第 27 卷 第 1 期
2006 年 1 月

半 导 体 学 报
CHINESE JOURNAL OF SEMICONDUCTORS

Vol. 27 No. 1
Jan. , 2006

A Novel Algorithm to Extract Weighted Critical Area *

Wang Junping†, Hao Yue, Zhang Huining, Zhang Xiaoju, and Ren Chunli

(*Key Laboratory of the Ministry of Education for Wide Band-Gap Semiconductor Materials and Devices, Microelectronics Institute, Xidian University, Xi'an 710071, China*)

Abstract : Inductive fault analysis is a technique for enumerating likely bridges that is limited by the weighted critical area computation. Based on the rectangle model of a real defect and mathematical morphology, an efficient algorithm is presented to compute the weighted critical area of a layout. The algorithm avoids the need to determine which rectangles belong to a net and the merging of the critical area corresponding to a net pair. Experimental results showing the algorithm's performance are presented.

Key words : bridge fault; defect rectangle model; layout analysis; mathematical morphology
EEACC : 1130B; 0240G; 2220C
CLC number : TN43 **Document code :** A **Article ID :** 0253-4177(2006)01-0024-06

1 Introduction

To remain competitive, designers must make fast and accurate fault diagnosis to obtain the set of fault sites and the ranked fault list[1]. The bridge fault site set and bridge fault sorted list can be extracted (the bridge faults accounting for the down total upwards of 50 %)[2] by calculating the weighted critical area in inductive fault analysis. To realize inductive fault analysis, a fast algorithm is needed to efficiently compute the weighted critical area.

When applied to large designs, current weighted critical area extraction algorithms do not meet the demands for the speed or accuracy[3]. At present, the standard methods consist of three main steps[4~7]. The first step is to divide the nets into many rectangles. The nets are composed of overlapping rectangular features that lie on the same layer or on adjacent layers and are connected by vias or contacts. Non_Manhattan geometry nets are approximated by a set of rectangles. The next step is to compute the weighted critical area between two rectangles belonging to different nets, merge the weighted critical areas, wipe off the redundancy of weighted critical area sections, and gain the approximate value. The last step is to rank the weighted critical areas of different nets and obtain the sorted fault lists. The algorithms above have serious disadvantages such as the calculation of weighted critical area, the approximation of the Non_Manhattan geometry, and the merging of weighted critical areas, which have a severe influence on the speed and accuracy of the algorithm. We present our weighted critical area solution and provide extraction results for some industrial layouts.

2 Inductive fault analysis

Previous works[8~12] show that the rectangle defect model more accurately estimate critical areas associated with a special layout than does the circular defect model. We give the following definition: for a real defect, the maximum possible extension of the real defect between two parallel straight lines is called the length (represented by the symbol H) of the real defect, which corresponds to a horizontal routing direction of the special layout and touches the defect. The maximum possible extension of the real defect between two parallel straight lines, which corresponds to a vertical routing direction of the special layout and touches the defect, is called the width (represented by the symbol

* Project supported by the National High Technology Research and Development Program of China (No. 2003AA1Z1630)

† Corresponding author. Email: jpwang@mail.xidian.edu.cn

Received 29 May 2005, revised manuscript received 2 October 2005

B) of the real defect, as shown in Fig. 1. Then we can define the rectangular model $D(H,B)$ of a real defect as a rectangle with length H and width B.

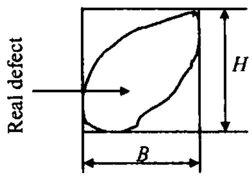

Fig. 1　Rectangular model of a real defect $D(H,B)$

In inductive fault analysis, the weighted critical area is used to model the likelihood of the occurrence of a fault. The critical area of two nets is defined as the area in which the center of the defect can lie and cause a bridge between the two nets. The weighted critical area of two nets is the product of the critical area and the probability of the occurrence of a defect. For the layout to be analyzed, we assume two kinds of defects: intra-layer and inter-layer. Accordingly, the weighted critical area of a net pair is the sum of its inter-layer and intra-layer weighted critical area constituents.

