Physical Design and Mask Optimization for Directed Self-Assembly Lithography (DSAL)

Seongbo Shim‡ and Youngsoo Shin§
‡ Samsung Electronics, Hwasung 445-330, Korea
§ Department of Electrical Engineering, KAIST, Daejeon 305-701, Korea

(Invited)

***Abstract*—In DSAL, contact holes are indirectly formed through guide patterns (GPs). Thus the integrity of GPs is very important, in particular when GP shape is large and complex. The limitation in GP shape calls for careful consideration in physical design stage. In mask optimization, synthesizing ideal GP shape and verifying whether synthesized GP is correct are important but difficult problem. Some challenges in physical design and mask optimization are reviewed in this paper with possible solutions.**

I. Introduction

It has recently been observed that the scaling of devices is approaching fundamental (i.e. atomic scale) and economic (i.e. cost per fabrication facility) limits, in large part because traditional lithography is facing substantial challenges for printing fine features while maintaining a reasonable cost. Researchers are actively searching for alternative patterning approaches for the next generation lithography. Potential solutions such as extreme ultraviolet lithography, electron beam lithography, and multiple patterning lithography have attracted much attention from the lithography community.

However, they all have practical limitations, e.g. extremely high cost or low throughput. Directed self-assembly lithography (DSAL) stands out due to its low cost and high throughput as well as good patterning resolution for sub-20nm features; it is thus considered as a most promising patterning solution for contact holes and vias in technology node of 7nm and below.

In DSAL, contacts (or vias) that are physically close are clustered as shown in Fig. 1(a), which are then patterned together by DSAL process through one guide pattern (GP) as shown in Fig. 1(b). The DSAL process consists of two steps: optical lithography to form GP on a wafer, and DSA to form contacts (or vias) within each GP. A GP, which is a key component of DSAL, poses a few challenges. As a cluster contains more contacts, the corresponding GP is more sensitive to lithography variations and is more likely to cause a pattern failure during DSA process; for example, DSA patterns from three contact GP are more irregular than those from two contact GP (see Fig. 1(b)). The limitation in contact topologies should be taken into account in layout design. In mask synthesis, challenge lies on runtime since a set of GP images for a given contact layout can only be synthesized through lengthy simulations. Since GP is created by optical lithography, GP may have errors due to lithography variations.

978-1-4673-9140-5/15/$31.00

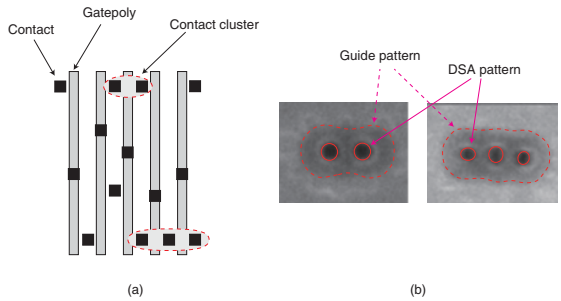

Fig. 1. (a) Contact clusters and (b) SEM images of corresponding GP and DSA pattern on a wafer.

Therefore, once GP images are obtained, they should be verified to see whether target contacts are correctly formed, which is also a difficult problem.

This paper presents the basics of DSAL technology together with current issues and a few state-of-the-art in design optimization and mask synthesis as follows:

- **Overview of DSAL technology:** Patterning mechanism and flow of DSAL is briefly introduced and compared with those of standard optical lithography in Section II, which will also cover recent development issues that affect circuit design.
- **DSAL-Aware Physical Design Optimization:** In Section III, we will present standard cell layout optimization and post-placement optimization to avoid undesired GP with high defect probability within cell region and in between adjacent cells.
- **DSAL Mask Synthesis:** Conventional mask synthesis and verification (for optical lithography) are obsolete in DSAL. New problems in mask synthesis and verification will be introduced and addressed in Section IV. Inverse DSA problems is to synthesize ideal GPs for a given contact layout, but it takes too much time due to lengthy simulation. Verification of GP also requires a number of simulations, so alternative is required to handle GPs on full-chip layout.

