

TROY: Track Router with Yield-driven Wire Planning

Minsik Cho, Hua Xiang*, Ruchir Puri*, David Z. Pan
ECE Dept. Univ. of Texas at Austin, Austin, TX 78712

*IBM T. J. Watson Research Center, Yorktown Heights, NY 10598

thyeros@cerc.utexas.edu, huaxiang@us.ibm.com, ruchir@us.ibm.com, dpan@ece.utexas.edu

ABSTRACT

In this paper, we propose TROY, the first track router with yield-driven wire planning to optimize yield loss due to random defects. As the probability of failure (*POF*) computed from critical area analysis and defect size distribution strongly depends on wire ordering, sizing, and spacing, track routing plays a key role in effective wire planning for yield optimization. TROY formulates wire ordering into a preference-aware minimum Hamiltonian path problem. For simultaneous wire sizing and spacing optimization, TROY solves it optimally by formulating the problems into a second order conic programming (SOCP). Experimental results show that TROY can reduce the random-defect yield loss by 18% on average without any overhead in wirelength, compared with the widely used greedy approach.

Categories and Subject Descriptors

B.7.2 [Hardware, Integrated Circuit]: Design Aids

General Terms

Algorithms, Design, Performance

Keywords

VLSI, Track Routing, Yield, Manufacturability

1. INTRODUCTION

Smaller feature size makes nanometer VLSI designs vulnerable to ever-growing yield loss due to both random and systematic causes [17]. While it is believed that the yield loss due to systematic sources is greater than that due to random defects during the technology and process ramp-up stage, the systematic yield loss can be largely eliminated when the process becomes mature and systematic variations are extracted/compensated. On the other hand, the random defects which are inherent due to manufacturing limitations will still be there even for mature fabrication process [17]. Thus, its relative importance will indeed be much bigger for mature process with systematic variations designed in. Among random defects, the density of back-end-of-line (BEOL) defects (i.e., interconnect defects) is increasing, compared to that of front-end-of-line (FEOL) defects (i.e., device defects) [16]. Since the random BEOL defects mainly occur either between physically adjacent interconnects (short defects) or on interconnect itself (open defects), routing and interconnect optimization should be the suitable stage for random-defect yield optimization [4, 17, 21].

This work is supported in part by SRC, IBM Faculty Award, Fujitsu, and equipment donations from Intel.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

DAC 2007, June 4–8, 2007, San Diego, California, USA.

Copyright 2007 ACM ACM 978-1-59593-627-1/07/0006 ...\$5.00.

In general, routing consists of two steps, global routing and detailed routing. Global routing plans an approximate path for each net, while detailed routing finalizes the exact DRC-compatible pin-to-pin connections. Track routing, as an intermediate step between global and detailed routing, can expedite detailed routing by embedding major trunks from each net within a panel (a row/column of global routing cells) in DRC-friendly manner [3]. Such track routing is an appealing stage to optimize critical area for yield enhancement, as decent flexibility in routing optimization exists with wire adjacency information [8, 17, 26], while global routing is lack of wire adjacency information for accurate critical area estimation and detailed routing cannot provide enough flexibility for significant yield improvement.

Due to the criticality of yield, there have been considerable amount of efforts to enhance yield by reducing critical area in routing or post-routing. However, there are a few drawbacks in these prior works: (a) one single defect size is considered, rather than a defect size distribution [18, 22], (b) the trade-off between open and short defects due to fixed routing area is ignored [1, 2, 16, 18, 22], (c) localized/greedy optimization is performed, which may be suboptimal [2, 4, 7, 16, 23], (d) wire adjacency information is not available for accurate critical area estimation [14, 20].

In this work, we propose TROY, a track router with yield-driven wire planning (wire ordering, sizing, and spacing) to optimize yield w.r.t random defects. TROY orders wires first to minimize overlapped wirelength between adjacent wires based on *preference-aware minimum Hamiltonian path*, and then performs optimal wire sizing and spacing for the ordered wires with efficient *second order conic programming*. As a result, globally optimal wire width and spacing as well as minimal overlapped wirelength decreases critical area, making a design robust to random defects. The major contributions of this paper include the following.

