nets.

Conditioned on timestep t, the Hetero GNN iteratively updates node embeddings via message passing. Formally, for each layer l, the update process for a macro node m aggregating messages from neighboring net nodes $n \in \mathcal{N}(m)$ is defined as:

$$\mathbf{e}_{n\to m}^{(l+1)} = \text{AGGREGATE}_{n\to m}^{(l)}(\{h_n^{(l)} : n \in \mathcal{N}(m)\}, t), \quad (2)$$

$$h_m^{(l+1)} = \text{UPDATE}^{(l+1)}(h_m^{(l)}, \mathbf{e}_{n\to m}^{(l+1)}). \quad (3)$$

In our implementation, the AGGREGATE function employs the GATv2 attention mechanism [Brody *et al.*, 2021]. This allows the model to dynamically weigh the importance of information from different neighboring nodes based on their features, enabling a more nuanced understanding of connectivity than methods with static weights. The UPDATE function then combines the resulting aggregated message with the previous node embedding $h_m^{(l)}$ through a non-linear transformation and a residual connection. The residual connection provides a direct path for gradients, mitigating the vanishing gradient problem in deep Hetero GNNs. A symmetric process is applied to update net embeddings (h_n) by aggregating information from macro nodes. Through L layers of such message passing, the model learns the structural importance of each net and macro, producing topologically rich embeddings H_{macro} and H_{net}.

The net embeddings H_{net} serve as the Net-Centric Noise (ϵ_{net}), a high-dimensional vector defined on net nodes that encodes the connection intent, the learned importance of each net at the current denoising step. Conceptually, this allows the model to directly incorporate the critical HPWL objective into noise prediction, identifying which connections are most critical to optimize. However, a key distinction is that despite being termed noise by analogy with the standard diffusion formulation, ϵ_{net} resides in an abstract topological space

rather than the physical coordinate space, representing connection intent rather than spatial displacement. To convert this intent into geometric updates, our fusion phase introduces a projection mechanism, described next.

Transformer Network

Complementing the Hetero GNN, the Transformer network serves as the geometric branch. It is built upon a U-Net architecture integrated with Transformer blocks. The hierarchical structure of the U-Net processes spatial features at multiple scales, while the embedded Transformer blocks perform two critical attention operations to resolve geometric constraints. First, *Self-attention* captures long-range dependencies among all macros, enabling the model to detect and diffuse high-density clusters effectively. Second, *Cross-attention* injects physical conditioning information C into the latent space. Here, C corresponds to the macro size features (w, h) (see Appendix B for detailed input features), ensuring the generation process remains cognizant of physical dimensions. Specifically, the macro embeddings H_{macro} from the Hetero GNN form the initial sequence $Z^{(0)}$. Formally, the operation sequence within each Transformer block at layer l is defined as:

$$Z'^{(l)} = \text{Self-Attn}(Z^{(l-1)}, t), \tag{4}$$

$$Z''^{(l)} = \text{Cross-Attn}(Z'^{(l)}, C), \tag{5}$$

$$Z^{(l)} = \text{FFN}(Z''^{(l)}). \tag{6}$$

Here, each function includes residual connections and normalization. Finally, a projection head converts the output $Z^{(L)}$ into the Macro-Centric Noise (ϵ_{cell}).

Unlike the net-centric prediction from the GNN branch, this macro-centric prediction is a direct coordinate-based noise prediction inferred from the global spatial context. Its primary strength lies in generating well-distributed layouts that respect chip boundaries and density constraints. However, a significant limitation is that this prediction is agnostic to the netlist topology; it may produce valid but wirelength-suboptimal placements. Therefore, it serves as the geometric counterpart to the topological insight provided by the Hetero GNN, necessitating the synergistic fusion mechanism we detail in the next section.