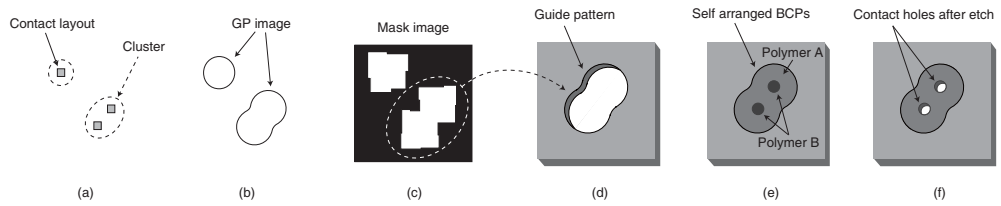

Fig. 2. Mask synthesis: (a) contact layout and clusters, (b) GPs, and (c) mask image after OPC; DSA process: (d) a GP created by optical lithography, (e) a GP filled with BCPs, and (f) contact holes after polymer B is etched away.

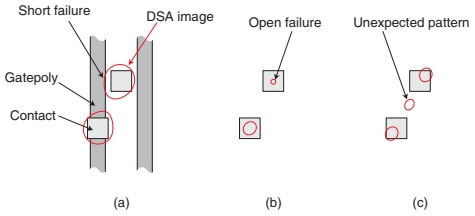

Fig. 3. Examples of DSA defect: (a) too large contact may cause electrical short, (b) actual contact is not open, and (c) unexpected contact appears.

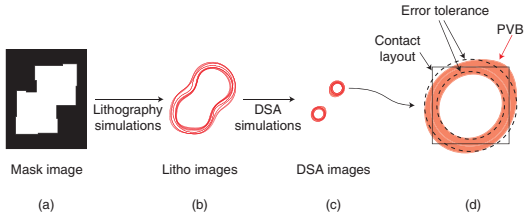

Fig. 4. (a) Mask image of a GP, (b) expected shapes of GP after optical lithography (lithography images) under various conditions, (c) expected shapes of contacts (DSA images) after DSA process, and (d) PVB of a contact with error tolerance.

II. Preliminaries

A. DSAL process

Block copolymers (BCPs) are unique soft materials that assemble by themselves through microphase separation into various periodic nanostructures such as cylinders, spheres and lamellas [1] [1]. The feature size of these nanostructures is dependent on the length of the block copolymers and therefore not limited by the same factors that limit optical lithography such as ultraviolet light wavelength. In addition, the self-assembly could be controlled by a simple thermal annealing process, which significantly reduces the cost and improves the throughput. Among all the varieties of nanostructures, the cylindrical self-assembled patterns are especially suitable for patterning contacts and vias in integrated circuits (ICs).

The standard procedure of DSAL is illustrated in Fig. 2.

- **Mask synthesis:** Neighboring contacts in a layout are clustered with respect to their physical distance [2] (Fig. 2(a)); a GP is obtained for each cluster through the process named inverse DSA or GP synthesis (Fig. 2(b)); a mask image is then synthesized through OPC with GPs as targets (Fig. 2(c)).
- **DSA process:** GPs are created by the optical lithography process such as 193nm immersion lithography, which has enough pitch resolution to pattern the sparsely located GPs (Fig. 2(d)); each GP (a large hole) is filled with BCPs, which are self arranged due to forces between the polymers and the GP (Fig. 2(e)); one of the polymers (polymer B) is then etched away leaving small contact (or via) holes (Fig. 2(f)).

B. DSA Defect due to Lithography Variations

Even though an ideal GP and its mask image are synthesized (Fig. 2(b) and (c)), GP may have errors on a wafer due to the lithography variations, which may cause a pattern failure, called DSA defect. Some defect examples are shown in Fig. 3: too large contact that may cause electrical short, target contact is not open due to too small contact image on a wafer, and unexpected contact appears during DSA process. Since contacts are patterned through two independent steps as above, small variations on some critical boundaries of GP may result in huge interference on the final DSA image.