- We propose TROY, a track router with yield-driven wire planning. To our best knowledge, this is the first work that yield is optimized during track routing.
- We propose a simple model of probability of failure due to random defects. This simple, yet effective model enables our second order conic programming.
- We show that wire ordering within a panel (the first step of wire planning in TROY) can be efficiently solved by preference-aware minimum Hamiltonian path. TROY considers the interaction between adjacent panels to overcome any disadvantage from isolated panel-by-panel approach.
- We show that wire sizing and spacing for an entire layer (the second step of wire planning in TROY) can be formulated as second order conic programming which can be solved optimally in $O(N^{1.3})$.

2. PRELIMINARIES

2.1 Notations

Table 1 shows a list of notations in this paper. Fig. 1 shows an example of track routing where six wires from W_1 to W_6 are assumed to be already routed (thus, p_1 to p_6 are known) within a panel P_i which is bounded by T_i and B_i . M_i is the median of x/y positions of all the pins in the panel where W_i exists. If $p_i \neq M_i$, we can use the deviation $|p_i - M_i|$ as a metric for possible wirelength increase, because the shortest trunk-steiner tree can be built with the median of pins [6]. Thus, p_i should be as close as possible to M_i for shorter wirelength and less defects.

Table 1: The notations in this paper.

W_i	wire i
M_i	preferred position of W_i for minimal wirelength
L_i	wirelength of W_i
L_{ij}	overlapped wirelength between W_i and W_j
p_i	x/y position of the center of W_i
n_i	a set of wires adjacent to W_i
l_{ij}	adjacent and overlapped wirelength between W_i and W_j
w_i	wire width of W_i
s_{ij}	spacing between W_i and W_j , $ p_i - p_j - \frac{w_i + w_j}{2}$
P_i	the i -th panel
T_i	the top position of P_i
B_i	the bottom position of P_i
W_{min}	minimum wire width of the layer
W_{max}	maximum wire width of the layer
S_{min}	minimum wire spacing of the layer

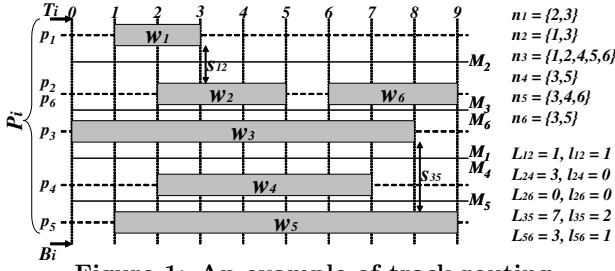


Figure 1: An example of track routing

2.2 Critical Area and Probability of Failure

Critical area for a defect is equal to the area where the center of the defect must fall in order to cause a circuit failure for a given defect size distribution. Probability of failure (POF) based on critical area analysis with defect size distribution is a widely used metric for yield prediction and optimization [8, 17]. The defect size distribution $F(x)$ is widely modelled as follows [8, 19]:

$$F(x) = kx^{-r} \quad \text{for } x_{min} \leq x < \infty \quad (1)$$

where x is the defect size, x_{min} is the minimum resolvable lithographic feature size, k is a coefficient to ensure $\int_{x_{min}}^{\infty} F(x) dx = 1$, and $r \approx 3$ [9]. When the end effect is ignored [14], the critical area $A_i^o(x)$ for open defects on a wire W_i and the critical area $A_{ij}^s(x)$ for short defects between two parallel wires W_i and W_j can be approximated as follows [8, 15, 19]:

$$A_i^o(x) = \begin{cases} 0 & \text{for } 0 \leq x < w_i \\ L_i(x - w_i) & \text{for } w_i \leq x < 2w_i + S_{min} \\ L_i(w_i + S_{min}) & \text{for } 2w_i + S_{min} \leq x < \infty \end{cases}$$

