Redundant Via Insertion in Directed Self-Assembly Lithography

Woohyun Chung, Seongbo Shim, and Youngsoo Shin
School of Electrical Engineering, KAIST
Daejeon 34141, Korea

***Abstract*— In directed self-assembly lithography (DSAL), vias that are located close are clustered and patterned together. A large and complex cluster, however, is not allowed in this process due to its potential danger of pattern failure. We address redundant via insertion in DSAL. The goal is to insert maximum number of redundant vias while adjacent vias do not form a large and complex cluster. The problem is formulated as maximum independent set (MIS) of a conflict graph. Experiments demonstrate 13% more redundant vias inserted compared to simple-minded approach, basic insertion with no consideration of DSAL followed by removal of redundant vias in large and complex clusters. We also introduce DSA defect probability in order to quantitatively define which clusters should be allowed during insertion process.**

I. Introduction

Redundant via insertion is a standard practice to preserve the connectivity that via provides under potential via pattern failure. It has become more important as optical lithography approaches its resolution limit to pattern a fine feature [1]. Original- and redundant-via together forms a double via. A few researches [2]–[5] have been performed to address redundant via insertion problem.

We address redundant via insertion in directed self-assembly lithography (DSAL) [6], [7], which is regarded as a patterning solution for via and contact layers in technology of 10 nm or below. DSAL process is illustrated in Fig. 1. Vias that are orthogonally or diagonally adjacent are clustered (Fig. 1(a)). A guide pattern (GP) [8], [9], which surrounds each cluster, is synthesized; its photomask image is also obtained, which is then used for patterning on a wafer through optical lithography (Fig. 1(b)). Each GP on a wafer is filled with block copolymers (BCPs), strings of two types of polymers, which are arranged due to forces between polymers and GP wall; one type of polymer is etched away leaving final vias (Fig. 1(c)). Since GP goes through two complicated steps, optical lithography and DSA, larger and complex shape containing many vias is not allowed. An example is shown in Fig. 2, where 5-via cluster causes pattern failure, short between two vias in this case.

In DSAL redundant via insertion, therefore, the goal is to insert maximum number of redundant vias while adjacent vias do not form a large and complex cluster. A necessary component in this problem is to identify which clusters are allowed and which are not. We introduce DSA defect probability for this purpose, which is the probability that cluster causes pattern failure during DSA process.

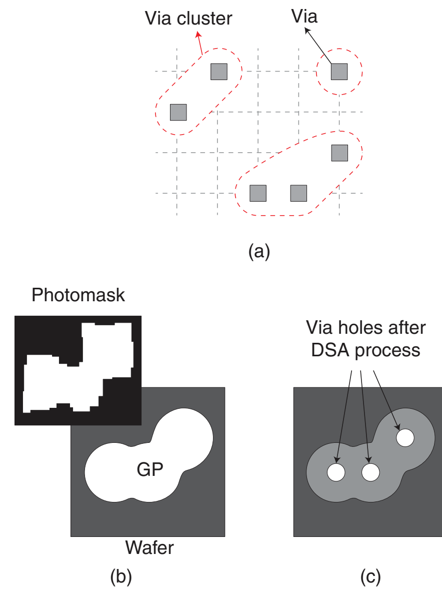

Fig. 1. DSAL process: (a) via clustering, (b) synthesizing GP mask image and optical lithography, and (c) DSA to pattern final vias.

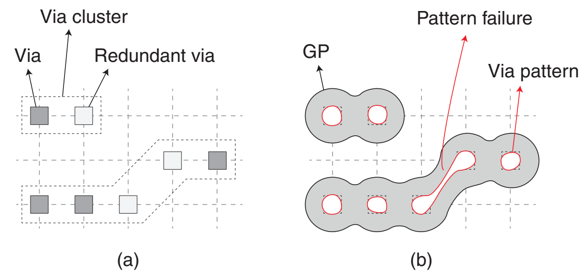

Fig. 2. (a) Via clustering and (b) pattern failure in 5-via cluster.

