An AI-based approach to Noise Reduction in Electronic Circuits

Parthasarathi Dasgupta
Management Information Systems Group
Indian Institute of Management Calcutta
Kolkata – 700 104
E-mail: partha@iimcal.ac.in

Kashi Nath Dey and Samar Sensarma
Department of Computer Science
University of Calcutta
Kolkata – 700 009
E-Mail: sssarma2001@yahoo.com

Abstract. A channel is a rectangular region of VLSI layout, between pair of adjacent blocks and is used for interconnecting the nets of signal, clock, and ground. With increased scaling of the chips, the coupling capacitance between the interconnect lines becomes predominant. This capacitance along with the increased interconnect lengths, clock frequencies and reduced supply voltages lead to noise in nets. A high-performance channel router needs to minimize noise. We apply the A* algorithm for a channel router to find a minimum-track solution with minimum noise. Empirically, the proposed algorithm yields an average of 37.5% noise reduction over a standard minimum-track routing solution, which is highly encouraging.

I. INTRODUCTION

Increased scaling of fabrication geometries and low voltage levels has increased the importance of crosstalk between adjacent wires in channel routing, particularly in high-speed designs [1, 2]. Reduced geometries tend to bring the interconnect lines in close proximity. The fringing capacitances thus increase, resulting in significantly increased noise due to crosstalk. Crosstalk affects a circuit in at least three ways: (i) inducing undesired signals in noisy environment, (ii) changing the signal delay due to a switching transition, and (iii) increasing the switching power dissipation. Crosstalk depends on the routing of the lines, and hence the routing algorithms take this into consideration. In this paper, we propose an Artificial Intelligence (AI) – based method for channel routing with the objective of minimizing crosstalk noise within given constraints.

A. Crosstalk in Channel Routing. Channel routing problem (CRP) [3] is the interconnection of a set of nets through a substrate. A channel (Figure 1) is a region with a pair of lists of terminals on its top and bottom boundaries respectively. Each terminal belongs to a net, and all the terminals of every net are to be connected. Generally, multiple layers of substrate and conductor connected by vias, are used. Traditional channel routing models include the HV, HVH and the VHV models [3]. Two graphs associated with CRP are the horizontal (HCG) and the vertical constraint graphs (VCG). The HCG indicates interference of routing of the trunks, and the VCG indicates the relative order of the trunks of different nets along channel height. Doglegs help to avoid cycles in the VCG. The CRP is NP-hard, and hence lacks optimal algorithm for all cases [3, 18].

Crosstalk distortion between two adjacent nets is determined by several factors like coupling capacitance between the nets, driving capacity of the nets, and the timing of the signals [4]. The coupling capacitance (C_c) provides a first order approximation for crosstalk [6]. Due to the reduced chip sizes and higher operating frequencies, the impact of crosstalk is growing fast. If ε is the dielectric constant of the substance between the two wires, w is the width of each conductor, l is the length of overlap of the conductors, and d is their separation, then coupling capacitance between the adjacent wires is given by : $C_c = \varepsilon . w. l / d$ (1)
Equation (1) shows that the overlap across two adjacent wires can be used to represent the coupling capacitance, and hence the crosstalk. CMP even without considering vertical constraints is NP-hard [15, 20].

The rest of the paper is organized as follows. Section II presents a brief survey of notable works in crosstalk minimization. In Section III, we discuss the model of the crosstalk problem considered in this paper. Section IV introduces the proposed AI-based method of channel routing with minimum crosstalk. Section V provides the empirical results. Finally, Section VI concludes the paper and discuses the scope for further work.

II. RELATED WORKS

CMP have been considerably studied in recent past. The method in [11] for the HVH model uses post-processing, and iteratively modifies the layout without increasing its area. A notable post-processing method based on wire perturbation is reported in [10]. The method in [9] discusses a channel router that starts with a solution produced by the left-edge algorithm (LEA) [3]. Each net is assigned a weight depending on its susceptibility to crosstalk, or its chance of generating crosstalk, and accordingly, the initial LEA solution is modified. [12] applies Genetic Algorithm for routing of Channels and Switchboxes under crosstalk constraints. Other notable works are reported in [4, 5, 7].

III. CROSSTALK MINIMIZATION PROBLEM

The proposed crosstalk minimization problem (CMP) is based on the following assumptions: (i) Vertical constraints are ignored, (ii) each net has only two terminals, and a net occupies exactly one track, and (iii) the total number of tracks available for routing in the channel is given.