$$A_{ij}^s(x) = \begin{cases} 0 & \text{for } 0 \leq x < s_{ij} \\ l_{ij}(x - s_{ij}) & \text{for } s_{ij} \leq x < 2s_{ij} + W_{min} \\ l_{ij}(s_{ij} + W_{min}) & \text{for } 2s_{ij} + W_{min} \leq x < \infty \end{cases} \quad (2)$$

where L_i , w_i , l_{ij} , s_{ij} are as in Table 1. Since $A_i^o(x)$ and $A_{ij}^s(x)$ cannot keep increasing, it saturates at a defect size of $2s_{ij} + W_{min}$ and $2w_i + S_{min}$, respectively [19]. The probability of failure due to open defects on W_i (POF_i^o) and due to short defects between W_i and W_j (POF_{ij}^s) on a given layer can be obtained as follows [8, 19]:

$$POF_i^o = \int_{x_{min}}^{\infty} F(x) \frac{A_i^o(x)}{A_{chip}} dx = \frac{kL_i}{2A_{chip}} \left(\frac{w_i + S_{min}}{2w_i^2 + S_{min}w_i} \right)$$

$$POF_{ij}^s = \int_{x_{min}}^{\infty} F(x) \frac{A_{ij}^s(x)}{A_{chip}} dx = \frac{kl_{ij}}{2A_{chip}} \left(\frac{s_{ij} + W_{min}}{2s_{ij}^2 + W_{min}s_{ij}} \right) \quad (3)$$

where A_{chip} is the total chip area. As POF_i^o and POF_{ij}^s indicate the chance of having a random defect, yield can be improved by minimizing POF_i^o and POF_{ij}^s together, which can be accomplished by maximizing wire width (w_i) and wire spacing (s_{ij}) respectively. However, minimizing POF_i^o and POF_{ij}^s are two conflicting objectives, as larger w_i to decrease POF_i^o leads to smaller s_{ij} which increases POF_{ij}^s with a fixed routing area.

3. YIELD-DRIVEN TRACK ROUTING

In this section, we show yield-driven track routing in mathematical formulation. To maximize yield, we need to minimize POF^o and POF^s for all the wires in the design. At the same time, the deviation (see Section 2.1) resulted from track routing should be taken into account as potential victim of open defects. Assuming such deviation is captured by POF_i^{o*} , yield-driven track routing can be formulated as in Fig. 2 where the objective is the weighted total probability of failure, and α is a user-defined parameter ($0 \leq \alpha \leq 1$) to control the trade-off between open and short defects. The constraint (a) is about the deviation of W_i from M_i (possible wirelength increase), and the constraint (b) is to guarantee $s_{ij} \geq S_{min}$ for any adjacent wires. The constraint (c) is to keep wires within the corresponding panel, and the constraint (d) is to control wire width w_i .

$$\begin{aligned} \min : & \alpha \sum_i (POF_i^o + POF_i^{o*}) + (1 - \alpha) \sum_{i,j>i} POF_{ij}^s \\ \text{s.t. :} & \text{(a) } |p_i - M_i| \leq d_i \quad \forall i \\ & \text{(b) } S_{min} \leq s_{ij} \leq |p_i - p_j| - \frac{(w_i + w_j)}{2} \quad \forall i, \forall j \in n_i \\ & \text{(c) } B_k + \frac{w_i}{2} \leq p_i \leq T_k - \frac{w_i}{2} \quad \forall i \in P_k \\ & \text{(d) } W_{min} \leq w_i \leq W_{max} \quad \forall i \end{aligned}$$

Figure 2: Yield-driven track routing formulation

The objective in Fig. 2 is non-linear, and the constraint (b) is non-convex. In fact, this formulation has high combinatorial complexity, as neither the order of wires is fixed nor p_i is identified. We can convert the formulation in Fig. 2 into an integer non-linear programming as in Fig. 3 by reformulating the constraint (b) with a binary integer variable o_{ij} , which is set to 1 if $p_i > p_j$, otherwise 0. N is a huge constant.

