An ECO Algorithm for Eliminating Crosstalk Violations

Hua Xiang[1,3] Kai-Yuan Chao[2] Martin D.F. Wong[1]
xiangh@cadence.com kchao@ichips.intel.com mdfwong@uiuc.edu
[1] Coordinated Science Lab, University of Illinois at Urbana-Champaign, IL 61801
[2] Intel Corporation, Hillsboro, Oregon 97124
[3] Cadence Design Systems Inc, San Jose, CA 95134

ABSTRACT

ECO changes are almost inevitable in late stages of a design process. Based on an existing design, incremental change is favored since it can avoid considerable efforts of re-doing the whole process and can minimize the disturbance on the existing converged design. In this paper, we address the CVE (Crosstalk Violation Elimination) problem. Due to the changes in a multiple layer routing design, the total capacitive crosstalk on some signal wire segments on a layer may be larger than their allowable bounds after post-layout timing/noise analysis. The target is to find a new routing solution without crosstalk violations under certain constraints which help to keep the new design close to the original one. We propose a two-stage algorithm to solve CVE problems, and present optimization strategies to speed up the execution. Experimental results demonstrate the efficiency and effectiveness of our algorithm.

Categories & Subject Descriptors: B.7.2 [Integrated Circuits]: Design Aids - Placement and routing; J.6 [Computer Applications]: Computer-Aided Engineering - Computer-aided design
General Terms: Algorithms, Design, Theory
Keywords: ECO, Crosstalk, Routing

1. INTRODUCTION

ECO due to frequency push and design/market requirement change is very important for producing high-end and high-volume main stream products in semiconductor industry. It is a highly constrained design optimization based on an existing design with tight design scheduling due to time-to-market consideration. However, any changes on existing routing design may cause design rule violations and it is necessary to develop efficient and graceful algorithms to resolve these violations.

In this paper, we address the problem of eliminating crosstalk violations to a given routing design. One possible application of our algorithm is as follows. Recently [4] addressed an ECO problem which solves design spacing rule violations between power rails and signal wires due to the re-design of power rails on the top layer of a multiple layer routing region. This problem is usually caused by design changes in power delivery or package - both can be requested from performance, noise, reliability, or marketing considerations. However, in that problem, the spacing between any two signal wires only need to satisfy the minimum spacing requirement and this may cause large capacitive crosstalk for some sensitive signal wires. Therefore CVE algorithm can be used to eliminate crosstalk violations upon the output of the previous ECO wire legalization problem.

In this paper, we propose a two stage CVE (Crosstalk Violation Elimination) algorithm to eliminate crosstalk violations for a given routing design as well as minimizing the total deviation. The first step FCVE processes signal wire segments on the layer L one by one and tries to find a clean routing solution satisfying all constraints. Then in the second stage SCVE, we make efforts to minimize the total deviation based on the shortest path algorithm. Experimental results demonstrate that our approach is efficient and effective.

*This work was partially supported by the National Science Foundation under grants CCR-0244236 and CCR-0306244.

ISPD'04, April 18-21, 2004, Phoenix, Arizona, USA.

2. CROSSTALK VIOLATION ELIMINATION

Given a routing solution S with N signal nets, there are P power rails on layer L. Without loss of generality, we assume the metal layer L is used for horizontal tracks, and the layers below and above L, which are $\hat{L}$ and $\tilde{L}$ respectively, are used for vertical tracks. Any changes on L may lead to changes on other layers. However, the changes should not propagate to all layers. Therefore, we confine the changes to L, $\hat{L}$ and $\tilde{L}$, and treat all connections to the three layers from other layers as fixed pins.

For convenience, let the coordinate of the left bottom corner of the routing region be $(0,0)$. Suppose s is the half minimum wire separation of a metal layer. For a horizontal segment, it can be represented by (x_1,x_2,y,w,c,d) where (x_1,y) and (x_2,y) are the end point coordinates of the center line $(x_1 < x_2)$, and w is the half-width of the segment, c is called crosstalk threshold, i.e., the total capacitive crosstalk to the segment should not exceed this bound, and d is allowable deviation bound, i.e., when the segment moves up/down, its new position $(x_1,x_2,\bar{y},w,c,d)$ should satisfy $|\bar{y}-y| \leq d$. Similarly, a vertical segment can be represented as (y_1,y_2,x,w,c,d). Sometimes, we can simplify the representation. For example, a horizontal segment can be represented by (x_1,x_2,y) if we do not care other factors.

Since the crosstalk on some sensitive segments in S exceeds the given bounds, the target is to modify the existing routing solution S so that the new routing solution $\underline{S}$ is a clean routing solution which satisfies the following constraints:

1. The power rails P on L are not changed.

2. Horizontal signal wire segments on L can only move up/down, i.e., the x-coordinates of the two end points of the segment keep unchanged.

3. The total crosstalk on a wire segment should not exceed its capacitive crosstalk threshold c, i.e., the total crosstalk from its neighbor wires should not exceed the bound.

