OPTIMAL VIA-SHIFTING IN CHANNEL COMPACTION*

Yang Cai and D. F. Wong

Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712 U.S.A.

Abstract

We study the problem of shifting vias to obtain more compactable two-layer channel routing solutions. Let $S$ be a grid-based two-layer channel routing solution. Let $v_c$ be the number of grid points on column $c$ that are occupied by vias. Let $w_h$ be the number of grid points on column $c$ that are occupied by horizontal wires. We define the height of a column $c$ to be the quantity $h_c = A_v + B_w + C$, where $A$, $B$, $C$ are some design rule dependent constants. A column is said to be critical if it is a column with maximum height. Let $H_S$ be the height of the critical column(s) in $S$. In general, $H_S$ is a good measure of the channel height after compaction. We show that the problem of shifting vias to minimize $H_S$ can be solved optimally in polynomial time. The complexity of our optimal via-shifting algorithm is $O(WL(V+L)\log(V+L))$ where $W, L,$ and $V$ are the number of tracks in $S$, the number of columns in $S$, and the number of vias in $S$, respectively.

1. INTRODUCTION

In VLSI layout design, a significant portion of area is used for channel routing. There are several grid-based two-layer channel routers which can consistently produce channel routing solutions which are at most one or two tracks within optimal solutions [YoKu82, RiFi82, BuPe83, ReSS85]. Recent studies [Deut85, ChDe88, CoWo88] showed that the routing solutions of these routers could be compacted to obtain further area reduction. From the experimental results of different channel compactors, it was observed that the amount of area reduction is closely related to both the routing solution and the design rule used. Appropriate modification of a given channel routing solution could result in a significant amount of area reduction. Techniques developed to modify routing solutions include via-shifting [Deut85, ChDe88], via-offsetting [Deut85, XiKu87, ChDe88], track-permutation [WoLi86, CoWo88], and local-rerouting [CoWo88, ThWC91]. In this paper, we restrict our attention to the technique of via shifting only.

Let $S$ be a grid-based two-layer channel routing solution. Let $v_c$ be the number of grid points on column $c$ that are occupied by vias. Let $w_h$ be the number of grid points on column $c$ that are occupied by horizontal wires. Let $f_x = v_c + w_h$. We shall refer to $v_c$ as the via count at column $c$, $w_h$ as the wire count at column $c$, and $f_x$ as the feature count at column $c$. We define the height of a column $c$ to be the quantity $h_c = A_v + B_w + C$, where $A$, $B$, $C$, are some design rule dependent constants, with $A > B$. A column is said to be critical if it is a column with maximum height. Let $H_S$ be the height of the critical column(s) in $S$. In general, $H_S$ is a good measure of the channel height after compaction.

In a typical grid-based two-layer channel routing solution, the vias have some freedom to move along some rectilinear trees as shown in the example in Figure 1. In this example, the rectilinear tree associated with each via along which the via can move is indicated by heavy lines. If the rectilinear trees associated with two vias share a common grid point, then the vias can be merged into a single via. For example, vias $v_1$ and $v_2$ in Figure 1 can be merged, so are vias $v_3$ and $v_4$. Hence we may assume that the rectilinear trees associated with the vias are all disjoint. Note that shifting vias can change the amount $H_S$, as shown in Figure 2. Before we shift vias, columns 7, 8 are the critical columns, and $H_S = 3A + 3B + C$. After shifting vias as indicated by the arrows, we get a new channel routing solution $S'$ with $H_{S'} = 2A + 4B + C$. Since $A > B$, the expected height of the channel is decreased by $H_S - H_{S'} = A - B$.

In this paper, we present an efficient optimal algorithm to shift vias to minimize $H_S$. The complexity of our optimal via-shifting algorithm is $O(WL(V+L)\log(V+L))$ where $W, L,$ and $V$ are the number of tracks in $S$, the number of columns in $S$, and the number of vias in $S$, respectively.

*This work was partially supported by the Texas Advanced Research Program under grant 4096, by the National Science Foundation under grant MIP-8909586, and by an IBM Faculty Development Award.

2024/0/0000/0186801.00©1990 IEEE
The rest of this paper is organized as follows. In Section 2, we introduce a new measure of the channel height after compaction. Section 3 analyzes the effects of via shifting. The optimal via-shifting algorithm is given in Section 4. Finally, Section 5 concludes the paper.

2. CHANNEL HEIGHT MEASURE

Let \( \alpha \) be the height of a via, \( \beta \) be the width of a wire, and \( \gamma \) be the minimum spacing between adjacent features (i.e., vias or wires). According to current fabrication technology, \( \alpha > \beta \). For example the values of \( \alpha, \beta, \gamma \) used in [Deu85, ChDe88, CoWo88] are: \( \alpha = 2.0, \beta = \gamma = 1.0 \).

