Litho-aware redundant local-loop insertion framework with convolutional neural network

Qu, Tong, Lin, Yibo, Gai, Tianyang, Su, Xiaojing, Wang, Shuhan, et al.

Tong Qu, Yibo Lin, Tianyang Gai, Xiaojing Su, Shuhan Wang, Bojie Ma, Yajuan Su, Yayi Wei, "Litho-aware redundant local-loop insertion framework with convolutional neural network," Proc. SPIE 11855, Photomask Technology 2021, 118550L (12 October 2021); doi: 10.1117/12.2601685

SPIE.

Event: SPIE Photomask Technology + EUV Lithography, 2021, Online Only

Invited Paper

Litho-Aware Redundant Local-Loop Insertion Framework With Convolutional Neural Network

Tong Qu[a,b], Yibo Lin[c], Tianyang Gai[a,b], Xiaojing Su[a,b], Shuhan Wang[a,b], Bojie Ma[a,b], Yajuan Su[a,b,d], and Yayi Wei[a,b,d]

[a]Institute of Microelectronics of Chinese Academy of Sciences, Beijing 100029, China
[b]University of Chinese Academy of Sciences, Beijing 100049, China
[c]CS Departmen, Peking University, Beijing, 100080, China
[d]Guangdong Greater Bay Area Applied Research Institute of Integrated Circuit and Systems, Guangzhou Guangdong 510535, China

ABSTRACT

With the VLSI technology shrinking to 7nm and beyond, the Redundant Local Loop (RLL), also known as via pillar, becomes a promising candidate of redundant via insertion due to its compatibility with the unidirectional layout style. Existing RLL insertion approaches only leverage rule-based heuristics for manufacturing constraints, which can no longer obtain a large enough Process Window (PW) in advanced technology nodes. It is imperative to develop new techniques to optimize lithography process window while inserting RLL to achieve a good yield. In this paper, we propose a machine learning-based litho-aware RLL insertion framework. Conventional lithography simulation requires tremendous computational resources to evaluate the lithography quality accurately, which is not feasible for process window exploration. We formulate the lithography simulation as a regression task and develop a customized Conventional Neural Network (CNN) architecture to predict the Depth of Focus (DOF), a standard metric for evaluating process window. We propose a complete flow for litho-aware RLL insertion based on the CNN model for process window evaluation. The commercial lithography simulator evaluates the effectiveness of the proposed framework. Experimental results demonstrate that our lithography model can predict the DOF with high accuracy and generalize well on unseen patterns while achieving orders of magnitude speedup compared to conventional lithography simulation. Our litho-aware RLL insertion framework can effectively improve the lithography process window with comparable runtime and insertion rate compared to the state-of-the-art method.

Keywords: redundant local loop, conventional neural network, lithography, DOF

1. INTRODUCTION

With the continuous scaling of semiconductor technology nodes, redundant via insertion becomes a pivotal technology to improve yield. In advanced technology nodes with the unidirectional routing style, conventional methods of inserting redundant vias have become obsolete because they introduced metal shapes in the non-preferred direction. Fig. 1(b) shows that traditional redundant via insertion introduces Metal-3 (M3) wire bending in the non-preferred direction. To overcome this issue, Redundant Local Loop (RLL), also known as via-pillar, is proposed to ensure the consistent direction of each metal wire with the design rules while introducing redundant vias (Fig. 1(c)). Recent works[1,2] are proposed to optimize the delay and performance of chips in RLL insertion. Xu et al.[3] propose a rule-based algorithm for RLL insertion considering advanced manufacturing constraints.

Nevertheless, such approaches can no longer obtain a large enough process window in advanced technology nodes. It is imperative to develop new techniques to optimize lithography while inserting RLL to achieve a good yield. In this paper, we propose a machine learning-based litho-aware RLL insertion framework. Conventional

The first two authors contributed equally to this work.
Yajuan Su, Yayi Wei are the corresponding authors ({suyajuan, weiyayi}@ime.ac.cn).