   Since this is an ECO task for post-layout converged design, threshold for capacitive crosstalk can be derived after timing analysis for ECO area by calculation or heuristics for crosstalk re-budgeting/specification.

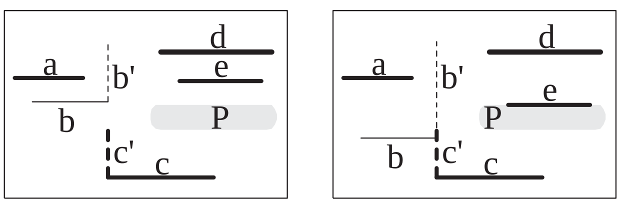
4. The relative positions of two segments on all layers should not be changed.

   For example, for any two horizontal signal wire segments on one layer (x_1, x_2, y) and (x_1', x_2', y') (assume $y > y'$), their new positions are $(x_1, x_2, \overline{y})$ and $(x_1', x_2', \overline{y}')$ respectively. If $(x_1 - s, x_2 + s) \cap (x_1' - s, x_2' + s) \neq \phi$, $\overline{y} > \overline{y}'$ must hold. Similar requirements for vertical segments. This property is called "order consistency".

5. The difference between the new position of a wire segment and its old location should not exceed its allowable deviation bound d.

   d is defined to constrain that one segment does not derive too much from its original position. At the same time, it helps to prevent introducing new crosstalk violations to other layers. When horizontal segments on L are changed, the length of vertical segments on $\hat{L}$ or $\tilde{L}$ may also be changed. However, the length change is no more than $2d$ since each vertical segment connects to at most two horizontal segments on L. Then the crosstalk introduced by length increase is also limited. Therefore, by setting appropriate deviation bounds, new crosstalk violations on layer $\hat{L}$ or $\tilde{L}$ can be avoided.

Although the CVE problem deals with crosstalk violations on one layer, it can be applied layer by layer to resolve violations on all layers to a given multiple layer routing design.



(a) (b)

Figure 1: (a) A routing solution with 5 horizontal signal wire segments and 1 power rail on L, and 2 vertical signal wire segments on $\hat{L}$; (b) A routing solution with overlap violations

As we notice that, once a signal wire segment is moved, the total capacitive crosstalk on both this segment and its neighbor segments may be changed. At the same time, design spacing rule violations must be avoided. For convenience, if the spacing between two segments is less than the minimum spacing requirement, we say the two segments overlap. Figure 1 (a) gives a routing solution with five horizontal signal wire segments, two vertical wire segments and one power rail. Suppose segments b and e violate the capacitive crosstalk requirement, i.e., the total crosstalk on b and e exceeds defined thresholds. As illustrated in Figure 1 (b), if e is moved down, it overlaps with the power rail P. Also, if b is moved down, vertical overlap between b' and c' on $\hat{L}$ is introduced.

3. PRELIMINARIES

3.1 FP-Range

If we arbitrarily move one horizontal signal wire segment up or down, not only overlaps between horizontal segments on L, but also vertical overlaps on $\hat{L}$ or $\tilde{L}$ may be introduced as illustrated in Figure 1 (b). Therefore, similar to [4], FP-Range is introduced and if horizontal signal wire segments move within the range, no vertical wire separation violations are introduced.

Suppose the wire separation requirement is $2s$, and W and H are the width and height of the routing region respectively. A horizontal wire segment $R = (x_1, x_2, y_r)$ on L belongs to net n_r. Its two end points are $r_1 = (x_1, y_r)$ and $r_2 = (x_2, y_r)$, and they are connected to layer L' and L'' respectively. L' (L'') is $\hat{L}$ or $\tilde{L}$. Then calculate two pin sets $\tilde{P}$ and $\tilde{Q}$. Let $\tilde{P}$ be the set of fixed pins on L and L' whose x-coordinates fall in $(x_1 - 2s, x_1 + 2s)$ and do not belong to net n_r, and $\tilde{Q}$ be the set of fixed pins on L and L'' whose x-coordinates fall in $(x_2 - 2s, x_2 + 2s)$ and do not belong to net n_r. Let $U = min\{\{y - 2s | y \in \tilde{P} \cup \tilde{Q} \wedge y \geq y_r\} \bigcup \{H - 2s\}\}$ and $V = max\{\{y + 2s | y \in \tilde{P} \cup \tilde{Q} \wedge y \leq y_r\} \bigcup \{2s\}\}$. The range $[V, U]$ is called "FP-Range". Figure 2 shows the FP-Range of a horizontal segment R.

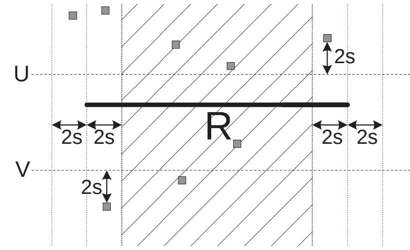


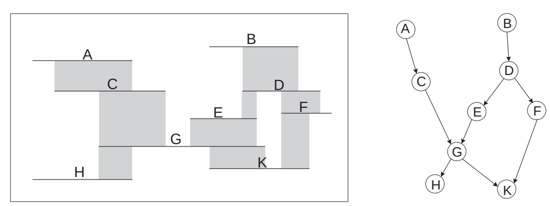
Figure 2: FP-Range illustration. Tiny squares are fixed pins.