$$\begin{aligned} \min : & \alpha \sum_i (POF_i^o + POF_i^{o*}) + (1 - \alpha) \sum_{i,j>i} POF_{ij}^s \\ \text{s.t. :} & |p_i - M_i| \leq d_i \quad \forall i \\ & S_{min} \leq s_{ij} \leq p_i - p_j - \frac{(w_i + w_j)}{2} + (1 - o_{ij})N \quad \forall i, j \\ & S_{min} \leq s_{ij} \leq p_j - p_i - \frac{(w_i + w_j)}{2} + o_{ij}N \quad \forall i, j \\ & o_{ij} \in \{0, 1\} \quad \forall i, j \\ & B_k + \frac{w_i}{2} \leq p_i \leq T_k - \frac{w_i}{2} \quad \forall i \in P_k \\ & W_{min} \leq w_i \leq W_{max} \quad \forall i \end{aligned}$$

Figure 3: Yield-driven track routing formulation in integer non-linear programming

Optimally solving the formulation in Fig. 3 maximizes yield w.r.t random defects in track routing. However, this formulation is unacceptably expensive to compute even with a linearized objective function by Taylor expansion (not to mention that this linearization can introduce significant suboptimality).

4. TROY ALGORITHM

In this section, we present our track routing algorithm, TROY to solve the formulation in Fig. 3. The key observation we make is that if POF_i^o and POF_{ij}^s in Eq. (3) are simplified into simpler convex forms as in Eq. (4) and o_{ij} (thus, n_i as well) is known, wire sizing and spacing which significantly impact immunity to random defects can be optimally solved by *second order conic programming* (SOCP), which provides a global optimal solution as efficiently as linear programming. The model in Eq. (3) shows over 99% regression coefficient for a wide range of wire width/spacing.

$$POF_i^o \approx \frac{kL_i}{2A_{chip}} \left(a \frac{S_{min}}{w_i} - b \right) \quad \left(1 \leq \frac{w_i}{S_{min}} \leq 20 \right)$$

$$POF_{ij}^s \approx \frac{kl_{ij}}{2A_{chip}} \left(a \frac{W_{min}}{s_{ij}} - b \right) \quad \left(1 \leq \frac{s_{ij}}{W_{min}} \leq 20 \right) \quad (4)$$

4.1 Motivation and Strategy

The wire width (w_i) needs to be larger to minimize POF_i^o , while the overlapped wirelength (l_{ij}) and wire spacing (s_{ij}) need to be smaller and larger respectively to minimize POF_{ij}^s . Minimizing POF_{ij}^s is much harder than POF_i^o , because the former