2.1　Intra-layer weighted critical area

For each layer I, let M_I be the set of probable rectangle defects $D(H,B)$ and let $P_I(H,B)$ be the probability of each defect's occurrence, where $D(H,B) \in M_I$. The intra-layer weighted critical area for net pair (N_1, N_2), denoted by RWCA(N_1, N_2, I) is defined as

$$\mathrm{RWCA}(N_1, N_2, I) = \sum_{D(H,B) \in M_I} (\mathrm{CA}(N_1, N_2, D(H,B), I) \times P_I(H,B)) \tag{1}$$

where CA$(N_1, N_2, D(H,B), I)$ is the intra-layer critical area for net pairs (N_1, N_2), which is computed by first expanding the nets (N_1, N_2) on all sides by $H/2$ in the horizontal routing direction and the nets (N_1, N_2) on all sides by $B/2$ in the vertical routing direction and then computing the overlap region (see Fig. 2).

2.2　Inter-layer weighted critical area

Every inter-layer defect is assumed to be a parallelepiped, of which the vertical projection is a rectangle with length H and width B, as shown in Fig. 3. For adjacent layers I and J, let M_{IJ} be the set of probable defects, where $D(H,B) \in M_{IJ}$, and $P_{IJ}(H,B)$ the probability of the occurrence of D

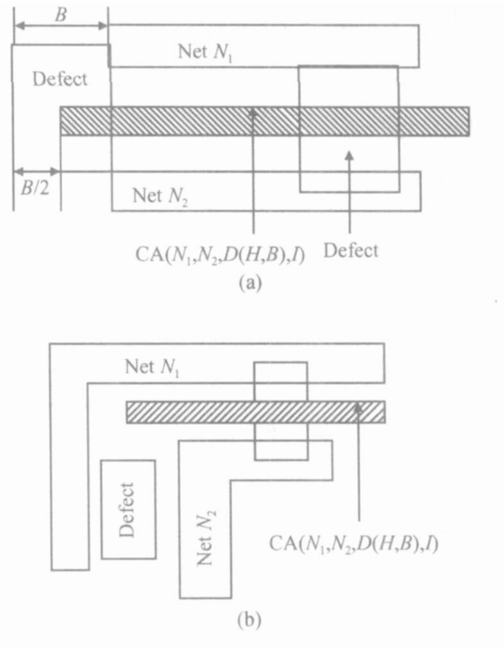

Fig. 2　(a) Critical area of the horizontal net pairs (N_1, N_2); (b) Critical area of net pairs (N_1, N_2)

(H,B). Then the inter-layer weighted critical area for net pairs (N_1, N_2), denoted by EWCA(N_1, N_2, I, J), is defined

$$\mathrm{EWCA}(N_1, N_2, I, J) = \sum_{D(H,B) \in M_{IJ}} (\overline{\mathrm{CA}}(N_1, N_2, D(H,B), I, J) \times P_{IJ}(H,B)) \tag{2}$$

where $\overline{\mathrm{CA}}(N_1, N_2, D(H,B), I, J)$ is the inter-layer critical area for net pairs (N_1, N_2) for adjacent layers I and J, which is determined by first expanding the horizontal routing direction of the nets (N_1, N_2) on all sides by $H/2$ and the vertical routing direction of the nets (N_1, N_2) on all sides by $B/2$ and then computing the overlap region (see Fig. 3).

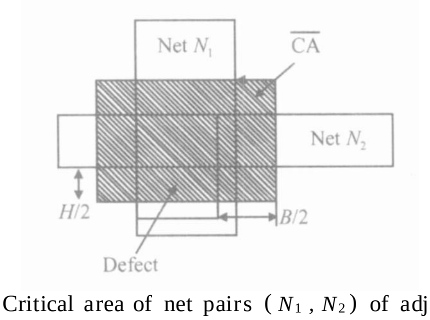

Fig. 3　Critical area of net pairs (N_1, N_2) of adjacent layers I, J

3 Algorithm and experimental results

3.1 Inductive fault analysis based on mathematical morphology

Mathematical morphology was developed in the 1970's by Matheron[13] and Serra[14]. It has several advantages over other techniques especially when applied to image processing, but offers no way of extracting weighted critical area. There are four foundational operations in morphology: dilation, erosion, opening, and closing. The dilation operation of binary morphology is defined as

$$\mathrm{DILATE}(S, E) = \mathrm{UTRAN}(E : i, j), \quad (i, j) \in D_S \tag{3}$$

where D_S represents the S domain, which is the set of known pixel positions of the image S. E is the structure element, which is converted by S. TRAN $(E : i, j)$ is defined as placing the center of E at the position (i, j) of image S and then copying the image E.

