RuleLearner: OPC Rule Extraction From Inverse Lithography Technique Engine

Ziyang Yu, Su Zheng, Wenqian Zhao, Shuo Yin, Xiaoxiao Liang, *Graduate Student Member, IEEE*, Guojin Chen, Yuzhe Ma, *Member, IEEE*, Bei Yu, *Senior Member, IEEE*, and Martin D. F. Wong, *Life Fellow, IEEE*

***Abstract*—Model-based optical proximity correction (OPC) with subresolution assist feature (SRAF) generation is a critical standard practice for compensating lithography distortions in the fabrication of integrated circuits at advanced technology nodes. Typical model-based OPC and SRAF algorithms involve the selection of user-controlled rule parameters. Conventionally, these rules are heuristically determined and applied globally throughout the correction regions, which can be time consuming and require expert knowledge of the tool. Additionally, the correlations of rule parameters to the objectives are highly nonlinear. All these factors make designing a high-performance OPC engine for complex metal designs a nontrivial task. This article proposes RuleLearner, a comprehensive mask optimization system designed for SRAF generation and model-based OPC in real industrial scenarios. The proposed framework learns from the guidance of an information-augmented inverse lithography technique engine, which, although expressive for complex designs, is expensive to generate refined masks for a whole set of design clips. Considering the nonlinearity and the tradeoff between local and global performance, the extracted rule value distributions are further optimized with customized natural gradients. The sophisticated SRAF generation, the edge segmentation and movements are then guided by the rule parameter. Experimental results show that RuleLearner can be applied across different complex design patterns and achieve the best lithographic performance and computational efficiency.**

***Index Terms*—Design automation, design for manufacture, optical proximity correction.**

I. Introduction

In THE past decades, advances in semiconductor manufacturing technology have continuously pushed the boundaries of chip design, driving the need for more sophisticated computational lithography techniques. Applying resolution enhancement techniques (RETs), such as subresolution assist feature (SRAF) insertion and optical proximity correction (OPC), to adjust layout patterns has therefore become critical for acquiring high pattern fidelity and mask manufacturability.

OPC as a key RET, addresses unwanted wafer image distortions by predistorting lithography masks. Nowadays fast-developed OPC methods are mainly the model-based OPC [1], [2], [3], [4], inverse lithography technique (ILT) [5], [6], [7], [8] and deep learning-based OPC [9], [10], [11], [12]. Model-based OPC methods are grounded in sophisticated simulations, and meticulously adjust mask layouts through the edge segmentation and movement, augmented by the strategic placement of SRAFs [13], [14], [15], [16] to optimize the photolithographic process. Later on, ILT treats mask optimization as a pixel-based inverse imaging problem and iteratively refines the mask layout until the simulated wafer image closely aligns with the target. It has proven to be invaluable, particularly for its ability to handle intricate design patterns and generation of masks that account for optical and process effects. Finally, deep learning-based OPC accelerates the mask optimization process by predicting initial mask solutions using deep learning models, followed by post-refinement through the ILT engine.

These techniques, however, face challenges in complicated large-scale industrial patterns. ILT's complex nature demands significant computational resources, resulting in extended runtime (RT) unsuitable for large-scale designs. Besides the ILT masks are hard to manufacture in practice owing to pixel-based behaviors. For deep learning-based OPC, the accuracy of the generative model heavily depends on the quantity and quality of training data. These models may not generalize well across different pattern regions, and their opaque nature can lead to difficulties in understanding and diagnosing OPC results.

Currently, model-based OPC methods, which involve mask edge segmentation and fragment movement, are mainstream in real industry [17], [18], primarily due to their ability to handle complex pattern geometries and adapt to advanced lithography technologies. Key to these methods is the optimization of mask edge segmentation and fragments movement rules [1]. The rules mainly control how to segment the edge into different kinds of fragments and the displacement of fragments in every OPC iteration. For the SRAF insertion process, the rules can be categorized mainly into two groups: 1) the distance between SRAF and main features and 2) the size of SRAF. Optimizing these rule parameters is crucial for achieving the desired OPC solution, characterized by superior lithographic performance and high efficiency. In the industrial context, the prevalent approach mainly relies on OPC engineer manually configuring

Received 18 June 2024; revised 16 September 2024; accepted 28 October 2024. Date of publication 15 November 2024; date of current version 23 April 2025. This work was supported in part by the Research Grants Council of Hong Kong, SAR, under Grant CUHK14208021, and the MIND Project under Grant MINDXZ202404. This article was recommended by Associate Editor V. Pavlidis. *(Corresponding author: Bei Yu.)*

Ziyang Yu, Su Zheng, Wenqian Zhao, Shuo Yin, Guojin Chen, Bei Yu, and Martin D. F. Wong are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, SAR (e-mail: byu@cse.cuhk.edu.hk).

Xiaoxiao Liang and Yuzhe Ma are with the Microelectronics Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou 511458, China.

Digital Object Identifier 10.1109/TCAD.2024.3499909

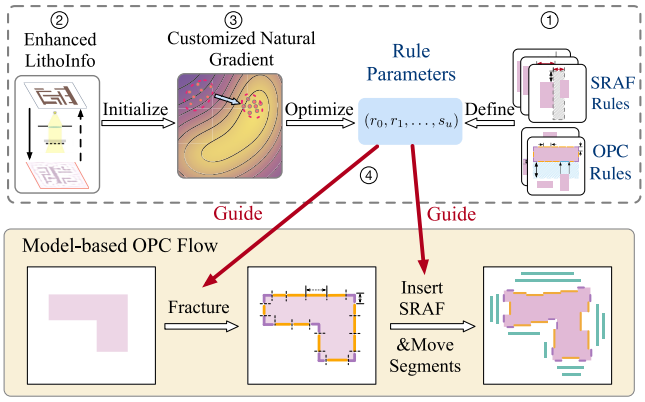

Fig. 1. Flow of RuleLearner: ① the considered rule set is predefined, ② the value is initialized from CTM, using enhanced lithography information, and ③ the rule candidates are sampled and the rule value distribution is updated through the natural gradient. ④ After that, the optimized rule is applied to guide the SRAF insertion and model-based OPC.

rule values, necessitating both sufficient domain knowledge and a significant amount of human labor. However, fine-tuning these parameters in high-dimension space is intricate and challenging, mainly due to their indirect and nonanalytical impact on lithographic performance and the delicate tradeoff between local and global performance. The genetic algorithm has been explored for tuning the parameters in SRAF generation rules [19], [20] and OPC recipe parameter tuning [21], revealing potential solutions that are difficult to achieve with simply hand-tuned recipes. These methods model the rule parameter tuning as a total opaque optimization problem and do not consider the information contained in lithography process. The flow could potentially undergo drastic changes in the gene composition of a generation, making it likely for the search algorithm to get stuck in local minima.

Recognizing these challenges, we assert it necessary to combine the information obtained from ILT engine to optimize rule values for complex 2-D metal design. In this work, we propose RuleLearner, a comprehensive mask optimization system aiming to optimize rule values for better lithography performance across various mask clips. We developed a customized ILT engine that leverages enhanced information in a hierarchical manner to quickly generate quasi-optimized continuous transmission mask (CTM) in only a few iterations for a small subset of design clips, from which the expressive intensity map could guide initial rule value distributions. Contrasting with binary masks that use binary values to indicate blockage or transparency at each grid site, CTM [22], [23], [24] allocate a floating intensity value to each site, providing a spectrum from complete blockage (value 0) to full transparency (value 1). The gradient of intensity values in CTM enables a more refined representation than traditional binary encoding. To address the lack of a direct and analytical relationship between the rule values and the resultant lithography performance, and achieve better tradeoff between local and global performance, we incorporate the exponential extension of lithography-aware natural gradient [25], [26]. Instead of employing plain gradient search algorithms, this approach maintains and iteratively updates the search distribution toward enhanced expected pattern fidelity, guided by natural gradients. We utilize this parameterized search distribution to generate batches of rule parameters for pattern clips. Subsequently, SRAFs are created, and model-based OPC is applied to these clips based on the sampled rule parameters. The lithographic performance of each pattern clip is then evaluated as the fitness, and the natural gradient is estimated to update the rule parameter distribution accordingly. By identifying reusable lithography-aware mask updating and SRAF generation guidelines, we can reduce the complexity of mask generation to the application of these rules. This enables efficient, scalable, and manufacturable OPC correction without compromising accuracy.

TABLE I
GENERAL RULE PARAMETERS IN THIS ARTICLE

Rule parameter	Description
r_0	SRAF insertion forbidden range
r_1	Single level SRAF insertion range
r_2	Double level SRAFs insertion range
w_0	Single level SRAF width
w_1, w_2	Double level 1st, 2nd SRAF width
l_{SRAF0}	Single level SRAF length ratio
l_{SRAF1}, l_{SRAF2}	Double level 1st, 2nd SRAF length ratio
d_1, d_2	Double level 1st, 2nd SRAF distance
d_p	Projection consideration range
L_{tr}	Length threshold for corner segmentation
L_c, L_u	Corner and uniform fragment length
s_p, s_c, s_u	Projection, corner and uniform fragment step size

The major contributions are summarized as follows.

1) We propose a comprehensive modern mask optimization framework for both SRAF generation and model-based OPC.
2) We develop an information-augmented hierarchical ILT engine to better extract latent lithography knowledge during the optimization process.
3) We customize a lithography-aware natural gradient method to optimize the distribution of industrial-like mask optimization rules.
4) We enhance precision and generalizability in rule optimization through clip overlapping and adaptive sampling.
5) We conduct experiments on complex first metal layer clips, the performance shows the learned rules can generalize well on complex industrial-like designs.