Dual-Branch Fusion and Noise Prediction

To harness the strengths of both branches, we formulate the final noise prediction as a synergistic fusion within the Noise Formulation module. This fusion utilizes the **Gradient Projector Matrix** as a bridge between the topological and geometric domains (cf. Fig. 2). Since the net-centric noise ϵ_{net} resides in an abstract topological space, it cannot directly update macro coordinates. To resolve this, we employ a projection mechanism using the gradient of the HPWL. Specifically, we project ϵ_{net} into the coordinate space via the Gradient Projector Matrix (implemented as a Jacobian vector-product) and combine it with the macro-centric noise ϵ_{cell} as follows:

$$\epsilon_\theta(x_t, t) = \lambda_{cell} \cdot \epsilon_{cell} + \lambda_{net} \cdot (\nabla_{x_t} \text{HPWL}(x_t))^T \cdot \epsilon_{net}. \tag{7}$$

Here, the term $(\nabla_{x_t} \text{HPWL}(x_t))^T$ corresponds to the Gradient Projector Matrix shown in Fig. 2. It represents the sensitivity of wirelength to macro displacements, effectively translating the learned connection importance into physical force

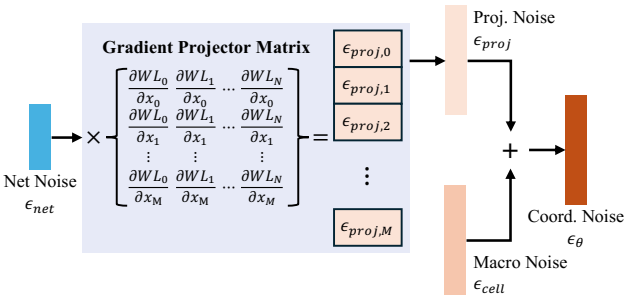


Figure 2: Detailed schematic of the Noise Formulation.

vectors. The scalars λ_{cell} and λ_{net} are hyperparameters that balance the influence of the two branches. This formulation (Eq. 7) mathematically realizes the synergy proposed in our framework, balancing global spatial arrangement (density) with fine-grained connectivity objectives (wirelength).

3.3 Physics-Guided Sampling

While the dual-branch model learns the distribution of high-quality placements, probabilistic generation alone may yield samples violating strict physical constraints. We thus employ Physics-Guided Sampling (cf. Fig. 3), which refines each predicted placement to better satisfy physical constraints during generation.

The procedure operates by augmenting the standard reverse diffusion process. At each timestep t, we first use our trained dual-branch denoising model ϵ_θ to obtain an initial prediction of the clean macro placement, denoted as $\hat{x}_0$. This prediction is derived from the current noisy state x_t using the standard diffusion formulation:

$$\hat{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t, t)). \tag{8}$$

This initially predicted placement $\hat{x}_0$ serves as the starting point for a brief, physics-driven optimization. We define a composite cost function, $\mathcal{L}_{guide}$, that mathematically represents the quality of the placement:

$$\begin{aligned}\mathcal{L}_{guide}(\hat{x}_0) =& w_{HPWL} \cdot \mathcal{L}_{HPWL}(\hat{x}_0) \\ &+ w_{overlap} \cdot \mathcal{L}_{overlap}(\hat{x}_0),\end{aligned} \tag{9}$$

where $\mathcal{L}_{HPWL}$ estimates the wirelength and $\mathcal{L}_{overlap}$ penalizes illegal overlaps. The scalars w_{HPWL} and $w_{overlap}$ follow a two-phase schedule: the first phase emphasizes wirelength minimization, and once HPWL improvement plateaus, the second phase down-weights L_{HPWL} to focus on overlap resolution. This scheduling avoids gradient interference between conflicting objectives that arises under fixed weights. To ensure the guidance is physically meaningful, we perform this optimization directly in the clean data space ($\hat{x}_0$) rather than the noisy latent space. The goal is to find a modification vector, Δx_0, that minimizes the cost:

$$\min_{\Delta} \mathcal{L}_{guide}(\hat{x}_0 + \Delta). \tag{10}$$

In practice, we approximate this minimization via gradient descent steps starting from $\Delta = 0$. This step effectively pulls