Let \( S \) be a grid-based channel routing solution. There are three major factors that contribute to the height of a column \( c \) after compaction: (1) sum of the heights of vias on \( c \), i.e. \( \alpha v \); (2) sum of the widths of horizontal wires on \( c \), i.e. \( \beta w \); (3) total minimum spacings between adjacent features on \( c \), i.e. \( \gamma(v + w + 1) \). We define the height of a column \( c \) to be

\[
\text{height}(c) = \alpha v + \beta w + \gamma(v + w + 1) = \left(\alpha + \beta + \gamma\right) c + \gamma \]

Let \( H_s \) be the expected height after compaction. We shall refer to \( H_s \) as the expected height of \( S \) after compaction.

Figure 3 shows an example of a channel before and after channel compaction. In this example, we use \( \alpha = 2.0, \beta = \gamma = 1.0 \). In Figure 3(a), the critical column is column 5. Clearly \( v_5 = 2, w_5 = 0 \). Therefore the expected height \( H_5 \) after compaction is \( 2\alpha + (2 + 0 + 1)\gamma = 2\alpha + 3\gamma \). As we can see from Figure 3(b) that, for this example, the height of the channel after compaction is also \( 2\alpha + 3\gamma \).

3. VIA SHIFTING

Figure 4 illustrates the different ways to shift vias. Note that via shifting may introduce overlap of wires of the same net but never introduces overlaps of wires from different nets. We consider the following two kinds of via shiftings.

1. Horizontal Via Shifting

Suppose we shift a via \( v \) horizontally from column \( c \) to another column. Let \( v'_c, w'_c, \) and \( f'_c \) be the new via count, wire count, and feature count at column \( c \), respectively; and \( h'_c \) be the new height of column \( c \). Since the grid point where \( v \) was located is now replaced by a horizontal wire, we have

\[
v'_c = v_c - 1, \\
w'_c = w_c + 1.
\]

We also have,
vertical wire segment). Hence, without loss of generality, we may assume that the original solution has the property that if a via is not at a corner, then it is not possible to shift it to a corner. It follows that we only need to consider three cases of vertically shifting a via: (a) corner to corner; (b) non-corner to non-corner; and (c) corner to non-corner. (See Figure 5.)

When we vertically shift a via along a column c of v stays at column c. Let v, w, f, h and h' be the via count, wire count, feature count, and height at column c, respectively, after shifting a via v. We have,

\[ f' = f + 1 \]  \quad \text{Cases (a) and (b)}

\[ h' = h + (\alpha + \beta) \]  \quad \text{Case (c)}

It follows from the last formula that shifting vias vertically cannot decrease the height of a channel. Since only horizontal shifting of vias can decrease the height of a channel, and hence the height of the channel. There is no point of shifting a via vertically unless it will eventually be shifted horizontally from that column, in that case the effect is identical to a horizontal shifting. We can therefore associate with each via v a closed interval I_v, of consecutive columns, whose leftmost column is the leftmost column that intersects the rectilinear tree associated with v, and whose rightmost column is the rightmost column that intersects the rectilinear tree associated with v. The via v can be arbitrarily shifted within I_v (it may move from one track to another, or introduces overlapping wires of the same net but they are immaterial here). For the example in Figure 1, we have \( I_v = I_{v_1} = [1,12] \) and \( I_{v_2} = I_{v_3} = [10,12] \).

The rectilinear tree associated with a via v and hence \( I_v \) can be constructed by using depth-first search [AHH74] to search for all points on the net of v that are reachable from v. Thus, we can easily detect when the rectilinear trees associated with two vias share a common grid point and merge them into a single via by changing the layer assignment of some wires. Consequently, we may assume that no vias are mergeable in the given grid-based two-layer channel routing solution S.

4. AN OPTIMAL VIA-SHIFTING ALGORITHM

Given a grid-based two-layer channel routing solution S, another solution \( S' \) is said to be derivable from S if it can be obtained from S by shifting vias. Because the feature count at each column is not decreased by shifting vias, for any channel routing solution \( S' \) derivable from S, we must have

\[ H_{S'} = \max \{ (\alpha + \gamma) v' + (\beta + \gamma) w' + \gamma \} \]

\[ = \max \{ (\beta + \gamma) v' + \gamma + (\alpha - \beta) w' \} \]

\[ = \max \{ (\beta + \gamma) v' + \gamma + (\alpha - \beta) v' \} \]

Our objective is to obtain a new channel routing solution \( S' \) derivable from S with minimum expected height \( H_{S'} \). A key procedure in our algorithm is called FEASIBLE, which given any \( H > 0 \), determines whether there is a solution \( S' \) derivable from S such that \( H_{S'} \leq H \).