The remainder of this article is organized as follows. Section II briefly reviews the basic concepts and formulates the rule learning problem. Section III provides a detailed discussion of the RuleLearner framework. Section IV demonstrates the effectiveness of our method, followed by the conclusion in Section V.

II. PRELIMINARIES

A. Compact Rule Set

In this work, we focus on a compact rule set suitable for industrial-like SRAF generation and model-based OPC. The considered rule parameters and descriptions are listed in Table I. It is worth mentioning that this framework can be extended to consider other rule parameters without much more effort.

SRAF Generation Rule: The SRAFs are strategically placed adjacent to isolated target patterns, enhancing spatial frequency components of the main features without being

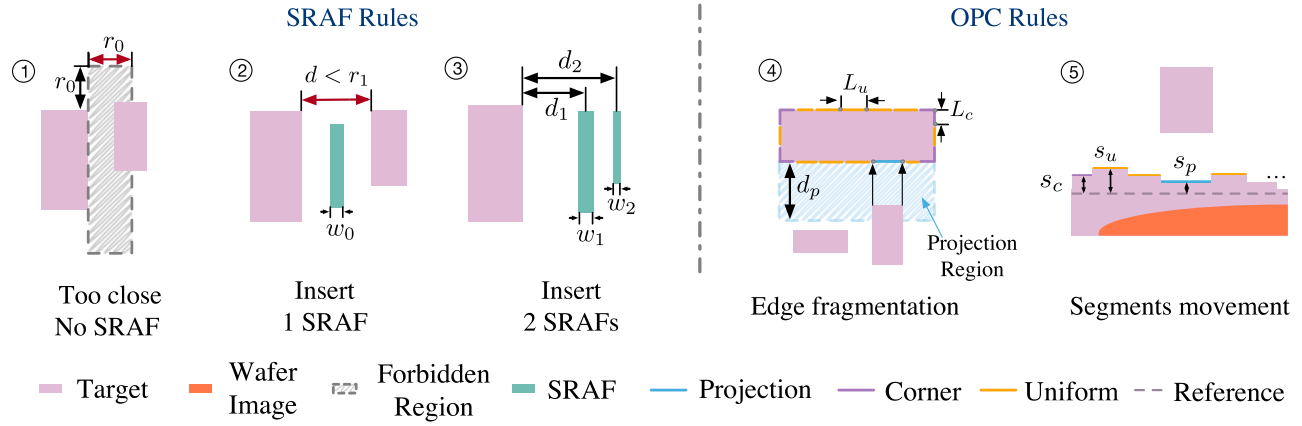

Fig. 2. Illustration of Considered rule cases. (a) Left part depicts three SRAF rule cases considered: ① no SRAF insertion, ② one level SRAF insertion, and ③ two levels of SRAF insertion. (b) Right part illustrates the two fundamental stages of model-based OPC: ④ edge segmentation and ⑤ fragments moving.

physically printed on the wafer. This technique significantly improves the imaging fidelity of target patterns. According to the left part of Fig. 2, this article follows an industrial procedure for SRAF generation [27], the deployment of SRAFs for an edge of length L is conditional as follows.

1) SRAFs are not inserted if a main pattern lies within a perpendicular distance r_0, which constitutes the SRAF insertion forbidden region.
2) Otherwise, if a main pattern is within distance r_1, a rectangular SRAF with dimensions w_0 in width and $l_{\text{SRAF1}} \times L$ in length is placed in the center between the nearest opposite edges.
3) In the absence of other main patterns within r_2, two SRAFs, each of width w_1 and w_2 and length $l_{\text{SRAF2}} \times L$ and $l_{\text{SRAF3}} \times L$ are positioned at distances d_1 and d_2, respectively.

The rule parameter for SRAF generation is thus defined as a vector containing above-mentioned parameters: $\boldsymbol{m}_{\text{SRAF}} = [r_0, r_1, r_2, w_0, w_1, w_2, l_{\text{SRAF1}}, l_{\text{SRAF2}}, l_{\text{SRAF3}}, d_1, d_2]$.

Model-Based OPC Rule: The considered model-based OPC consists of two steps: 1) edge segmentation and 2) fragments moving, as can be seen in the right part of Fig. 2.

1) During the segmentation phase, projection segments are created if another main pattern is present within the projection region with perpendicular threshold distance d_p. Subsequently, if the edge length L exceeds the threshold L_{tr}, corner fragments of length L_c are generated for precise control around corners, while the remainder of the edge is evenly divided into uniform fragments, each of length L_u. Conversely, for $L <= L_{tr}$, the edge is segmented into equal parts without considering the corner case. After the segmentation phase, fragment lengths are fixed.
2) In the subsequent movement stage, each iteration involves fragment adjustments either inward or outward to correct distortions in the printed wafer image. If the printed contour fits well with the target, the corresponding segment remains unchanged. The movement step sizes for projection, corner, and uniform fragments are denoted as s_p, s_c, and s_u, respectively.

Thus, the rule set defining the OPC process is encapsulated as $\boldsymbol{m}_{\text{OPC}} = [d_p, L_{tr}, L_c, L_u, s_p, s_c, s_u]$.

The complete rule set is a composite of the SRAF and OPC rules, and the rule vector is represented as a concatenation of two parts: $\boldsymbol{m} = [\boldsymbol{m}_{\text{SRAF}}^\top, \boldsymbol{m}_{\text{OPC}}^\top]^\top$.

B. Search Gradient Method

The general idea of the search gradient method is utilizing the sampled gradient of expected loss as the search gradient for updating search distribution parameters [28]. Without loss of generality, we assume a multinormal distribution for the search process of rule parameters. Let $\boldsymbol{\theta}$ denote the parameters of the probability density function $P(\boldsymbol{m}|\boldsymbol{\theta})$, and use $l(\boldsymbol{m})$ represent lithography cost of sampled rule vector $\boldsymbol{m}$, which will be illustrated in detailed in Section III-A. The expected cost under the search distribution is

$$L(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[l(\boldsymbol{m})] = \int l(\boldsymbol{m}) P(\boldsymbol{m}|\boldsymbol{\theta}) d\boldsymbol{m}. \tag{1}$$

To update distribution density parameter $\boldsymbol{\theta}$ toward lower expected cost, the direct derivative of expected loss $L(\boldsymbol{\theta})$ with respect to the distribution parameter can be derived using the log-likelihood trick

$$\begin{aligned} \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) &= \nabla_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{\theta}}[l(\boldsymbol{m})] = \nabla_{\boldsymbol{\theta}} \int l(\boldsymbol{m}) P(\boldsymbol{m}|\boldsymbol{\theta}) d\boldsymbol{m} \\ &= \mathbb{E}_{\boldsymbol{\theta}}\left[l(\boldsymbol{m}) \nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}|\boldsymbol{\theta})\right] \\ &\approx \frac{1}{N} \sum_{n=1}^{N} l(\boldsymbol{m}_n) \nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}_n|\boldsymbol{\theta}). \end{aligned} \tag{2}$$

In the final step, the search gradient is estimated through the Monte Carlo sampling strategy with sample size n. This gradient of expected cost enables a direct gradient descent approach for iterative search distribution updates

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) \tag{3}$$

where η is the learning rate.

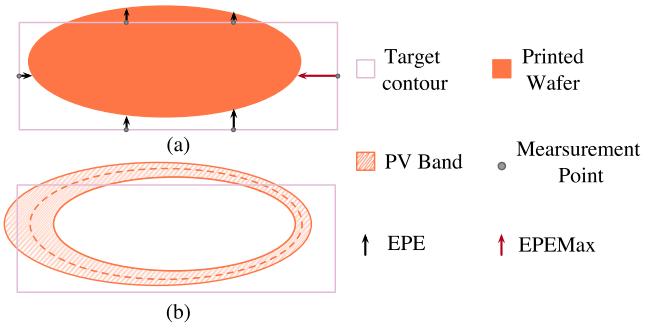

Fig. 3. (a) Measurement of EPE, EPESum is the summation of EPE over all measurement points and EPEMax is the EPEMax. (b) Measurement of PVBand.

C. Problem Formulation

In this work, we build a comprehensive mask optimization system focusing on extracting and optimizing rule values to improve the lithography performance of SRAF and model-based OPC. To evaluate the quality of the final lithographic simulation results, we employ the edge placement error and process variation band (PVBand) as criteria.

Definition 1 (Edge Placement Error): Edge Placement Error refers to the vertical or horizontal misalignment between the lithography contour under nominal condition and the desired contour of the target pattern, measured as the Manhattan distance between these two contours, as is shown in Fig. 3(a). The measurement points are evenly distributed along the contours of the target shape. We adopt two metrics for comprehensive pattern fidelity evaluation as follows.

1) *Total EPE Lengths (EPESum):* The sum of EPE distances over all measurement points, offering a comprehensive quantification of overall mask fidelity.
2) *Maximum EPE (EPEMax):* The largest EPE value among all measurement points, serving as a critical indicator of the most severe pattern distortions.

Definition 2 (PVBand): In practical lithography, process variations can induce deviations in the printed images, risking printing failures. The PVBand is delineated by the XOR region among multiple contours under variant process conditions, as can be seen in Fig. 3(b).

Given the above lithography performance evaluation metrics, we can formulate our SRAF and the model-based OPC rule optimization problem.

Problem 1 (Rule Learning-Based Mask Optimization Problem): Given a small set of metal layer clips from large-scale design patterns and specific rule patterns, the objective is to design a mask optimization system that can extract and optimize rule values $\boldsymbol{m} = [\boldsymbol{m}_{\text{SRAF}}^{\top}, \boldsymbol{m}_{\text{OPC}}^{\top}]^{\top}$ and applied to other unseen metal clips, with the EPESum, EPEMax, and *PVBand* area minimized, and OPC RT as short as possible.