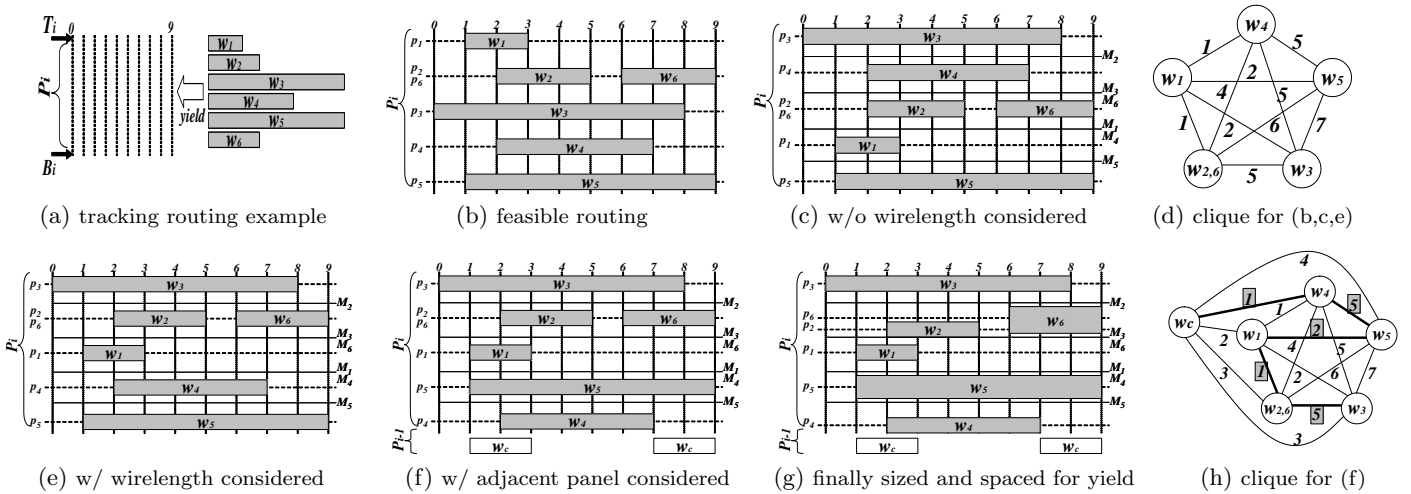


Figure 4: TROY example

has two variables and l_{ij} depends on the order of wires (intuitively, optimizing POF_{ij}^s involves two adjacent wires). Finding the optimal order of wires (thus, o_{ij}) is NP-hard, as it can be deduced from minimum Hamiltonian path problem. On the top of that, w_i and s_{ij} are two conflicting objectives. Thus, minimizing POF_i^o and POF_{ij}^s in Fig. 3 is prohibitively expensive.

Interestingly, we discover that POF_i^o can be translated into a rotated conic constraint [13], and due to the similarity between POF_i^o and POF_{ij}^s , POF_{ij}^s can be also translated into a rotated conic constraint, if l_{ij} is known. This translation enables fast SOCP which can find globally optimal w_i and s_{ij} with $O(N^{1.3})$ bound where N is the number of variables [5]. This observation motivates our two step wire planning for TROY as follows:

- Wire Ordering:** The goal of wire ordering is to compute o_{ij} (thus l_{ij}). In TROY, wire ordering is done in each panel such that total overlapped wirelength between adjacent wires is minimized by finding *preference-aware minimum Hamiltonian path*. This is discussed in Section 4.2.
- Wire Sizing and Spacing:** The goal of wire sizing and spacing is to tune wire width and spacing such that the maximum immunity to random defects, thus maximum yield can be achieved. As wire sizing and spacing are conflicting objectives due to fixed routing area, the optimal trade-off is found by *second order conic programming*. This is discussed in Section 4.3.

4.2 Wire Ordering Optimization

The goal of wire ordering is to find an order of wires such that the overlapped wirelength (l_{ij}) between adjacent wires is minimized. Although the critical area is not explicitly targeted, shorter overlapped wirelength will effectively reduce POF^s . We first identify a set of disjoint sub-panels within each panel such that there is no shared wire between any two identified sub-panels. Then, wire ordering is performed from the lowest panel to the highest panel for each sub-panel in each panel.

Wire ordering for each sub-panel to minimize total overlapped wirelength can be achieved by well-known minimum Hamiltonian path (MHP) [22, 26]. Consider the example in Fig. 4 where six wires ($W_1 \sim W_6$) are to be routed within a sub-panel of a panel P_i for maximum yield. Fig. 4 (a) illustrates the problem in this example. First, assuming minimum wire width and spacing, a feasible track routing needs to be found through interval packing [10]. Then, a clique as in Fig. 4 (d) can be constructed by regarding each row as a vertex, and edge weight E_{ij} between two rows (thus, two vertices), V_i and V_j can be computed as follows:

$$E_{ij} = \sum_{W_i \in V_i, W_j \in V_j} L_{ij} \quad (5)$$

Since finding MHP from the clique is well-studied, we skip the details. From the MHP, a routing solution like Fig. 4 (c) can be found. However, naive MHP approach has two drawbacks regarding yield, which we further address in TROY.