The remainder of this paper is organized as follows. In Section II, DSA defect probability is introduced, first for individual via and then for via cluster. DSAL redundant via insertion problem is then addressed in Section III. Experimental results are presented in Section IV, followed by summary of paper in Section V.

978-3-9815370-6-2/DATE16/©2016 EDAA

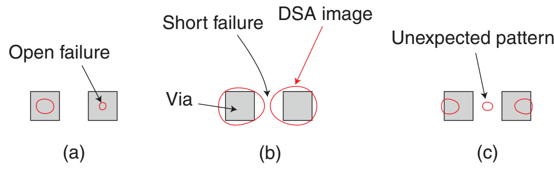

Fig. 3. DSA defects: (a) open failure, (b) short failure, and (c) unexpected pattern.

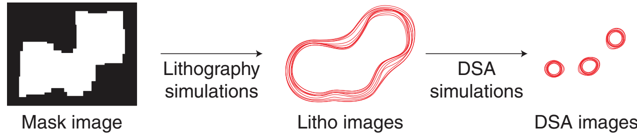

Fig. 4. Mask image of GP, expected shapes of GP after optical lithography (called litho images) under lithography variations, and expected shapes of vias after DSA process (called DSA images).

II. DSA Defect Probability

A key component in DSAL redundant via insertion is to define a list of via clusters that are manufacturable and so are allowed. For this purpose, we introduce DSA defect probability (or defect probability for short), which is the probability that a particular via causes a DSA defect after DSAL process.

A. Definition

Three types of DSA defects are illustrated in Fig. 3. Target via hardly forms due to too small via image on a wafer, called open failure, in (a). Too large and close vias may cause electrical short as shown in (b). Unexpected via forms during DSA process as shown in (c).

The probability that a particular via (or a via pair) has DSA defect is obtained through repeated lithography- and DSA-simulations. As illustrated in Fig. 4, a mask image of GP is submitted to lithography simulation, which yields expected GP shape on a wafer, called litho image. To account for lithography variations, the simulation is repeated while lithography parameters such as scanner focus, exposure energy, and mask size, are varied [10]–[12]; the results is a set of litho images. Each litho image is then handed out to DSA simulator [13], [14], which outputs an expected shape of via (or vias) after DSA process, called DSA image. The region bounded by the outermost- and innermost-contours of multiple DSA images is called process variation band (PVB), which is also illustrated in Fig. 5.

Open failure is identified by examining the area of inner bound of PVB (see A_{in} in Fig. 5). The defect probability of open failure is modeled by

$$D_{\text{open}} = \frac{M_A - A_{in}}{M_A - m_A} \times 100\%, \quad (1)$$

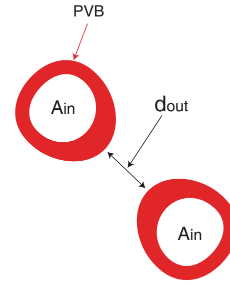

Fig. 5. PBVs, PVB distance between adjacent PBVs (d_{out}), and area of PVB inner bound (A_{in}).

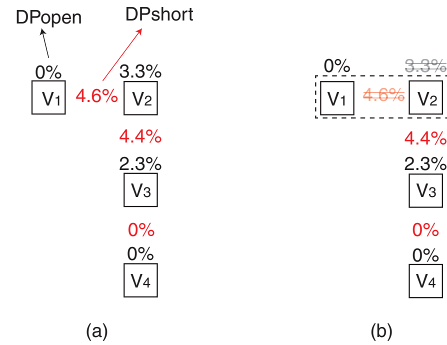

Fig. 6. (a) A 4-via cluster with DP_{open} and DP_{short} values of member vias and (b) calculation of DP_{open} and DP_{short} of via cluster when it contains a double via.

where M_A is an area of inner bound of PVB beyond which open failure never occurs (i.e. 0% defect probability), and m_A is an area of inner bound of PVB below which open failure occurs in 100%. The values of M_A and m_A are typically available from foundry fab [11], [12], [15], [16].