DSA defect can be predicted through repeated lithography and DSA simulation. As illustrated in Fig. 4, a lithography simulation with a mask image of a GP yields an expected GP shape on a wafer, called litho image. To account for lithography variations, the simulation is in fact repeated while lithography parameters (e.g. scanner focus, exposure energy, and mask manufacturing error) are varied [3]; the result is a set of litho images. Each litho image is then submitted to a DSA simulator [4], which outputs an expected shape of a contact (or contacts), called DSA image. Each contact, in the end, is associated with multiple DSA images as shown in Fig. 4(c), and we can verify whether DSA defect occurs by examining the size and location of the DSA images.

C. DSA Defect Probability

The probability that DSA defect occurs, called DSA defect probability (or simply defect probability), is proposed to quantitatively measure how sensitive a GP is to the lithography variations [5]. The region bounded by the outermost- and innermost-contours of the multiple DSA images is called DSA process variation band (PVB) as illustrated in Fig. 4(d). The size of DSA image mainly determines the defect, and PVB indicates how much that size varies. So, defect probability can be defined by comparing the PVB area (red region) with some target region, called error tolerance as shown in Fig. 4(d) [5]. If

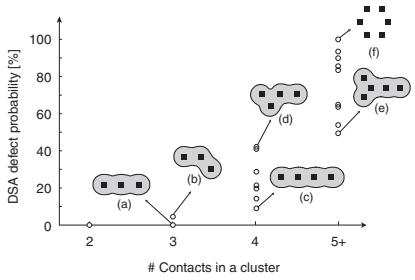

Fig. 5. DSA defect probability for various contact clusters.

whole PVB is contained within the target region, defect is not expected (i.e. zero defect probability). Otherwise, we define DSA defect probability by area ratio of PVB beyond tolerance to PVB area. The defect probability of a contact cluster (or GP) is assumed to be the maximum defect probability of its member contacts.

Experiments using 7-nm technology indicate that the defect probability increases as a cluster contains more contacts as shown in Fig. 5 [5]. There is a substantial variation of defect probability even for the clusters of the same number of contacts. Cluster (a), (c), and (e) have lower defect probability. This is because of symmetry that GP has, which makes the cluster less sensitive to lithography variations because BCPs tend to be aligned periodically for lower energy. Due to the absence of symmetry, cluster (b) and (d) have high defect probability. Cluster (f) cannot be patterned by DSAL since corresponding GP is not synthesized, so its defect probability is 100%.

III. DSAL-Aware Physical Design Optimization

As we discussed in Section II, the DSA defect probability usually increases as a cluster contains more contacts because corresponding GP contour becomes more complex. To avoid such undesired clusters, contact layout is carefully modified in physical design stages.

An alphabet approach [6] has been proposed to optimize standard cell layout for this, in a way that such large clusters are split into smaller ones by relocating some contacts as shown in Fig. 6. The approach is to use a minimal set of alphabet, which are associated with allowable contact clusters (or GPs), to pattern all contact configurations on integrated circuits. This method successfully avoids all large clusters within a cell region.

However, when two cells are located side by side (e.g. during placement), large clusters can still form across the cells as illustrated in Fig. 7. Verifying the correctness of all inter-cell clusters is very important but difficult. Applying lithography- and DSA-simulations on whole layout may be considered as a possibility, but impractical amount of time does not allow this approach.

Key ideas of our approach is to determine defect probabilities of all cell pairs in advance, when two cells of each pair are placed side by side (Section III-A); and after standard placement, we perform post-placement optimizations with the

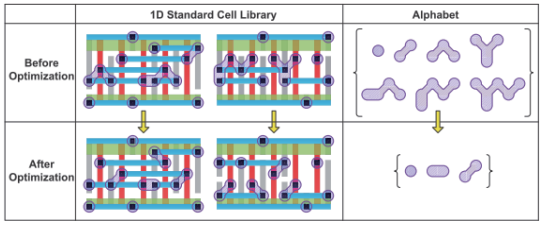

Fig. 6. Layouts and GPs for a standard cell before and after optimization [6].

