A layout defect-sensitivity extractor

Citation for published version (APA):
https://doi.org/10.1109/ICCAD.1989.77008

DOI:
10.1109/ICCAD.1989.77008

Document status and date:
Published: 01/01/1989

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.
Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
ABSTRACT

A new method based on a deterministic geometrical construction of critical areas is presented to determine the sensitivity of layouts to spot defects. The models for fatal faults considered are bridges and cuts related to patterns in one layer. Our approach, based on the new concept of "susceptible sites", has a complexity $O(N \log N)$, where $N$ is the number of line segments. Moreover, only two scans are necessary to extract all "susceptible sites" which then are used to compute the "critical areas" for a whole set of points in a domain of defect sizes.

2. FINDING THE CRITICAL AREAS IN LAYOUTS.

In this paper we restrict ourselves to catastrophic faults related to patterns in one layer, namely, the bridge (joining patterns unintendedly) and the cut (breaking patterns unintendedly).

It is difficult to model the exact shape of defects since in reality they are rough-edged splodges. However, we propose to model defects as square shaped objects. We stipulate that this approximation is sufficiently correct. It can be shown that it implies very simple and fast algorithms.

It has been shown [11] that the critical areas can be found geometrically. Under this approach the critical areas for bridges are found by expanding each pattern by an amount equal to half of the defect size, and then by checking if the expansions intersect. If so, then the amount of intersection corresponds to the critical area between the patterns. In the case of critical areas for cuts every pattern is shrunk by half of the defect size, and a critical area is established only when on shrinking parallel edges pass over each other. This solution has a quadratic complexity, in the case of bridges, because pairs of patterns have to be investigated, and thus it is computationally prohibitive for very large layouts. Also, the expansion of the patterns implies that a layout extraction has to be executed for each defect size. For the areas sensitive to cuts the problem is simpler since only individual patterns have to be considered.

3. MATHEMATICAL FOUNDATION.

A point set $A_i$ is a connected open set of points in a mask plane. The boundary of $A_i$ is a closed contour of cartesian straight lines called edges. The point set $A_i$ divides the layout mask in two disjoint regions, the interior and the exterior that are separated by the contour of $A_i$. The point sets $A_i$ may be the set of opaque zones of a mask.

We say that the horizontal line segments $s_1$ and $s_2$ are comparable at abscissa $x$ if there exists a vertical line that intersects them. The complex layouts are based on a statistical Monte Carlo simulation, and analytical methods [8] are restricted only to simple and regular layouts. We present a layout verifier capable of identifying the critical areas in complex layouts. Unlike the Monte Carlo approaches our method is deterministic. The implementation is based on a simple scan line algorithm. Given a defect type the algorithm performs only one layout extraction for any span of defect sizes.
relation above at x is defined as: s₁ is above s₂ at x if s₁ and s₂ are comparable at x and the intersection of s₁ with the vertical line lies above the intersection of s₂ with that line [10]. The relationships to the left of y and to the right of y are defined correspondingly by rotating the configuration by 90 degrees.

Our algorithm is based on the geometrical approach explained in section 2. However, instead of proceeding directly to establish the critical areas we determine first those regions where a defect of any size may potentially cause a fault. These regions are parts of the empty spaces between patterns, for the case of bridges, and the patterns themselves, for the case of cuts. Therefore only a single layout extraction is needed for any span of defect sizes, and thus the problem is reduced to determining the shape of subsets constituting the critical area from these regions for any given defect size.

Definition 1 Let (y(u); r(u)), r(u)) denote the coordinates of a horizontal line segment u with ordinate y having its left and right abscissae respectively. Assume three line segments α, β, and γ, such that (α) > (γ), y(α) > y(γ) and that (y(α)) > (y(γ)) (y(α)) > y(γ)). We say that α and γ are “diagonally neighboring” if in the open rectangle S\text{diag}1 with comers (d₁,d₂) there exist no points of any other point set of the same mask, where d₁=(r(α), y(α)) and d₂=(r(γ), y(γ)). We call S\text{diag}1 a left corner susceptible site for bridging faults. Horizontal susceptible sites are defined analogously.