Defect probability of short failure is modeled by a distance between outer bounds of adjacent PVBs (see d_{out} in Fig. 5) and is given by

$$D_{\text{short}} = \frac{M_d - d_{out}}{M_d - m_d} \times 100\%, \quad (2)$$

where M_d and m_d are defined similarly to M_A and m_A, respectively.

Defect probability of unexpected pattern shown in Fig. 3(c) is binary. If unexpected pattern is observed in DSA images, the probability is 100%; otherwise, the probability is 0%.

B. Defect Probability of Via Cluster

Double via is still functional even if open failure occurs to one of its member vias, but not to both, or even if short failure occurs between the two member vias. Consider Figure 6(b), in which V_1 and V_2 comprise a double via. DP_{short} between them is 4.6%, but because they are members of a double via,

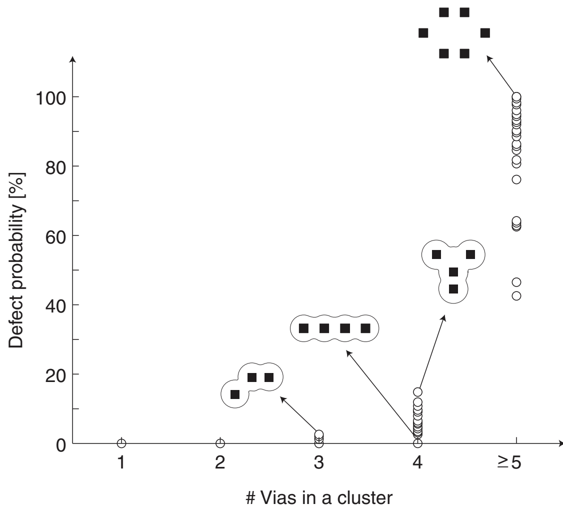

Fig. 7. Defect probability of a variety of cluster structures.

DP_{short} is ignored. Open failure of double via is when both V_1 and V_2 are associated with open, so DP_{open} of double via is 0%, smaller value of DP_{open} of V_1 and V_2.

Once defect probabilities of double vias are identified, defect probability of a via cluster can easily be computed. DP_{short} of a via cluster is the maximum DP_{short} of all its member vias and double vias (4.4% in Figure 6(b)); DP_{open} is calculated similarly (2.3%). Defect probability of a via cluster is then the maximum of its DP_{short} and DP_{open} (4.4%). Note that defect probability of a via cluster is determined as above if defect probability of unexpected pattern is 0%; on the other hands, it is determined as 100% regardless of DP_{short} and DP_{open} of member vias, if defect probability of unexpected pattern is 0%.

C. Experiments

We demonstrate the defect probability using 10-nm synthetic via layouts of a few circuits from OpenCores [17] and ITC'99 [18]. Logic synthesis, placement, and routing are performed with 15-nm NanGate library [19]; Via1 layouts are appropriately shrunk so that they can follow the gridded design rule (GDR) of 10-nm technology [20], in which via size is 25nm by 25nm with minimum via pitch of 50nm. Redundant vias are inserted with basic insertion method [2] that does not account for DSAL. Any vias that are one grid apart (either in orthogonal- or diagonal-direction) are clustered.

Defect probability calculation for each individual via cluster is impractical due to large number of clusters and lengthy simulations. Fortunately, DSA process is localized within GP region, so defect probability is same for the same GP image shape; in addition, GDR limits via positions which in turn limits the number of cluster structures. This allows us to compile a list of cluster structures and calculate defect probability of each cluster beforehand.