Then we have the following theorem. The proof is similar to [4] and it is omitted here.

THEOREM 1. *If all horizontal segments on layer L move up/down within their FP-Ranges $[V, U]$ and satisfy horizontal wire separation requirement and order consistency, the new routing solution has no vertical wire separation violations.*

3.2 Consistency Graph

An important property of CVE problem is to keep "order consistency". Given any two horizontal segments $A = (x_1, x_2, y)$ and $B = (x_1', x_2', y')$, if $(x_1 - s, x_2 + s) \cap (x_1' - s, x_2' + s) \neq \phi$ ($2s$ is the wire separation requirement) and no segments fall in the region with left bottom corner $(min\{x_1, x_1'\}, min\{y, y'\})$ and right upper corner $(max\{x_2, x_2'\}, max\{y, y'\})$, we define segments A and B adjacent segments. According to order consistency, we can construct "consistency graph": each horizontal signal wire segment is represented by a node; for any two adjacent segments A and B, if A is above B, one edge (A, B) is added. Figure 3 illustrates an example.



(a) (b)

Figure 3: (a) A routing solution of signal wires on L; (b) Consistency graph

3.3 Crosstalk Model

In general, each segment has coupling effect to all other segments. However, the coupling capacitance decreases dynamically if the segment is out of the neighborhood of the other segment [5, 3]. Therefore, we only consider the capacitive crosstalk between two neighboring parallel wires and suppose the neighborhood distance is $D = \gamma \cdot 2s$, $(0 < \gamma < 2)$. Then the capacitive crosstalk between two segments can be expressed by the following formula:

$$c = \begin{cases} \alpha \cdot \frac{l}{t^2} & t \leq D \\ 0 & t > D \end{cases}$$

where α is the coupling parameter, l is the coupling length, and t is the distance between two segments. Furthermore, power rails act as shields and do not cause crosstalk to their adjacent segments.

4. CVE ALGORITHM

To solve the CVE problem, we develop a two-stage algorithm. The first stage FCVE processes signal wire segments on L one by one and tries to find a clean routing solution satisfying all constraints. Then in the second stage SCVE, efforts are made to minimize the total deviation based on the shortest path algorithm.

4.1 FCVE Algorithm

For convenience, for any two nodes A and B in G, if there is a path from A to B, we say A is B's parent, and B is A's child.

The main idea of FCVE algorithm is as follows: each time, select the nodes which have no parent nodes and try to move them to their highest available positions. These positions are their new locations. Then remove these nodes from the graph. Repeat this process until no nodes are left.

For each segment, its available position is related to its FP-range, allowable deviation bound, crosstalk threshold, the distribution of power rails and the positions of its parents. Let the wire separation requirement be $2s$. Suppose segment $A = (x_1, x_2, y, w, c, d)$ has an FP-range $[V, U]$. Also A records a value $Ubound$. $Ubound = min\{y_p - 2s - w_p | y_p$ is the y-coordinate of an A's parent node and w_p is its half width$\}$. Then if A moves in the range $[0, Ubound - w]$, the order consistency is guaranteed. Let $[\bar{V}, \bar{U}] = [V, U] \cap [y - d, y + d] \cap [0, Ubound - w]$. Check tracks t starting from $\bar{U}$. If t is not occupied by any power rails and no crosstalk violations are introduced to A's parents and itself if A is put at track t, t is assigned as A's new position. Otherwise, check the next track below t. Repeat this process until a feasible position is found or the track goes beyond $\bar{V}$. The latter case means no feasible solution is found. Once the position of A is decided, the crosstalk bounds of A and A's parents have to be adjusted accordingly, i.e., minus the crosstalk between A and its parent from the crosstalk bounds of A and its parent.

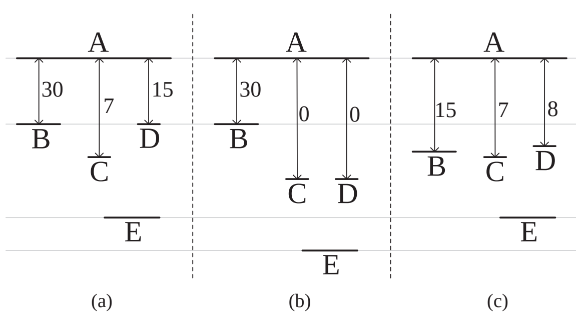


Figure 4: (a) B**,** C **and** D **are the children of** A**. The position of** A **is fixed. All segments have a crosstalk bound** 30**. The length ratio of** B**,** C **and** D **is** $2:1:1$**. (b)** B **is first selected and put to its highest available position. But** C **and** D **have to be placed lower. (c) A solution according to our approach.**