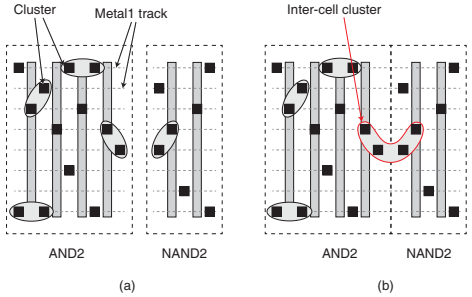

Fig. 7. Contact clusters (a) before- and (b) after-placement.

goal of minimizing the use of whitespace, which is inserted in between a cell pair whose defect probability is unacceptably high (Section III-B).

A. Fast Identification of Defect Probability

Fortunately, DSA process is localized within a GP [7], which implies that the identical clusters are associated with the same GP. And there are many identical clusters, which have identical contact topology (the number of contacts, distances between adjacent contacts, and the angle between lines connecting the contacts) because Gridded Design Rules (GDR) limits diversity of contact topology. So, the number of unique (inter-cell) clusters is quite manageable, and they are submitted to defect probability computation. The defect probability of a cell pair (see Fig. 7(b)) is now represented by the maximum defect probability of inter-cell clusters.

In the experiments using a 7-nm synthetic library, in which the minimum contacted-poly pitch (CPP) is 44nm, size of contact is 22nm, and metal 1 track pitch is 36nm, we identify about 40000 inter-cell clusters, but unique clusters are only 19. Defect probability calculation takes about 50 hours, which however is required only once. About 35% of pairs have zero defect probability: 17% pairs have no inter-cell clusters; 11% have clusters of two contacts, which have zero defect probability as shown in Fig. 5; the remaining 7% have clusters of three contacts that are linearly aligned (cluster (a) in Fig. 5), which also have 0% defect.

B. Post-Placement Optimization

Once standard placement has been performed, if there is a cell pair whose defect probability exceeds the threshold (usually 0%), a whitespace needs to be inserted in-between so that no inter-cell cluster can form. The goal of post-placement optimization is to minimally perturb the placement

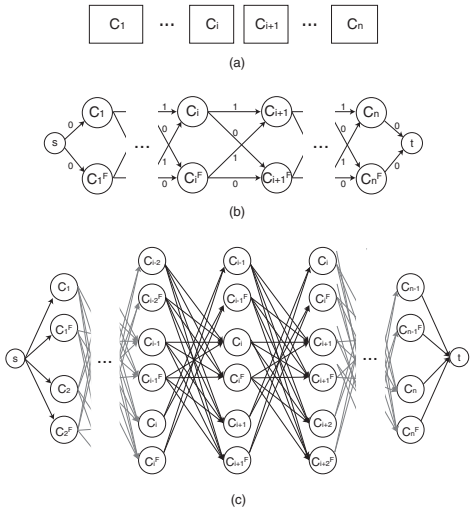

Fig. 8. (a) A placement row of n cells, (b) graph modeling to determine optimal cell flipping, and (c) graph modeling to determine optimal cell swapping and flipping.

so that the amount of whitespace we insert is minimized. Our optimization has two options: (1) flip some cells (along y-axis) and (2) swap some adjacent cells as well as flip some.

Our problem is modeled as the shortest path problem in a directed acyclic graph (DAG) with nonnegative weights. For a given cell row as shown in Fig. 8(a), each position in the row (i.e. each cell) corresponds to two vertices as shown in Fig. 8(b), where superscript F indicates that the cell is flipped. If two adjacent cells can be swapped, each position now corresponds to 6 vertices since three cells can be located at one position; note that the positions at the end of the row corresponds to 4 vertices. Two adjacent cells have 4 edges, which represents 4 different paring configurations of the two cells. If a paring configuration requires a whitespace due to high defect probability, corresponding edge is associated with weight of 1; otherwise weight is 0. The shortest path from s to t is a list of verticies, which denotes a single row placement that requires smallest number of whitespace.

We demonstrate our method using a few test circuits from Open Cores [8]. Cell flipping alone achieves 6% increase of placement density with marginal increase of wirelength (2%). Cell swapping and flipping allows 11% increase of placement density, which comes at the cost of 9% increase of wirelength. The same comparison is also made when the threshold increases to 5% and 10%. Placement density generally increases with increasing threshold because we need whitespaces in-between less number of cell pairs. Since our optimizations are applied to less number of cell pairs, the benefit clearly decreases.