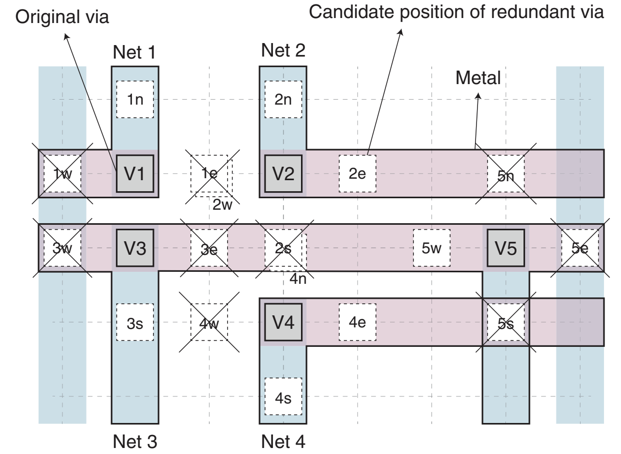

Fig. 8. Original vias (V_1, $\cdots$, V_5) and candidate positions of redundant vias.

We identify 48 different cluster structures and calculate their defect probabilities, which are shown in Fig. 7. As cluster size increases (i.e. as cluster includes more vias), defect probability also increases, e.g. cluster (a) vs cluster (b). Note that cluster (c) does not have any valid GP image that can yield the desired via pattern, so it is strictly prevented.

III. Redundant Via Insertion

The goal of redundant via insertion in DSAL is to insert maximum number of redundant vias while all via clusters are manufacturable or their defect probability is all zero. We construct a conflict graph, in which a vertex corresponds to a candidate position of redundant via and an edge represents a conflict between two candidate positions. The problem is then formulated as maximum independent set (MIS) of the conflict graph.

A. Graph Modeling

Consider Fig. 8, in which five vias (V_1, V_2, $\cdots$, V_5) are shown. For each original via, at most one redundant via can be inserted at one of north, east, west, and south side with minimum pitch distance from original via. But, there are some prohibited positions of redundant vias, which are removed from candidate positions, as follows:

- A redundant via cannot be inserted to the position occupied by an original via. For instance, V_1 cannot have candidate position on its south side due to V_3, and vice versa.
- A redundant via should not be placed on a different net due to electrical short. For example, if $2s$, which are originated from V_2, is inserted, Net 2 and Net 3 are shorted, so $s2$ is not selected as a candidate position; similarly, redundant via cannot be inserted to $1w$, $3w$, $4n$, $5n$, $5e$, and $5s$.

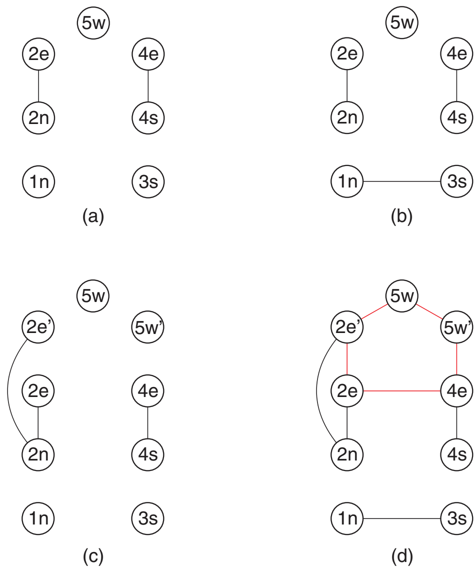

Fig. 9. Procedure of conflict graph modeling for layout in Fig. 8.

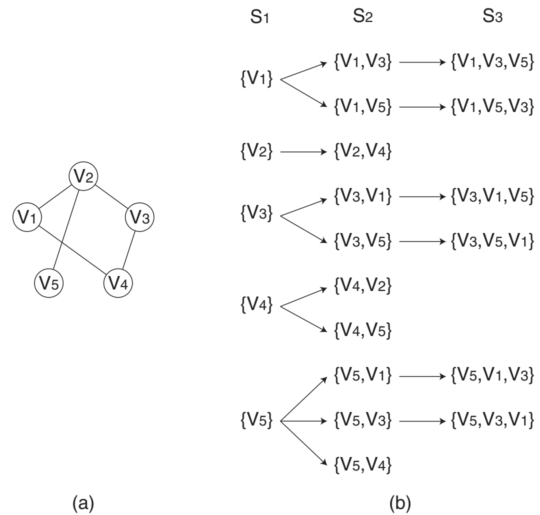

Fig. 10. (a) An example of conflict graph and (b) list of independent sets: S_1, S_2, and S_3 corresponds to the superset of the independent sets with size 1, 2, and 3 respectively.