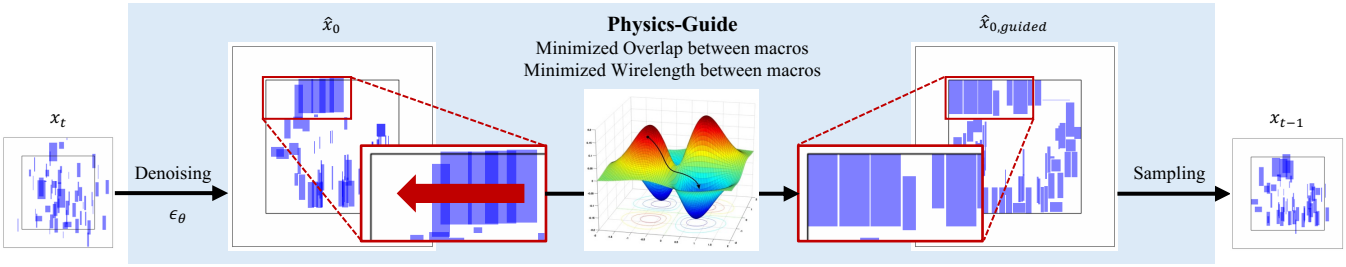


Figure 3: Detailed workflow of the Physics-Guided Sampling employed during inference.

the predicted placement towards a configuration that simultaneously reduces wirelength and resolves overlaps. The refined placement is obtained as:

$$\hat{x}_{0,guided} = \hat{x}_0 + \Delta x_0. \tag{11}$$

Crucially, this refined placement $\hat{x}_{0,guided}$ is not the final output but is used to guide the sampling trajectory. We derive an implicitly guided noise, ϵ_{guided}, based on this physically optimized prediction:

$$\epsilon_{guided} = \frac{1}{\sqrt{1-\bar{\alpha}_t}}(x_t - \sqrt{\bar{\alpha}_t}\hat{x}_{0,guided}). \tag{12}$$

By replacing the original network prediction ϵ_θ with ϵ_{guided} in the update rule for x_{t-1}, we ensure that every step of the generation is not only grounded in the learned data distribution but is also actively steered by explicit physical forces. This procedure, outlined in Algorithm 1, allows MacroDiff+ to produce layouts that are both topologically coherent and physically valid.

Algorithm 1 Physics-Guided Sampling from MacroDiff+

Require: Learned model ϵ_θ, graph G with static features, step size η, guidance steps K, weights $w_{\text{HPWL}}, w_{\text{overlap}}$, weight schedule $W(\cdot)$

$x_T \sim \mathcal{N}(0, I)$
Construct graph G_T using x_T
for $t = T, \ldots, 1$ **do**
  $z \sim \mathcal{N}(0, I)$ if $t > 1$, else $z = 0$
  $\epsilon_{\text{pred}} \leftarrow \epsilon_\theta(G_t, t)$ {Predict noise}
  // Step 1: Estimate clean data (Eq. 8)
  $\hat{x}_0 \leftarrow \frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_{\text{pred}}\right)$
  // Step 2: Physics-Guided Optimization (Eq. 9-11)
  $\hat{x}_{0,\text{guided}} \leftarrow \hat{x}_0$
  for $k = 1, \ldots, K$ **do**
    $(w_{HPWL}, w_{overlap}) \leftarrow W(k, L_{HPWL})$
    $L_{\text{guide}} \leftarrow w_{\text{HPWL}} \cdot L_{\text{HPWL}} + w_{\text{overlap}} \cdot L_{\text{overlap}}$
    $\nabla \leftarrow \nabla_{\hat{x}_{0,\text{guided}}} L_{\text{guide}}$ {Compute gradient}
    $\hat{x}_{0,\text{guided}} \leftarrow \hat{x}_{0,\text{guided}} - \eta \cdot \nabla$ {Gradient descent}
  end for
  // Step 3: Compute guided noise (Eq. 12)
  $\epsilon_{\text{guided}} \leftarrow \frac{1}{\sqrt{1-\bar{\alpha}_t}}\left(x_t - \sqrt{\bar{\alpha}_t}\hat{x}_{0,\text{guided}}\right)$
  // Step 4: Denoise one step
  $x_{t-1} \leftarrow \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_{\text{guided}}\right) + \sigma_t z$
  Construct graph G_{t-1} using x_{t-1}
end for
return x_0