We omit the formal definition of susceptible sites for cutting faults.

Definition 2 Assume that we have a defect of size δ, a lateral susceptible site S₁, and a corner susceptible site S₂. Referring to our previous notation assume that y(α) < y(β) < δ and that (l(α)) < (r(γ)) < δ, (y(α)) < y(γ) Then the rectangular open point set R\text{AB} with comers (r₁,r₂) given as r₁=\max((l(α)), y(α)) and r₂=\min((r(γ)), y(γ)), and the rectangular open point set R\text{BC} with comers (q₁,q₂) given as q₁=(l(α), y(α)) and q₂=(r(γ), y(γ)) are called critical regions. The area enclosed in a critical region is called the critical area.

We now give two theorems which reduce the time complexity of the computation of critical areas. The idea behind these theorems is that some critical regions for bridges are contained in others and thus need not be considered.

Theorem 1 Let S₁ and S₂ be two corner susceptible sites, and R\text{A}(S₁) and R\text{B}(S₂) their corresponding critical regions for a defect δ, respectively. If \( S_1 \subseteq S_2 \) then \( R\text{A}(S_1, R\text{B}(S_2) \). \)

Proof: Assume three line segments α, β, and γ, such that (α) > (γ), y(α) > y(γ) and that (y(α)) > (y(γ)) (y(α)) > y(γ)). We say that α and γ are “diagonally neighboring” if in the open rectangle S\text{diag}1 with comers (d₁,d₂) there exist no points of any other point set of the same mask, where d₁=(r(α), y(α)) and d₂=(r(γ), y(γ)). We call S\text{diag}1 a left corner susceptible site for bridging faults. Horizontal susceptible sites are defined analogously.

We omit the formal definition of susceptible sites for cutting faults.

Theorem 2 Let \( \alpha, \beta, \) and \( \gamma \) be four maximal horizontal line segments of the open connected point sets A, B, and C, respectively, such that the interior of A is below \( \alpha \), the interior of B is above \( \beta \) and below \( \gamma \), and the interior of C is above \( \gamma \). Suppose also that the four line segments are comparable at a closed interval \([a,b]\) of the domain of points of the abscissae. Assume that the following order is imposed \( y(\alpha) < y(\beta) < y(\gamma) \). Let \((q₁,q₂),(r₁,r₂)\), and \((s₁,s₂)\) be the corner points of the lateral susceptible sites \( S\text{AB, SBC, and SCA} \), respectively, expressed as \((q₁,q₂),(r₁,q₂),(s₁,q₂),(r₁,s₁),(s₁,s₂),(r₂,s₂),(q₂,s₂)\). It can be shown that a defect of size \( \delta \) placed at any point of \( y(\alpha) \) is effectively placed at \((y(\alpha))\). The area enclosed in a critical region is called the critical area.

Proof: Let \( \alpha, \beta, \) and \( \gamma \) be four maximal horizontal line segments of the open connected point sets A, B, and C, respectively, such that the interior of A is below \( \alpha \), the interior of B is above \( \beta \) and below \( \gamma \), and the interior of C is above \( \gamma \). Suppose also that the four line segments are comparable at a closed interval \([a,b]\) of the domain of points of the abscissae. Assume that the following order is imposed \( y(\alpha) < y(\beta) < y(\gamma) \). Let \((q₁,q₂),(r₁,r₂)\), and \((s₁,s₂)\) be the corner points of the lateral susceptible sites \( S\text{AB, SBC, and SCA} \), respectively, expressed as \((q₁,q₂),(r₁,q₂),(s₁,q₂),(r₁,s₁),(s₁,s₂),(r₂,s₂),(q₂,s₂)\). It can be shown that a defect of size \( \delta \) placed at any point of \( y(\alpha) \) is effectively placed at \((y(\alpha))\). The area enclosed in a critical region is called the critical area.

We outline now the steps involved in the computation of critical areas.

