Reducing Channel Density by Routing Over The Cells

Kuo-En Chang

Department of Information and Computer Education,
National Taiwan Normal University

Abstract

A linear time algorithm for greater reduction of maximum channel density by routing over the cells is presented. The algorithm first models the problem as a new scheme of zone representation and two intersected graphs for two sides of the channel. Then, a feasible independent set of each graph is found to represent the subnets that are efficient to be routed over the cells at the corresponding side of channel. The algorithm is implemented and evaluated by the tests of famous benchmarks. In comparison with previous research, our results are satisfactory while the algorithm takes less CPU time than those of previous works. For Deutsch's difficult example, the previous algorithm takes about 29.25 seconds while we need only 5.6 seconds.
1. Introduction

In VLSI design automation, the channel routing is one of the most important steps to finish the circuit design. A channel router receives two lists of terminals specified at both the top and the bottom sides of channel, and finds all interconnections of terminals in order to minimize the channel width, length of interconnections, and the number of vias. The conventional channel routing assumes that areas used for interconnections are restricted to the two routing layers within the channel. The type of channel routing is a well-known problem which has been extensively studied by many approaches referring to [1-4]. To further reduce the channel routing area or channel width, the channel router is allowed to use extra areas over the cells for interconnections. These channel routers are called over-the-cell routing. This type of channel routing was studied in the papers [5-7].

The basic idea behind the over-the-cell routing is that two routing areas, one layer over the cells (outside the channel) and two layers inside the channel, are used to connect all nets. For the one layer over the cell, the router finds some planar connections of subnets so that the number of remaining nets needed for the routing inside the channel is reduced. Fig. 1 is an example of over-the-cell routing. In general, an over-the-cell routing can be divided into three steps [7]:

(1) Routing over the cells,
(2) Choosing net segments within the channel,
(3) Routing within the channel.

During the first step, they try to connect terminals at one side of channel using the over-the-cell routing area on that side. The goal is to find as many nets to be routed over the cells as possible and thus fewer nets to be left within the channel. After the completion of the first step, a net has been partially routed over the cells. In the second step, a proper subset of terminals which have been connected over the cells are selected for completing the interconnection of a net
within the channel. Finally, the physical connections of all nets are achieved in the third step. Obviously, the third step can be accomplished using a conventional channel router.

In this paper, we only consider the problem of the first step. In the abovementioned approach, it is based on the intuition that if fewer nets are routed within the channel, fewer tracks are used in the channel. Hence, the goal of this approach is to find a maximum planar subset of nets to be routed over the cells. Unfortunately, it does not meet the property of channel routing. It is well-known that the number of tracks needed in the channel is proportional to the channel density of routing problem. The selection of subnets, which contribute to reduce the maximum channel density after they were routed over the cells, is more critical than the selection of maximum planar subset of subnets. The approach that blindly routes as many nets as possible over the cells only contributes to the slight reduction in channel width. The critical nets routed over the cells are those whose removal will contribute to the reduction of the channel density. The selection of these critical nets are essentially addressed in our consideration.

Lin, Perng, Hwang, and Lin [8, 9] have studied the channel density reduction by routing over the cells. They transformed this problem into a constrained covering problem and formulated as an integer linear programming problem. In comparison with previous research, their approach reduces more channel densities while using fewer tracks over the cells. However, the drawback of the integer linear programming formulation is its computational inefficiency, and the CPU consumption is not proportional to the channel size. When the problem size becomes large, the CPU time grows very rapidly. A heuristic algorithm is needed to replace the integer linear programming for CPU time consideration. In this paper, we present a heuristic algorithm to route the nets over the cells. The experiment showed that our algorithm is efficient to the reduction of channel width and the CPU time consideration.
In the following section, we model the problem of routing over the cell as a new scheme of zone representation. The graph model of the problem is presented in Section 3. Based on the zone representation and graph model, a heuristic algorithm to determine the critical subnets for routing over the cell is addressed in Section 4. Finally, results and conclusions are considered.

2. Zone Representations

To facilitate the understanding of our algorithm, some terminologies and definitions should be introduced first.

The channel routing is defined by the two lists of terminals numbered at both the top and the bottom sides of channel. A net is an interconnection of terminals with the same number in the channel. A segment is an interval between two adjacent terminals of a net at the same side in the channel. The segments can be classified into three types: top, bottom and dummy segments. Top segments are the segments defined on the top side of channel, and bottom segments are at the bottom side. Dummy segment is an interval of two terminals which are located at the different sides of channel respectively. For net $j$ with $k$ terminals on the top side of channel, there are $k-1$ top segments denoted by $h_{j1}^t$, $h_{j2}^t$, ..., $h_{j(k-1)}^t$ respectively. The definition of bottom segments is the same as top segments except the bottom side of channel is considered. Let $h_{ji}^b$ be the $i$-th bottom segment of net $j$ and $h_{j}^d$ be a dummy segment of net $j$. Fig. 2 specifies the definition of segments, in which $h_{51}^t$ and $h_{52}^t$ are two top segments of net 5, $h_{51}^b$ is a bottom segment, and the dummy segment of net 4 is $h_{4}^d$.