4 Experimental Evaluations

4.1 Experimental Setup

Our software and hardware environments are as follows: UBUNTU 18.04 LTS, PYTHON 3.8.10, PYTORCH 1.8.1 and AMD EPYC 7513 CPU, and NVIDIA RTX A6000. The mean and variance of 10 runs are reported for model evaluation. Hyperparameter settings are detailed in Appendix C.

Benchmarks and Dataset Construction

We conduct our experiments on the Modern Mixed-Size (MMS) benchmarks [Yan *et al.*, 2009], modified from ISPD2005 [Nam *et al.*, 2005] with movable macros and fixed I/O pads whose dimensions are set to zero. Detailed statistics for each benchmark circuit are summarized in Appendix D.

To effectively train our diffusion model to capture generalized topological features, we constructed a dedicated training dataset via data augmentation. Since the original benchmarks provide limited samples, we generated 1,000 augmented netlists for each of the eight ISPD2005 designs, resulting in a total dataset of 8,000 samples. The augmentation was performed using a degree-preserving configuration model [Newman, 2003]. In this process, the degrees of macro and net nodes are preserved to maintain the fundamental circuit complexity, while edges are randomly rewired to create diverse connectivity patterns. For each augmented netlist, a high-quality reference macro placement was generated using DREAMPlace to serve as the ground truth (x_0) for training.

Evaluation Protocol and Metrics

To assess placement quality, we adopt a unified pipeline across all methods: (i) macro sampling using the benchmark models (e.g., MacroDiff+), (ii) macro legalization, (iii) standard cell placement via DREAMPlace with macros held fixed, and (iv) final legalization. The primary metric is the Half-Perimeter Wirelength (HPWL), where lower values indicate better quality.

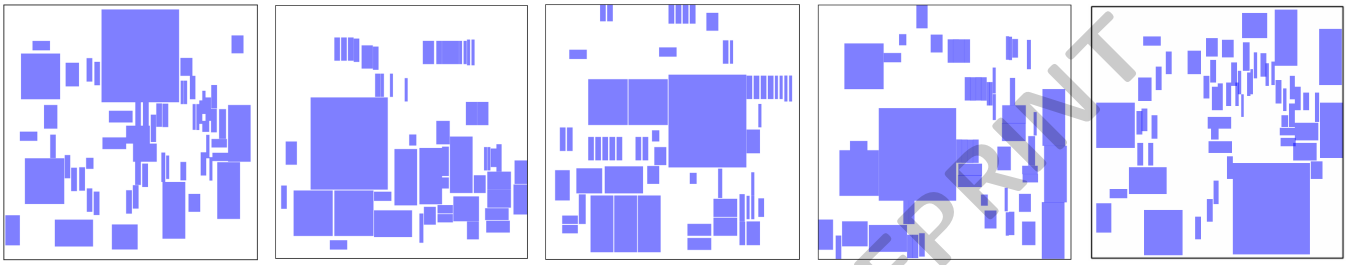
Baselines

We evaluate our framework by comparing it against four representative macro placement methodologies:

- DREAMPlace [Chen *et al.*, 2023b]: A widely adopted analytical placer serving as the *performance upper bound* for wirelength quality in our benchmarks.
- MaskPlace [Lai *et al.*, 2022]: A representative RL-based