III. RuleLearner Framework

The workflow of our RuleLearner framework is illustrated in Fig. 1. After the considered rule set is predecided, the sampled subset of metal layer clips are processed through the customized ILT engine and quasi-optimized into CTM for rule value initialization. The rules are then evaluated based on the quality of corrected masks, and rule distribution is optimized using the lithography-aware exponential natural evolution strategy. Finally, the optimized rule can be applied to guide the SRAF and model-based OPC on other test pattern clips.

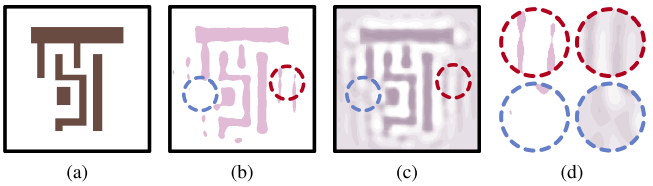

Fig. 4. (a) Target pattern of a complex metal design, (b) corresponding optimized binary mask, and (c) information-enhanced CTM. (d) Comparisons in blue circles indicate that transmissivity information is lost in the binary mask, while comparisons in red circles show that SRAFs generated in the binary mask are unstructured.

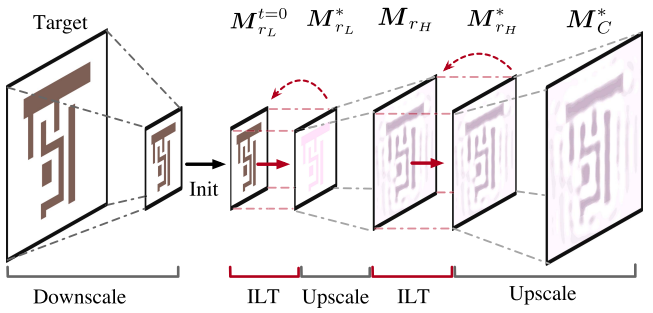

Fig. 5. Illustration of hierarchical information-enhanced CTM generation flow.

A. Latent Information Extraction

As can be seen in Fig. 4, compared with commonly seen binary mask which has either total blockage or transparency at each grid site, the information-enhanced CTM provides a more sophisticated spatial frequency indication, allowing higher frequency diffraction components to be involved in the imaging process [24]. Additionally, it preserves the latent information for otherwise unstructured SRAFs [23]. In our framework, the generation of CTM depends on a customized ILT engine.

Forward lithography simulation models the transformation of a mask $\boldsymbol{M}(x, y)$ into a wafer image $\boldsymbol{R}(x, y)$, where the light passing through the mask creates aerial intensity $\boldsymbol{I}(x, y)$ on the wafer. The aerial image $\boldsymbol{I}(x, y)$ is then transformed into the wafer image $\boldsymbol{R}(x, y)$ through the resist model by comparing the aerial intensity to the photo-resist intensity threshold

$$\boldsymbol{R}(x, y) = \begin{cases} 1, \text{ if } \boldsymbol{I}(x, y) \geq I_{\text{th}} \\ 0, \text{ if } \boldsymbol{I}(x, y) < I_{\text{th}} \end{cases} \tag{4}$$

where I_{th} is the intensity threshold controls the binary image on the wafer plane. This forward lithography process can be generally expressed with

$$\boldsymbol{R} = \Gamma(\boldsymbol{M}) \tag{5}$$

where Γ is the lithography engine. ILT then takes into account the desired pattern on the wafer and the lithography system parameters to compute the optimal transmission values for each mask site. The objective considers minimizing both the

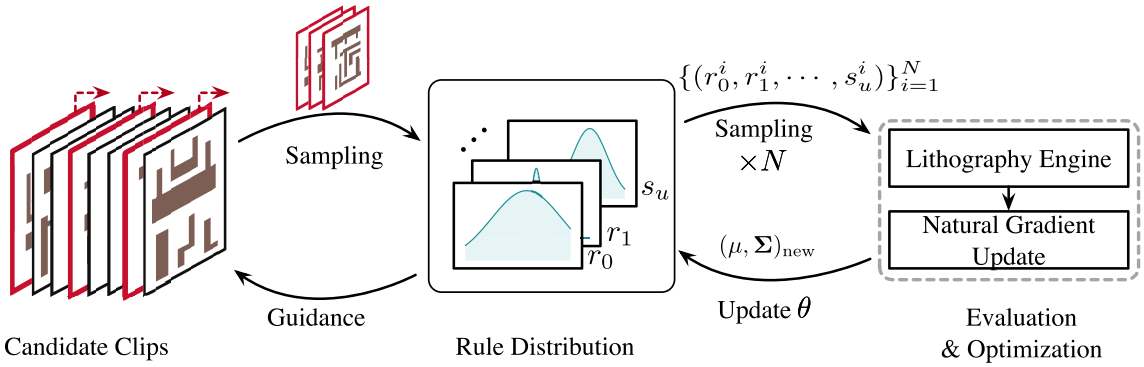

Fig. 6. Illustration of rule distribution update flow.

deviation between the wafer image under nominal condition and the target image $\boldsymbol{R}^*(\boldsymbol{M})$, and the PVBand area. The loss function $l(\cdot)$ in the right hand side of (1) for a fixed rule vector $\boldsymbol{m}$ is also a function of the binary mask $\boldsymbol{M}$

$$l(\boldsymbol{M}) = \left\|\boldsymbol{R}(\boldsymbol{M}) - \boldsymbol{R}^*(\boldsymbol{M})\right\|^2 + \beta\|\boldsymbol{R}_{\text{out}}(\boldsymbol{M}) - \boldsymbol{R}_{\text{in}}(\boldsymbol{M})\|^2 \quad (6)$$

where $\boldsymbol{R}_{\text{out}}(\boldsymbol{M})$ denotes the outermost contour, $\boldsymbol{R}_{\text{in}}(\boldsymbol{M})$ is the innermost contour, and β is the weighting coefficient. For binary mask updating using gradient descent, an unconstrained intermediate variable $\boldsymbol{M}'$ is utilized

$$\boldsymbol{M} = \text{sig}(\boldsymbol{M}') = \frac{1}{1 + \exp\left[-\alpha_{\boldsymbol{M}'} * \boldsymbol{M}'\right]}. \quad (7)$$

Due to foundry manufacturing constraints, only binary masks are producible. Previous works [6], [29] used a large steepness for a distinct binary transition. Flattening the transition with smaller $\alpha_{\boldsymbol{M}'}$ enhances the light transmissivity distribution expression and enriches the gradient detail.

The information-enhanced CTM is optimized with $\boldsymbol{M}'$ iteratively updated to minimize the cost function

$$\boldsymbol{M}' \leftarrow \boldsymbol{M}' - \Delta t \frac{\partial l(\boldsymbol{M})}{\partial \boldsymbol{M}'}. \quad (8)$$

To facilitate the backpropagation of loss gradients, in this section we adopt the accurate deep lithography simulator (DLS) [30] as the forward lithography simulator to update the CTM.

To better balance accuracy and efficiency in spatial transmissivity calculations, updates to the information-enhanced CTM are conducted hierarchically, as can be seen in Fig. 5. The target design initially undergoes downscaling with 8×8 average pooling for fast optimization at low resolution r_L, followed by a twofold upscaling for detailed optimization at a higher resolution r_H. We denote the optimization at resolution r ($r \in \{r_L, r_H\}$) as $\boldsymbol{M}_r^* = \min l(\boldsymbol{M}_r)$. The initial mask is represented by $\boldsymbol{M}_r^{t=0}$. Optimization at the lower resolution r_L enables rapid exploration, setting initial parameters for higher resolution r_H refinement. And the information-enhanced CTM $\boldsymbol{M}_C^*$ is acquired by a fourfold upscaling to restore the original size. This approach is encapsulated in the ILT methodology as

$$\begin{aligned} \boldsymbol{M}_C^* &= \text{Upscale}\left(\boldsymbol{M}_{r_H}^*, 4\right) \\ &\text{s.t. } \boldsymbol{M}_{r_H}^{t=0} = \text{Upscale}\left(\boldsymbol{M}_{r_L}^*, 2\right). \end{aligned} \quad (9)$$

In the above equation, $\boldsymbol{M}_{r_L}^*$ and $\boldsymbol{M}_{r_H}^*$ are the CTM optimized at lower and high resolution using (8), respectively. Function $\text{Upscale}(\boldsymbol{M}, k)$ upscale the mask $\boldsymbol{M}$ using nearest neighbor interpolation with scale factor k.

The optimized CTM reveals significantly enhanced latent information. The transmissivity distribution pattern near the main pattern's edge demonstrates dynamic shrinking and expansion, aligning closely with fragment-level changes. The parameter $\boldsymbol{m}_{\text{OPC}}$ on the ith clip is initialized to approach the shape of generated CTM on the ith clip. Accordingly, The SRAF parameter $\boldsymbol{m}_{\text{SRAF}}$ is initially tailored to match isolated highly transmissive regions in the respective CTM. The transmissivity distribution guides the rule value initialization, greatly improving the efficiency of the optimization process.

B. Lithography-Aware Natural Evolution Strategy for Rule Parameter Optimization

The generated CTM is a quasi-optimized mask, yet the latent information it yields might be implicit and potentially nongeneralizable across different pattern clips, which needs further consideration. Another primary challenge is the absence of a direct analytical relationship between rule parameters and final lithographic performance. The effects of rule parameter values, characterized by their nondifferentiability, discontinuity, and high stochasticity, preclude the direct use of gradient descent-based algorithms for direct updates on the individual rule vector $\boldsymbol{m}$.