Photomask Technology 2021, edited by Stephen P. Renwick, Proc. of SPIE Vol. 11855, 118550L · © 2021 SPIE · CCC code: 0277-786X/21/$21 · doi: 10.1117/12.2601685

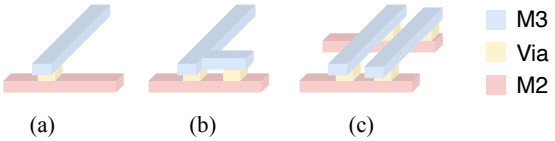

Figure 1. (a) single via, (b) traditional redundant via insertion with wire bending, (c) redundant local loop that compatible with one-dimensional routing.

lithography simulation requires tremendous computational resources to accurately evaluate the lithography quality, which is not feasible for process window exploration. Considering the fact that machine learning approaches have demonstrated superior computational efficiency to traditional simulation methods, we formulate the lithography process window simulation as a regression task and develop a customized conventional neural network (CNN) architecture to predict the Depth of Focus (DOF), a standard metric for evaluating lithography process window. This proposed framework can trade-off between accuracy and runtime. The major contributions of this paper are highlighted as follows.

- The lithography process window prediction problem is formulated as a regression task without lithography simulation.
- The CNN network is developed to achieve both high accuracy and efficiency.
- Experimental results demonstrate that our framework can increase the average lithography process window by 1.8 % for benchmarks at 10 nm technology node with comparable insertion rate to sate-of-work.[3]

The rest of this paper is organized as follows. Section 2 reviews the basic concepts and gives the problem formulation. Section 3 provides a detailed explanation of the proposed framework. Section 4 reports the experimental results. Finally, Section 5 concludes the paper.

2. PRELIMINARIES

In modern VLSI redundant via insertion, the optimization usually includes multiple objectives, such as wirelength and the number of vias. A larger number of vias and wirelength leads to a considerable timing impact of a local loop structure[4] and difficulty for post stages. In practice, the solution that neglected the above metrics may result in congestion and failure. Hence, the RLLs with less redundant vias and wirelength are preferred. We adopt the cost metric in this work, considering both wirelength and the number of vias.

Definition 1 (Cost). Cost $c \in \mathbb{R}$ evaluates the cost of an RLL structure with N metal layers and $N-1$ via layers, which is defined as follows:

$$c = \sum_{i=0}^{N} \alpha_i m_i + \sum_{i=0}^{N-1} \beta_i v_i \,, \tag{1}$$

where α, β are user-defined parameters, m_i denotes the redundant wirelength on the ith metal layer, v_i denotes the number of redundant vias on the ith via layer.

In advanced technology nodes, the cost is not enough to evaluate an RLL structure. Different RLLs with similar costs may lead to distinctive yield impacts. So, we select the *depth of focus* (DOF) metric to evaluate the lithography of a pattern.

Definition 2 (DOF). DOF $\in \mathbb{R}$ evaluates the performance of optical lithography. It can be defined as the range of focus that keeps the resist profile of a given feature within all specifications over a specified exposure range.

In practical semiconductor lithograph, DOF generally depends on resist, process parameters, and imaged patterns. Therefore, DOF is generally obtained by lithography simulation. In this work, we introduce a lithography machine learning model to speed up the simulation flow, evaluated by Mean Absolute Percentage Error (MAPE).

Definition 3 (MAPE). MAPE $\in \mathbb{R}$ evaluates the prediction accuracy of the proposed lithography model:

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{y_i} \right| , \tag{2}$$

where y_i is the actual DOF value obtained by lithography simulation, and $\hat{y}_i$ is the predicted DOF value.

With all the metrics defined, the redundant local loop insertion in the unidirectional layout is defined as follows:

Problem 1 (Lithography Model). Given a dataset containing the labelled data, pairs of layout patterns, and corresponding DOFs obtained by lithography simulation, train a model that can accurately predict a given layout pattern's DOF (i.e., minimize MAPE).

Problem 2 (Redundant Local Loop Insertion). Given the unidirectional routing design and design rules, produce a legal RLL insertion solution with optimized insertion rate, total cost, and lithography quality.

3. LITHO-AWARE RLL INSERTION FRAMEWORK

3.1 Data Preparation

For training the proposed lithography model, a labelled dataset is needed. The dataset includes 900 randomly selected $1.04 \times 1.04\,\text{mm}^2$ clips of each metal layer, and Mask Optimization (MO) has been applied to those clips to obtain the corresponding DOFs. The original layout data format (GDS II) is composed of succeeding vertex coordinate lists. Therefore, we encode these vertex coordinates into pixels. In our work, the routing solutions are based on a routing grid model, and the $1.04 \times 1.04\,\text{mm}^2$ clips can be pixeled into binary images of size 52×52 pixels without loss. For a better representation under the optical proximity effect, $5.04 \times 5.04\,\text{mm}^2$ clips centred on selected clips are pixeled into binary images of size 252×252 pixels. The dataset is divided into two parts: 50 % are preprocessed for CNN model training, while 50 % are used for validation. Rotation and flipping are applied to the training dataset to obtain various layout patterns further.