Design	DREAMPlace	MaskPlace		WireMask		ChipDiffusion		MacroDiff+ (Ours)	
		Mean ± Std	Best	Mean ± Std	Best	Mean ± Std	Best	Mean ± Std	Best
adaptec1	64.7	72.1 ± 1.5	69.3	71.0 ± 2.1	**68.0**	70.1 ± 1.1	68.3	**69.1 ± 1.0**	68.1
adaptec2	75.8	109.0 ± 12.3	90.8	108.8 ± 8.1	95.6	96.0 ± 20.9	80.5	**81.6 ± 2.6**	**79.4**
adaptec3	153.3	185.3 ± 5.8	178.3	180.8 ± 4.8	173.3	165.2 ± 1.3	164.0	**163.9 ± 1.1**	**162.4**
adaptec4	142.4	162.5 ± 1.9	160.6	164.7 ± 4.4	159.0	150.6 ± 1.2	149.0	**147.9 ± 0.6**	**146.9**
bigblue1	85.3	89.0 ± 1.7	87.3	88.2 ± 0.6	87.2	**87.2 ± 0.5**	86.8	87.3 ± 0.5	**86.7**
bigblue2	125.4	T/O	T/O	T/O	T/O	**134.9 ± 0.5**	**134.4**	139.5 ± 1.0	138.1
bigblue3	273.8	325.7 ± 13.7	307.1	327.1 ± 17.4	313.2	319.6 ± 5.0	310.3	**299.3 ± 4.3**	**293.8**
bigblue4	643.2	T/O	T/O	T/O	T/O	702.8 ± 10.4	688.8	**678.9 ± 7.0**	**672.0**

Table 1: Comparison of mixed-size placement results on HPWL ($\times 10^6$). Each method performs macro placement using its respective approach, followed by standard cell placement using DREAMPlace. Lower values are better.



(a) DREAMPlace (b) MaskPlace (c) WireMask (d) ChipDiffusion (e) Ours

Figure 4: Qualitative comparison on adaptec3. (e) MacroDiff+ achieves a more uniform distribution of macros and whitespace compared to (b-d), ensuring sufficient spatial resources for downstream standard cell placement (see Appendix F for additional visualizations of our method across all benchmark designs).

method that formulates placement as a sequential decision process using visual masks.

- WireMask-BBO [Shi *et al.*, 2023]: An optimization-centric approach utilizing wire-mask-guided black-box optimization.
- ChipDiffusion [Lee *et al.*, 2025]: A recent generative method applying diffusion models to chip placement, serving as a direct generative baseline.

4.2 Macro Placement Results

Following the evaluation protocol described in Section 4.1, we report the final Half-Perimeter Wirelength (HPWL) after standard cell placement. Table 1 summarizes the quantitative comparison between MacroDiff+ and state-of-the-art baselines across all benchmark designs.

Superior Quality and Scalability

Across all benchmarks, MacroDiff+ consistently outperforms existing learning-based and optimization-based baselines. Compared to the RL-based MaskPlace and the optimization-based WireMask-BBO, our framework achieves substantial HPWL reductions of 6.1% and 6.2% on average, respectively. More importantly, our method demonstrates superior scalability. As shown in Table 1, both MaskPlace and WireMask-BBO failed to converge (T/O) on large-scale designs such as *bigblue2* and *bigblue4* due to computationally prohibitive sequential rollouts and heuristic exploration. In contrast, MacroDiff+ successfully scales to these massive netlists with superior efficiency. Specifically, while the generative baseline ChipDiffusion demonstrates competitive placement quality, it incurs substantial computational overhead on large designs. MacroDiff+ alleviates this bottleneck and maintains tractable runtime across all benchmarks (see Appendix E for detailed runtime analysis).