4.1. PROCEDURE FEASIBLE

In order to determine whether there is a channel routing solution \( S' \) derivable from S with \( H_{S'} \leq H \), we need to construct a network \( G_\psi = (N,E,\psi) \), where \( G = (N,E) \) is a directed graph and \( \psi \) is a non-negative integer-valued capacity function defined on the set of edges E, such that

\[ N = \{ s,t \} \cup \{ v \mid v \text{ is a via of } S \} \cup \{ c \mid c \text{ is a column of } S \}, \]

\[ E = \{(s,v) \mid v \text{ is a via of } S \} \cup \{(c,t) \mid c \text{ is a column of } S \} \cup \{(v,c) \mid v \text{ is a via, } c \text{ is a column of } S \}, \]

\[ \psi(s,v) = 1 \text{ for all vias } v \text{ of } S, \]

\[ \psi(v,c) = 1 \text{ for all vias } v \text{ of } S \text{ and all columns } c \text{ of } S \text{ with } (v,c) \in E, \]

\[ \psi(c,t) = \left( H - (\beta + \gamma) f - \gamma / (\alpha - \beta) \right) \]

An example of the construction of \( G_\psi \) is shown in Figure 6. The two nodes s and t are the source and the sink of \( G_\psi \), respectively. For each column c, \( \psi(c,t) \) is the maximum number of vias which can be placed in column c such that the expected height of the channel does not exceed H. (By formula (1), \( H_{S'} \leq H \) if and
Lemma 1. \( G_w = (N, E, \psi) \) has an integer-valued flow of value \( V \) if and only if there exists a channel routing solution \( S' \) derivable from \( S \) such that \( H_{S'} \leq H \).

Note that the capacity function of \( G_w \) is integer-valued, therefore there always exists an integer-valued maximum flow in \( G_w \), and it can be computed in \( O(|N| |E| \log |V|) \) time [Sle80].

According to Lemma 1, the procedure \( \text{FEASIBLE} \) can be implemented as a boolean function \( \text{FEASIBLE}(S, H) \) such that \( \text{FEASIBLE}(S, H) \) is true if and only if \( G_w = (N, E, \psi) \) has an integer-valued flow of value \( V \).

4.2. THE ALGORITHM

Let \( S' \) be a solution derivable from \( S \). It follows from formula (1) that \( L \leq H_{S'} \leq U \) where \( L = (\beta + \gamma) (\max_{v \in V} f_v) + \gamma \) and \( U = L + (\max_{v \in V} f_v) (\alpha - \beta) \).

Another fact that is needed in our algorithm is that for any channel routing solutions \( S' \) and \( S'' \) derivable from \( S \) with \( H_{S'} \leq H_{S''} \), we have \( |H_{S'} - H_{S''}| \geq \delta \) where \( \delta = \min[ |i (\alpha + \gamma) - j (\beta + \gamma)| : 1 \leq i, j \leq W] \). This fact follows from the observation that both \( H_{S'} \) and \( H_{S''} \) are of the form \( A (\alpha + \gamma) + B (\beta + \gamma) + \gamma \), where \( 1 \leq A, B \leq W \).

Our algorithm consists of two stages. At the first stage, we determine the minimal index \( i \) such that \( \text{FEASIBLE}(S, L + i (\alpha - \beta)) = \text{TRUE} \) by using binary search on the interval \( \text{[L, U]} = \text{(L, L + (max}_{v \in V} f_v) (\alpha - \beta)} \). Let \( l_0 \) be the minimal index found at the first stage, and let \( h = L + l_0 (\alpha - \beta) \). At the second stage, we determine the maximal index \( j \) such that \( \text{FEASIBLE}(S, h + j \delta) = \text{TRUE} \). Let \( j_0 \) be the maximal index found at the second stage. Let \( H = h + j_0 \delta \). By Lemma 1, there is a solution derivable from \( S \) with expected height \( H \) and there is no solution derivable from \( S \) with expected height \( \leq H - \delta \). Thus we have \( H \) is the minimum expected height. The corresponding solution can be easily obtained from the maximum network flow of \( G_w \).