3.2 Convolutional Neural Network Architecture

Convolutional Neural Networks (CNN) have been proved capable of image classification and recognition.[5] Convolutional layer, pooling layer, and Fully Connected (FC) layer are three main components of CNN architecture. The convolutional layer's parameters consist of a set of learnable filters (or kernels), with a small receptive field and apply a convolution operation to the input, passing the result to the next layer. As a result, the network learns filters that activate when it detects some specific features. Pooling layers extract the statistical summary of the previous layer's local regions reducing the feature map dimension. Fully connected layers are used to flatten the feature maps extracted from multiple convolution and pool operations into a one-dimensional vector to predict the final results. The CNN architecture for the DOF prediction problem is summarized in Table 1, consisting of five convolution blocks and three FC layers. The first convolutional layer filters the input vectors of size 252×252 with a kernel of size 5×5. The remaining convolutional layers with kernels size of 3×3 to obtain a more profound representation. Max-pooling with filter size 2×2 and stride 2 is applied after each convolution block. Three FC layers are applied to flatten high-dimensional feature vectors to the final result.

3.3 ILP Formulation

Problem 2 can be formulated as an assignment problem. In this work, we extend the ILP formulation developed in Xu et al.,[3] and add a DOF item in the objective function to improve the lithography process window. Our

Table 1. The CNN architecture.

Layper	Kernel	Stride	Output Size	Layper	kernel	Stride	Output Size
Conv1-1	$5 \times 5 \times 4$	2	$124 \times 124 \times 4$	Pool1	2×2	2	$62 \times 62 \times 4$
Conv2-1	$3 \times 3 \times 8$	1	$62 \times 62 \times 8$	Conv2-2	$3 \times 3 \times 8$	1	$62 \times 62 \times 8$
Conv2-3	$3 \times 3 \times 8$	1	$62 \times 62 \times 8$	Pool2	2×2	2	$31 \times 31 \times 8$
Conv3-1	$3 \times 3 \times 16$	1	$31 \times 31 \times 16$	Conv3-2	$3 \times 3 \times 16$	1	$31 \times 31 \times 16$
Conv3-3	$3 \times 3 \times 16$	1	$31 \times 31 \times 16$	Pool3	2×2	2	$15 \times 15 \times 16$
Conv4-1	$3 \times 3 \times 32$	1	$15 \times 15 \times 32$	Conv4-2	$3 \times 3 \times 32$	1	$15 \times 15 \times 32$
Conv4-3	$3 \times 3 \times 32$	1	$15 \times 15 \times 32$	Pool4	2×2	2	$7 \times 7 \times 32$
Conv5-1	$3 \times 3 \times 32$	1	$7 \times 7 \times 32$	Conv5-2	$3 \times 3 \times 32$	1	$7 \times 7 \times 32$
Conv5-3	$3 \times 3 \times 32$	1	$7 \times 7 \times 32$	Pool5	2×2	2	$3 \times 3 \times 32$
FC1	-	-	1024	FC2	-	-	512
FC3	-	-	1				

modifications are highlighted in blue.

$$\max \quad \delta \sum_{x_i} n_i x_i - \epsilon \sum_{x_i} c_i x_i + \zeta \sum_{x_i} p_i x_i \tag{3}$$

$$\text{s.t.} \quad \sum_{x_i \in X_j} x_i \leq 1 \qquad \forall X_j \in X \tag{3-c1}$$

$$\sum_{x_i \in A} x_i \leq 1 \qquad \forall A \in G \cup \text{SA} \tag{3-c2}$$

$$\sum_{llc_i \in W_k} n_{ik} \cdot x_i \leq \text{DB}_k \qquad \forall W_k \in W \tag{3-c3}$$

$$x_i \in \{0, 1\} \qquad \forall x_i \in X_j \tag{3-c4}$$

Eq. (3) consists of three terms. The first term $\sum_{x_i} n_i x_i$ is the total number of redundant vias, which improves the insertion rate; The second term $\sum_{x_i} c_i x_i$ aims to reduce the overall cost of inserted RLLs; The third term $\sum_{x_i} p_i x_i$ is used to improve the lithography process window of the target design. The custom parameters θ, ϵ, and ζ can be flexibly set to trade-off those items.