However, direct implementation of the plain search gradient, as discussed in Section II-B, leads to unstable and unsatisfactory performance, even in simple quadratic cases [25]. It seeks the steepest descent direction in the space of rule parameters $\boldsymbol{\theta}$, and treats the parameter space as a Euclidean space, using the Euclidean metric to measure the distance between parameter vectors. This metric's dependence on the parameterization implies divergent gradients and updates under reparameterization for the complex rule optimization problem.

In addressing this, the natural gradient method [25] offers a robust alternative, optimizing the distribution parameters of rule values instead to address these challenges effectively. To this end, this section describes how to incorporate lithography performance guidance into rule parameter updating process, to further improve the efficiency of mask optimization task.

Lithography-Aware Natural Gradient: The key idea is to remove the dependence on the parameterization by relying on a more "natural" measure of distance $D(\boldsymbol{\theta}'||\boldsymbol{\theta})$ between probability distributions $P(\boldsymbol{m}|\boldsymbol{\theta})$ and $P(\boldsymbol{m}|\boldsymbol{\theta}')$. The update of $\boldsymbol{\theta}$ is expected to be in a direction that maximally decreases the expected lithography cost, while imposing the constraints on the information gain at each step. The natural gradient can then be formalized as the solution to the constrained lithography rule value optimization problem

$$\begin{aligned} \min_{\delta\boldsymbol{\theta}} \quad & L(\boldsymbol{\theta}-\delta\boldsymbol{\theta}) \approx L(\boldsymbol{\theta}) - \delta\boldsymbol{\theta}^{\top}\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) \\ \text{s.t.} \quad & D(\boldsymbol{\theta}-\delta\boldsymbol{\theta}||\boldsymbol{\theta}) = \epsilon \end{aligned} \tag{10}$$

where $L(\boldsymbol{\theta})$ is the expected lithography cost for rule distribution parameter $\boldsymbol{\theta}$. One such natural distance measure between two probability distributions is the Kullback–Leibler (KL) divergence. Since we have $\lim \delta\boldsymbol{\theta} \to 0$, the nature measure of distance is

$$D_{KL}(\boldsymbol{\theta}-\delta\boldsymbol{\theta}||\boldsymbol{\theta}) = \frac{1}{2}\delta\boldsymbol{\theta}^{\top}\boldsymbol{F}(\boldsymbol{\theta})\delta\boldsymbol{\theta}. \tag{11}$$

In the above equation, $\boldsymbol{F}(\boldsymbol{\theta})$ is Fisher information matrix of the given parametric family of search distributions, which is represented as

$$\begin{aligned} \boldsymbol{F}(\boldsymbol{\theta}) &= \int P(\boldsymbol{m}|\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}|\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}|\boldsymbol{\theta})^{\top} d\boldsymbol{m} \\ &= \mathbb{E}\Big[\nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}|\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}|\boldsymbol{\theta})^{\top}\Big]. \end{aligned} \tag{12}$$

The solution to the constrained optimization problem in (10) can be found using a Lagrangian multiplier λ, and generate the necessary condition

$$\boldsymbol{F}(\boldsymbol{\theta})\delta\boldsymbol{\theta} = \gamma\nabla_{\boldsymbol{\theta}} L. \tag{13}$$

The natural gradient can be determined to be the same direction with $\delta\boldsymbol{\theta}$ calculated

$$\tilde{\nabla}_{\boldsymbol{\theta}} L = \boldsymbol{F}^{-1}\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}). \tag{14}$$

The parameter can thus be updated with the learning rate η

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \cdot \boldsymbol{F}^{-1}\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}). \tag{15}$$

By combing (15), (12), and (2), the lithography performance can be utilized to update the rule parameter distribution, which would otherwise be prohibited due to the absence of an analytical relationship between lithography performance and rule parameters.

Rank-Based Fitness Shaping: Mask optimization problem is a highly nonconvex problem, the optimization path for different rule vectors $\boldsymbol{\theta}$ on the same mask clip could be different. Additionally, different design pattern clips have varying geometrical complexities, leading to distinct final lithography performance after mask optimization across pattern clips. These variabilities in mask optimization can distort the gradient due to fluctuating lithography costs.

To mitigate this issue, our RuleLearner transforms lithography costs into utility values $u_1 \geq \cdots \geq u_N$, ranking individuals by cost, where $\boldsymbol{m}_i$ is the ith highest-ranking individual in the population when sorted by ascending cost, $\boldsymbol{m}_1$ is the rule parameter with the lowest cost, and $\boldsymbol{m}_N$ is the one with the highest. This approach revises the gradient estimate in (2) for scale invariance and rank preservation among diverse pattern clips

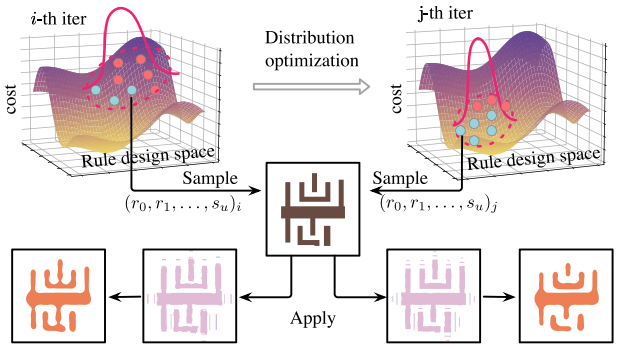

Fig. 7. Rule distribution iteratively optimizes toward a lower expected cost. Rule vectors sampled from the updated rule distribution will lead to better lithography performance.

$$\nabla_{\boldsymbol{\theta}} L(\boldsymbol{\theta}) \approx \frac{1}{N}\sum_{n=1}^{N} u_n \nabla_{\boldsymbol{\theta}} \log P(\boldsymbol{m}_n|\boldsymbol{\theta}). \tag{16}$$

The selection of u_i s based on insights from the previous covariance matrix adaptation evolution strategy (CMA-ES) [31]

$$u_n = \frac{\max(0, \log(N/2+1) - \log n)}{\sum_{m=1}^{N} \max(0, \log(N/2+1) - \log m)} - \frac{1}{N}. \tag{17}$$

Natural Exponential Extension: In d-dimensional multivariate Gaussian distribution $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$, the updated $\boldsymbol{\Sigma}$ should keep positive definite, which is not guaranteed a priori. Besides, the distribution parameter $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ has $d + d(d+1)/2 \in \mathcal{O}(d^2)$ components, the Fisher information matrix in (12) consists of $\mathcal{O}(d^4)$, and its inversion requires $\mathcal{O}(d^6)$ operations. To mitigate these challenges, we adopt an exponential map of the covariance matrix and update the distribution vector using natural coordinates. Instead of calculating new $(\boldsymbol{\mu}', \boldsymbol{\Sigma}')$ directly, we factorize $\boldsymbol{\Sigma} = \boldsymbol{Q}^{\top}\boldsymbol{Q}$ and represent the updated search distribution using the tangent space of the parameter manifold

$$(\boldsymbol{\delta}, \boldsymbol{W}) \mapsto (\boldsymbol{\mu}', \boldsymbol{Q}') = \left(\boldsymbol{\mu} + \boldsymbol{Q}\boldsymbol{\delta},\ \boldsymbol{Q}\exp\left(\frac{1}{2}\boldsymbol{W}\right)\right). \tag{18}$$

This coordinate system is natural in the sense that the Fisher matrix $\boldsymbol{F}(\boldsymbol{\theta})$ w.r.t. an orthonormal basis of $(\boldsymbol{\delta}, \boldsymbol{W})$ is the identity matrix. The current search distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{Q}\boldsymbol{Q}^{\top})$ is encoded as $(\boldsymbol{\delta}, \boldsymbol{W}) = (0, 0)$. In the new coordinate system, the log density becomes

$$\log P(\boldsymbol{m}|\boldsymbol{\delta}, \boldsymbol{W}) = -\frac{d}{2} - \mathrm{tr}(\boldsymbol{Q}) - \frac{1}{2}\left\|\exp\left(\frac{\boldsymbol{W}}{2}\right)\boldsymbol{Q}^{-1}\cdot(\boldsymbol{m}-\boldsymbol{\mu})\right\|^2. \tag{19}$$

Consider a sample $m_n = \boldsymbol{\mu} + \boldsymbol{Q}\cdot\boldsymbol{t}_n$ with $\boldsymbol{t}_n \sim \mathcal{N}(0,\ I)$, the gradient on the current distribution with respect to the $\boldsymbol{\delta}$ is