Furthermore, if one segment has several children, then the children selected first always have higher priority. For example, in Figure 4, suppose the position of A has been fixed. B, C and D are the children of A. The coupling length ratio of B, C and D is $2:1:1$. The crosstalk bound of all segments is 30. The numbers in the figure indicate the capacitive crosstalk if the segment is placed at their highest available positions. Suppose B is first selected and it is placed as Figure 4 (b). Then the crosstalk bound of A is reduced to 0. Therefore C and D have to be placed lower, which pushes E down too. In order to avoid one segment consuming all or most of the crosstalk budget, we use the following approach. Suppose a segment R is fixed and its crosstalk bound is c_r. Also its total coupling length with all of its unfixed children is l_r. Let $d_r = \min\{D, \sqrt{\alpha \cdot (l_r / c_r)}\}$. Then the distance between R and its first selected child T must be no less than d_r. Once T is fixed, c_r is adjusted accordingly, i.e., minus the crosstalk between R and T from c_r. Then the new c_r is used for R's other children in the same way. Figure 4 (c) shows a solution with this approach. According to the crosstalk budget, the crosstalk between A and B, A and C, A and D should be 15, 7.5 and 7.5 respectively. Suppose segment B is first selected, then the crosstalk upper bound of A is reduced to 15. Since the lengths of C and D are the same, C and D get a crosstalk budget 7.5. And the new position of C can be calculated. However, the highest available position of C is lower than the calculated position. Therefore, C is put on its highest available position and the crosstalk to A is 7. Finally, D takes all of the crosstalk budget.

In FCVE, we always try to put a horizontal segment upwards. This leaves more room for other segments since once one segment is processed, its location is fixed and other segments below it cannot take the places above it. If we arbitrarily assign a segment to one of its available positions, some segments may have no place to put.

As we notice that, even if there are no crosstalk violations in the given input routing design, segments may still be moved in the above procedure. However, our targets are not only to eliminate crosstalk violations, but also to minimize the total deviation. Therefore, we start with a zero allowable deviation bound and each time increase the bound by a certain percentage. For each deviation value, we calculate the positions of all segments according to the above procedure. Repeat this process until a feasible solution is found or the deviation bound exceeds the pre-defined value. For the latter case, no feasible solution is found.

4.2 SCVE

If FCVE returns a solution, then the solution must be a feasible solution satisfying all of the constraints. However, FCVE tends to place segments to their "highest" available positions while some segments do not need to deviate so much from their original positions. In this section, we first consider a special case of CVE problem (CVEP) and propose an exact polynomial-time algorithm to decide wire segment positions with minimum total deviation under all constraints. Then by applying this algorithm repeatedly on the output of FCVE, we can greatly reduce the total deviation.

PROBLEM CVEP is a special case of CVE problem when all horizontal segments on layer L are placed in a line, i.e., the corresponding consistency graph is a path.

Figure 5 (a) shows an example. There are 4 signal wire segments and 1 power rail. (b) is its consistency graph and it is a single path from node A_4 to A_1. For convenience, the segments in a CVEP problem are indexed as A_1, ..., A_n from bottom to top.

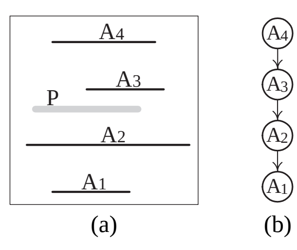


Figure 5: (a) A CVEP problem. There are 4 **signal wire segments** A_1, A_2, A_3 **and** A_4**, and** 1 **power rail** P**. (b) The consistency graph is a path.**

To solve the CVEP problem, we first construct a "Segment Position" (SP) graph, and then apply the shortest path algorithm to get the solution. The SP graph is constructed in two steps. The first step graph (FSP) $G = (V, E)$ is formed as follows.

Nodes: Since the allowable deviation of segment A_i is d_i, totally there are $2d_i + 1$ possible positions for A_i. Let node set $V' = \{v_i^j | i \in [1, n], j \in [-d_i, d_i]\}$ representing possible positions of A_i,