ALGORITHM Via-Shifting

INPUT: A grid-based two-layer channel routing solution \( S \);

OUTPUT: A new routing solution \( S' \) derivable from \( S \) with minimum height;

begin [* Initialization *]
  \( l := 0 \);
  \( u := \max_v f_v \);
  \( b := (\beta + \gamma) u + \gamma \);
  \( \delta := \min \{ |i (\alpha + \gamma) - j (\beta + \gamma)| : 1 \leq i, j \leq W \} \);
Compute \( f_v \) for all vias \( v \) of \( S \);
Construct \( G = (N, E, \psi) \);
for all vias \( v \) of \( S \) do
  \( \psi(\psi, v) := 1 \);
for all vias \( v \) and columns \( c \) of \( S \) such that \( c \in I_v \) do
  \( \psi(\psi, v) := 1 \);
while \((l \leq u)\) do [* Stage 1 *]
  begin
    \( H := b + (\alpha - \beta) (l + u) / 2 \);
    for all columns \( c \) of \( S \) do
      \( \psi(c, v) := (H - (\beta + \gamma) f_v - \gamma (\alpha - \beta)) \);
  end
end

Figure 6

only if \((\beta + \gamma) f_v + \gamma + (\alpha - \beta) v \leq H \), for all \( v \). Solving this inequality, we have \( v \leq \psi(\psi, v) \).

The intuition behind the construction of \( G_w \) is that if \( G_w \) has an integer-valued flow \( \phi \) of value \( V \) (= total number of vias), then for each via \( v \) of \( S \), we must have \( \phi(v) = 1 \), meaning that all the vias of \( S \) will be placed in some column in the new solution. Hence for each via \( v \), there exists exactly one column \( c \) in \( I_v \), such that \( \phi(v, c) = 1 \) and all other edges leaving \( v \) has flow value 0; this forces each via to be placed in at most one column in the new solution. Thus the new solution so generated is guaranteed to be valid. Finally, for each column \( c \), \( \phi(c, j) \) is the number of incoming edges to \( c \) with flow value 1, is the number of vias placed in column \( c \) in the new solution. The expected height of the new solution is

\[
H_{S'} = \max_c (\phi(c, j) (\alpha - \beta) + (\beta + \gamma) f_v + \gamma) \quad \text{(by formula (1))}
\]

\[
\leq \max_c (\psi(c, j) (\alpha - \beta) + (\beta + \gamma) f_v + \gamma) \quad \text{for all } v.
\]

On the other hand, if there exists \( S' \) derivable from \( S \) with \( H_{S'} \leq H \), then we can define \( \phi \) as follows:

\[
\phi(\psi, v) := 1 \quad \text{for all } v,
\]

\[
\phi(v, c) := \begin{cases} 
1 & \text{if via } v \text{ is at column } c \in I_v \land \text{column } c \text{ is free}, \\
0 & \text{otherwise},
\end{cases}
\]

\[
\phi(c, j) := \sum_{v \in E_c} \phi(v, c).
\]

We now show that \( \phi \) is a valid flow. If \( \phi \) is not a valid flow, there exists \( c \) such that \( \phi(c, j) > \psi(c, j) \geq (H - (\beta + \gamma) f_v - \gamma (\alpha - \beta)) \). Since \( \phi(c, j) \) is an integer, we have \( \phi(c, j) > (H - (\beta + \gamma) f_v - \gamma (\alpha - \beta)) \). Thus

\[
H_{S'} > \phi(c, j) (\alpha - \beta) + (\beta + \gamma) f_v + \gamma \geq H,
\]

which is a contraction. Hence \( \phi \) is a valid flow and its value is clearly \( V \). We have thus proven the following lemma.
if \text{FEASIBLE}(S, H) = \text{TRUE} \\
\{ \mathcal{G}_\Phi \text{ has an integer flow } \phi \text{ of value } V \}
\text{then } u := \lceil (l + u)/2 \rceil - 1 \\
\text{else } l := \lceil (u + l)/2 \rceil + 1 \\
end;

\text{repeat} \{ \text{Stage 2} \}
\begin{align*}
&h := \max \{ (b + \gamma f_c + \gamma + \Phi(c,t)) (\alpha - \beta) \} \\
&H := h - 8; \\
&\text{for all columns } c \text{ of } S \text{ do} \\
&\Psi(c,t) := \lfloor (H (b + \gamma f_c - \gamma) (\alpha - \beta)) \rfloor \\
&\text{until } \text{FEASIBLE}(S, H) = \text{FALSE}; \\
&\text{Shift vias according to the maximum flow } \Phi. \\
\text{end.}
\end{align*}

\textbf{Theorem 1.} The algorithm \textit{Via-Shifting} produces an optimal solution in \(O(WL(V+L)\log^2(V+L))\) time.