IV. DSAL Mask Optimization

DSA simulation [4] predicts DSA image for a GP, which therefore can be employed to synthesize GP for a given contact cluster (i.e. inverse DSA) or to verify DSA defect. However, the extremely low efficiency makes it impossible to be adopted

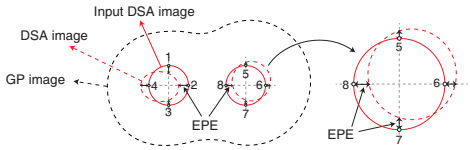

Fig. 9. GP image and its DSA image; each contact of input DSA image has 4 EPEs at certain measurement points.

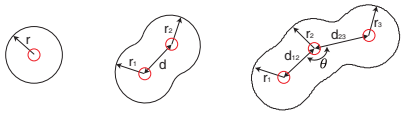

Fig. 10. Geometry parameters of GP image.

in the full-chip level implementation. For example, 10000 GPs on $100\mu\text{m} \times 100\mu\text{m}$ layout, verification is estimated to reach more than 6 months, if 27 lithography settings are considered [9].

In this section, we introduce and address two new problems in DSAL mask optimization: (1) inverse DSA problem, which is to synthesize perfect GP image for a contact cluster (Fig. 2(b)) and (2) fast verification problem of GP on a full-chip level.

A. Inverse DSA

There have been a few studies for the opposite process of inverse DSA, e.g. form various shapes of GP images on a wafer and check how contacts are formed for each GP [10], [11]. But inverse DSA has never been studied before. We propose an iterative solution, in which a GP image is progressively refined while the resulting DSA image, obtained through DSA simulation, is assessed against a target DSA image. A GP image is defined as a function of a few geometry parameters; it is refined by using sensitivity matrix, which contains the extent of how sensitive the DSA image is to each parameter change.

The algorithm receives one contact cluster in DSA image (input DSA image in Fig. 9), and returns corresponding perfect GP image. An initial GP image is constructed by placing circles (each is a concentric circle for a circle in DSA image with radius being set to the length of one BCP string), taking only the boundary arc of merged circles, and smoothing the boundary (see GP image in Fig. 9). A DSA simulation is performed on the initial GP image to obtain DSA image, which is, in most cases, different from input DSA image. The edge placement errors (EPEs) between the two DSA images are measured at certain points (four for each circle), and they are arranged as a vector $\mathbf{e} = (e_1, e_2, ..., e_m)$ (see Fig. 9).

Our goal is to refine the GP image so that resulting DSA image resembles the input DSA image as much as possible; this is achieved through iteration. To guide the refinement, GP image is defined as a function of geometry parameters, i.e.

$$\mathcal{G} = f(\mathbf{g}) = f(g_1, g_2, g_3, \cdots, g_n). \quad (1)$$

The parameters include the radius of each constituent circle, the center-to-center distance between adjacent circles, and the

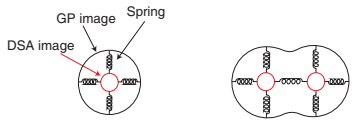

Fig. 11. Compact DSA model using springs [12], [13].

angle between adjacent center-to-center lines (see Fig. 10). We perturb each parameter to obtain a new GP image, perform a DSA simulation that returns corresponding DSA image, and assess its EPEs; a set of EPE changes allows us to introduce the sensitivity matrix:

$$\mathbf{M} = \begin{bmatrix} \frac{\partial e_1}{\partial g_1} & \frac{\partial e_1}{\partial g_2} & \cdots & \frac{\partial e_1}{\partial g_n} \\ \frac{\partial e_2}{\partial g_1} & \frac{\partial e_2}{\partial g_2} & \cdots & \frac{\partial e_2}{\partial g_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial e_m}{\partial g_1} & \frac{\partial e_m}{\partial g_2} & \cdots & \frac{\partial e_m}{\partial g_n} \end{bmatrix}. \quad (2)$$