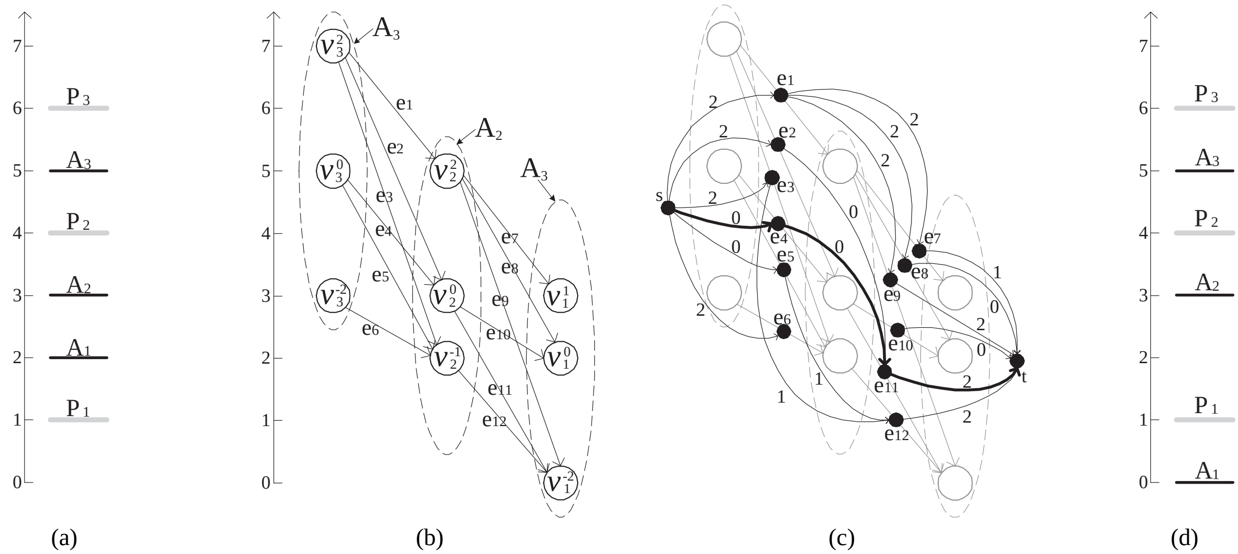


Figure 6: (a) A CVEP problem. There are 3 signal wire segments A_1, A_2, and A_3, and 3 power rails. The allowable deviation of A_1, A_2 and A_3 is 2, and their crosstalk upper bound is 0. A_1 and A_2 violate the crosstalk requirement. (b) FSP graph G of the CVEP problem. (c) SP graph $\bar{G}$ of the CVEP problem. (d) A feasible solution to the CVEP problem.

i.e., v_i^j refers to the position $y_i + j$. For convenience, we call v_i^j a node of A_i. Also for any possible position, if it is occupied by a power rail or it is outside A_i's FP-Range, then A_i cannot put there. Suppose nodes corresponding to this kind of positions form the set V''. $V = V' - V''$.

Edge: $E = \{(v_i^j, v_{i+1}^k) \mid v_{i+1}^k - w_{i+1} - (v_i^j + w_i) \geq 2s, i \in [1, n-1], j \in [-d_i, d_i], k \in [-d_{i+1}, d_{i+1}], v_i^j \in V, v_{i+1}^k \in V\}$. For each node of A_i, it is connected to the nodes of A_{i+1} such that the distance between two nodes satisfies the minimum spacing requirement.

Cost: Each edge (v_i^j, v_{i+1}^k) is assigned a cost which is the capacitive crosstalk between A_i and A_{i+1} supposing the two segments are placed at v_i^j and v_{i+1}^k respectively.

Figure 6 shows an example. (a) is a CVEP problem with 3 signal wire segments A_1, A_2, A_3 and 3 power rails. For simplicity, suppose all wires have the same length, and the deviation bounds of signal wire segments are all 2. Also the crosstalk thresholds are all 0, i.e., the distance between any two signal wire segments must be larger than 1 unit. In Figure 6 (a), since segments A_1 and A_2 are adjacent to each other, the capacitive crosstalk between them exceeds the crosstalk bounds of both A_1 and A_2.

Figure 6 (b) shows the corresponding CVEP graph G for (a). Due to the overlap with power rails, the available positions of each segment are only 3 and they are represented by 3 nodes respectively. The costs of all edges are 0 except two edges e_6 and e_{10}.

In FSP graph, the allowable deviation bound is reflected by nodes, and the edge cost records the crosstalk between two segments. However, the crosstalk constraint is not included. Therefore, based on FSP graph, we derive the SP graph $\bar{G} = (\bar{V}, \bar{E})$ so that the shortest path algorithm can be applied to find the solution. $\bar{G}$ is formed as follows.

1. Nodes: Each edge in G is represented by a node. For convenience, an edge (u, v) in FSP graph also refers to a node in SP graph. Also two nodes s and t are added representing the starting and ending nodes respectively.

2. Edges: For any two edges (v_i^j, v_{i+1}^k) and (v_{i+1}^k, v_{i+2}^l) in FSP graph, if the total cost of the two edges is less than c_{i+1} which is the crosstalk bound of segment A_{i+1}, an edge is added between the two corresponding nodes in $\bar{G}$. Also connect s to all of the nodes corresponding to the edges related A_1 in FSP graph, and all of the nodes corresponding to the edges related to A_n are connected to t.

3. Cost: If edge $\bar{e}$ connects two nodes (v_i^j, v_{i+1}^k) and (v_{i+1}^k, v_{i+2}^l), the cost of $\bar{e}$ is $|k|$ (i.e., the deviation of A_{i+1}); if edge $\bar{e}$ starts from s, i.e., $\bar{e}$ connects s and (v_1^j, v_2^k), the cost is $|j|$; if edge $\bar{e}$ ends at t, i.e., $\bar{e}$ connects (v_{n-1}^j, v_n^k) and t, the cost is $|k|$.

Figure 6 (c) illustrates the SP graph $\bar{G}$ for the given CVEP problem. Each edge in G is represented by a node in $\bar{G}$. For edges e_6 and e_{10} in G, since their cost is 1, edges (e_6, e_{12}), (e_2, e_{10}) and (e_4, e_{10}) are not included in $\bar{G}$. Based on $\bar{G}$, we apply the shortest path algorithm to find the shortest path from s to t. In Figure 6 (c), the shortest path is indicated by thick curves. It is easy to derive a CVEP solution as shown in Figure 6 (d).