Assumption (i) over-constrains the problem, but makes the proposed method applicable to HVH-model. Since the number of tracks in the CMP solution is bounded by the available tracks, the proposed method assigns the tracks conservatively. Our method outperforms LEA, and a method based on Travelling Salesman Problem (TSP) [21]. Thus, CMP is defined as: *For a given channel, find an assignment of nets into tracks with minimum crosstalk, such that the connection requirements by channel specification are preserved without violating the horizontal constraints.*

IV. AN A*-BASED METHOD FOR CROSSTALK MINIMIZATION

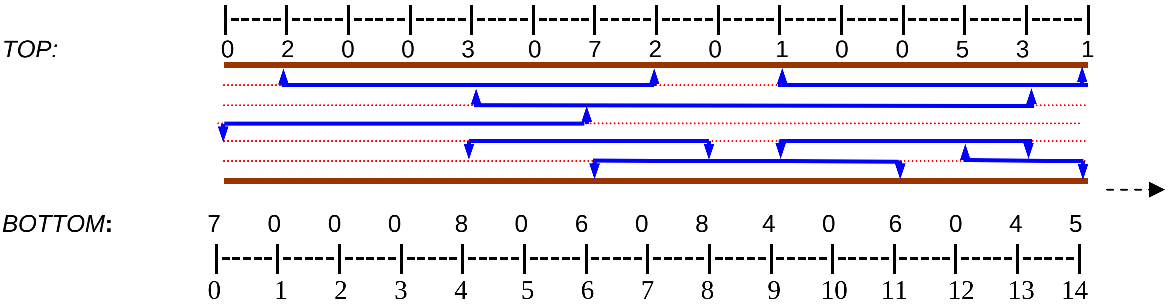


Figure 1

Net number:	1	2	3	4	5	6	7	8
start:	9	1	4	9	12	6	0	4
end:	14	7	13	13	14	11	6	8

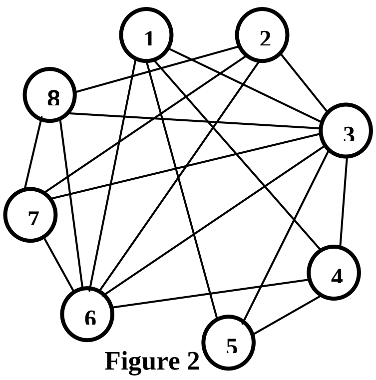


Figure 2

For the example CRP in Figure 1, let *start* and *end* be two lists containing the two ends of each net. Let *T, N* and *Column* denote respectively the total number of tracks, number of nets, and the number of columns. Here, $N = 8$, $T = 5$ and *Column* = 15, and *channel density* $d_{max} = 5$. A standard CRP algorithm for this example yields crosstalk = 16. We intend to obtain a minimum-crosstalk routing solution for the channel. Procedure *find_hcg()* constructs the *HCG* (Figure 2). Nets are assigned to the tracks in a conservative manner, and the crosstalk is used to guide the A* [17] search. Assignment of nets to tracks requires finding the *Maximal Independent Set (MIS)* of vertices in the HCG.

A. Finding Maximal Independent Set (MIS) from HCG. Procedure *find_crosstalk_nets(i, k)* finds the overlap between nets *i* and *k*. To obtain the *MIS*s from the *HCG,* we apply the largest-first sequential algorithm [13] for vertex colouring. As HCG is a perfect graph [14, 3] its vertices can be coloured in polynomial time. Let *vertex_colour( )* denote the corresponding procedure (Figure 3). For the *HCG* of Figure 2, the descending order of vertices is: 3,6,1,2,4,7,8,5, and the five *MIS*s obtained by *vertex_colour( )* are : {(3), (6,5), (1,2), (4,7), (8)}. Nets in one *MIS* are assigned to the same track. Once the *MIS*s are obtained, they are assigned to a specified number of tracks, one track at a time, so that the total crosstalk is minimum.

```
Procedure vertex_colour( )
begin
List vertices of HCG as v1,…,vn, s.t. degree(v1) ≥ degree(v2) ≥ … ≥ degree(vn)
List the available colours as 1, 2…n
 For i = 1,…,n, let Ci = {1,2,…,i} (* set of colours available for colouring vertex vi *)
 Set i = 1
 While i+1 ≤ n do
  Let ci be the first colour in Ci and assign this to vertex vi
  For each k with k > i and vk in HCG
   Set Ck = Ck – {ci} (* vk will not be given the same colour as vi *)
  i = i+1
 Record each vertex and its colour
end
```