4. EXPERIMENTAL RESULTS

4.1 Experiment setup

We adopt Pytorch[6] to implement the CNN model. The experiments ran on a 64-bit Linux machine with two 20-core Intel Xeon@2.1 GHz CPUs and 64 GB RAM. The commercial lithography software Tachyon runs on a 64-bit Linux machine with four Intel Xeon@2 GHz CPUs and 220 GB RAM.

The benchmarks from Xu et al.[3] are listed in Table 3. Those benchmarks are shrunk to 10 nm technology node. This shrinkage will not affect the algorithm's behaviour since Xu et al.[3] adopts a grid-based solution strategy.

4.2 Lithography Model Validation

We use Adam[7] as the gradient descent optimizer for model training. The learning rate is set to 0.01, the batch size is set to 40, and the maximum number of iterations is 1000. The dropout rate is set to 0.5 to prevent overfitting. The Mean Squared Error (MSE) is used as the loss function. We set α_i ($i \in [0, N]$) to 1 and β_i ($i \in [0, N-1]$) to 5 in Eq. (1). δ and ϵ in Eq. (3) is set to 500 and 1 respectively. ζ is defined as $\zeta = e^{4(1-\frac{p}{120})}$, where p is the predicted lithography.

Table 2. Notations.

rll_i	ith RLL candidate
c_i	the cost of rll_i
v_i	the number of redundant vias that llc_i covers
n_i	the number of vias that llc_i covers
p_i	the lithography (i.e., DOF) of llc_i
x_i	the binary variable for llc_i
X_j	the variable set for the RLLCs covering v_i
G_k	the kth set for RLL candidates occupying the same grid
SA_k	the kth set for RLL candidates occupying conflicting SAV grids
X, G, SA	set for X_i, G_k and SA_k, respectively
W_K	the kth density window
$n_{i,k}$	the number of vias of llc_i in W_k
DB_k	via density upper bound for W_k
W	the set of density window W_k
δ, ϵ, ζ	custom parameters

Table 3. Benchmark Statistics.[3]

Metric	ecc	efc	ctl	alu	div	top
#via	4013	4619	5873	6683	12 878	48 847
#nets	1539	1322	2062	2138	3792	12 988
#RLLC per via	47.3	39.0	43.7	32.6	36.0	35.2

We trained two CNN models for M2 and M3, respectively, to further improve the prediction accuracy. This will not introduce too much runtime overhead. The maximum iteration of each model is set to 1000. Their performances on training and testing datasets are reported in Table 4. It can be seen that the prediction accuracy of the two metal layers of M2 and M3 are both about 3 %. The runtime to obtain the process window using the lithography simulation tool exceeds 30 minutes, which is related to the scale of the layout. However, it takes about 5 minutes to train a CNN model, and the runtime of predicting process window is about 3.2 ms. This means that the proposed CNN model is more than $10^5\times$ faster than the simulation tool, and the accuracy loss is still within a reasonable range. On the other hand, the precision losses can still be further reduced. Since the process windows of most benchmarks are $80 \sim 100$, the training suffers from a date imbalance issue, hindering the achievement of high accuracy. Techniques such as data augmentation, and optimized sampling strategies can address this concern. Due to this accuracy meets the requirements of our framework, we leave the exploration in the future.

4.3 Framework Validation

As mentioned in Problem 2, the goal of the RLL is to maximize the inserting rate to improve the yield and reduce the timing impact of the introduced redundant vias. With the proposed CNN model, we can predict the process window of each RLL candidate. In this way, the trade-off between the insertion rate and the lithography can be achieved. The insertion rate of the proposed framework tends to be reduced compared to Xu et al.,[3] due to that their method takes the insertion rate as the only metric to be evaluated. We added a control group (denoted as IR-R) whose insertion rate is randomly reduced to the same level as ours to control variables. The comparison of our work, IR-R, and Xu et al.[3]is reported in Table 5.

The "Ratio" is based on Xu et al.[3] as the baseline. It can be seen that we can increase the average lithography process window by 4 % with comparable runtime. Compared with IR-R, we can achieve a 2 % more

Table 4. Experimental result of the CNN models on training dataset and testing dataset.

Dateset	Model	MAPE (%)	TpS (ms) *
Training	M2	2.77	3.42
	M3	2.46	4.82
	Avg.	2.61	4.12
Testing	M2	3.3	3.26
	M3	3.08	2.94
	Avg.	3.19	3.10

* TpS: The average runtime of each sample in the datasets.