- The possible wirelength increase due to deviation from the median (See Section 2.1) is not considered, which in turn increases the density of random defects.
- The interaction between adjacent panels is ignored. As short defects can occur on the boundary of adjacent panels, it is required to take this into account.

We observe that there can be multiple optimal MHP solutions, as the distribution of edge weights are rather narrow. Thus, we need to find the minimum deviation solution estimated by $\sum_i |p_i - M_i|$ among all the optimal MHP solutions. We call our modified MHP as *preference-aware minimum Hamiltonian path* (pMHP). For example, although both Fig. 4 (c) and (e) are the MHP of Fig. 4 (d) (the same overlapped wirelength), one can recognize that Fig. 4 (e) shows less deviation from the preferred positions ($\sum_i |p_i - M_i|$), which can result in less random defects.

We further improve our wire ordering by considering the contour of adjacent panel. Consider the example in Fig. 4 (f) where W_c are the wires from a panel P_{k-1} , assuming the wires in P_{k-1} are already ordered. Fig. 4 (f) shows a better wire ordering than Fig. 4 (e), when the interaction between P_k and P_{k-1} is considered. This can be done with a new clique in Fig. 4 (h) where W_c is added and set as a starting vertex, and the bold lines indicate the pMHP. The edge weight between W_c and other vertices can be computed with Eq. (5) as well.

4.3 Globally Optimal Wire Sizing and Spacing

After wires in every panel on a layer are ordered, the formulation in Fig. 3 can be further deduced into the formulation in Fig. 5 after plugging in Eq. (4), filling all the integer variables (o_{ij}) with the corresponding values, and eliminating constant terms from the objective. Auxiliary variable γ_{ij} and δ_{ij} are introduced to translate the non-linear objective into rotated conic constraints which enable second order conic programming (SOCP) [13]. Due to the space limitation, we omit the derivation details. The formulation in Fig. 5 can be solved optimally and efficiently by interior-point method with $O(N^{1.3})$ bound where N is the number of variables [5, 13], thus the solution will provide the optimal wire width and position (thus, spacing) for maximum yield. The optimal wire sizing and spacing for an entire layer by SOCP can find the optimal trade-off between open defects and short defects in

$$\begin{aligned} \text{min : } & \alpha \sum_i \{ \delta_i + (1 - \frac{b}{a}) d_i \} + (1 - \alpha) \sum_{i,j} \gamma_{ij} \\ \text{s.t : } & |p_i - M_i| \leq d_i \quad \forall i \\ & S_{min} \leq s_{ij} = p_i - p_j - \frac{w_i + w_j}{2} \quad \forall o_{ij} = 1, \forall j \in n_i \\ & l_{ij} W_{min} \leq s_{ij} \gamma_{ij} \quad \forall i, \forall j \in n_i \\ & L_i S_{min} \leq w_i \delta_i \quad \forall i \\ & B_k + \frac{w_i}{2} \leq p_i \leq T_k - \frac{w_i}{2} \quad \forall i \in P_k \\ & W_{min} \leq w_i \leq W_{max} \quad \forall i \end{aligned}$$

Figure 5: Yield-driven track routing in SOCP

Table 2: Comparison between greedy track router and TROY ($\alpha = 0.6$).