Let the rectangle defect be $D(H, B)$. By using the definitions of mathematical dilation and critical area, we derive the structure element $D(H, B)$ from defect $D(H, B)$ and the source image from net N_1. The dilation of defect $D(H, B)$ to net N_i for layer I is represented by $\mathrm{DILATE}(N_i, D(H, B), I)$. For layer I the intra-layer weighted critical area $\mathrm{MCA}[(N_1, N_2, D(H, B), I]$ for the net-pair (N_1, N_2) and the inter-layer weighted critical area $\overline{\mathrm{MCA}}(N_1, N_2, D(H, B), I, J)$ for the net pair (N_1, N_2) for adjacent layers I and J are:

$$\mathrm{MCA}[(N_1, N_2, D(H, B), I](i, j) = \begin{cases} 1, & \text{if } \mathrm{DILATE}(N_1, D(H, B), I) + \mathrm{DILATE}(N_2, D(H, B), I) = 2 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$\overline{\mathrm{MCA}}[(N_1, N_2, D(H, B), I, J](i, j) = \begin{cases} 1, & \text{if } \mathrm{DILATE}(N_1, D(H, B), I) + \mathrm{DILATE}(N_2, D(H, B), J) = 2 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Thus, Equations (1) and (2) can be modified to the following:

$$\mathrm{RWCA}(N_1, N_2, I) = \sum_{D(H,B) \in M_I} (\mathrm{MCA}(N_1, N_2, D(H, B), I) \times P_I(H, B)) \tag{6}$$

$$\mathrm{EWCA}(N_1, N_2, I, J) = \sum_{D(H,B) \in M_{IJ}} (\overline{\mathrm{MCA}}(N_1, N_2, D(H, B), I, J) \times P_{IJ}(H, B)) \tag{7}$$

The above equations show that the computation of weighted critical area for the net pair (N_1, N_2) based on mathematical morphology avoids dividing the nets into many rectangles and extracting and merging the critical area of rectangle pairs. As a result, there is no influence on the accuracy of the weighted critical area of the net pairs.

3.2 Algorithm for weighted critical area

A standard layout file can be converted into a flattened layout that consists of many layers. Each layer is a binary image. We assume that defect data for each layer I consists of the set M_I and the probability distribution $\{P_I(D(H, B)) / D(H, B) \in M_I\}$ and that the defect data for adjacent layers I and J consists of the set M_{IJ} and the probability distribution $\{P_{IJ}(D(H, B)) / D(H, B) \in M_{IJ}\}$. Our intra-layer weighted critical area algorithm (the algorithm for inter-layer weighted critical area is similar to that for intra-layer) based on mathematical morphology and the known defect data is shown ALGORITHM ca_area.

```
ALGORITHM ca_area
Input: a binary layout image S(b, w) with b rows and w
       columns, a defect D(H, B)
Output: the rank fault list A(n, n), n is the number of
       nets in layout image S.
n = bwlabel(S);
for i = 1 to n
    for j = 1 to n
        area[i][j] = 0;
    end for
end for
for i = 1 to n - 1
    m1 = imdilate(si, D(H, B))
    for j = i + 1 to n
        m2 = imdilate(sj, D(H, B))
        m1 = m1 + m2;
        A[i][j] = area(m1 = 2);
    end for
end for
A[n][n] = QKSORT(A[n][], n);
return(A[n][n])
```

The running time of the algorithm is analyzed as follows. Since there are bw pixels in image S, step 1 takes bw iterations to output the n nets. Steps 2～6 are executed nn times, which sets $A[i][j] = 0$, where $i = 1, 2, \cdots, n$, $j = 1, 2, \cdots, n$. Steps 10～12 compute the critical area $A[i][j]$ nn times for net pairs (N_i, N_j) using Eq. (4). Step 15 sorts the $A[n][n]$ by the QKSORT method, it easy to see that costs $O(n \times \lg(n))$ computations. From above analysis, it follows that the time complexity of the algorithm is $O(n^2)$. Usually it corresponds to the presence of an intersect region of one dilation net with the other $n-1$ dilation nets, a procedure which seldom happens here due to the nature of the layout. The minimum time complexity of the algorithm is $O(n)$, corresponding to the presence of an intersect region of dilation nets with another of the $n-1$ dilation nets. In general, if the dilation net intersects K dilation nets, the average time complexity is $O(Kn)$. Even in the fast algorithm at present[3], apart from the time complexity of the algorithm, most of the time is spent in the merging of the dilation rectangle regions, which has a time complexity of $O(n\lg(n))$, where n is the number of the intersect regions of the rectangles.