- Unmanufacturable via cluster should not occur when a redundant via is inserted. For instance, redundant via cannot be inserted to $1e$ because it results in an unmanufacturable cluster, which is composed of V_1, V_2, V_3, and the redundant via at $1e$, is generated; similarly, redundant via cannot be inserted to $2w$, $3e$, and $4w$.

In this example, seven candidate positions of redundant vias are resulted in as shown in Fig. 8.

We then insert some edges to represent a few basic conflicts of design rule violations. Two vertices have an edge if they are originated from the same original via. For example, $2n$ and $2e$ are originated from V_2, so they have an edge in-between as shown in Fig. 9(a); similarly, $4s$ and $4e$ also have an edge. Two vertices also have an edge if their positions are physically overlap in a layout.

In addition to this, an edge is inserted between two vertices if an unmanufacturable cluster occurs when two corresponding redundant vias are inserted together. For example, vertices $1n$ and $3s$ have an edge as shown in Fig. 9(b), because concurrent insertion of redundant vias to $1n$ and $3s$ results in an unmanufacturable cluster composed of $1n$, V_1, V_2, and $3s$.

There may be a set of three redundant vias that occur unmanufacturable cluster when they are inserted at the same time, but inserting any two of them does not cause unmanufacturable cluster. For example, if redundant vias are inserted to $2e$, $4e$, and $5w$, the unmanufacturable cluster composed of V_2, $2e$, $5w$, $4e$, and V_4 occurs; but inserting any two of them does not cause an unmanufacturable cluster[1].

[1]The cluster composed of V_4, $4e$, $5w$, and $V5$ (or V_2, $2e$, $5w$, and $V5$) is in fact manufacturable because V_4 and $4e$ (and $5w$ and $V5$) comprise a double via so their defect probability of short failure is zero and the final defect probability of the cluster is zero.

Imagine that $2e$, $4e$, and $5w$ in Fig. 8 are all occupied by redundant vias, which cause unmanufacturable cluster that is composed of V_2, $2e$, $5w$, $4e$, and V_4. But, if redundant vias are inserted to only two of them, unmanufacturable cluster does not occur[2]. In this case, we modify our conflict graph partly near the three vertices. We randomly pick two out of the three vertices, duplicate them as new vertices, and add relevant edge such that $2n$ is connected to $2e'$ as well as $2e$ as shown in Fig. 9(c)). Additional edges are then inserted in cyclic fashion as shown in Fig. 9(d); note that the duplicate vertices (e.g. $2e$ and $2e'$ should be connected. At most two of the three vertices can be selected if we solve MIS on this modified graph. Note that we do not consider a set of four or more vertices, since any via cluster containing five or more vias is not manufacturable as we presented in Section II-C.

B. Algorithm

Our problem is now to find a maximum independent set (MIS) of an input conflict graph. Independent set is a set of vertices that have no edge between them, which implies that there is no conflict in the vertex set, and MIS thus corresponds to candidate positions of maximum number of redundant vias without any conflict.

For a given conflict graph, there may be various independent sets whose sizes are same. Let $s_{k(i)}$ be the i-th independent set whose size is k, and S_k be a superset of the independent sets of size k. We start from S_1, whose elements are each individual vertex in the graph; for the graph shown in Fig. 10(a), for

[2]The cluster composed of V_4, $4e$, $5w$, and $V5$ is in fact manufacturable because V_4 and $4e$ (and $5w$ and $V5$) comprise a double via so their defect probability of short failure is zero and the final defect probability of the cluster is zero; similarly, cluster composed of V_2, $2e$, $5w$, and $V5$ is also manufacturable.