Figure 3. Algorithm for finding MISs from a given HCG

B. Computing Crosstalk between two nets. A pair of nets will have a crosstalk distortion if and only if there exists an edge between the corresponding vertices in the *HCG* and the nets are in consecutive tracks. Let *max(a, b)* be the maximum of two numbers *a* and *b*. We call unit column span or unit overlap as unit crosstalk and calculate it as follows:

crosstalk(n_i,n_k) = *max*(0, (smallest value of the *end* points of two nets n_i and n_k – largest value of the *start* points of two nets n_i and n_k)) = *max*(0, end_{min} - $start_{max}$). Procedure *find_crosstalk_nets(n_i, n_k)* returns the crosstalk between nets n_i and n_k assigned to consecutive tracks. In Figure 1, crosstalk between the net pairs {n_6(6,11), n_4(9,13)} and {n_1(9,14), n_2(1,7)} are respectively *max*(0, (11 –9)) = 2, and *max*(0, (7 –9)) = 0.

C. Computing Crosstalk between two MISs. Crosstalk between two *MISs* is the sum of the crosstalks between all possible unique pairs of nets, taking one from each *MIS***.** The procedure *find_crosstalk_MIS(m_i,m_j)* (Figure 4) computes the total crosstalk between the *MIS*s m_i and m_j. For the channel of Figure 1, the five *MIS*s are $m_1(3)$, $m_2(6,5)$, $m_3(1,2)$, $m_4(4,7)$ and $m_5(8)$, and $M = 5$. Thus, *crosstalk(m2,m4)* = *find_crosstalk_nets(6,4)* + *find_crosstalk_nets(6,7)* + *find_crosstalk_nets(5,4)* + *find_crosstalk_nets(5,7)* = 2 + 0 + 1 + 0 = 3. In our proposed method, at each stage of search, we compute an estimate of the minimum crosstalk among the *MIS*s that are yet to be assigned.

```
Procedure find_ crosstalk_MIS(mi,mj)
begin
  crosstalk = 0;  ct = 0;
  for each nets in mi do
    neti ← a net in mi;
    for each nets in mj do
      netj ← a net in mj;   ct = find_crosstalk_nets(neti,netj);
      crosstalk = crosstalk + ct;
  return crosstalk;
end
```

Figure 4. Computing crosstalk between MISs

D. Computing Lower bound of crosstalk.

Theorem 1. The lower bound of crosstalk for the unassigned MISs is given by the sum of the crosstalks over the entire range of column spans, where a column span is the length between a pair of adjacent columns.
Proof. We skip the proof here for the sake of brevity.
For Figure 1, *Column* = 15 and maximum value of *column_span* is 14. Procedure *find_lower_bound*$(m_1,m_2,\ldots,m_n,n)$ computes the lower bound of crosstalk for the MISs $m_1,m_2,\ldots,m_n$ to be assigned to *n* tracks, and procedure *count_net*$(m_1, m_2,\ldots, m_n, column\_span)$ calculates the number of nets in the MISs $m_1,m_2,\ldots,m_n$ within a specified *column_span*. Thus, *lower_bound*$(m_1,m_3,m_4,3)$ = 0+ 0 + 0+ 0 + 2 + 2 + 0 + 0 + 0 + 2 + 2 + 2 + 2+ 0= 12.

E. The A*-based algorithm. The *CMP* will involve assignment of several *MIS*s to the tracks one by one. Consider a search tree, with *source node* for no assignment of *MIS*, and the goal node for assignment of all *MIS*s. A node at level *h* represents a set of *MIS*s assigned to tracks *1, 2, …, h*. To obtain an assignment with minimum crosstalk, we find the minimum-cost path from *Source* to *Goal*, with appropriately defined cost. Proposed algorithm AI_CMP (Figure 5) finds the *optimal solution path*. Each node of the tree has a cost = crosstalk *g()* obtained so far due to the assignment of *MIS*s + estimate *h()* of the minimum crosstalk between the remaining *MIS*s. This implies the following result.
Lemma 1. The lower bound of crosstalk *h()* is an *admissible heuristic*.
Theorem 2[17] A*returns an optimal solution with an admissible heuristic.
Lemma 1 and Theorem 2 leads to the following result.
Corollary. For sufficient number of given tracks, the algorithm *AI_CMP* terminates, and always produces a solution with minimum crosstalk.
In the following description of the algorithm - *OPEN* is a list containing the un-expanded nodes, *CLOSED* is a list containing the already-expanded nodes, *g(n)* is crosstalk from the *Source* to node *n*, *h(n)* is the estimated crosstalk from *n* to *Goal*, and $f(n) = h(n) + g(n)$ is the total estimated crosstalk from *Source* to *Goal* passing through node *n*.