circuit			Greedy (discrete wire width)				TROY (discrete wire width)				TROY (continuous wire width)							
name	grid	wires	wlen (μm)	wlen inc ^a (μm)	yield loss			cpu (sec)	wlen inc (μm)	yield loss			cpu (sec)	wlen inc (μm)	yield loss			cpu (sec)
					open	short	sum			open	short	sum			open	short	sum	
ibm01	64x64	35K	135172	19606.6	527	521	1048	1	19119.7	561	320	881	13	19119.7	547	310	857	13
ibm02	80x64	68K	1049388	97720.2	660	513	1173	1	94747.6	511	445	956	90	94747.6	539	407	946	90
ibm03	80x64	56K	899694	69128.0	666	457	1123	1	68808.3	519	447	966	50	68808.3	532	396	928	51
ibm04	96x64	73K	714612	86668.5	692	527	1219	1	88547.1	618	399	1017	50	88547.1	612	379	991	50
ibm05	128x64	110K	4917756	290296.0	568	406	974	2	292148.4	389	293	682	217	292148.4	391	278	669	218
ibm06	128x64	112K	1693308	139420.2	776	507	1283	2	139782.8	671	491	1162	87	139782.8	679	463	1142	86
ibm07	192x64	141K	2652531	224507.6	587	453	1040	4	221714.3	459	351	810	131	221714.2	461	328	789	136
ibm08	192x64	160K	2456022	201960.8	691	481	1172	4	205223.8	562	425	987	117	205223.8	582	393	975	116
ibm09	256x64	156K	2094085	168111.3	681	539	1220	5	169543.3	551	423	974	110	169543.2	553	408	961	108
ibm10	256x64	216K	4701936	344649.2	599	456	1055	7	344612.4	441	376	817	214	344612.4	447	344	791	219
total			1642068.3	6447	4860	11307	27	1644247.8	5282	3970	9252	1079	1644247.5	5343	3706	9049	1087	
norm.			1.00	1.00	1.00	1.00	1.00	1.00	0.82	0.82	0.82	39.96	1.00	0.83	0.76	0.80	40.3	

^a wirelength increase (the summation of the deviation of each wire from its preferred location).

terms of yield. Thus, TROY is much superior to traditional local or iterative approaches. Fig. 4 (g) shows a track routing solution after wire sizing and spacing are done by SOCP.

5. EXPERIMENTAL RESULTS

We implement TROY in C++. The initial global routing results are generated from the publicly available BoxRouter binary [12]. All the experiments are performed on a 3.0 GHz Pentium machine with 1GB RAM. A solver in [11] is properly modified to find preference-aware minimum Hamiltonian path for wire ordering in Section 4.2, and MOSEK 4.0 [13] is used to solve second order conic programming for wire sizing and spacing in Section 4.3. We assume $0.13\mu\text{m}$ technology to use the defect size distribution parameter in [8], and set $S_{min}=W_{min}=0.2\mu\text{m}$. We further assume that $W_{max}=0.4\mu\text{m}$, and 0.2, 0.3 and $0.4\mu\text{m}$ are only allowed wire widths. The first set of columns in Table 2 shows the detail for each benchmark. Since the benchmarks are lack of detailed pin locations, 1-5 pins for each global routing cell on each wire are randomly generated, to define the preferred position of each wire (M_i). For comparison, we implement a greedy algorithm similar to [24], the only track routing algorithm which can handle arbitrary wire spacing to our best knowledge. As the original algorithm in [24] optimizes crosstalk and timing without wire sizing, we modify the optimization objective and add wire sizing function for greedy minimization of POFs. Monte-Carlo simulation [25] with 10K random defects based on Eq. (1) is performed to estimate yield loss.

Table 2 shows the comparison between the greedy yield-driven track router similar to [24] and TROY with $\alpha = 0.6$ (one in continuous wire width and the other one in discrete wire width). It clearly shows that TROY in discrete wire width can significantly reduces yield loss by 18% on average, and it can be even over 30% for ibm05, compared with the greedy approach in discrete wire width. The discrete wire width incurs only 2.2% more yield loss on average, when TROY in discrete wire width and continuous wire width are compared.

6. CONCLUSION

In order to cope with yield loss due to random defects in the advance technology, we present TROY, an efficient track router with yield-driven wire planning. With effective wire ordering and wire sizing/spacing optimization, experimental results show that TROY reduces yield loss significantly.