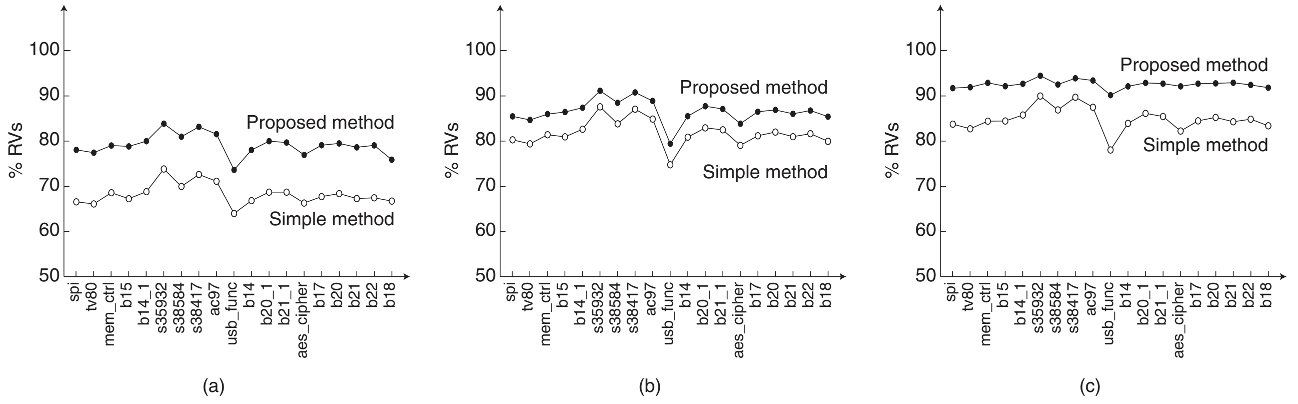

Fig. 11. Percentage of redundant via after simple- and proposed-method are applied: (a) when clusters of non-zero defect probability are not allowed, (b) when clusters with defect probability less than 5% are allowed, and (c) when clusters with defect probability less than 10% are allowed.

instance, $S_1 = \{\{v_1\}, \{v_2\}, \cdots, \{v_5\}\}$ and $s_{1(1)} = \{v_1\}$ as shown in Fig. 10(b). We then find S_2, in a way that for each $s_{1(i)}$, a set of vertices that do not connect with $s_{1(i)}$ is found, which we name $I_{1(i)}$, and each element of $I_{1(i)}$ is added to $s_{1(i)}$ to form an independent set whose size is 2. If we look $s_{1(1)} = \{v_1\}$, v_1 is not adjacent to v_3 and v_5, thus $I_{1(1)} = \{v_3, v_5\}$, so we can find two independent sets of size 2: $\{v_1, v_3\}$ and $\{v_1, v_5\}$. We repeat this to other elements in S_1 to find S_2. Now, we find S_3 from S_2 in similar way. For example, $s_{2(1)} = \{v_1, v_3\}$ as shown in Fig. 10(b). Since v_5 is the only vertex not adjacent to both v_1 and v_3, $I_{2(1)} = \{v_5\}$. From this we can find an independent set of size 3: $\{v_1, v_3, v_5\}$. We repeat this to other elements in S_2 to find S_3. We repeat this until there is no element in S_{m+1}, which means that every element in S_m is maximum independent set. In our example, since there is no element in S_4, maximum independent sets can be found in S_3, e.g. $\{1, 3, 5\}$.

IV. Experiments

The proposed redundant via insertion algorithm is tested using a few circuits from OpenCores [17] and ITC99 [18]; they are listed in Table I with the number of cells in column 2 and the number of vias in column 3. Each circuit is synthesized using 15-nm NanGate library while GDR (gridded design rule) is assumed. Metal layers up to M4 are used and placement density is set to 70%. Design rules are intentionally modified so that the final layout after routing is free from via clusters with non-zero defect probability. The layout is then appropriately shrunk for 10-nm technology, which we assume in this paper. Via size is set to 25nm by 25nm with minimum pitch of 50nm.