It can be shown that the following holds:

$$\mathbf{M} \times \Delta\mathbf{g}^{\mathrm{T}} = \Delta\mathbf{e}^{\mathrm{T}}, \quad (3)$$

where $\Delta\mathbf{e}$ is EPE variations due to some extents of the parameter changes ($\Delta\mathbf{g}$). Rearranging (3) yields

$$\Delta\mathbf{g}^{\mathrm{T}} = \mathbf{M}^{-1} \times \mathbf{e}^{\mathrm{T}}, \quad (4)$$

where $\Delta\mathbf{g}$ now indicates how much we should adjust each parameter to compensate current EPEs to obtain a new GP image[1]. DSA simulation and EPE measurement then follow. If the largest EPE does not exceed a certain threshold, iteration stops; otherwise iteration continues until user defined maximum iterations are reached.

This algorithm synthesizes a GP image for one cluster, so it has to be applied to all clusters (of contacts and vias) in a layout. Fortunately, there are many identical clusters and DSA process is localized within a GP image, so we classify all contact clusters in a layout by examining contact topology and apply inverse DSA to a few unique clusters.

We implement our method in MATLAB and C++, and DSA simulator that we used is based on self consistent field theory [4]. To compare with the state-of-the-art of DSAL mask synthesis [14], we combine our inverse DSA method with conventional OPC tool, which receives resulted perfect GP as a target. Test via layout with 44μm$\times 44\mu$m size is prepared by synthesizing a circuit from Open Cores [8] using 15-nm NanGate library [15] and layout was scaled down so that the layouts follow the design rules of 7-nm technology. The layout contains 1263 clusters, only 14 (or about 1%) required inverse DSA in our approach while all clusters are individually concerned in the state-of-the-art approach. It takes 25 hours (using 42 cpu cores) [14] to obtain DSAL mask in the state-of-the-art, but our method finishes it in 1 hours (using 42 cpu cores), which is understandable consequence of applying lengthy inverse DSA to only a few unique clusters in our approach.

[1]If $\mathbf{M}$ is not square, $\mathbf{g}^{\mathrm{T}} - (\mathbf{M}^{\mathrm{T}} \times \mathbf{M})^{-1} \times \mathbf{M}^{\mathrm{T}} \times \mathbf{e}^{\mathrm{T}}$ replaces $\mathbf{g}^{\mathrm{T}}$ or singular value decomposition (SVD) is applied.

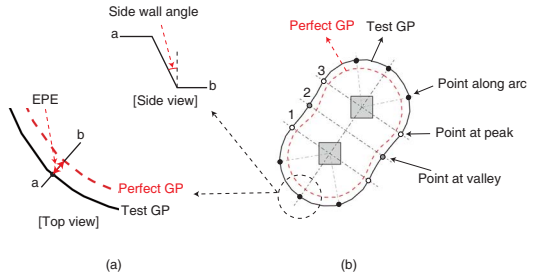

Fig. 12. Parameterization of two contact GP: (a) EPE and side wall angle and (b) measurement points.

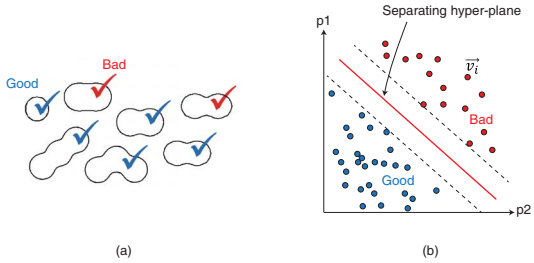

Fig. 13. Construction of decision function: (a) DSA simulations on test GPs and (b) identifying hyper-plane that separates good and bad GPs.

B. DSA Verification

Once GP and mask images are obtained, we should verify whether the mask image results in DSA defect due to lithography variations. This can be straightforwardly performed by repeated lithography and DSA simulation with various lithography settings, which is however not practical due to huge runtime.

One of the alternative approaches is to make fast model instead of the rigorous simulator. In [12], [13], force between contacts and GP is modeled as springs as shown in Fig. 11; the spring can be compressed and stretched but has an equilibrium position where the force of the spring is zero, which denotes the equilibrium state of BCPs after annealing. This method can predict center locations of contacts with reasonable errors, but a pattern failure is not correctly predicted because size variation of DSA image is not taken into account.

