

CROSSTALK DRIVEN ROUTING RESOURCE ASSIGNMENT

Hailong Yao, Qiang Zhou, Xianlong Hong, Yici Cai

EDA Lab., Dept. of Computer Science & Technology
Tsinghua University, Beijing, 100084, China

ABSTRACT

Crosstalk noise is one of the emerging issues in deep sub-micrometer technology which causes many undesired effects on the circuit performance. In this paper, a CDRRA algorithm, which integrates the routing layers and tracks to address the crosstalk noise issue during the track/layer assignment stage, is proposed. The CDRRA problem is formulated as a weighted bipartite matching problem and solved using the linear assignment algorithm. The crosstalk risks between nets are represented by an undirected graph and the maximum number of the concurrent crosstalk risking nets is computed as the max-clique of the graph. Then the nets in each max-clique are assigned to disadjacent tracks. Thus the crosstalk noise can be avoided based on the clique concept. The algorithm is tested by a set of bench mark examples and the experimental results show that it can improve the final routing layout a lot with little loss of the completion rate.

1. INTRODUCTION

As VLSI technology advances, crosstalk noise has emerged to be critical in determining the performance of the overall chip. Crosstalk noise profoundly affects the performance of a circuit because it causes signal delays, logic hazards and even circuit malfunctioning [1].

Traditionally, the routing problem is divided into two sequential stages: global routing and detailed routing. The global router determines wirings for each net in a rough scale across the Global Routing Cells (GRC), while the detailed router decides the exact connections for nets inside the individual GRCs each at a time. Most previous works on crosstalk control are performed during the detailed routing stage [2-5] where the estimation of crosstalk can be accurate but the flexibility to control it is restricted. However, the crosstalk control during the global routing stage can not be accurate enough though it has more freedom. With enough flexibility and fairly accurate net routing information, an intermediate stage proves to be an ideal place to solve the problem [6]. In literature, Track Assignment (TA) and Cross Point Assignment (CPA) stages have been taken as the intermediate ones. In [6], the track assignment problem is studied to improve the routing results and the running time of the whole routing stage, but it does not consider the crosstalk issue. Some works on crosstalk avoidance during the CPA stage have been done in [7-9]. Then

the crosstalk-aware track/layer assignment which can estimate the crosstalk more accurately is study in [10-12]. In [10-11], the track/layer assignment is integrated into global routing and in [12] it is formulated as an ILP problem and solved using a constructive randomized rounding technique.

In this paper, a new crosstalk-aware track/layer assignment heuristic algorithm called Crosstalk Driven Routing Resource Assignment (CDRRA) is proposed. It fully utilizes the routing resources including the routing layers and tracks to resolve the crosstalk noise issue. The rest of the paper is organized as follows. In Section 2, the crosstalk noise model is given. And in Section 3, details about the CDRRA algorithm are discussed. Then in Section 4, the experimental results are presented. Finally, a conclusion to our work is drawn in Section 5.

2. CROSSTALK NOISE MODEL

2.1 Preliminaries

After global routing, the routing area is divided into $m \times n$ global routing cells. Then during the CDRRA stage, the GRCs are merged into horizontal and vertical slices, which can be represented as $HS_i = \{grc_{i1}, grc_{i2}, \dots, grc_{in}\}$ and $VS_j = \{grc_{1j}, grc_{2j}, \dots, grc_{mj}\}$ respectively. Then the slices are dissected into routing tracks which serve as the routing resources onto which net segments can be assigned. Thus a slice S_k with r routing tracks can be represented as $S_k = \{t_{k1}, t_{k2}, \dots, t_{kr}\}$. Fig.1 gives an example of a horizontal slice with 4 GRCs and 6 routing tracks.

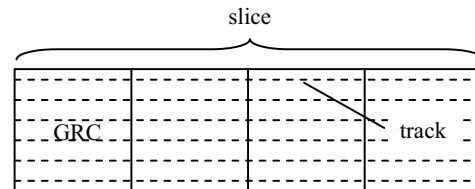


Fig. 1 A horizontal slice: A row of GRCs

For each slice, the net segments are constructed from the global nets across different GRCs within the slice. Since the detailed routing results are not decided, the end points of the net segments are set to the centers of the GRCs. In Fig. 2, a Steiner-tree based global route for net ABCDE spans through three slices, two vertical (4, 8) and one horizontal (7). Then the global net is decomposed into three net segments: AC, BD and DE. Note that A, B, C, D and E are the center points of the corresponding GRCs.