3.3 Experimental results

The above algorithm is implemented using an Intel Celeron AT COMPATIBLE CPU 1.2GHz (128M memory) on a 4×4 shift register metal layout. A $1/x^3$ defect size distribution and a random distribution of defect lengths and widths are assumed. By the nature of the real layout of the 4×4 shift register layout, eleven differently sized and shaped defects are selected: $D1$ $(11e, 11e)$, $D2(9e, 9e)$, $D3(2e, 9e)$, $D4(9e, 4e)$, $D5(7e, 7e)$, $D6(7e, 3e)$, $D7(3e, 7e)$, $D8(5e, 5e)$, $D9(5e, 2e)$, $D10(4e, 4e)$ and $D11(3e, 3e)$ (where e is the grid resolution, $e = 0.3\mu m$). Correspondingly, the probability distributions are such that P_{D1} is 0.1; the sum of P_{D2}, P_{D3}, and P_{D4} is 0.3; the sum of P_{D5}, P_{D6}, and P_{D7} is 0.3; the sum P_{D8} and P_{D9} is 0.2; P_{D10} is 0.1; and P_{D11} is 0.1. In Table 1, we present the last sections of the sorted fault lists and the full list is shown in Fig 5. The codes of the nets are assumed to be sorted by their y coordinate. Table 1 illustrates the maximum fault probability of the nets in the net pair (net 1, net 6). It can be seen that the maximum probability of a bridge fault corresponds to net 6. Figure 5 shows the extensions of net pairs as well as their weighted critical areas.

For this experiment, the number of nets is 74 and the number of net pairs with existing faults is 90 (K is about 1.2). Therefore, the time complexity of the algorithm is $O(1.2n)$, which is efficient and practical.

Table 1　Partial fault sorted lists of metal layer

Net N_1	Net N_2	WCA/e^2	Time/s
6	31	137.78	0.0940
6	47	137.78	0.0780
6	17	145.07	0.0930
6	33	1666.2	0.0780
2	17	1685.2	0.0940
4	33	1695.1	0.0780
1	6	2521.8	0.0940

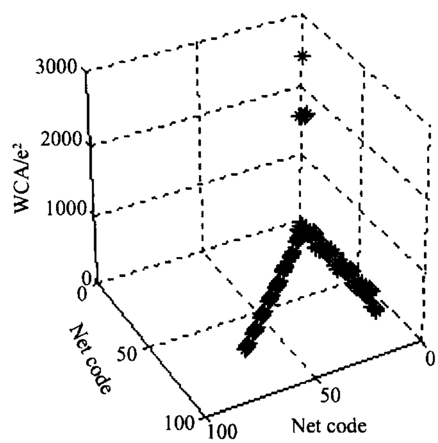

Fig. 5　WCA sorted list of net pairs

For large layouts, a divide-and-conquer strategy[15,16] can be used to reduce the CPU time and space complexity of the algorithm. The main idea of the strategy is to divide the input layout into M segments, depending on the size of the layout, find the weighted critical area of each segment, and then merge them to obtain the weighted critical area of the total layout. For example, the first column in Table 1 lists the the CPU time of a segment layout. If a large layout consists of 1000 segments, the total CPU time required to extract the weighted critical area of the layout is less than 2min.

Table 2 summarizes the performance comparison of the current algorithm with the new one. In columns two to five, the defect model, shape of layout routing, CPU time, and accuracy are given. From Table 2, it can be seen that the new algorithm may extract the weighted critical area of the layout more accurately than the current algorithm, so the proposed algorithm is highly efficient.

Table 2 Performance comparison of current algorithm with new one

Algorithm	Defect model	Shape of layout routing	CPU time	Accuracy
Our algorithm	Rectangle	Random	Practical	Error free
Current algorithm	Circular	Rectangle	Practical	Inaccurate

4 Conclusion

We report the designs and implementation of a new algorithm for extracting the weighted critical area associated with a layout. The rectangle model of real defects assures the accuracy of the computation of the critical area caused by real defects. For layout nets, the mathematical morphological model is presented which assures the accuracy and efficiency of the layout. The experimental results show the efficiency of the new algorithm, which provides a reliable foundation for fault diagnosis at high speed and greater accuracy.