On the other hands, machine learning (ML) has demonstrated its effectiveness and efficiency for GP verification problem [9], [16], [17]. This approach is to construct a decision function of n variables, which correspond to geometry parameters of GP (e.g. edge placement error (EPE) and side wall angle at various points along GP contour as shown in Fig. 12 [9]); the function is evaluated for each GP in a given test layout and decides whether GP can faithfully produce desired contacts (good) or not (bad).

The function type depends on machine learning algorithm we use. In artificial neural network (ANN), a set of weight values in hidden layers plays a role of a decision function [16], [17], and in support vector machine (SVM), hyper-plane in n-dimensional space is used as a decision function [9]. For example, a decision function of SVM is determined as follows:

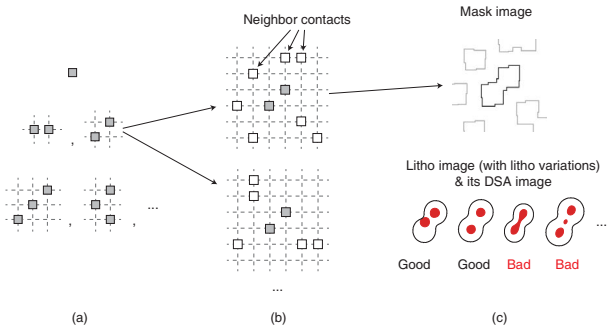

Fig. 14. Test GPs: (a) generating all possible contact topologies, (b) introducing random neighbor contacts, and (c) mask synthesis, litho- and DSA-simulation, and assessing GPs.

a set of test GPs is prepared (Fig. 13(a)); after DSA simulations, each GP is marked good or bad in n-dimensional space; the hyper-plane that separates good marks and bad marks is determined through machine learning process (Fig. 13(b)), which corresponds to our decision function.

The accuracy and coverage of a decision function is affected by how comprehensive test GPs are. Introducing random noise to a perfect GP has been tried to generate various test GPs [16], [17], but this may yield many unrealistic GPs, e.g. too curvy and fine GP contour that lithography cannot support. To prepare more comprehensive and realistic test GPs, we explore all possible contact topologies under GDR, and generate corresponding deformed GPs (as well as original GP) by performing lithography simulations with various lithography settings, as shown in Fig. 14.

How effectively geometry parameters are chosen is also important to construct stable model. A few popular parameters from image processing and computer vision, e.g. histogram of oriented gradients, have been employed in [16], [17], but a number of parameters may cause over fitting issue during constructing decision function. We therefore propose to analyze the parameters and try to remove redundant ones, which can be performed through principal component analysis (PCA). For example, we first measure more than 50 parameters (e.g. EPEs along a GP contour) so a GP is marked in 50-dimension space, but we can extract only 12 dominant and orthogonal new parameters, which are obtained by a linear combination of the initial 50 parameters, implying that a GP is now marked in 12-dimension space. Our experiment indicates that this reduction of parameter space increase fitting error (by 11%) when constructing a decision function, but increases prediction accuracy for new (unknown) GPs (by 27%).

To apply a decision function to new GPs, the new GPs should be parameterized using the same number of parameters used in function construction. This implies that a decision function should be constructed for each GP type (e.g. two contact and three contact GPs), and it is not possible to apply a decision function to new GP type, which limits coverage of this approach. A single global model that can be applied to all GP types has been proposed in [9], which is made possible by rearranging the parameters for different GP types and unifying them.

V. Summary

We have presented some new challenges posed by DSAL in physical design and mask optimization. To avoid lithography-induced DSA defect, standard cell layout and its placement have been optimized, in which DSA defect probability is proposed and its fast identification is realized by grouping identical contact topologies. In mask optimization, inverse DSA problem has been addressed for the first time, and machine learning has been applied to fast GP verification.

References

[1] M. Muramatsu, M. Iwashita, T. Kitano *et al.*, "Nanopatterning of diblock copolymer directed self-assembly lithography with wet development," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 11, no. 3, pp. 1–6, Nov. 2012.