Algorithm 1 Rule Value Optimization via Exponential NES

Input: Lithography cost l, initial parameter $\boldsymbol{\mu}_0$, $\boldsymbol{\Sigma}_0 = \boldsymbol{Q}^\top \boldsymbol{Q}$.
Output: The optimized distribution parameter $\boldsymbol{\theta}^* = (\boldsymbol{\mu}^*, \boldsymbol{\Sigma}^*)$.

```
Initialize σ ← ∛|det(Q)|, W ← det(Q)/σ;
repeat
    for n in 1, ..., N do
        Draw sample t_n ~ N(0, I) and pattern clips;
        Rule vector m_n ← μ + σ B^⊤ t_n;
        Conduct mask optimization with rule m_n on clips;
        Evaluate average lithography cost l(m_n);
    end for
    Sort (t_n, m_n) with respect to l(m_n);
    Calculate u_η                                   ▷ (17);
    ∇_δ L ← Σ_{n=1}^{N} u_n · t_n;
    ∇_W L ← 1/2 Σ_{n=1}^{N} u_n · (t_n t_n^⊤ − I);
    ∇_σ L ← tr(∇_W L)/d;
    ∇_U L ← ∇_W L − ∇_σ L;
    μ ← μ + η_μ · σ U · ∇_δ L;
    σ ← σ · exp(η_σ)/2 · ∇_σ L;
    U ← U · (η_U/2·)∇_U;
until Stopping criterion is met
```

$$\begin{aligned}\nabla_{\boldsymbol{\delta}} L(0,0) &= \sum_{n=1}^{N} u_n \cdot \nabla_{\boldsymbol{\delta}}|_{\boldsymbol{\delta}=0} \log P(\boldsymbol{m}_n|\boldsymbol{\delta}, \boldsymbol{W}=0) \\ &= \sum_{n=1}^{N} u_n \cdot \nabla_{\boldsymbol{\delta}}|_{\boldsymbol{\delta}=0}\left[-\frac{1}{2}\left\|\boldsymbol{Q}^{-1}\cdot(\boldsymbol{m}_n - (\boldsymbol{\mu}+\boldsymbol{\delta}))\right\|^2\right] \\ &= \sum_{n=1}^{N} u_n \cdot \boldsymbol{Q}^{-1}\cdot(\boldsymbol{m}_n - \mu) = \sum_{n=1}^{N} u_n \cdot \boldsymbol{t}_n. \end{aligned} \tag{20}$$

Similarly, the gradient with respect to $\boldsymbol{W}$ can be derived

$$\begin{aligned}\nabla_{\boldsymbol{W}} L(0,0) &= \sum_{n=1}^{N} u_n \cdot \nabla_{\boldsymbol{W}}|_{\boldsymbol{W}=0} \log P(\boldsymbol{m}_n|\boldsymbol{\delta}=0, \boldsymbol{W}) \\ &= \frac{1}{2}\sum_{n=1}^{N} u_n \cdot \left(\boldsymbol{t}_n \boldsymbol{t}_n^\top - I\right). \end{aligned} \tag{21}$$

To further improve the efficiency, the transformation matrix $\boldsymbol{Q} = \sigma \cdot \boldsymbol{U}$ is decomposed into the step size $\sigma \in \mathbb{R}^+$ and the normalized matrix $\boldsymbol{U}$ with $\det(\boldsymbol{U}) = 1$. The gradient for σ and $\boldsymbol{U}$ are

$$\nabla_\sigma L(0,0) = \operatorname{tr}(\nabla_{\boldsymbol{W}} L(0,0))/d \tag{22}$$

$$\nabla_{\boldsymbol{U}} L(0,0) = \nabla_{\boldsymbol{W}} L(0,0) - \nabla_\sigma L(0,0). \tag{23}$$

Consequently, the natural gradient can be computed in $\mathcal{O}(d^3)$. The learning rate for $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$, and $\boldsymbol{U}$ are set to $\eta_{\boldsymbol{\mu}} = 1$ and $\eta_{\boldsymbol{\sigma}} = \eta_{\boldsymbol{U}} = ([3\cdot(3+\log(d))]/[5d^{1.5}])$, respectively, as adopted in CMA-ES. Given the above discussion, the general algorithm is shown in Algorithm 1.

C. Adaptive Sampling on Unseen Clips

The sampling-based rule distribution update, considering geometric properties, aims to boost performance in complex metal patterns. Yet, due to computational limits, mask optimization with this rule is clip-based, and optimizing a full mask for complex very large scale integration (VLSI) designs on a single system is unfeasible.

In lithography, the optical projection system's capabilities, dictated by factors like light wavelength and numerical aperture, set limits on linewidth and depth of focus. As design nodes shrink, diffraction effects become more significant [32], [33]. Consequently, mask optimization for a layout tile also involves considering the influence of adjacent tiles. Accordingly, we partition our layout into overlapping tiles as considered in [34] and [35], calculating each tile's width and height based on these considerations

$$k_w, k_h \leftarrow \left\lfloor \frac{\Lambda_w - s_w}{\lambda_w - s_w} \right\rfloor, \left\lfloor \frac{\Lambda_h - s_h}{\lambda_h - s_h} \right\rfloor \tag{24}$$

where Λ_w and Λ_h represent the width and height of the full layout, while λ_w and λ_h represent the width and height of the clipped tiles. The term s_w and s_h represent the stride in the width and height directions, which are predefined constants. $k_w \times k_h$ denotes the number of tiles, respectively.

The rule distribution is updated following lithography engine simulation and natural evolution optimization. To effectively generalize across a wide range of pattern complexities, pattern clip sampling must adapt to the distribution properties. If the rule favors denser SRAFs and coarser segmentation with more aggressive fragment movements, the resulting sampled clips may be sparse and simple. Consequently, the next iteration's sampling should prefer more complex clips with a higher metal area ratio. Conversely, if the rule involves smaller SRAFs and a more sophisticated model-based OPC process, the subsequent sampling should focus on sparser and simpler clips to maintain balance.

D. General Mask Optimization Flow

In this article we apply the rule into the real industrial-like mask optimization flow consists of the rule-based SRAF insertion and the model-based OPC.

The key components for SRAF generation is the position and the size information. Compared with simple one level rule-based SRAF insertion [1], the considered rules are capable of adapting to the intricate geometrical relationships inherent in complex main patterns. Since assist features are essential for isolated patterns, within our rule set, the degree of sparsity proximal to each boundary of the main pattern is examined. The optimized r_1–r_3 characterize this isolation and decide whether we could add SRAF and how many levels of SRAF should be added. Depending on the unique circumstances of each case, positional rule parameters, d_1 and d_2 are employed to define the SRAFs' proximity to the edges of the main pattern. Subsequently, the configuration of the SRAFs is delineated by shape rule parameters w_1–w_3 and l_{SRAF1}–l_{SRAF3}. The generated SRAFs are fixed in the following model-based OPC process. Notably, the RuleLearner can be seamlessly extended to accommodate more elaborate rules without much effort.

After SRAF generation, model-based OPC modifies complex main patterns to compensate for imaging distortions. Within a new complex clip, the edge of main pattern is segmented in a adaptive manner: the proximal effect from neighboring main patterns may have influence on the lithography result, Lithographic outcomes are significantly influenced

by proximity effects from adjacent patterns within a threshold distance d_p, necessitating distinct treatment of segmented projection fragments. If the length is smaller than l_{tr} we neglect the corner case, otherwise two corner fragments with length l_c are segmented, and the remaining part is segmented in equal length l_u. In every iteration, wafer image deviations from the target edge guide fragment offsetting: if the wafer image shrinks, fragments expand by fixed step sizes s_p, s_c, and s_u, respectively, based on fragment type. If protrusion occurs, fragments shrink accordingly. It is worth mentioning that, in addition to our mask optimization process design, the lithography-aware natural evolution strategy in RuleLearner can be extended to consider other lithography-related rule parameters in specific mask optimization processes.

This iterative process will stop if the cost is smaller than a predefined threshold or maximum iteration number is met.

IV. Experiments

A. Experimental Setup

Our RuleLearner is implemented in PyTorch, all the experiments are conducted on Linux system with an Nvidia GeForce RTX 3090 GPU. The Calibre-compatible lithography simulator is adopted to conduct the lithography process, which consists of an optical plus a compact resist model from the industry, the reference threshold is 0.25. The corresponding Calibre OPC scripts are from an industry partner. Following state-of-the-art works on OPC, such as AdaOPC [36] and LithoBench [37], we crop abundant layout tiles from the 45-nm designs synthesized by the IC design tool, OpenROAD [38]. Given a GDS-II layout file, we crop the layout into 1024 nm $\times$ 1024 nm tiles with a stride of 256 nm $\times$ 256 nm using KLayout [39] and extract the first metal layer. Each clip contains part of the entire design pattern, the local geometrical properties could be either simple (metal area ratio $<$ 0.25) or complex (metal area ratio $>$ 0.35). We select five simple clips and five complex clips to test the performance of the optimized rules. The detailed pattern area in nm^2 and area ratio of each benchmark are listed in Table II.

Following previous work, the performance is evaluated based on the final printed wafer image using the process variation band area (PVB) in square nanometers (nm^2), the summation of edge placement errors (EPESum) and maximum edge placement errors (EPEMax) are in nanometer (nm), and RT in seconds (s) as evaluation metrics. The EPE probe points are set evenly along every target edge with the distance of 40 nm, following the conventional strategy illustrated in [40]. The total number of probe points on every test case is listed in the final column in Table II. The PVB is the XOR region of wafer contours among different process conditions. The defocus values are chosen from the set $\{-10, 0, 10$ nm$\}$ where the nominal focus corresponds to 0 nm, and exposure dose values are chosen from the set $\{0.95, 1.0, 1.05\}$ with the nominal dose value set as 1.0.

TABLE II
Benchmark Statistics

ID	Area(nm^2)	Ratio	#Points
S1	189058	0.1803	68
S2	218209	0.2081	84
S3	240753	0.2296	155
S4	243899	0.2326	106
S5	261934	0.2498	139
C1	371930	0.3547	178
C2	385142	0.3673	198
C3	408525	0.3896	197
C4	428972	0.4091	210
C5	440926	0.4205	286
L1	859832	0.205	354
L2	872415	0.208	424
L3	947913	0.226	459
L4	952107	0.227	487
L5	1098908	0.262	570

B. Compare With Different Model-Based OPC Methods

In the first experiment, we evaluate the optimized mask qualities of RuleLearner in comparison with other model-based OPC methods. We compare our results with the industrial tool Calibre [18] and two state-of-the-art model-based OPC methods that can adapt to metal layer patterns. AccOPC [2] conducts mask optimization through the insertion of single-layer SRAFs, along with length-adaptive edge fragmentation and segment movements. CAMO [4] is a reinforcement learning-based OPC method that employs graph-based mask encoding to consider geometric correlations and RNN-based sequential segment movements.