This work is supported by the National Natural Science Foundation of China (NSFC) 60176016, High-Tech Research & Development (863) Program Of China 2002AAIZ1460 and Specialized Research Fund for the Doctoral Program of Higher Education (SRFDP) 20020003008

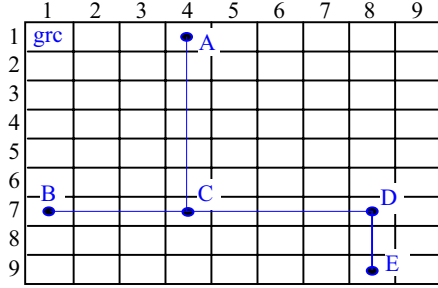


Fig. 2 Decomposition of a global net in the slices

2.2 Crosstalk Model

According to [13], if a switch event on signal net N_i causes signal net N_j to malfunction, then N_i and N_j are regarded to be sensitive to each other, where N_i is called the aggressor and N_j the victim. While the sensitivity rate is defined as the ratio of the number of aggressors for N_i to the total number of signal nets. Then we can define a sensitivity matrix $S = [s_{ij}]_{N \times N}$ where N is the total number of the signal nets and $s_{ij} = 1$ if N_i and N_j are sensitive to each other, otherwise $s_{ij} = 0$. Since when N_i is sensitive to N_j , N_j is also sensitive to N_i , the sensitivity matrix is symmetric. In the implementation, we randomly generate the sensitive nets for each signal net for a given sensitivity rate and then construct the sensitivity matrix.

Crosstalk noise depends on the coupling capacitances, the driver resistances, the load capacitances and the input waveforms. In the CDRRA algorithm, only the capacitive crosstalk noise is considered. Here we adopt the coupling model same as [5] to calculate the coupling capacitance between sensitive nets.

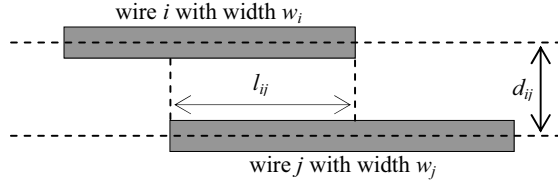


Fig. 3 The coupling capacitance between two nets

In Fig. 3, the coupling capacitance between the two nets i and j can be calculated as formula (1) where w_i and w_j are the widths of wires i and j ($w_i, w_j > 0$), f_{ij} is the unit length fringing capacitance between wires i and j , l_{ij} is the overlap length of wires i and j and d_{ij} is the distance from the center line of wire i to the center line of wire j .

$$C_c(i, j) = \frac{f_{ij} \times l_{ij}}{d_{ij}} \times \frac{1}{1 - \frac{w_i + w_j}{2d_{ij}}} \quad (1)$$

In formula (1), we assume that w_i , w_j and f_{ij} are constant parameters. So in order to achieve capacitive crosstalk-free layouts, we should control l_{ij} and d_{ij} , which means to avoid adjacent sensitive nets from running in parallel for a long distance. In the implementation, we assume that only adjacent sensitive nets will violate the crosstalk constraint when their overlap length exceeds a predefined constant MAXOL and disadjacent nets will never run into trouble with crosstalk violation.

3. THE CDRRA ALGORITHM

3.1 Problem Formulation

For the CDRRA algorithm, only the global nets which span more than one GRC are considered. The algorithm is performed in slice-by-slice manner. For each slice S_k , the assignment of net segments to the routing tracks can be formulated as follows:

$$\Phi: N \times T \rightarrow C \quad (2)$$

where N is the set of net segments inside the slice S_k and T the set of the routing tracks of S_k . C is the set of costs which indicates the consumption of the assignment pairs $\langle n_i, t_j \rangle$ ($1 \leq i \leq |N|$, $1 \leq j \leq |T|$). The track assignment problem is to find a feasible set $\Phi' = \{c_{i,j} \mid c_{i,j} = \Phi(\langle n_i, t_j \rangle), 1 \leq i \leq |N|, 1 \leq j \leq |T|\}$ for all the elements in N , where the objective

$$\sum_{c_{i,j} \in \Phi'} c_{i,j} \text{ is minimized.}$$

The cost for assigning net segment n_i to the routing track t_j is mainly composed of the following items:

- **Layer Cost:** During the CDRRA procedure, the tracks on all the routing layers with the same routing direction (horizontal or vertical) are projected onto one plane. A net segment can be assigned to any of the tracks with the same routing direction. Since longer net segments on lower routing layers will block the connections between other pins and net segments, higher costs are assigned to longer net segments for lower routing layers.
- **Obstacle Cost:** In order to make the CDRRA result feasible for the detailed router, we should consider the via draw-up issue during the assignment procedure. When there is an obstacle between the pins of a net and a routing track, then the assignment of the net's segments onto the routing track should be avoided because it will trouble via draw-up procedure in the following detailed routing.
- **Net Length Cost:** When the segments of a net are not assigned to the routing tracks simultaneously, the pre-routed net components make the routing tracks in the neighborhood more preferable for the minimization of the total net length. So we assign lower costs for these routing tracks.