7. REFERENCES

- [1] G. A. Anah, Targeted Layout Modifications for Semiconductor Yield/Reliability Enhancement. *IEEE Trans. on Semiconductor Manufacturing*, 17(4), Nov 2004.
- [2] C. Bamji and E. Malavasi. Enhanced network flow algorithm for yield optimization. In *Proc. Design Automation Conf.*, 1996.
- [3] S. Batterywala, N. Shenoy, W. Nicholls, and H. Zhou. Track Assignment: A Desirable Intermediate Step Between Global Routing and Detailed Routing. In *Proc. Int. Conf. on Computer Aided Design*, 2002.
- [4] Y. Bourai and C.-J. R. Shi. Layout compaction for yield optimization via critical area minimization. In *Proc. Design, Automation and Test in Europe*, 2000.
- [5] S. Boyd and L. Vandenberghe. *Convex optimization*. In Cambridge, 2004.
- [6] H. Chen, C. Qiao, F. Zhou, and C.-K. Cheng. Refined Single Trunk Tree: A Rectilinear Steiner Tree Generator For Interconnect Prediction. In *Proc. System-Level Interconnect Prediction*, April 2002.
- [7] V. K. I. Chiluvuri and I. Koren. Layout-Synthesis Techniques for Yield Enhancement. *IEEE Trans. on Semiconductor Manufacturing*, 8(2), May 1995.
- [8] P. Cristie and J. P. de Gyvez. Prelayout Interconnect Yield Prediction. *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 11(1), Feb 2003.
- [9] R. Glang. Defect Size Distribution In VLSI Chips. *IEEE Trans. on Semiconductor Manufacturing*, 4(4), Nov 1991.
- [10] U. I. Gupta, D. T. Lee, and J. Y.-T. Leung. An Optiaml Solution for the Channel-Assignment Problem. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, c-28(11), Nov 1979.
- [11] <http://www.akira.ruc.dk/~keld/research/LKH>.
- [12] <http://www.cerc.utexas.edu/utda>.
- [13] <http://www.mosek.com>.
- [14] E. P. Huijbregtz, H. Xue, and J. A. Jess. Routing for Reliable Manufacturing. *IEEE Trans. on Semiconductor Manufacturing*, 8(2), May 1995.
- [15] T. Iizuka, M. Ikeda, and K. Asada. Exact Wiring Fault Minimization via Comprehensive Layout Synthesis for CMOS Logic Cells. In *Proc. Int. Symp. on Quality Electronic Design*, Mar 2004.
- [16] A. B. Kahng, B. Liu, and I. I. Mandoiu. Non-tree routing for reliability and yield improvement. In *Proc. Int. Conf. on Computer Aided Design*, Nov 2002.
- [17] I. Koren. Should Yield be a Design Objective? In *Proc. Int. Symp. on Quality Electronic Design*, Mar 2000.
- [18] S.-Y. Kuo. YOR: a yield-optimizing routing algorithm by minimizing critical areas and vias. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 12(9):1303–1311, Sept 1993.
- [19] W. Maly. Modeling of Lithography Related Yield Losses for CAD of VLSI Circuits. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, July 1985.
- [20] D. Muller. Optimizing yield in global routing. In *Proc. Int. Conf. on Computer Aided Design*, 2006.
- [21] E. Papadopoulou and D. T. Lee. Critical Area Computation via Voronoi Diagrams. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 18, 1999.
- [22] A. Pitaksanonku, S. Thanawastien, C. Lursinsap, and J. Gandhi. Dtr: A defect-tolerant routing algorithm. In *Proc. Design Automation Conf.*, 1989.
- [23] J. Z. Su and W. Dai. Post route optimization for improved yield using a rubber-band wiring model. In *Proc. Int. Conf. on Computer Aided Design*, Nov 1997.
- [24] H.-P. Tseng, L. Scheffer, and C. Sechen. Timing- and Crosstalk-Driven Area Routing. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 20(4), April 2001.
- [25] D. Walkter. *Yield Simulation for Integrated Circuits*. 1987.
- [26] D. Wu, J. Hu, M. Zhao, and R. Mahapatra. Timing Driven Track Routing Considering Coupling Capacitance. In *Proc. Asia and South Pacific Design Automation Conf.*, 2005.