As illustrated in Table III, RuleLearner demonstrates superior mask optimization performance. Benefiting from more sophisticated SRAF insertion decisions and segment control mechanisms, our results outperform AccOPC with an 8% drop in PVB area, a significant 18% reduction in aggregate EPE lengths, and a 15% reduction in EPEMax. Compared with CAMO, which employs a reinforcement learning strategy, RuleLearner exhibits enhanced scalability for intricate 2-D patterns, yielding reductions of 6% in average PVB area, 8% in EPE summation, and 21% in EPEMax length. Even compared with the commercial tool, RuleLearner achieves improvements of 6%, 3%, and 7% in PVB area, EPE summation, and EPEMax, respectively.

The mask optimization is a complex iterative process in a highly nonconvex space, and different methods require various numbers of iterations to reach the optimal results. Beyond mask quality metrics, the adaptive flexibility inherent in RuleLearner facilitates expedited mask optimization convergence. RuleLearner attains speedups of 1.55$\times$, 2.23$\times$, and 1.51$\times$ compared with Calibre, AccOPC, and CAMO, respectively.

We also evaluated the mask shots number of different methods, which represents the average number of rectangular shots needed to accurately replicate the optimized mask for the test cases. The result is shown in Fig. 8. Compared to other model-based OPC engines, our method results in more mask fracturing shots. This is because RuleLearner incorporates more sophisticated edge segmentation and SRAF insertion processes to achieve better wafer image quality.

C. Compare With Different Rule Optimization Methods

In the second experiment, we evaluate the quality of rules generated from RuleLearner from the quality of final optimized masks.

TABLE III
COMPARISON WITH DIFFERENT OPC ENGINES

ID	Calibre				AccOPC [2]				CAMO [4]				RuleLearner			
	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT
S1	18288	84	10	8.09	19558	89	10	11.21	18452	86	15	7.89	19171	72	10	5.14
S2	19105	60	12	8.47	18959	72	12	8.58	17549	60	15	6.79	18831	68	12	4.25
S3	52305	216	13	8.59	54029	217	14	12.14	52223	181	13	6.03	52800	172	12	5.5
S4	26790	122	12	8.21	27983	121	14	13.74	27549	128	16	9.2	21888	107	8	2.15
S5	35110	156	12	8.93	37673	200	15	14.08	37801	169	13	8.77	33024	139	11	4.58
C1	46046	271	15	9.02	44933	204	14	10.1	47330	265	15	8.62	45713	193	13	4.96
C2	61934	245	13	8.48	62132	314	14	9.33	59789	278	13	7.67	57424	315	13	5.24
C3	66240	258	14	8.77	66539	354	16	15.35	63785	253	14	8.8	54528	245	14	8.76
C4	56309	230	16	8.92	55095	244	14	12.77	55020	287	16	9.74	50816	237	14	8.13
C5	81510	214	13	8.76	82306	327	17	16.56	80445	241	18	10.21	81510	263	15	6.89
Avg.	46364	185.6	13	8.63	46920.7	214.2	14	12.39	45994.3	194.8	14.8	8.37	43570.5	181.1	12.2	5.56
Ratio	1.06	1.03	1.07	1.55	1.08	1.18	1.15	2.23	1.06	1.08	1.21	1.51	**1.00**	**1.00**	**1.00**	**1.00**

PVB unit: nm^2; EPE / EPEMax unit: nm; RT unit: s

TABLE IV
COMPARISON WITH DIFFERENT RULE OPTIMIZATION METHODS

ID	Expert				CMA-ES [31]				GA-Rule [19]				RuleLearner			
	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT
S1	19074	92	10	6.99	19168	91	11	7.21	18687	87	10	9.47	19171	72	10	5.14
S2	19175	69	12	6.04	18729	67	12	5.92	1903	72	15	5.41	18831	68	12	4.25
S3	54494	218	13	9.26	55486	183	13	6.42	64159	206	14	8.2	52800	172	12	5.5
S4	26995	142	16	10.61	26920	113	12	7.3	32351	111	8	8.52	21888	107	8	2.15
S5	36930	178	12	8.33	34975	157	12	6.09	42894	152	12	6.32	33024	139	11	4.58
C1	46841	268	14	9.29	47688	261	14	9.44	44008	307	21	6.51	45713	193	13	4.96
C2	62273	237	14	9.26	65192	322	15	7.69	60233	324	16	8.74	57424	315	13	5.24
C3	67874	365	14	9.54	61743	290	14	9.8	69411	311	12	10.26	54528	245	14	8.76
C4	57545	241	14	8.42	54520	288	14	8.93	54772	299	15	9.54	50816	237	14	8.13
C5	82963	309	16	10.26	82306	248	17	10.92	85012	375	22	8.72	81510	263	15	6.89
Avg.	47370.4	211.9	13.5	8.8	46672.7	202	13.4	7.97	49055.9	224.4	14.5	8.17	43570.5	181.1	12.2	5.56
Ratio	1.09	1.17	1.11	1.58	1.07	1.12	1.1	1.43	1.13	1.24	1.19	1.47	**1.00**	**1.00**	**1.00**	**1.00**

PVB unit: nm^2; EPESum / EPEMax unit: nm; RT unit: s

To the best of our knowledge, this is the first model-based OPC framework that considers rule optimization of both the SRAF generation stage and OPC stages. Considering this, we compare the developed RuleLearner with different parameter optimization methods, and test the generated rules using the same rule-based SRAF and model-based OPC. Experts sample different possible rule value candidates manually based on the parameter tuning experience and heuristically keep the best-performing one. This method is set to mimic the expert's behavior when selecting the proper rule value. CMA-ES [31] focuses on adapting the covariance matrix of the sampling distribution, allowing the algorithm to learn the shape of the objective function's landscape. GA-Rule [19] adopts a genetic algorithm to optimize only the SRAF insertion rule. For a more fair comparison, we reimplement the algorithm and optimize both SRAF and OPC rules. The update iterations for the above-mentioned methods are set to 50.

As shown in Table IV, RuleLearner achieves the best mask quality, optimizing the rules for superior performance across these ten cases. Our RuleLearner outperforms the expert tuning with 9% less PVB area, 17% less EPESum, and 11% less EPEMax. Compared with CMA-ES, the use of natural gradient and fitness shaping in RuleLearner led to more efficient updates on the rule value, which reduced PVB area by 7% with 12% less total EPE lengths and 10% less EPEMaxs. Besides, our RuleLearner utilizes the customized ILT engine to guide the optimization process, and the result is better than the previous GA-rule with 13%, 24% improvements in PVB area and EPESum lengths, together with 19% less EPEMax.

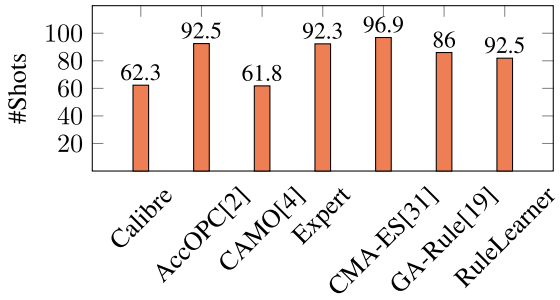

Fig. 8. Shot numbers comparison on normal layouts.

The rule generated with RuleLearner could also lead to the fastest mask optimization convergence time. Compared with the listed methods, our mask optimization RT could achieve 58%, 43%, and 47% speedup, respectively. This advantage could be more substantial when implementing in large set of mask clips.

The mask shot numbers of different rule optimization methods are comparable, as shown in Fig. 8. The difference mainly arise from the selected segment lengths and SRAF insertion decisions.

D. Visualization of Lithography Performance

Fig. 9 displays examples of SRAF insertion and mask optimization for two simple cases (S3 and S4) and two complex cases (C1 and C5) formulated using RuleLearner, alongside the resultant wafer images and PVB. An examination of the figure reveals that the mask produced by

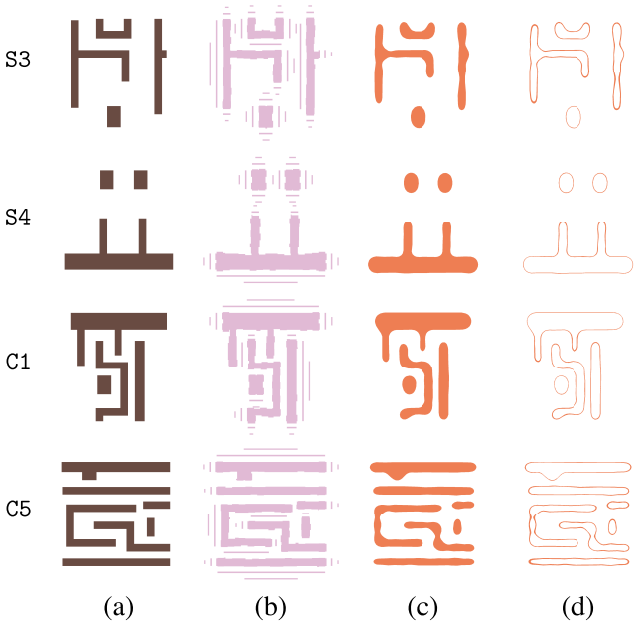

Fig. 9. Mask optimization results for four test cases with different complexities. (a) Target. (b) Optimized masks. (c) Wafer images. (d) Process variational bands.

the RuleLearner algorithm closely aligns with conventional industrial mask design principles, particularly evident in the gradation of SRAFs; those proximal to the main pattern are notably larger than their distal counterparts. During our optimization process, we prioritize minimizing distortions of wafer images with reduced EPE and clear hotspots under nominal conditions, and reducing the differences between different contours under varying process conditions. The printed wafer image under nominal condition could avoid critical hotspots, such as disruptive parts or unintended connection, even in complex cases.