Suppose totally there are n wire segments and M is the max allowable deviation. The number of nodes in FSP graph is $O(n \cdot M)$. For each node in FSP graph, it connects to at most M nodes. Therefore, the number of nodes and edges in SP graph $\bar{G}$ are $O(n \cdot M^2)$ and $O(n \cdot M^3)$ respectively. Since $\bar{G}$ is a directed acylic graph, the shortest path algorithm can be accomplished in $O(|\bar{V}| + |\bar{E}|)$ [1, 2], i.e., $O(n \cdot M^3)$.

We now summarize the CVEP algorithm as follows.

Algorithm CVEP(P)
1. Construct SP graph $\bar{G}$ for the input path P;
2. Apply the shortest path algorithm on $\bar{G}$;
3. Derive the solution to the given CVEP problem

The construction of SP graph $\bar{G}$ takes $O(n \cdot M^3)$ and the derivation from a shortest path in $\bar{G}$ to a CVEP solution takes $O(n)$. Therefore, CVEP algorithm can solve CVEP problems in $O(n \cdot M^3)$. Furthermore, the algorithm guarantees to return a feasible solution with minimum deviation as long as there is a solution to the given CVEP problem.

Based on CVEP algorithm, we have the following SCVE algorithm. SCVE algorithm performs as the second stage of CVE algorithm since its input is the output of FCVE algorithm, which is a feasible solution to the given CVE problem. The target of SCVE is to reduce the total deviation.

Based on the consistency graph, each time we select a path and apply CVEP algorithm to find the optimal solution corresponding to the selected path. Once a path is processed, all nodes along the path are marked “Processed” and their positions are not changed any more. Since FCVE algorithm traverses a consistency graph from top to bottom, and many segments may put on a position higher than their original positions, SCVE algorithm selects paths from the bottom of a consistency graph. For each path, the first node u must either have no child or all of its children are marked. Then trace up to its parents. If one of its parents p has u as the only unmarked child, p is selected. Continue this procedure until no nodes satisfy the selection rule. Once a path is selected, we treat all other nodes unchanged and apply the CVEP algorithm. Note that the crosstalk of each possible position of a signal wire segment is also affected by other segments which are not incident on the path.

Algorithm SCVE()
1. Set all nodes in the consistency graph “UnProcessed”
2. While($\exists$ “UnProcessed” nodes)
3. Select a path P from the consistency graph
4. Apply CVEP algorithm on P
5. Mark all nodes on P as “Processed”

5. OPTIMIZATION

CVEP algorithm is the kernel part of SCVE algorithm. However, the number of possible positions of wire segments may be quite large and it makes SP graph include a lot of nodes and edges, which requires not only much memory but also long running time. In order to speed up the execution, we develop the following optimization strategies.

5.1 Node Clustering

When the deviation of a wire segment is large, the corresponding FSP graph and SP graph must include a large number of nodes. In order to facilitate the process of huge CVEP problems, we propose the following node clustering method to speed up the computation.

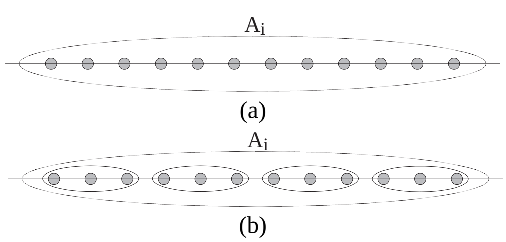


Figure 7: (a) A_i is a wire segment and it has 12 available positions. (b) Every three nodes are clustered as a “super-node”.

For any wire segment A_i, suppose the number of its possible positions is M. Then by grouping neighbor positions together, we can greatly reduce the number of nodes in FSP graph, consequently reduce the size of SP graph. Once several nodes are grouped together, we can use the average coordinate as the location of the new “super-node”. Figure 7 illustrates an example. A_i includes 12 feasible positions. When clustering 3 nodes as a “super-node”, there are only 4 “super-nodes”. Accordingly, the size of SP graph can be greatly reduced.

5.2 Edge Omitting

The construction of SP graph $\bar{G}$ is based on FSP graph G. During the transformation from FSP graph to SP graph, if we know that some edges will not appear in the final solution, then these edges can be omitted in SP graph. Therefore, the target of this optimization strategy is to identify this kind of edges.

Suppose a path $P = (A_1, ..., A_n)$ is the input of a CVEP problem, where $A_i (i = 1, ..., n)$ is a wire segment. Let $A_i = (x_1^i, x_2^i, y_i, w_i, c_i, d_i)$, and its FP-range be $[V_i, U_i]$. For convenience, we call a position is a feasible position of A_i if it is not occupied by any power rail, and its y-coordinate falls in $[V_i, U_i] \cap [y_i - d_i, y_i + d_i]$. For a feasible position p of A_{i-1}, suppose there is one edge e connecting to p either from s or a feasible position of A_{i-2} as illustrated in Figure 8 (a). In (a), A_i includes 7 feasible positions. e connects to p and p connects to all feasible positions of A_i. Based on this FSP graph, the corresponding SP graph is Figure 8 (b), assuming (e, e_i) $(i = 1, ..., 7)$ satisfies the crosstalk constraint. However, in some cases, some of these edges may not be needed.