Step 0. For some mask assign unique identification numbers to the open connected segments. Decompose them into line connected segments.

Step 1. Sweep the layout horizontally and vertically, to extract, from the pre-processed mask, all the susceptible sites for bridges and cuts. Store them in two different data structures, one for bridges and one for cuts. We denote these data structures as "susceptibility structures".

Step 2. For every defect size defined in the range of sizes traverse the "susceptibility structure". The coordinates of areas sensitive to bridges, or cuts, are obtained by shrinking the abscissae (for horizontal sites), or the ordinates (for vertical sites), or both abscissae and ordinates (for corner sites) of the related susceptible sites. See Fig. 1.

Step 3. For every defect size compute the total critical area enclosed in the set of coordinates found above. The total critical area per defect size is the area of the union of the individual critical regions found in step 2.

Sections 4 and 5 give details of the algorithms.
4. EXTRACTION OF SUSCEPTIBLE SITES.

We classify all line segments as belonging to two types. The horizontal (vertical) line segments are of type BEGIN when the interior of their open connected point set is above (to the right) of the line segment. Similarly, the line segments are of type END when the interior of the open connected point set is below or to the left of the line segment. Each vertical line segment is specified by its x-coordinate and the y-values of the lower and upper endpoints. Each horizontal line segment is similarly specified by its y-coordinate and the x-values of its left and right endpoints. We store the horizontal and vertical line segments in two different data structures.

The extraction of susceptible sites is based on the principle of the scanline algorithm [10]. We perform two orthogonal layout sweeps: a bottom-up and a left-right sweep that cover all the susceptible sites parallel to the scanline. The bottom-up sweep, or VERTICAL sweep, scans the data structure with the horizontal line segments. The left-right sweep, or HORIZONTAL sweep, scans the one with the vertical segments. As the algorithms for finding susceptible sites for bridges and cuts are very similar the explanation to follow is restricted to bridges.

Suppose now that the vertical sweep is carried on. Let \( P = \{p_1, \ldots, p_n\} \) be the set of line segments in ascending y-order. \( M \) a set of line segments sorted in ascending x-order, and \( S \) a set to store the susceptible sites found. The main loop of the algorithm sweeps a scanline through the set \( P \). Every END line segment swept is stored in the set \( M \) ordered by its left coordinate. Initially \( M \) is empty. Whenever a BEGIN line segment is encountered we check for intersections with the (horizontal line) segments of \( M \). For every intersection a new susceptible site is made and stored in the set \( S \) if the identification numbers of the BEGIN and END line segments are different. Based on theorem 1, we also look, in \( M \), for the nearest line segments to the left (predecessor) and to the right (successor) of the endpoints of the BEGIN line segment. If these lines exist the corresponding left and right corner susceptible sites are created too.

Lateral susceptible sites are labeled as VERTICAL, or HORIZONTAL, depending on the sweep in which they were found. Corner susceptible sites are always labeled as CORNER. The justification of labeling the susceptible sites is because in step 2 we need to know what coordinates to shrink, either the abscissae or the ordinates before we can deduce if a critical region is established.

The set \( M \) is updated in such a form that only the intersected sections of the intersected line segments are deleted. Theorem 2 guarantees that these sections are no longer necessary to create critical regions with other line segments ahead of the scanline position.

The extraction of susceptible sites for cuts is essentially the same except that in the algorithm the BEGIN line segments are the ones that are stored in the set \( M \), and only the BEGIN and END line segments with the same identification number are processed. In some situations the algorithm will apparently find the same susceptible site for both vertical and horizontal sweeps. This is because we assumed that the "current cross edges" of the open connected point sets can be either vertical or horizontal. Since we are dealing with independent masks there is no way to know the signal flow (left to right, top to bottom?) in order to determine if the pattern was broken in two nonequipotential regions. See Fig. 2 for an illustration of this case.

5. CREATION OF CRITICAL REGIONS.

The critical regions related to bridges and cuts are found by traversing their corresponding susceptibility data structures.