References

[1] Jee A L, Ferguson F J. CARAFE: an inductive fault analysis tool for CMOS VLSI circuit. IEEE VLSI Test Symp, 1992: 92

[2] Zachariah S T, Chakravarty S. Algorithm to extract two-node bridges. IEEE Trans Very Large Scale Integration Systems, 2003, 11(4): 741

[3] Nag P K, Maly W. Hierarchical extraction of critical area for shorts in very large ICs. IEEE Int Defect and Fault Tolerance in VLSI Systems Workshop, 1995: 19

[4] Goncalves F M, Teixeira I C, Teixeira J P. Integrated approach for circuit and fault extraction for VLSI circuits. Proc Int Defect and Fault Tolerance Symp, 1996: 96

[5] Zachariah S T, Chakravarty S, Roth C D. A novel algorithm to extract two-node bridges. ACM Design and Automation Conference, 2000: 790

[6] Zachariah S T, Chakravarty S. A scalable and efficient methodology to extract two-node bridges from large industrial circuits. IEEE Test Conference, 2000: 750

[7] Walker D, Stanojevic Z. FedEx: fast bridging fault extractor. IEEE Test Conference, 2001: 696

[8] Jing Ming'e, Hao Yue. A geometry explanation of center of gravity algorithm of VLSI parametric yield. Chinese Journal of Semiconductors, 2004, 25(5): 594 (in Chinese) [荆明娥, 郝跃. VLSI 成品率重心游移算法的一个几何解释. 半导体学报, 2004, 25(5): 594]

[9] Ma Peijun, Hao Yue, Liu Hongxia. Analysis and simulation of effect of pinhole defects on integrated circuits functional yield. Chinese Journal of Semiconductors, 2001, 22(1): 102 (in Chinese) [马佩军, 郝跃, 刘红侠. 针孔缺陷对集成电路功能成品率影响分析与仿真. 半导体学报, 2001, 22(1): 102]

[10] Zhao Tianxu, Hao Yue, Ma Peijun. An effectual IC's yield estimation model. Chinese Journal of Semiconductors, 2002, 23(2): 198 (in Chinese) [赵天绪, 郝跃, 马佩军. 一种有效的 IC 成品率估算模型. 半导体学报, 2002, 23(2): 198]

[11] Wang Junping, Hao Yue. Yield modeling of IC based on distribution of defect shapes. Chinese Journal of Semiconductors, 2005, 26(5): 1054 (in Chinese) [王俊平, 郝跃. 考虑缺陷形状分布的 IC 成品率模型. 半导体学报, 2005, 26(5): 1054]

[12] Wang Junping, Hao Yue. Yield modeling of rectangle defect outline. Chinese Journal of Semiconductors, 2005, 26(8): 1514

[13] Serra J. Image analysis and mathematical morphology. New York: Academic Press, 1992

[14] Castleman K R. Digital image processing. Prentice Hall, 1996

[15] Wang X, Fu Q. A frame for general divide and conquer recurrences. Information Processing Letters, 1996, 59: 45

[16] Alsuwaiyel M H. Algorithm design techniques and analysis. Beijing: Publishing House of Electronics Industry, 2003: 161

提取带权关键面积的新算法*

王俊平† 郝 跃 张会宁 张晓菊 任春丽

(西安电子科技大学微电子学院 宽禁带半导体材料与器件教育部重点实验室, 西安 710071)

摘要: 导体故障分析是一种列举与缺陷有关的集成电路版图中可能出现桥连的技术,计算带权关键面积是限制其性能的主要因素. 文中提出了一种基于数学形态学和真实缺陷矩形模型提取带权关键面积的新算法,该算法不需要将版图上的线网拆分为矩形,也不需要合并矩形对的带权关键面积. 实验结果验证了新算法的有效性.

关键词: 桥形故障; 矩形缺陷模型; 版图分析; 数学形态学

EEACC: 1130B; 0240G; 2220C

中图分类号: TN43 **文献标识码**: A **文章编号**: 0253-4177(2006)01-0024-06

* 国家高技术研究发展计划资助项目(批准号:2003AA1Z1630)

† 通信作者. Email:jpwang@mail.xidian.edu.cn

2005-05-29 收到,2005-10-02 定稿