Suppose q is the lowest feasible position of A_{i+1}. Let $B_u = \min\{q - 2s - w_i - w_{i+1}, q - D - w_i - w_{i+1}\}$, where $2s$ is the minimum spacing between two segments and if the distance of two segments is larger than D, there is no crosstalk between the two segments. Also let $B_l = \max\{p + 2s + w_i + w_{i-1}, p + D + w_i + w_{i+1}\}$. Then we have the following cases.

Case 1. $B_l \le y_i \le B_u$

Let $r = \min\{y_i - B_l, B_u - y_i\}$. Start from y_i, and search within $[y_i - r, y_i + r]$. If y_i is occupied by power rails, check $y_i - 1$, $y_i + 1$, $y_i - 2$, $y_i + 2$ ... until a feasible position u is found or it is out of the range. If u is found, then only one edge is needed in the SP graph, i.e., connecting the two nodes corresponding to e and (p, u) in the FSP graph. In Figure 8 (Case 1), a feasible position u is found in the range and only one edge (e, e_5) is needed in SP graph.

Consider other feasible positions v of A_i. Given an optimal solution S of a CVEP problem, suppose positions p, v and w (w is a feasible position of A_{i+1}) are selected for A_{i-1}, A_i and A_{i+1} respectively. Then p, u and w are also feasible positions of the three wire segments since the crosstalk of (p, u) and (u, w) is zero. However, u is the closest feasible position to y_i and it has the least deviation among all feasible positions of A_i. Therefore, a solution with p, u and w as the positions of A_{i-1}, A_i and A_{i+1} should have less deviation. But this contradicts that S is an optimal solution.

Case 2. $B_l \le B_u \le y_i$

Start from B_u, and search within $[B_l, B_u]$. If B_u is occupied by power rails, then check $B_u - 1$, $B_u - 2$ ... until a feasible position u is found or it is out of the range. If u is found, then add edges $(e, (p, u))$ and $(e, (p, \bar{u}))$ where $\bar{u} \in (B_u, 2y_i - u)$. As illustrated in Figure 8 (Case 2), the feasible position in $(B_u, 2y_i - u)$ is y_i. Therefore, only two edges (e, e_3) and (e, e_4) are added in SP graph.

As to other feasible positions v of A_i, it must be outside the range $[u, 2y_i - u)$. If p, v and w (w is a feasible position of A_{i+1}) are selected for wire segments A_{i-1}, A_i and A_{i+1} respectively in a solution S, there must exist a solution $\bar{S}$ with less total deviation. In $\bar{S}$, the positions of all segments are the same as those in S except that A_i is placed at u instead of v.

Case 3. $y_i \le B_l \le B_u$

Start from B_l, and search within the range $[B_l, B_u]$. If B_l is occupied by power rails, then check $B_l + 1$, $B_l + 2$ ... until a feasible position u is found or it is out of the range. If u is found, then add edges $(e, (p, u))$ and $(e, (p, \bar{u}))$ where $\bar{u} \in (2y_i - u, B_l)$. As shown in Figure 8 (Case 3), there are two feasible positions in $(2y_i - u, B_l)$. Therefore, edges (e, e_3), (e, e_4) and (e, e_6) are added in SP graph.

If the conditions in the above three cases are not satisfied, then just connect nodes in the original way.

6. EXPERIMENTAL RESULTS

Our algorithms were implemented in C++ on PC (733MHz) with 128M memory. We tested CVE algorithms for four test files. These circuits were obtained from industry files. For all of the test circuits, the allowable derivation of each signal wire segment is bounded as 2% of the height of the ECO region area. After applying the FCVE algorithm, we can find clean routing solutions for all four files, and the max deviations are much smaller than the given bound. Then based on the output of the FCVE algorithm, we use SCVE to further improve the total deviation. The test results show that SCVE can greatly reduce the total deviation, for example, the total deviation is reduced to less than 5% of the original total deviation for both N3 and S6.