[2] H. Yi, X. Bao, R. Tiberio, and P. Wong, "Design strategy of small topographical guiding templates for sub-15nm integrated circuits contact hole patterns using block copolymer directed self assembly," in *Proc. SPIE*, Mar. 2013, pp. 1–9.

[3] L. Liebmann, S. Mansfield, G. Han, J. Culp, J. Hibbeler, and R. Tsai, "Reducing DfM to practice: the lithography manufacturability assessor," in *Proc. SPIE*, Feb. 2006, pp. 786–798.

[4] N. Laachi, K. T. Delaney, B. Kim *et al.*, "Self-consistent field theory investigation of directed self-assembly in cylindrical confinement," *Journal of Polymer Science: Part B Polymer Physics*, vol. 53, no. 2, pp. 142–153, Jan. 2015.

[5] S. Shim, W. Chung, and Y. Shin, "Defect probability of directed self-assembly lithography: fast identification and post-placement optimization," in *Proc. Int. Conf. on Computer Aided Design*, Nov. 2015, to be presented.

[6] Y. Du, D. Guo, M. Wong, H. Yi, H. Wong, H. Zhang, and Q. Ma, "Block copolymer directed self-assembly (dsa) aware contact layer optimization for 10nm 1D standard cell library," in *Proc. Int. Conf. on Computer Aided Design*, Nov. 2013, pp. 186–193.

[7] H. Yi, A. Latypov, and P. Wong, "Computational simulation of block copolymer directed self-assembly in small topographical guiding templates," in *Proc. SPIE*, Mar. 2013, pp. 1–7.

[8] "Opencores," http://www.opencores.org/.

[9] S. Shim, W. Chung, and Y. Shin, "Synthesis of lithography test patterns through topology-oriented pattern extraction and classification," in *Proc. SPIE*, Mar. 2014, pp. 1–10.

[10] H. Yi, Y. Bao, J. Zhang *et al.*, "Flexible control of block copolymer directed self-assembly using small, topographical templates: potential lithography solution for integrated circuit contact hole patterning," *Advanced Materials*, vol. 14, no. 23, pp. 3107–3114, Jul. 2012.

[11] Y. Seino, H. Yonemitsu, H. Sato, M. Kanno, H. Kato, K. Kobayashi, A. Kawanishi, T. Azuma, S. Nagahara, T. Kitano, and T. Toshima, "Contact hole shrink process using graphoepitaxial directed self-assembly lithography," *Journal of Micro/Nanolithography, MEMS, and MOEMS*, vol. 12, no. 3, pp. 1–6, Dec. 2013.

[12] J. Torres, S. Kyohei, D. Fryer, Y. Granik, Y. Ma, P. Krasnova, G. Fenger, S. Nagahara, S. Kawakami, B. Rathsack, G. Khaira, J. Pablo, and J. Ryckaert, "Physical verification and manufacturing of contact/via layers using grapho-epitaxy DSA process," in *Proc. SPIE*, Mar. 2014, pp. 1–8.

[13] G. Fenger, A. Burbine, J. Torres, Y. Ma, Y. Granik, P. Krasnova, G. Vandenberghe, R. Gronheid, and J. Bekaert, "Calibration and application of a dsa compact model for grapho-epitaxy hole processing using contour-based metrology," in *Proc. SPIE*, Mar. 2014, pp. 1–12.

[14] W. Wang, L. Azat, Y. Zou, and T. Coskun, "A full-chip DSA correction framework," in *Proc. SPIE*, Mar. 2014, pp. 1–11.

[15] "Nangate 15nm open cell library," http://www.nangate.com/.

[16] Z. Xiao, Y. Du, H. Tian, and M. Wong, "Directed self-assembly (DSA) template pattern verification," in *Proc. Design Automation Conf.*, Jun. 2014, pp. 1–6.

[17] Z. Xiao, Y. Du, M. Wong, H. Yi, H. Wong, and H. Zhang, "Contact pitch and location prediction for directed self-assembly template verification," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2015, pp. 644–651.