\textbf{Proof.} Since we have already proved the optimality of the algorithm, it only remains to analyse its time complexity. To compute \(\Phi\), we need \(O(W^2)\) time. Since we assume that the rectilinear trees associated with the vias are all disjoint, we have the size of \(G = |N| + |E| = (V + L + 2) + (V + L + \sum_i |I_i|) = O(WL)\).

Thus \(O(WL)\) time suffices to construct \(\mathcal{G}_\Phi\). Because \(u \leq W\), the while-loop requires \(O(\log u) \leq O(\log W)\) iterations. Note that the amount of work for each iteration is dominated by that of finding a maximum integer flow in \(G_\Phi\), which is equal to \(O(|N|E\log|N|) = O((V + L)WL\log(W + L))\). Hence the while-loop requires \(O(WL(V+L)\log^2(V+L))\) time. By Lemma 1, at the end of the while-loop, we have \(b + (l-1)(\alpha - \beta) < H_L \leq b + l(\alpha - \beta)\) where \(S'\) is the optimal solution. In each iteration of the repeat-loop, at least one of the capacities \(\Psi(c,t)\) is decreased by 1, and no \(\Psi(c,t)\) can be decreased by more than 1 unless all of them have been decreased by 1. Hence the loop must terminate in \(O(L)\) iterations, otherwise we will have a solution derivable from \(S'\) with height less than or equal to \(b + (l-1)(\alpha - \beta)\), a contradiction. In each iteration of the repeat-loop, we have to call a procedure to compute a maximum integer flow, but since the difference between the networks in two consecutive iterations is that for some column \(c\), the capacity of the edge \((c,t)\) is decreased by 1, and the total number of such changes summed over the whole loop is \(O(L)\). Hence in order to compute a maximum flow for the current network, all we have to do is to take a maximum flow for the network of the previous iteration and for those columns \(c\) such that \(\Psi(c,t)\) is decreased, reset the flow value of one of the incoming edge to \(c\) to 0, and start searching for an augmenting path. Thus the whole loop can be done in \(O(WL^2)\) time. Hence the complexity of the whole algorithm is \(O(WL^2 + WL^2 + WL(V+L)\log^2(V+L)) = O(WL(V+L)\log^2(V+L))\) since \(W \leq L\).

5. \textbf{CONCLUDING REMARKS}

In this paper we study the problem of modifying given grid-based two-layer channel routing solutions by shifting vias to obtain more compactable channel routing solutions. We presented a polynomial time optimal algorithm for this problem. This appears to be the first attempt to study the via shifting problem from a global point of view, i.e., simultaneously shift all vias optimally.

We showed in Section 4 that the complexity of our algorithm is \(O(WL(V+L)\log^2(V+L))\). Since it is almost always true that the average length of \(L\) is a small constant, hence in almost all cases the size of the network \(G\) is \(O(V+L)\) rather than \(O(WL)\) and consequently our algorithm runs in \(O(V+L)\log^2(V+L))\) time.

It is well known that the height of the critical column(s) is the most significant factor that determines the final channel height after compaction. Note that there is another factor called \textit{bump-propagation} [Deut85] that can also affect the final channel height. Our proposed measure of channel height does not take the effect of bump-propagation into consideration. Further research is needed to design efficient algorithms that can minimize a new channel height measure which considers both the height of the critical column(s) and the effect of bump-propagations.

\textbf{REFERENCES}

[ChDe88] Cheng, C.-K. and D. Deutsch, "Improved Channel Routing by Via Minimization and Shifting,''
[CoWo88] Cong, J. and D. F. Wong, "How to Obtain More Compactable Channel Routing Solutions,''
[Deut85] Deutsch, D.N., "Compacted Channel Routing,''
[ReSS85] Reed, J., A. Sangiovanni-Vincentelli and M. Santomauro "A New Symbolic Channel Router: YACR2,''
[RiFi82] Rives, R. L. and C. M. Fiduccia, "A Greedy Channel Router,''
[Slea80] Sleator D. D., "An O(mn \log n) Algorithm for Maximum Network Flow,''
[TWNC89] The, K.S., D. F. Wong and J. Cong, "Via Minimization by Layout Modification,''
[WoLi86] Wong, D. F. and C.L. Liu, "Compacted Channel Routing with Via Placement Restrictions,''
[XiKu87] Xiong, X. and E.S. Kuh, "Netcracker: An Efficient and Intelligent Channel Router,''
[YoKu82] Yoshimura, T. and E.S. Kuh, "Efficient Algorithms for Channel Routing,''

190