Stability and Topology Awareness

Compared to ChipDiffusion, a state-of-the-art generative baseline, our method reduces HPWL by 1.2% on average. While this margin might appear modest, a closer inspection reveals a significant advantage in stability. For instance, in *adaptec2*, our method exhibits a standard deviation of 2.6, whereas ChipDiffusion shows a high variance of 20.9. This indicates that ChipDiffusion's reliance on geometric patterns alone leads to unstable convergence. As observed from the generated samples (cf. Fig. 4), MacroDiff+ produces a remarkably uniform distribution of macros across the canvas. Unlike RL-based or optimization-centric baselines that often suffer from severe macro clumping—creating massive routing blockages in localized regions—our method distributes whitespace evenly. This balanced layout is critical as it secures the necessary spatial slack for the subsequent placement of millions of standard cells, effectively providing favorable conditions for downstream routing. By explicitly incorporating circuit connectivity via our dual-branch architecture, ours ensures that generated layouts are not only valid but also consistently optimized for global wirelength.

Config.	Displacement		HPWL	
	Value ↓	Degr. (%)	Value ↓	Degr. (%)
MacroDiff+	8.7	-	299.3 ± 4.3	-
w/o ϵ_{net}	11.8	36.3	297.7 ± 8.0	-0.5
w/o Hetero GNN	18.3	111.7	313.7 ± 12.0	4.8
w/o Transformer	17.0	95.4	310.5 ± 21.3	3.7

Table 2: Ablation Study on Dual-Branch Architecture. Comparison of Macro Legalization Results on Displacement ($\times 10^3$) and Mixed-Size Placement Results on HPWL ($\times 10^5$) for *bigblue3*.

Config.	Displacement		HPWL	
	Value ↓	Degr. (%)	Value ↓	Degr. (%)
MacroDiff+	8.7	-	299.3 ± 4.3	-
Overlap Guidance	18.5	114.0	314.1 ± 28.7	5.0
No Guidance	108.1	1147.4	305.7 ± 8.0	2.1

Table 3: Ablation Study on Gradient-Guided Sampling. Comparison of Macro Legalization Results on Displacement ($\times 10^3$) and Mixed-Size Placement Results on HPWL ($\times 10^5$) for *bigblue3*.

Closing the Gap with Analytical Solvers

Finally, our results significantly narrow the gap between learning-based methods and the analytical upper bound set by DREAMPlace. MacroDiff+ achieves placement quality within a gap of 3.2%–7.3% relative to DREAMPlace on most benchmarks. This shows that a physics-guided geometric diffusion model can approach the wirelength quality of traditional analytical optimization while offering the generative diversity inherent to data-driven approaches.

4.3 Ablation Studies

To validate the effectiveness of each component in MacroDiff+, we conducted extensive ablation studies. We analyze the impact of our dual-branch architecture and the physics-guided gradient mechanism separately.

Effectiveness of Dual-Branch Architecture

To assess the contribution of each architectural component, we compare the full MacroDiff+ framework against three ablated variants: (1) **w/o Net-Centric Noise (ϵ_{net})**, (2) **w/o Hetero GNN**, (3) **w/o Transformer**, where each variant disables the corresponding module.

Table 2 summarizes the legalization displacement and post-legalization HPWL results. The **MacroDiff+** achieves the best overall balance between legality, wirelength, and stability. Interestingly, the **w/o Net-Centric Noise** configuration results in a slight HPWL decrease (-0.5%) but incurs a 36.3% increase in displacement. This suggests that without net-driven gradients, the model over-clusters macros to minimize local wirelength, ignoring global density constraints.

On the other hand, removing the geometric branch (**w/o Transformer**) results in severe instability. This configuration exhibits the highest variance in HPWL (Std: 21.3), nearly $5\times$ that of the full model, and a 95.4% increase in displacement. This implies that while the GNN captures connectivity, it fails to construct a valid global floorplan, leading to massive overlaps that require drastic legalization. Finally, the absence of the Hetero GNN (**w/o Hetero GNN**) leads to the worst wirelength degradation (+4.8%) and displacement (+111.7%), confirming that the macro–net distinction is the most critical factor for topology-aware sampling.