6. COMPUTATION OF CRITICAL AREAS.

The algorithm to compute the areas is an adaptation of the algorithm presented in [10]. The implementation runs a scanline from left to right in the critical mask. We denote the left side of a critical region as BEGIN and the right side as END. Let \( C = (c_1, \ldots, c_d) \) be the set of vertical BEGIN and END line segments of the critical regions in ascending x-order, and also let \( (key \text{bottom} \text{top})_i \) represent an edge \( c_i \) situated at abscissa \( key \) having \( \text{bottom} \text{top} \) \( \text{bottom} \) as the ordinates of its endpoints. At any instance of \( c_i \) the total area is updated by evaluating the area of the union of the rectangle's sections that lie in the plane strip \( key(c_i) \) and \( key(c_{i+1}) \). A segment tree \( M \) is used to determine the length of the intercept of the scanline in the strip with the union of the rectangle's sections.

7. COMPLEXITY ANALYSIS.

Table 1 presents the time complexity analysis of each of the steps of our method. The nomenclature used is \( N \) for the number of line segments of the preprocessed layout, and \( k \) for the number of susceptible sites. The number of susceptible sites is dependent on the style of the IC artwork, in the worst case \( k = N \). Let \( #d \) be the number of defect sizes, then the total time complexity to find the
critical areas for a range of defect sizes, assuming worst case, is \(O(N^2+6d(1+\log N))\). The quadratic complexity of step 1 is because our current implementation uses a linear list data structure. However, if the data structure is changed to a balanced tree the complexity of this step would be \(O(k+N\log N)\) [9].

**TABLE 1.** Time complexity analysis of the proposed method

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Susceptible sites</td>
<td>(N^2)</td>
</tr>
<tr>
<td>2)</td>
<td>Critical regions</td>
<td>(k)</td>
</tr>
<tr>
<td>3)</td>
<td>Critical areas</td>
<td>(k\log(k))</td>
</tr>
</tbody>
</table>

No results of similar tools [14] have been presented in the literature as to make comparisons. We will assume a hypothetical case in which the critical areas are found according to: 1) growing the patterns, 2) finding overlaps of the grown patterns, and 3) computing the areas. Table 2 shows the ideal time complexity analysis. In this case \(u(\delta)\) represents the number of overlaps, as a function of the defect size \(\delta\), that were created in the "pattern expansion" process of step 1. Notice that as the defect size grows \(u(\delta)-N^2\) since most patterns will be intersecting each other. For this algorithm, the total time complexity to find the critical areas for a range of defect sizes, assuming also a worst case, is \(O(\max(N+1+\log N)+N^2(1+\log(N^2)))\).

**TABLE 2.** Time complexity analysis of the hypothetical case

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Pattern expansion</td>
<td>(N)</td>
</tr>
<tr>
<td>2)</td>
<td>Overlap search</td>
<td>(N\log(N)+u(\delta))</td>
</tr>
<tr>
<td>3)</td>
<td>Critical areas</td>
<td>(u(\delta)\log(u(\delta)))</td>
</tr>
</tbody>
</table>

Comparing both complexity analyses we can see that our method is \(N\log N\) and depends only on the geometry of the layout. The hypothetical case is dependent on both the defect size and the layout geometry. For very large defect sizes, our method is superior. The \(O(N\log N)\) complexity and defect size independence of our method provides an approach for interactive applications.

8. EXPERIMENTAL RESULTS.

The previous algorithms were implemented in a system aimed at layout yield prediction [12]. Fig. 3 shows the critical regions for bridges of a layout mask. The minimum resolution features are \(6\mu\). The defect size is \(20\mu\).

Fig. 4 shows the corresponding critical areas for bridges and cuts in the form of "sensitivity curves" (defined as the ratio of the total critical area to the layout area).

![Figure 3. Critical regions are highlighted in black. Critical regions for bridges for a defect size of \(20\mu\).](image)

![Figure 4. Mask Sensitivity.](image)

9. REFERENCES