Table 5. Comparison of inserting rate (IR) and lithography process window (PW) of different RLL inserting methods.

Design	Xu et al.[3]		Xu et al.[3] (IR-R*)		Ours	
	IR (%)	PW	IR (%)	PW	IR (%)	PW
top	83.00	79.22	79.24	80.92	79.62	82.3
Ratio	1.00	1.00	0.95	1.02	0.96	1.04

* IR-R: The insertion rate is randomly reduced to close to ours.

average lithography process window with a 1 % higher insertion rate, which means that our framework can effectively trade-off between insertion rate and the lithography quality.

Table 6 gives the detailed experimental result on benchmarks. One can find that the insertion rate reduction of our framework is within the acceptable range (average 4.3 %), and the runtime is comparable. Those experimental demonstrate that our litho-aware RLL insertion framework can effectively improve the lithography process quality with comparable runtime and insertion rate. As the setting of the ϵ is flexible, we can adjust the weight of a lithography item to the requirements of real-world applications. This paradigm provides a flexible framework to meet the challenge of the DTCO methodology, which is promising at advanced nodes.

Table 6. Detailed experimental result on benchmarks.

	Xu et al.[3]				Ours			
Design	IR (%)	#RLL	#RpR *	T (s)	IR (%)	#RLL	#RpR *	T (s)
ecc	98.26	2542	2.45	3.7	96.61	2733	2.58	3.7
efc	92.35	2799	2.45	3.9	87.66	2866	2.57	3.9
ctl	95.23	3543	2.42	5.4	92.56	3746	2.55	5.4
alu	80.40	3232	2.34	5.2	75.69	3242	2.52	5.3
div	88.12	7103	2.40	11.0	83.11	7315	2.55	11.2
top	83.00	24705	2.36	37.0	79.62	25092	2.53	37.5
Avg.	89.56	7321	2.40	11.03	85.21	7499	2.55	11.17
Ratio	1.00	1.00	1.00	1.00	0.96	1.03	1.06	1.01

* #RpR: redundant via number per RLL.

5. CONCLUSION

In this paper, we present a litho-aware RLL insertion framework. The proposed framework considers the lithography requirements in the RLL candidates selection stage. It achieves a trade-off between lithography process

window and insertion rate compared with the traditional insertion algorithm. We also propose a CNN model to estimate the process window, which can be $10^5\times$ faster than the rigorous simulation. The experiments show that the proposed framework can improve the average lithography process window by 1.8 % on benchmarks in 10 nm technology node. Future work includes developing algorithms to address data mismatch to improve the DOF prediction accuracy.

ACKNOWLEDGMENTS

This work is partly supported by the National Natural Science Foundation of China (Grant Nos. 61874002, 61804174, 62034007), Youth Innovation Promotion Association CAS (No. 2021115), and National Key Research and Development Program of China (2019YFB2205005).

REFERENCES

[1] Lu, L.-C., "Physical Design Challenges and Innovations to Meet Power, Speed, and Area Scaling Trend," in [*Proceedings of the 2017 ACM on International Symposium on Physical Design*], *ISPD '17*, 63, Association for Computing Machinery, Portland, Oregon, USA (Mar. 2017).

[2] Chen, X., Liu, G., Xiong, N., Su, Y., and Chen, G., "A Survey of Swarm Intelligence Techniques in VLSI Routing Problems," *IEEE Access* **8**, 26266–26292 (2020).

[3] Xu, X., Lin, Y., Li, M., Ou, J., Cline, B., and Pan, D. Z., "Redundant Local-Loop Insertion for Unidirectional Routing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* **36**, 1113–1125 (July 2017).

[4] Huang, W., Morris, D., Lafferty, N., Liebmann, L., Vaidyanathan, K., Lai, K., Pileggi, L., and Strojwas, A. J., "Local loops for robust inter-layer routing at sub-20 nm nodes," in [*Design for Manufacturability through Design-Process Integration VI*], **8327**, 83270D, International Society for Optics and Photonics (Mar. 2012).

[5] Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., and Xu, W., "Cnn-rnn: A unified framework for multi-label image classification," in [*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*], 2285–2294 (2016).

[6] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in [*Advances in Neural Information Processing Systems*], **32**, Curran Associates, Inc. (2019).

[7] Kingma, D. P. and Ba, J., "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980* (2014).