E. Ablation Study on CTM Engine Guidance and Fitness Shaping

An ablation study was conducted to evaluate the impact of CTM engine guidance on optimization efficacy, with results presented in Table V. In the first row, the rule values are randomly initialized, followed by the same natural evolution optimization process. The result clearly demonstrates that rules initialized using the customized CTM engine outperform those with random initialization, as evidenced by improvements in PVB area, EPE summation length, EPEMax length, and optimization time.

The effectiveness of fitness shaping within the RuleLearner framework was also validated, with results displayed in the second row of Table V. Incorporating fitness shaping in RuleLearner effectively prevents premature convergence and numerical instability during the rule optimization process, which finally leads to better mask optimization results.

F. Comparison on Larger Layouts

To further validate the scalability of the proposed RuleLearner, we evaluated it on larger layouts. We selected `L1`–`L5` as five large clips, each sized 2048 nm × 2048 nm with a stride of 512 nm × 512 nm, cropped from the 45-nm designs. Each clip contains a part of the original design that is four times larger and exhibits more complex geometrical properties with additional patterns. An example of `L3` can be seen in Fig. 10. The detailed pattern area in nm^2, the area ratio and the number of EPE probe points are listed in the bottom part of Table II. The rule optimization process in RuleLearner involves optimizing rule parameters across different pattern clips, enabling it to effectively consider various local geometrical features. As shown in Table VI, RuleLearner exhibits superior mask optimization performance compared with various model-based OPC engines, in terms of PVB area, EPE summation, EPEMax length, and RT. The optimized rules achieves the best mask quality among different rule optimization methods, as shown in Table VII.

TABLE V
ABLATION STUDY

CTM engine	Fitness Shaping	PVB	EPESum	EPEMax	RT
	✓	49254	196.3	12.9	7.2
✓		48849	201.4	13.5	8.82
✓	✓	**43571**	**181.1**	**12.2**	**5.56**

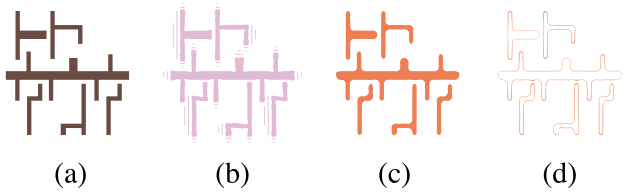

Fig. 10. Mask optimization results for large layout example. (a) Target. (b) Optimized masks. (c) Wafer images. (d) Process variational bands.

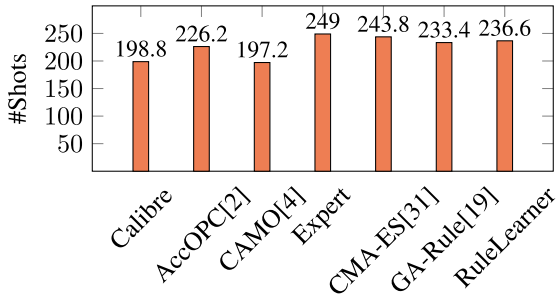

Fig. 11. Shot numbers comparison on large layouts.

The average shot numbers of different methods are also displayed in Fig. 11. Compared to layouts of normal size 1024 nm × 1024 nm, the shot numbers are larger due to increased pattern area and geometrical complexities. Similar to normal size results, the shot numbers of RuleLearner are slightly higher than those for Calibre, AccOPC [2], and CAMO [4], yet comparable with other rule optimization methods, such as Expert, CMA-ES [31], and GA-Rule [19].

V. CONCLUSION

We concentrate on optimizing rule values for intricate SRAF generation and model-based OPC processes. In this article, we propose RuleLearner, an innovative automated framework leveraging the ILT engine for the extraction of latent features and the optimization of rule values utilizing a natural evolution

TABLE VI
LARGE LAYOUT COMPARISON WITH DIFFERENT OPC ENGINES

ID	Calibre				AccOPC [2]				CAMO [4]				RuleLearner			
	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT
L1	90029	320	17	9.02	89488	382	17	11.31	93128	436	14	9.94	80080	359	14	8.08
L2	128073	641	13	8.97	124176	697	24	12.25	134112	754	16	8.85	127760	424	15	7.21
L3	141238	558	16	9.82	138336	468	18	11.48	144431	846	21	10.28	133920	419	13	8.47
L4	143198	499	16	9.23	131472	484	16	11.54	146708	674	14	13.21	129408	439	16	7.01
L5	150594	460	16	9.45	145104	517	19	13.69	152024	733	16	13.42	147856	515	15	7.86
Avg.	130626	495.6	15.6	9.3	125715	509.6	18.8	12.05	134081	688.6	16.2	11.14	123805	431.2	14.6	7.73
Ratio	1.06	1.15	1.07	1.2	1.02	1.18	1.29	1.56	1.06	1.60	1.11	1.44	**1.00**	**1.00**	**1.00**	**1.00**

PVB unit: nm^2; EPE / EPEMax unit: nm; RT unit: s

TABLE VII
LARGE LAYOUT COMPARISON WITH DIFFERENT RULE OPTIMIZATION METHODS

ID	Expert				CMA-ES [31]				GA-Rule [19]				RuleLearner			
	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT	PVB	EPESum	EPEMax	RT
L1	93684	392	18	10.52	86560	371	16	11.85	89824	382	17	13.20	80080	359	14	8.08
L2	128912	667	17	9.47	126096	690	18	10.76	130336	781	18	10.42	127760	424	15	7.21
L3	137056	709	18	10.90	136928	494	13	11.29	136736	481	16	11.44	133920	419	13	8.47
L4	134144	749	24	9.12	134864	542	16	9.46	139216	491	16	10.67	129408	439	16	7.01
L5	146384	641	19	11.04	138832	592	17	13.64	136768	571	16	12.14	147856	515	15	7.86
Avg.	128036	631.6	19.2	10.21	124656	537.8	16	11.4	126576	541.2	16.6	11.58	123805	431.2	14.6	7.73
Ratio	1.03	1.46	1.32	1.32	1.01	1.25	1.1	1.47	1.02	1.26	1.14	1.50	**1.00**	**1.00**	**1.00**	**1.00**

PVB unit: nm^2; EPESum / EPEMax unit: nm; RT unit: s

strategy. To the best of our knowledge, this is the first work to enable rule extraction from small sections of complex designs and generalize it to other parts. Experimental results, drawn from more complex metal layer, demonstrate the practical superiority of our methods compared with other methods. We believe that the strategic extraction of rules, as realized in RuleLearner, will play a crucial role in the evolution of modern lithographic mask optimization.

REFERENCES

[1] J. Kuang, W.-K. Chow, and E. F. Y. Young, "A robust approach for process variation aware mask optimization," in *Proc. IEEE/ACM Design, Autom. Test Europe (DATE)*, 2015, pp. 1591–1594.

[2] B. Jiang, H. Zhang, J. Yang, and E. F. Y. Young, "A fast machine learning-based mask printability predictor for OPC acceleration," in *Proc. IEEE/ACM 24th Asia South Pac. Design Autom. Conf. (ASPDAC)*, 2019, pp. 412–419.

[3] X. Liang, Y. Ouyang, H. Yang, B. Yu, and Y. Ma, "RL-OPC: Mask optimization with deep reinforcement learning," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 43, no. 1, pp. 340–351, Jan. 2024.

[4] X. Liang, H. Yang, K. Liu, B. Yu, and Y. Ma, "CAMO: Correlation-aware mask optimization with modulated reinforcement learning," 2024, *arXiv:2404.00980*.

[5] A. Poonawala and P. Milanfar, "Mask design for optical microlithography—An inverse imaging problem," *IEEE Trans. Image Process.*, vol. 16, pp. 774–788, 2007.

[6] J.-R. Gao, X. Xu, B. Yu, and D. Z. Pan, "MOSAIC: Mask optimizing solution with process window aware inverse correction," in *Proc. 51st ACM/IEEE Design Autom. Conf. (DAC)*, 2014, pp. 1–6.

[7] Y. Ma, J.-R. Gao, J. Kuang, J. Miao, and B. Yu, "A unified framework for simultaneous layout decomposition and mask optimization," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2017, pp. 81–88.

[8] Z. Yu, G. Chen, Y. Ma, and B. Yu, "A GPU-enabled level-set method for mask optimization," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 42, no. 2, pp. 594–605, Feb. 2023.

[9] H. Yang, S. Li, Z. Deng, Y. Ma, B. Yu, and E. F. Y. Young, "GAN-OPC: Mask optimization with lithography-guided generative adversarial nets," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 10, pp. 2822–2834, Oct. 2020.

[10] B. Jiang, L. Liu, Y. Ma, H. Zhang, E. F. Y. Young, and B. Yu, "Neural-ILT: Migrating ILT to neural networks for mask printability and complexity co-optimizaton," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2020, pp. 1–9.

[11] G. Chen, Z. Yu, H. Liu, Y. Ma, and B. Yu, "DevelSet: Deep neural level set for instant mask optimization," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2021, pp. 1–9.

[12] B. Zhu et al., "L2O-ILT: Learning to optimize inverse lithography techniques," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 43, no. 3, pp. 944–955, Mar. 2024.