Besides the items mentioned above, the cost matrix also plays a role in preventing the already assigned nets from coupling with the latter ones, which will be covered in section 3.3. When the cost matrix is constructed with the above components, a linear assignment algorithm [14] is adopted to find a minimum cost matching for the assignment problem.

3.2 The Underlying Graph Model

• The Crosstalk Graph (XG)

According to the sensitivity rate and the sensitivity matrix defined in section 2.2, we can get a Crosstalk Graph $XG(V, E_{xg})$ where V represents the set of all the nets and E_{xg} represents the sensitive relationship between the corresponding nets. Where there are two nets sensitive to each other, there is an edge in E_{xg} between their corresponding vertexes in V .

- **The Interval Graph (IG)**

After global routing, each global net is represented by a set of global routing cells which indicates its routing path. To store the net segments' overlap information, we introduce the Interval Graph (IG). Different from XG which stores the information for all the nets, IG is slice-based and only has information for the net segments in the current slice. Each time a new slice is being processed, IG should be reconstructed. IG can be denoted as $IG(V_{ig}, E_{ig})$ where V_{ig} is the set of the net segments inside the current slice and E_{ig} stores the overlap information between each two net segments. Where there is overlap between two net segments, there is an edge in E_{ig} between the corresponding vertexes in V_{ig} . Note that IG is a weighted graph and the weights on the edges in E_{ig} are the values of the overlap lengths.

- **The Real Crosstalk Graph (RXG)**

According to the crosstalk model introduced in section 2.2, only adjacent sensitive nets with their overlap length exceeding the constant MAXOL will violate the crosstalk constraint. So we introduce another graph called the Real Crosstalk Graph (RXG) to represent the real crosstalk risks. In fact, RXG is the subgraph of XG and IG. In $RXG(V_{rxg}, E_{rxg})$, each vertex in V_{rxg} represents a net segment and each edge in E_{rxg} represents a crosstalk risk between the corresponding net segments. When two net segments are sensitive to each other according to XG and their overlap length exceeds MAXOL according to IG, then there is an edge in E_{rxg} between the corresponding vertexes in V_{rxg} . When such pair of net segments is assigned to adjacent tracks, a crosstalk violation is assumed to occur. By calculating the max-clique of RXG, the maximum set of concurrent crosstalk risking net segments can be acquired. These net segments should not be assigned adjacent to each other to observe the crosstalk constraint.

- **The Tracks' Adjacency Graph (TAG)**

When RXG has been constructed in a slice, the net segments in the max-clique of RXG should first be assigned to disadjacent tracks to avoid the crosstalk violation. To store the adjacency information of the routing tracks in a slice, we introduce the Tracks' Adjacency Graph (TAG). Like IG and RXG, TAG is also slice-based and should be reconstructed for each new slice. In $TAG(V_{tag}, E_{tag})$, V_{tag} is the set of routing tracks in the current slice and E_{tag} stores the adjacency information between the routing tracks. There is an edge in E_{tag} if and only if the two routing tracks of the corresponding vertexes are adjacent to each other. Then it is obvious that the maximum independent set in TAG stores the maximum number of the disadjacent routing tracks, to which real crosstalk risking net segments in RXG should be assigned.

Due to the characteristics of the routing tracks, the task of calculating the maximum independent set in TAG can be saved by a simple numbering technique. The routing tracks in each slice are numbered in the sequential order according to their relative positions and each time the real crosstalk risking net segments are assigned to the even numbered tracks or the odd ones iteratively. Thus, the crosstalk noise can be eliminated.

Fig. 4 illustrates the graphs defined above. The XG for all the nets named a, b, c, d and e is shown in a). Since net a does not pass through the current slice, it does not appear in the IG of b). In the IG, the overlap lengths are marked as the weights of the corresponding edges. We assume the constant MAXOL is set to

50, and then the RXG of c) is constructed from the XG and the IG of the current slice. f) gives the TAG of the tracks in the slice of d). From the TAG, we can compute the maximum independent set $MIS=\{1, 3, 5\}$. Then we calculate the max-clique of the RXG as $MC=\{b, c, d\}$. Now we can assign MC to MIS using the linear assignment algorithm. Finally we assign the net segment e to a routing track under the guidance of the IG and the cost matrix. Note that net segments that do not overlap with each other can share the same track, while those which overlap can not. Now we get the crosstalk-free assignment result shown in d).