Impact of Gradient Guidance

To evaluate the impact of our explicit physical steering, we analyze the loss components used in the sampling process (Eq. 9). We compare three strategies: (1) **MacroDiff+** (Full Guidance), applying both wirelength (HPWL) and overlap penalties; (2) **Overlap Guidance**, which focuses solely on resolving physical intersections; and (3) **No Guidance**, where the diffusion process proceeds without any gradient injection.

Table 3 reports the results. **MacroDiff+** yields superior placement quality, achieving both the lowest legalization displacement and downstream HPWL. In the **No Guidance** setting, the trajectory becomes physically unconstrained. While the HPWL degradation (+2.1%) appears moderate, the displacement explodes by over $11\times$ (+1147.4%). This indicates that without physical forces, the model blindly clusters macros to minimize wirelength, disregarding feasibility, which forces the legalizer to destroy the layout structure.

Conversely, **Overlap Guidance** reduces displacement compared to the unguided baseline but degrades HPWL by 5.0% and exhibits the highest variance (Std: 28.7). This suggests that focusing solely on legality causes macros to *over-separate* to avoid overlaps, breaking critical net connections. Thus, the joint physics guidance is essential: the HPWL gradient pulls connected macros together, while the overlap gradient ensures they remain valid, producing placements that are simultaneously legal and connectivity-efficient.

5 Conclusion

In this paper, we presented MacroDiff+, a physics-guided geometric diffusion framework that addresses the fundamental challenges of macro placement in modern VLSI design. Our dual-branch architecture synergizes topological insights from Heterogeneous GNNs with global geometric contexts from Transformers to capture the complex joint distribution of circuit connectivity and spatial constraints. Furthermore, the Physics-Guided Sampling strategy actively steers the generative trajectory using explicit physical gradients, ensuring that solutions are both statistically plausible and physically robust. Notably, the uniform distribution of whitespace in our placements leaves sufficient spatial slack for standard cell placement, as reflected in the consistent HPWL gains after the full mixed-size placement flow. Experimental results on ISPD2005 MMS benchmarks demonstrate that our approach outperforms existing baselines in stability and scalability, narrowing the gap between generative placement methods and analytical solvers. We believe this work takes a step toward topology-aware generative placement in physical design.

Acknowledgments

This research was supported by the Nano & Material Technology Development Program through the National Research Foundation of Korea (NRF) funded by Ministry of Science

and ICT (RS-2022-NR068233), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (RS-2024-00405991), and by the POSTECH Institute of Artificial Intelligence.

Contribution Statement

Jongho Yoon and Jinsung Jeon contributed equally to this work as co-first authors. Jinsung Jeon conducted this research while at the University of California, San Diego.

References

[Agnesina *et al.*, 2023] Anthony Agnesina, Puranjay Rajvanshi, Tian Yang, Geraldo Pradipta, Austin Jiao, Ben Keller, Brucek Khailany, and Haoxing Ren. Autodmp: Automated dreamplace-based macro placement. In *Proceedings of the 2023 International Symposium on Physical Design*, pages 149–157, 2023.

[Brody *et al.*, 2021] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021.

[Chan *et al.*, 2005] Tony Chan, Jason Cong, and Kenton Sze. Multilevel generalized force-directed method for circuit placement. In *Proceedings of the 2005 international symposium on physical design*, pages 185–192, 2005.

[Chen *et al.*, 2023a] Yifan Chen, Jing Mai, Xiaohan Gao, Muhan Zhang, and Yibo Lin. Macrorank: Ranking macro placement solutions leveraging translation equivariancy. In *Proceedings of the 28th Asia and South Pacific Design Automation Conference*, pages 258–263, 2023.

[Chen *et al.*, 2023b] Yifan Chen, Zaiwen Wen, Yun Liang, and Yibo Lin. Stronger mixed-size placement backbone considering second-order information. In *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pages 1–9. IEEE, 2023.