We apply basic redundant via insertion method [2], i.e. redundant via insertion without DSAL, and the percentage of vias that receive redundant vias (percentage of redundant vias, for short) is indicated in column 4; average is about 90%. The percentage of via clusters with non-zero defect probability is shown in the last column with average of 13.4%.

TABLE I
Percentage of redundant vias and percentage of via clusters with defects after basic redundant via insertion [2]

Circuits	# Cells	# Vias	Basic RV insertion	
			% RVs	% Clusters with defects
spi	1427	11714	89.3	14.2
tv80	4510	33310	88.3	14.0
mem_ctrl	4968	37541	88.9	13.0
b15	5044	36924	90.0	14.3
b14_1	5256	34687	90.3	13.3
s35932	5766	40877	93.4	11.9
s38584	6382	44683	91.5	13.1
s38417	6609	45838	93.2	12.6
ac97	7058	52720	91.6	12.7
usb_func	7222	63264	83.3	13.1
b14	7421	49630	89.0	14.0
b20_1	10567	69865	90.8	13.3
b21_1	10917	71561	90.1	13.5
aes_cipher	12302	92570	87.3	13.4
b17	15448	110787	89.9	14.0
b20	15608	104402	90.1	13.4
b21	15692	105602	89.4	13.6
b22	23301	155624	90.1	14.1
b18	33509	250170	89.1	13.7
Average			89.8	13.4

We then remove all the redundant vias in via clusters of non-zero defect probability. This method is named Simple in Fig. 11(a). Percentage of inserted redundant vias is now 75% on average, about 15% reduction from 4th column of Table I. Proposed method, also shown in Fig. 11(a), yields on average of 88% of redundant vias, which is very close to the number from basic insertion in Table I, even though there are now no via clusters of non-zero defect probability.

Let us assume that via clusters with defect probability smaller than 5% or 10% are allowed[3]. Via clusters with these assumptions are illustrated in Fig. 12. Proposed method is assessed with these assumptions in Fig. 11(b) and (c).

[3]Remind that our defect probability calculation is conservative, so clusters with very small defect probability may not actually cause any defects. Quantitative decision of how small is safe will be up to manufacturing details.

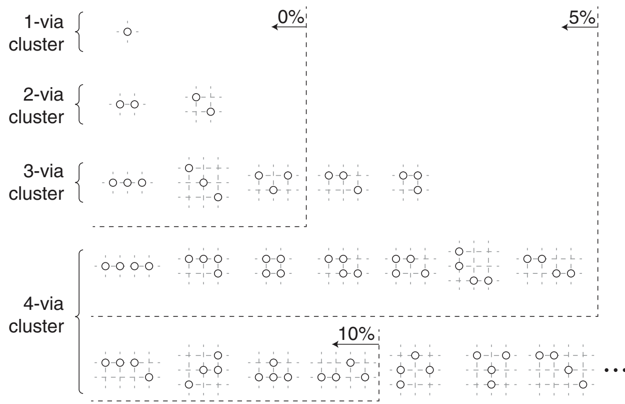

Fig. 12. Via clusters with their defect probability equal to 0% and less than 5% and 10%.

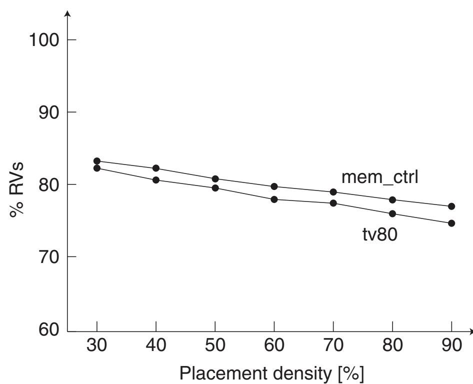

Fig. 13. Percentage of redundant vias with different placement density.

Percentage of redundant vias increases because more varieties of via clusters are accepted.