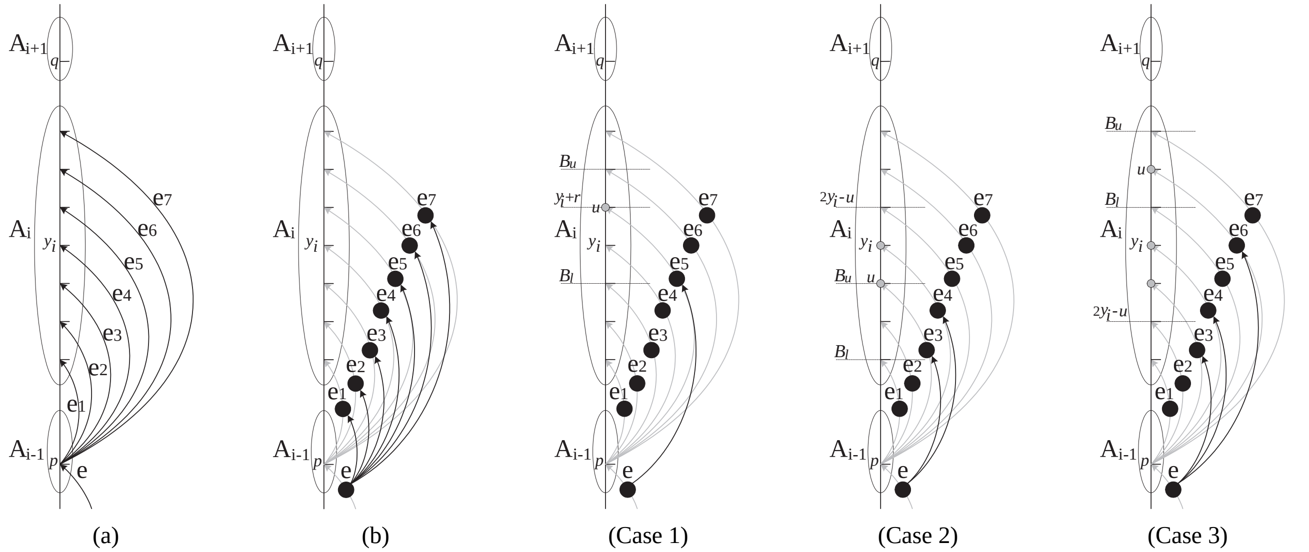


Figure 8: (a) FSP graph of a CVEP problem. p is a feasible position of A_{i-1}. (b) SP graph of the CVEP problem. (Case 1) u is the closest feasible position to y_i in $[y_i - r, y_i + r]$ where $r = \min(y_i - B_l, B_u - y_i)$. Only one edge (e, e_5) is needed in the SP graph. (Case 2) u is a feasible position in $[B_l, B_u]$, and y_i is the only feasible position in $(B_u, 2y_i - u)$. Edges (e, e_3), (e, e_4) are added. (Case 3) u is a feasible position in $[B_l, B_u]$, and there are two feasible positions in $(2y_i - u, B_l)$. Three edges (e, e_3), (e, e_4) and (e, e_6) are added.

Table 1: Test Results of CVE Problem

File		N3	S6	M8	F10
ECO Region Area(um^2)		4908.92x3295.52	3295.52x4908.92	10872.90x4799.54	4799.54x10872.90
Signal Segments		1601	2098	1266	726
Power Rail Segments		166	1128	631	747
Sensitive Segments		1439	1868	1085	683
Crosstalk Violation Segments		406	296	177	227
Allowable Deviation		2%	2%	2%	2%
Node Clustering for CVE		10	9	30	60
Test Results					
Max Deviation		0.12%	0.01%	0.28%	0.85%
Crosstalk Violation Segments		0	0	0	0
Time (second)	FCVE	3	2	2	1
	SCVE	5	1	9	42
Total Deviation	FCVE(*um*)	4533.54	768.15	11713.80	51930.80
	SCVE(*um*)	215.32	23.28	633.90	6820.99
	FCVE/SCVE	4.75%	3.03%	5.41%	13.13%

Table 2: Optimization for Test File N3

Node Clustering	Total Deviation	Time (second) NEO	Time (second) EO
2	160.23	251	149
4	176.56	49	32
6	198.77	22	14
8	209.17	12	8
10	215.32	8	5

Moreover, we tested the optimization strategies on the test file N3. Table 2 shows the test results of different granularity of node clustering. When more nodes are clustered as a “super-node”, the running time is much shorter although the total deviation is a little larger. At the same time, the experimental results show that edge omitting optimization strategy is also very effective such that the running time can be shortened by 1/3. NEO means no edge omitting is adopted; while EO refers to edge omitting.

7. CONCLUSION

In this paper, we present a two-stage algorithm to solve the CVE (Crosstalk Violation Elimination) problem. The first stage processes signal wire segments one by one and tries to find a clean routing solution. Then efforts are made in the second stage to minimize the total deviation. Furthermore, in order to facilitate the process of huge problems, we propose efficient optimization strategies to speed up the execution. Experimental results demonstrate the efficiency and effectiveness of our approach.

8. REFERENCES

[1] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. Network Flows, Prentice Hall, 1993.

[2] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. Introduction to Algorithms, The MIT Press, 1992.

[3] T. Sakurai and k. Tamaru. Simple formulas for two and three dimensional capacitance. IEEE Trans. Electron Devices, 1993.

[4] H. Xiang, K. Chao, and D.F. Wong. ECO Algorithms for Removing Overlaps between Power Rails and Signal Wires, IEEE/ACM ICCAD, pp 67-74, San Jose, CA, Nov. 2002.

[5] H. Zhou and D.F. Wong, Global Routing with Crosstalk Constraints, ACM/IEEE DAC, San Francisco, CA, 1998.