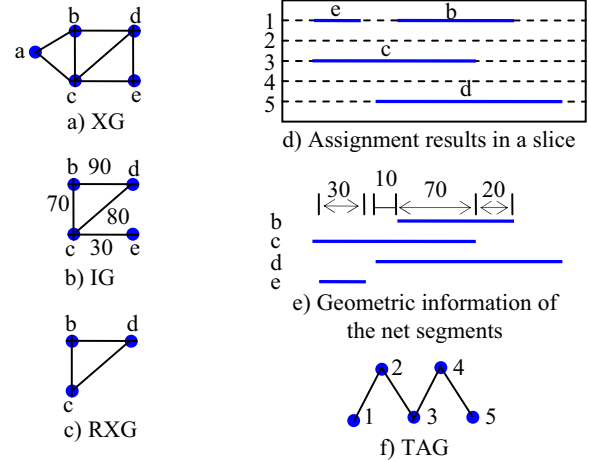


Fig. 4 Illustration of the Graph Model

3.3 Overview of the CDRRA Algorithm

The main steps of our algorithm can be summarized as in Fig. 5. For each slice, the assignment is an iterative procedure where the updated cost matrix eliminates the crosstalk risks between the already assigned net segments and the latter sensitive ones. After processing the net segments from the RXG, the algorithm tries to assign net segments as many as possible. Since the max-clique

- (1) Read in the sensitivity rate and construct the crosstalk graph (XG).
- (2) For all the horizontal and vertical slices, DO
- (3) Construct the Interval Graph (IG).
- (4) Construct the Real Crosstalk Graph (RXG).
- (5) Construct the tracks' adjacency graph (TAG).
- (6) Construct the cost matrix for the assignments of net segments onto the routing tracks.
- (7) Compute the maximum clique in RXG and the maximum independent set in TAG. Calculate the minimum cost assignment solution using the linear assignment algorithm.
- (8) Update IG, RXG, TAG and the cost matrix according to the assignment results. If RXG is NULL, then go to (9), else go to (7).
- (9) Compute the maximum clique from IG and assign the net segments onto the remaining routing tracks using the same algorithm until all the net segments are assigned or the routing tracks are not available.
- (10)END For

Fig. 5 Main Steps of the CDRRA Algorithm

problem and its equivalent one maximum independent set problem are both NP-complete [15], we adopt some heuristics to find the near optimal solution to save the running time. After the CDRRA procedure, the assignment results are fed to the detailed router as pre-routes. Note that the assignment results serve as a guide but not a constraint for the follow-up detailed router. During detailed routing stage, any pre-routed net segment can be ripped up and re-routed with a higher cost than ordinary rip-ups.

4. EXPERIMENTAL RESULTS

We have implemented the CDRRA algorithm in C programming language on SUN Enterprise E450 and have tested it by a set of IBM benchmarks. Parts of the experimental results are shown in Table 1, where the numbers of the routing layers and tracks are denoted in the column under "L/T". The total numbers of the net segments and nets are shown below "Segs" and "Nets" respectively. In the implementation, the sensitivity rate is set to be 0.5, and then the numbers of the sensitive net segment pairs and the sensitive net pairs are calculated as in column "S-SP's" and "S-NP's". The CDRRA algorithm runs in two modes named "XC" and "NXC". "XC" indicates that the crosstalk control is considered and "NXC" denotes no crosstalk control. Fig. 5 gives the main steps for the "XC" running mode. In "NXC" mode, only the Interval Graph is considered in the algorithm and step (9) in Fig. 5 is repeated until all the net segments have been assigned or no routing resources are available. From the experimental results, we can see that the overflow, i.e. the number of net segments or the number of nets failed in the assignment, is small compared to the total number to be assigned. However, at the expense of the little overflow, the CDRRA algorithm can greatly improve the routing layout in the crosstalk avoidance aspect. From the columns below "X-SP's" and "X-NP's", which means the number of the net segment pairs and the number of the net pairs violating the crosstalk constraints, we can see that the crosstalk violation pairs are greatly reduced. The improvement ratios are 37.5-69.8% for the sensitive net segment pairs and 36.8-68.8% for the sensitive net pairs respectively.