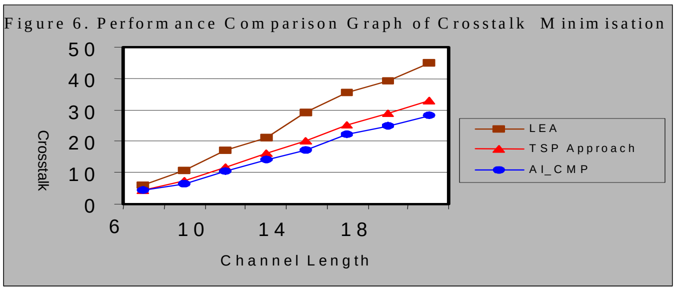
```
Procedure AI_CMP( )
1.  Find the HCG of the channel using Find_hcg();
2.  Find the MISs of the nets in the channel using vertex_colour().
3.  g(Source)=0; h(Source)=0; f(Source)=0;
4.  Put Source in OPEN; Found = False;
5.   while OPEN not empty and not(Found) do
6.      Select a node n from OPEN with minimum f() value;  Remove n from OPEN and put n in CLOSED;
7.      If n is a Goal node then Found = TRUE; else
8.       Get the list of MISs assigned so far from Source to the node n. Let, this MIS_assigned.
9.       Get the list (MIS_unassigned_n ) of MISs not assigned so far from Source to the node n.
10.      Expand n (* assign each successor MIS to the next track in MIS_not_assigned. *)
11.      For each immediate successor ni of n do
12.         g=g(n)+crosstalk(n,ni); find MIS_not_assigned_ni;
13.         h(ni)=find_lower_bound(MIS_not_assigned); g(ni)=g; f(ni)=g(ni)+h(ni);
14.         Put ni in OPEN; Direct backward pointer from ni to n;
15.  Find the last MIS (MIS_last ) assigned to the last track. Let MIS_prev be the last MIS assigned in the Goal node n;
16.  Find last = crosstalk(MIS_prev,MIS_last); Calculate crosstalk = f(n)+ last;
17.  Output crosstalk and solution path (* sequence of MISs assigned *) by tracing back  pointers;
```

Figure 5. The proposed A*-based crosstalk minimization algorithm

F. Time Complexity. Complexity of A*-based algorithm is obtained from its empirical performance. However, time complexities can be estimated for the heuristic computation, and node expansion. The worst-case complexity for computing *g()* for each node is $O(n^2)$. If *c* be the channel length, then the time complexity for calculating *h()*-value for each node expanded is $O(cn)$, in worst case. Thus a single node generation in *AI_CMP* requires time $O(n^2)+O(cn)$.

V. EMPIRICAL OBSERVATIONS.

AI_CMP and the LEA were implemented in C on an Intel Pentium III PC with clock speed 1 GHz. Random channels were generated such that (i) total number of nets is equal to the number of columns in the channel minus one, i.e. N =



Column - 1, to restrict the search space for AI_CMP, and (ii) the maximum number of tracks (T) is equal to the total number (N) of nets. For each channel length, 40 random instances were generated, and for each of them, crosstalk in routing solutions is obtained using *LEA* (initial solution) and *AI_CMP*. Average crosstalk was computed for each of these algorithms for a fixed channel length. The relative performances show that on an average, overall reduction in crosstalk is 37.5% for *AI_CMP*. Figure 6 shows the variation of crosstalk with increasing channel length for *LEA*, a TSP-based approach [21] and *AI_CMP*. Results for some benchmark channels are shown in Table 1. Multi-terminal nets were considered as two-terminal nets with the two terminals at extremities. For the CMP router in [9], for benchmark *Xtra* minimum-crosstalk solution used 4 tracks whereas the initial solution used 3 tracks. Our algorithm uses 3 tracks, but with larger crosstalk reduction than [9].