We vary placement density (from 30% to 90%) and, for each new layout, proposed redundant via insertion is applied. Resulting percentage of redundant vias is shown in Fig. 13 for two test circuits. Without surprise, as placement density increases, vias are more densely placed, which makes redundant via insertion more difficult (in particular, to avoid via clusters with non-zero defect probability) and there is a decline in the percentage of inserted redundant vias.

V. Conclusion

We have addressed DSAL redundant via insertion. The goal is to insert maximum number of redundant vias while adjacent vias and redundant vias do not form a large and complex cluster, which is hard to manufacture. We have introduced DSA defect probability in order to define a list of clusters that are allowed during insertion process.

In technology node of 7 nm or below, even DSAL is not enough and it is expected that DSAL is used together with multiple patterning (MP). Redundant via insertion in MP-DASL is another problem worth of pursue.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1A2A2A01008037).

References

[1] J. Pineda de Gyvez, "Yield modeling and beol fundamentals," in *Proceedings of the 2001 International Workshop on System-level Interconnect Prediction*, ser. SLIP '01. New York, NY, USA: ACM, 2001, pp. 135–163. [Online]. Available: http://doi.acm.org/10.1145/368640.368821/.

[2] K.-Y. Lee and T.-C. Wang, "Post-routing redundant via insertion for yield/reliability improvement," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2006, pp. 303–308.

[3] C.-W. Pan and Y.-M. Lee, "Redundant via insertion under timing constraints," in *Proc. Int. Symp. on Quality Electronic Design*, Mar. 2011, pp. 1–7.

[4] J.-T. Yan, Z.-W. Chen, B.-Y. Chiang, and Y.-M. Lee, "Timing-constrained yield-driven redundant via insertion," in *Proc. IEEE Asia Pacific Conference on Circuits and Systems.*, Nov. 2008, pp. 1688–1691.

[5] J. Pak, Y. Bei, and D. Z. Pan, "Electromigration-aware redundant via insertion," in *Proc. Asia South Pacific Design Automation Conf.*, Jan. 2015, pp. 544–549.

[6] H. Yi, Y. Bao, J. Zhang, C. Bencher, L. Chang, X. Chen, R. Tiberio, J. Conway, H. Dai, Y. Chen, S. Mitra, and H.-S. P. Wong, "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, July 2012.

[7] M. Muramatsu, M. Iwashita, T. Kitano, T. Toshima, M. Somervell, Y. Seino, D. Kawamura, M. Kanno, K. Kobayashi, and T. Azuma, "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.

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

[9] L. Azat, G. Grant, P. Moshe, S. Gerard, W. Wong, J. Xu, and Y. Zou, "Computational simulations and parametric studies for directed self-assembly process development and solution of the inverse directed self-assembly problem," *Japanese Journal of Applied Physics*, vol. 53, no. 6S, pp. 1–8, May 2014.

[10] J. Andres and T. Robles, "Integrated circuit layout design methodology with process variation bands," U.S. Patent 0 251 771A1, Nov. 10, 2005.

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

[12] J. P. Cain, "Design for manufacturability: a fabless perspective," in *Proc. SPIE Advanced Lithography*, Mar. 2013, pp. 1–9.

[13] H. D. Ceniceros and G. H. Fredrickson, "Numerical solution of polymer self-consistent field theory," *Multiscale Model. Simul.*, vol. 2, no. 3, pp. 452–474, Feb. 2004.

[14] N. Laachi, K. T. Delaney, B. Kim, S. Hur, R. Bristol, D. Shykind, C. J. Weinheimer, and G. H. Fredrickson, "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.

[15] Samsung Electronics Corp. DFM principal engineer, *personal communication*, June 2015.

[16] Samsung Electronics Corp. OPC principal engineer, *personal communication*, June 2015.

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

[18] "ITC99," http://www.cerc.utexas.edu/itc99-benchmarks/.

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

[20] 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 Advanced Lithography*, Mar. 2013, pp. 1–9.