5. CONCLUSION

In this paper, a CDRRA algorithm is proposed to address the crosstalk problem between the global routing and detailed routing stage. The basic idea of the algorithm is to calculate the crosstalk risking net segments by the clique heuristic and then assign them to disadjacent tracks using the minimum weighted bipartite matching problem formulation. The experimental results indicate that the CDRRA algorithm can greatly improve the final routing layout and eliminate most of the crosstalk violations.

6. REFERENCES

- [1] H.B. Bakoglu, "Circuits, Interconnections, and Packaging for VLSI," Addison Wesley, 1990.
- [2] T. Gao and C. L. Liu, "Minimum crosstalk channel routing," In Proc. Intl. Conf. on Computer-Aided Design, pages 692-696, Santa Clara, CA, November 1993.
- [3] T. Gao and C. L. Liu, "Minimum crosstalk switchbox routing," In Proc. Intl. Conf. on Computer-Aided Design, pages 610-615, San Jose, CA, November 1994.
- [4] H. Zhou and D. F. Wong, "An optimal algorithm for river routing with crosstalk constraints," In Proc. Intl. Conf. on Computer-Aided Design, San Jose, CA, 1996.
- [5] R. Kastner, E. Bozorgzadeh and M. Sarrafzadeh, "An Exact Algorithm for Coupling-Free Routing," in Proc. International Symposium on Physical Design, April 2001.
- [6] S. H. Batterywala, N. Shenoy, W. Nicholls, and Hai Zhou, "Track Assignment: A Desirable Intermediate Step Between Global Routing and Detailed Routing," IEEE International Conference on Computer Aided Design, San Jose, CA, 2002.
- [7] H.-P. Tseng, L. Scheffer, and C. Sechen, "Timing and crosstalk driven area routing," in Proc. 35th ACM/IEEE Design Automation Conf., June 1998, pp. 378-381.
- [8] C.-C. Chang and J. Cong, "Pseudo pin assignment with crosstalk noise control," in Proc. Int. Symp. on Physical Design, pp. 41-47, 2000.
- [9] H.L. Yao, Q. Zhou, X.L. Hong and Y.C. Cai, "Cross Point Assignment Algorithm with Crosstalk Constraint," Proceedings of the 5th International Conference on ASIC, October, 2003, pp. 352-355.
- [10] T. Xue and E.S. Kuh, "Post global routing crosstalk synthesis," TCAD, pp. 1418-1430, Dec. 1997.
- [11] H. Zhou and D.F. Wong, "Global Routing with Crosstalk Constraints," in Proc. ACM/IEEE Design Automation Conference, June 1998.
- [12] R. Kay, and R.A. Rutenbar, "Wire Packing: A Strong Formulation of Crosstalk-aware Chip-level Track/Layer Assignment with an Efficient Integer Programming Solution," ISPD '00, 61-68.
- [13] J.D. Ma and L. He., "Towards Global Routing with RLC Crosstalk Constraints," Proceedings. IEEE/ACM Design Automation Conference, New Orleans, Louisiana, USA, June 10-14 2002, pp. 669-62.
- [14] R.E. Burkard and U. Derigs, Assignment and Matching Problem: Solution Methods with Fortran-Programs, New York: Springer-Verlag, 1980.
- [15] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness (Freeman, New York, 1979), p.53-56.

Table 1 Experimental Results of the CDRRA Algorithm

circuit	L/T	Segs	overflow		S-SP's	X-SP's		Imp. (%)	Nets	overflow		S-NP's	X-NP's		Imp. (%)
			XC	NXC		XC	NXC			XC	NXC				
ibm1	4/5055	61310	0	0	113072	5233	17323	69.8	11753	0	0	48340678	5211	16681	68.8
ibm2	5/7318	109839	3	0	445515	28538	45667	37.5	18688	3	0	122232602	28091	44425	36.8
ibm3	5/10652	218784	23	0	776259	49827	85059	41.4	44681	23	0	698710308	49330	83483	40.9
ibm4	5/ 11231	248462	3	0	717704	36244	83724	56.7	50678	3	0	898885822	35833	81637	56.1
ibm5	5/14390	372000	4	0	1234360	80716	136888	41.0	64971	4	0	1477391812	79668	133512	40.3

L/T: No. of Layers/Tracks, Segs: No. of the net segments, XC: with crosstalk control, NXC: with no crosstalk control
S-SP's: No. of sensitive net segment pairs, X-SP's: No. of net segment pairs with crosstalk violations
S-NP's: No. of sensitive net pairs, X-NP's: No. of net pairs with crosstalk violations, Imp.: improvement ratio