Benchmark Channels	*No. of tracks*	*Crosstalk*					*CPU time (milliseconds)*
		LEA	*AI_CMP*	*% Reduction in AI_CMP*	*Nodes expanded*	*Nodes generated*	
Burstein[18]	4	13	8	38.46	8	22	5
Termcon[18]	4	4	2	50.00	6	18	5
Xtra[9]	3	6	2(4)	66.67	3	7	0
RKPC9[20]	6	76	65	14.47	49	178	5

Table 1. Performance of AI_CMP for some Benchmark channels

VI. CONCLUSIONS AND FUTURE SCOPE

This paper proposes an AI-based algorithm for crosstalk minimization in channel routing. The crosstalk for the proposed algorithm has been empirically compared with that obtained by LEA. The results are quite encouraging. The algorithm can be extended to multi-terminal nets by splitting such nets into several two-terminal nets. Also, it can be used for HVH model. Crosstalk may arise even without overlapping tracks. Future works may involve such cases.

REFERENCES

[1] H. B. Bakoglu, "Circuits, Interconnections, and Packaging for VLSI", Addison-Wesley Publishing Co., 1990.
[2] The International Technology Roadmap for Semiconductors, 2001 edition, International Sematech, Austin, Texas, December 2001, http://public.itrs.net/.
[3] N. A. Sherwani, "Algorithms for VLSI Physical Design Automation", 3rd Ed., Kluwer Acad.Pub., Boston, 1999.
[4] A. Vittal and M. Marek-Sadowska, "Crosstalk Reduction for VLSI", IEEE Trans. Computer-Aided Design, Vol. 16, No. 3, pp. 290-298, 1997.
[5] S. Thakur, K. –Y. Chao and D. F. Wong, "An Optimal Layer Assignment Algorithm for Minimizing Crosstalk for Three-Layer VHV Channel Routing", Proc. IEEE International Symp.on Circuits & Systems, pp. 207-210, 1995.
[6] T. Gao and C. L. Liu, "Minimum Crosstalk Channel Routing", IEEE Trans. CAD, Vol.15, No.5, pp. 465-474, May 1996.
[7] M. Hashimoto, M. Takahashi, H. Onodera, "Crosstalk Noise Optimization by Post-Layout Transistor Sizing", International Symposium on Physical Design, 2002, pp. 126 – 130.
[8] R. M. Smey, B. Swartz, and P. H. Madden, "Crosstalk Reduction in Area Routing", Design Automation and Test in Europe (DATE), 2003.
[9] N. Bird, E. Miller, P. Pfeiffer and S. Vemuru, "Channel Routing with Crosstalk Consideration", Proceedings of the International Conference on VLSI, June 23 - 26, 2003, Las Vegas, Nevada, USA. CSREA Press 2003, pp. 119-124.
[10] P. Saxena and C. L. Liu, "Crosstalk Minimization using Wire Perturbation", Design Autom. Conference, 1999.
[11] Z. Chen and I. Koren, "Crosstalk Minimization in Three-Layer HVH Channel Routing", Proceedings of the Workshop on Defect and Fault-Tolerance in VLSI Systems, pp. 1997, Paris, pp. 38-43, 1997.
[12] J. Lienig, "Channel and Switchbox Routing with Minimized Crosstalk- A Parallel Genetic Algorithm Approach", Proc. of the Tenth International Conference on VLSI Design, Hyderabad, pp. 27-31, 1997.
[13] J. Clark and D. A. Holton, "A First Look at Graph Theory". World Scientific, New Jersey, 1991.
[14] M. C. Golumbic, "Algorithmic Graph Theory and Perfect Graphs". Academic Press, New York, 1980.
[15] H. Zhou and D. F. Wong, "Global Routing with Crosstalk Constraints", 35th Design Automation Conference, San Francisco, 1998.
[16] T. H. Cormen, C.E. Leiserson & R.L. Rivest, "Introduction to Algorithms", Prentice-Hall of India, 1999.
[17] Nils J. Nilsson, "Artificial Intelligence: A new synthesis", Stanford Univ., Morgan Kauffmann Publishers, 1998.
[18] R. K. Pal, "Multi-Layer Channel routing: complexity and algorithms", Narosa Publishing House, 2000.
[19] M. R. Garey and D. S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness". San Francisco, W. H. Freeman, 1979.
[20] R. K. Pal, A. Pal and A. K. Datta, "Crosstalk Minimization Problem in Two-layer Channel Routing is NP-hard", University of Calcutta, Manuscript, 2004.
[21] P. Dasgupta, S. Roy, D. Das and S. Sensarma, "A Greedy Technique for Minimizing Crosstalk in Channel Routing", Manuscript, 2004.