[13] C. H. Wallace, P. A. Nyhus, and S. S. Sivakumar, "Sub-resolution assist features," U.S. Patent 7 632 610 B2, May 2009.

[14] R. Viswanathan, J. T. Azpiroz, and P. Selvam, "Process optimization through model based SRAF printing prediction," in *Proc. SPIE Adv. Lithogr.*, 2012, pp. 437–446.

[15] L. D. Barnes, B. D. Painter, and L. S. Melvin III, "Model-based placement and optimization of subresolution assist features," in *Proc. SPIE 19th Opt. Microlithogr.*, 2006, pp. 789–795.

[16] K. Sakajiri, A. Tritchkov, and Y. Granik, "Model-based SRAF insertion through pixel-based mask optimization at 32nm and beyond," in *Proc. SPIE 15th Photomask Next-Gener. Lithogr. Mask Technol.*, 2008, pp. 325–336.

[17] H. T. Vu, S. Kim, J. Word, and L. Y. Cai, "A novel processing platform for post tape out flows," in *Proc. SPIE 31st Opt. Microlithogr.*, 2018, pp. 219–224.

[18] (Siemens, Munich, Germany). *Calibre Design Solutions*. Aug. 2023. [Online]. Available: https://eda.sw.siemens.com/en-US/ic/calibre-design/

[19] Y. Xu, B. Zhang, C. Wang, W. Wilkinson, and J. Bolton, "The performance improvement of SRAF placement rules using GA optimization," in *Proc. SPIE Photomask Technol.*, 2016, pp. 158–164.

[20] C. Wang, N. Chen, C. Kallingal, W. Wilkinson, J. Liu, and A. Leslie, "Using heuristic optimization to set SRAF rules," in *Proc. SPIE 30th Opt. Microlithogr.*, 2017, pp. 32–41.

[21] A. Asthana, B. Wilkinson, and D. Power, "OPC recipe optimization using genetic algorithm," in *Proc. 29th SPIE Opt. Microlithogr.*, 2016, pp. 43–54.

[22] P. Gao, L. Zhang, and Y. Y. Wei, "SRAF generation based on SGM/CTM contour line," in *Proc. SPIE 34th Opt. Microlithogr.*, 2021, pp. 171–176.

[23] Z. Yu, P. Liao, Y. Ma, B. Yu, and M. D. F. Wong, "CTM-SRAF: Continuous transmission mask-based constraint-aware sub resolution assist feature generation," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 42, no. 10, pp. 3402–3411, Oct. 2023.

[24] P. Gao, X. Su, W. Shi, Y. Wei, and T. Ye, "Sub-resolution assist feature cleanup based on grayscale map," *IEEE Trans. Semicond. Manuf.*, vol. 32, no. 4, pp. 583–588, Nov. 2019.

[25] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 949–980, 2014.

[26] T. Glasmachers, T. Schaul, S. Yi, D. Wierstra, and J. Schmidhuber, "Exponential natural evolution strategies," in *Proc. 12th Annu. Conf. Genetic Evol. Comput.*, 2010, pp. 393–400.

[27] L. Yin et al., "Process window tripling by optimized SRAF placement rules: AP/DFM: Advanced patterning/design for manufacturability," in *Proc. 27th Annu. SEMI Adv. Semicond. Manuf. Conf. (ASMC)*, 2016, pp. 381–386.
[28] A. Berny, "Statistical machine learning and combinatorial optimization," in *Theoretical Aspects of Evolutionary Computing*. Berlin, Germany: Springer, 2001, pp. 287–306.
[29] Z. Yu, G. Chen, Y. Ma, and B. Yu, "A GPU-enabled level set method for mask optimization," in *Proc. IEEE/ACM Design, Autom. Test Europe Conf. Exhib. (DATE)*, 2021, pp. 1835–1838.
[30] G. Chen, W. Chen, Y. Ma, H. Yang, and B. Yu, "DAMO: Deep agile mask optimization for full chip scale," in *Proc. IEEE/ACM 39th Int. Conf. Comput.-Aided Design (ICCAD)*, 2020, pp. 1–9.
[31] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, Jun. 2001.
[32] C. Tabery et al., "In-design and signoff lithography physical analysis for 7/5nm (erratum)," in *Proc. SPIE 30th Opt. Microlithogr.*, 2017, pp. 26–31.
[33] X. Ma and G. R. Arce, *Computational Lithography*. Hoboken, NJ, USA: Wiley, 2011.
[34] H. Yang and H. Ren, "Enabling scalable AI computational lithography with physics-inspired models," in *Proc. 28th Asia South Pac. Design Autom. Conf.*, 2023, pp. 715–720.
[35] S. Yin et al., "FuILT: Full chip ILT system with boundary healing," in *Proc. ACM Int. Symp. Phys. Design (ISPD)*, 2024, pp. 13–20.
[36] W. Zhao et al., "AdaOPC: A self-adaptive mask optimization framework for real design patterns," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2022, pp. 1–9.
[37] S. Zheng, H. Yang, B. Zhu, B. Yu, and M. Wong, "LithoBench: Benchmarking AI computational lithography for semiconductor manufacturing," in *Proc. 37th Conf. Neural Inf. Process. Syst.*, 2023, pp. 1–12.
[38] T. Ajayi et al., "Toward an open-source digital flow: First learnings from the OpenROAD project," in *Proc. ACM/IEEE Design Autom. Conf. (DAC)*, 2019, pp. 1–4.
[39] "KLayout—Your mask layout friend." Feb. 2022. [Online]. Available: https://www.klayout.de/
[40] S. Banerjee, Z. Li, and S. R. Nassif, "ICCAD-2013 CAD contest in mask optimization and benchmark suite," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2013, pp. 271–274.

Ziyang Yu received the B.S. degree from the Department of Physics, University of Science and Technology of China, Hefei, China, in 2018, and the M.Phil. degree from the Department of Physics, University of Hong Kong, Hong Kong, in 2020. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong.

His current research interests include design space exploration in electronic design automation and machine learning on chips.

Su Zheng received the B.Eng. and M.S. degrees from Fudan University, Shanghai, China, in 2019 and 2022, respectively. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, under the supervision of Prof. Yu and Prof. Wong.

His research interest is to solve critical problems in electronic design automation with advanced artificial intelligence methods.

Wenqian Zhao received the B.Sc. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong, in 2019, where he is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering.

His research interests include machine learning for VLSI design automation and hardware-aware deep-learning acceleration.

Shuo Yin received the B.Eng. degree in computer science and engineering from Beihang University, Beijing, China, in 2022. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong.

His research interests include programming language, compiler design for hardware, and large scale GPU parallelization for EDA.

Xiaoxiao Liang (Graduate Student Member, IEEE) received the B.E. degree from the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, China, in 2020, and the M.S. degree from the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China, in 2021, where she is currently pursuing the Ph.D. degree with Microelectronics Thrust.

Her current research interests include computer-aided VLSI design and design for manufacturability.

Guojin Chen received the B.Eng. degree in software engineering from Huazhong University of Science and Technology, Wuhan, China, in 2019. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong.

His current research interests include machine learning in VLSI design for manufacturability and physics-informed networks for solving EDA area problems.

Yuzhe Ma (Member, IEEE) received the B.E. degree from the Department of Microelectronics, Sun Yat-sen University, Guangzhou, China, in 2016, and the Ph.D. degree from the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, in 2020.

He is currently an Assistant Professor with Microelectronics Thrust, Hong Kong University of Science and Technology (Guangzhou), Guangzhou. His research interests include agile VLSI design methodologies, machine learning-assisted VLSI design, and hardware-friendly machine learning.

Dr. Ma received the Best Paper Awards from ICCAD 2021, ASPDAC 2021, and ICTAI 2019, and the Best Paper Award Nomination from ASPDAC 2019.

Bei Yu (Senior Member, IEEE) received the Ph.D. degree from the University of Texas at Austin, Austin, TX, USA, in 2014.

He is currently an Associate Professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong.

Dr. Yu received the IEEE CEDA Ernest S. Kuh Early Career Award in 2021, the DAC Under-40 Innovator Award in 2024, the Hong Kong RGC Research Fellowship Scheme Award in 2024, six ICCAD/ISPD Contest Awards, and the 11 Best Paper Awards from ICCAD 2024, 2021, and 2013,IEEE TSM 2022, DATE 2022, ASPDAC 2021 and 2012, ICTAI 2019, Integration the VLSI Journal in 2018, ISPD 2017, and the SPIE Advanced Lithography Conference 2016. He has served as the TPC Chair of the ACM/IEEE Workshop on Machine Learning for CAD and on many journal editorial boards and conference committees.

Martin D. F. Wong (Life Fellow, IEEE) received the B.Sc. degree in mathematics from the University of Toronto, Toronto, ON, USA, in 1979, and the M.S. degree in mathematics and the Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA, in 1981 and 1987, respectively.

He was the Bruton Centennial Professor of Computer Science with the University of Texas at Austin, Austin, TX, USA, and an Edward C. Jordan Professor of Electronic Communication Engineering with UIUC, where he was the Executive Associate Dean of the College of Engineering from August 2012 to December 2018. From January 2019 to August 2023, he was the Dean of Engineering and the Choh-Ming Li Professor of Computer Science and Engineering with the Chinese University of Hong Kong, Hong Kong. He has published around 500 papers and graduated over 50 Ph.D. students in electronic design automation (EDA). Since August 2023, he has been with Hong Kong Baptist University as the Provost and the Chair Professor of Computer Science. His main research interest includes EDA.

Prof. Wong is a Fellow of ACM.