[Gao *et al.*, 2022] Xiang Gao, Yi-Min Jiang, Lixin Shao, Pedja Raspopovic, Menno E Verbeek, Manish Sharma, Vineet Rashingkar, and Amit Jalota. Congestion and timing aware macro placement using machine learning predictions from different data sources: Cross-design model applicability and the discerning ensemble. In *Proceedings of the 2022 International Symposium on Physical Design*, pages 195–202, 2022.

[Ho *et al.*, 2020] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[Lai *et al.*, 2022] Yao Lai, Yao Mu, and Ping Luo. Maskplace: Fast chip placement via reinforced visual representation learning. *Advances in Neural Information Processing Systems*, 35:24019–24030, 2022.

[Lai *et al.*, 2023] Yao Lai, Jinxin Liu, Zhentao Tang, Bin Wang, Jianye Hao, and Ping Luo. Chipformer: Transferable chip placement via offline decision transformer. In *International Conference on Machine Learning*, pages 18346–18364. PMLR, 2023.

[Lee *et al.*, 2025] Vint Lee, Minh Nguyen, Leena Elzeiny, Chun Deng, Pieter Abbeel, and John Wawrzynek. Chip placement with diffusion models. In *Forty-second International Conference on Machine Learning*, 2025.

[Mirhoseini *et al.*, 2021] Azalia Mirhoseini, Anna Goldie, Mustafa Yazgan, Joe Wenjie Jiang, Ebrahim Songhori, Shen Wang, Young-Joon Lee, Eric Johnson, Omkar Pathak, Azade Nova, et al. A graph placement methodology for fast chip design. *Nature*, 594(7862):207–212, 2021.

[Nam *et al.*, 2005] Gi-Joon Nam, Charles J Alpert, Paul Villarrubia, Bruce Winter, and Mehmet Yildiz. The ispd2005 placement contest and benchmark suite. In *Proceedings of the 2005 international symposium on Physical design*, pages 216–220, 2005.

[Newman, 2003] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[Qiu *et al.*, 2023] Yihang Qiu, Yan Xing, Xin Zheng, Peng Gao, Shuting Cai, and Xiaoming Xiong. Progress of placement optimization for accelerating vlsi physical design. *Electronics*, 12(2):337, 2023.

[Sechen and Sangiovanni-Vincentelli, 2003] Carl Sechen and Alberto Sangiovanni-Vincentelli. The timberwolf placement and routing package. *IEEE Journal of Solid-State Circuits*, 20(2):510–522, 2003.

[Shi *et al.*, 2023] Yunqi Shi, Ke Xue, Song Lei, and Chao Qian. Macro placement by wire-mask-guided black-box optimization. *Advances in Neural Information Processing Systems*, 36:6825–6843, 2023.

[Sun and Yang, 2023] Zhiqing Sun and Yiming Yang. Difusco: Graph-based diffusion solvers for combinatorial optimization. *Advances in neural information processing systems*, 36:3706–3731, 2023.

[Trippe *et al.*, 2023] Brian L Trippe, Jason Yim, Doug Tischer, David Baker, Tamara Broderick, Regina Barzilay, and Tommi S Jaakkola. Diffusion probabilistic modeling of protein backbones in 3d for the motif-scaffolding problem. In *ICLR*, 2023.

[Tseng, 2024] I-Lun Tseng. Challenges in floorplanning and macro placement for modern socs. In *Proceedings of the 2024 International Symposium on Physical Design*, pages 71–72, 2024.

[Xue *et al.*, 2024] Ke Xue, Ruo-Tong Chen, Xi Lin, Yunqi Shi, Shixiong Kai, Siyuan Xu, and Chao Qian. Reinforcement learning policy as macro regulator rather than macro placer. *Advances in Neural Information Processing Systems*, 37:140565–140588, 2024.

[Yan *et al.*, 2009] Jackey Z Yan, Natarajan Viswanathan, and Chris Chu. Handling complexities in modern large-scale mixed-size placement. In *Proceedings of the 46th Annual Design Automation Conference*, pages 436–441, 2009.