A net-segment is a set of segments belonging to the same net. For each column $i$, some net-segments are defined and a zone $z(i)$ is constructed. A zone $z(i)$ is defined as a set of net-segments in column $i$. The example shown in Fig. 3 is the zone representation of Fig. 2. For column 5 of Fig. 3, $z(5) = \{ (h_{21}^t), (h_{51}^t), (h_{51}^b), (h_{4}^d) \}$, where three net-segments $(h_{21}^t)$, $(h_{51}^t)$, $(h_{51}^b)$, and $(h_{4}^d)$ are defined. The local density $d(i)$ is defined as the number of net-segments in zone $z$.
(i) and is also shown in Fig. 3. The maximum channel density $d_{\text{max}}$ is then defined as the maximal one of $d(i)$ for all columns $i$. The $d_{\text{max}}$ of Fig. 3 is 4.

Furthermore, the net-segments are classified into three types. The single net-segment contains only one segment. The intersected net-segment contains at least two intersected segments. The dummy net-segment is a collection of segments in which at least one dummy segment is included. For the instance of Fig. 3, there are one intersected net-segment ($h^{t}_{52}, h^{b}_{51}$), one dummy net-segment ($h^{t}_{41}, h^{d}_{4}$), and two single net-segments $h^{b}_{31}$ and $h^{b}_{61}$ in $z(7)$.

Our study of routing over the cells is to find some critical net-segments for removing in order to significantly reduce the maximum channel density. For example, if the four net-segments ($h^{t}_{31}, h^{b}_{32}, h^{b}_{61}, h^{t}_{21}, h^{b}_{11}$ in Fig. 2 are selected to be routed over the cells, the maximum channel density will be reduced to 2 as shown in Fig. 4. Comparing to the result of $d_{\text{max}} = 3$ presented in Cong and Liu [6], our result is better. Since only one layer over the cell is permitted to be routed, the selected net-segments have to be routed for planar connections. In short, the aim of our algorithm is to find some critical net-segments that will be routed by the planar connections in order to minimize the $d_{\text{max}}$ of final result.

From above discussions, the problem for routing over the cells is transferred into how to select the net-segments for removal. However, not all net-segments are available to be removed for reducing the maximum channel density. For example, the dummy net-segment is instinctively unremovable. Therefore, in each zone, the only consideration of net-segments to be removed is single net-segments and intersected net-segments.

3. Graph Model

For removing the critical net-segments two graphs are used to define the relationships among top segments and bottom segments respectively. These two graphs are called Top-Segment Intersected Graph and Bottom-Segment
Intersected Graph, denoted by $\text{SIG}_T$ and $\text{SIG}_B$ respectively. Top-segment intersected graph is defined as follows. Vertices in the graph are represented by the top segments. Two vertices are linked by an edge when their corresponding top segments are intersected. Two top segments are said to be intersected when their intervals are partially rather than completely overlapped. For the instance of Fig. 2, the top segments $h^1_{41}$ and $h^1_{31}$ are intersected each other but $h^1_{52}$ and $h^1_{41}$ are not. The definition of bottom-segment intersected graph is similar to that of the top-segment intersected graph except it considers the bottom-segments. Fig. 5 shows these two graphs representing the relationships among the segments of Fig. 2.

From the definition of $\text{SIG}_T$ (or $\text{SIG}_B$), two segments can be planar routed over the cells at top (or bottom) side when their corresponding vertices are not connected in $\text{SIG}_T$ (or $\text{SIG}_B$). As a result, the vertices in any independent set in $\text{SIG}_T$ (or $\text{SIG}_B$) correspond to the segments that allow to be all routed in one layer over the cell at the top or the bottom side of the channel.

From the definition of the intersected net-segment, the removal of only a top segment (or bottom segment) in the net-segment is possibly useless in reducing the channel densities. For instance of Fig. 2, the only segment removed is $h^1_{52}$ instead of all segments in $(h^1_{52}, h^b_{51})$ in column 7, and the removal cannot reduce the local density $d(7)$. Therefore, the feasible consideration is the removal of net-segments but not for only top segments or bottom segments. A Net-segment Intersected Graph (or NIG) defined below is used to determine the removal of net-segments. Each vertex in NIG corresponds to the net-segment in the channel and two vertices are connected by edges when the segments in two corresponding net-segments are intersected. By this definition in an NIG, all segments in the corresponding net-segments of independent set can be routed simultaneously over the cells.

For constructing the graph NIG, two graphs of $\text{SIG}_T$ and $\text{SIG}_B$ are first linked by some edges. If two segments are in the same net, the two correspond-
ing vertices in $\text{SIG}_T$ and $\text{SIG}_B$ are connected by a dashed edge. The resulting graph is denoted as $\text{SIG}$ and shown in Fig. 6. Then, a set $S_{\text{max}}$ is defined to be an union of all top or bottom segments which are in the zones $j$ with $d(j)=d_{\text{max}}$. For the example of Fig. 3, $S_{\text{max}} = z(4) \cup z(7) \cup z(8) \cup z(9) = \{h_{11}^b, h_{21}^l, h_{51}^l, h_{52}^l, h_{51}^b, h_{41}^l, h_{61}^b, h_{31}^l, h_{31}^b\}$ since $d_{\text{max}} = 4$. Now, we construct a feasible net-segment intersected graph $\text{NIG}$ for maximum channel density from the graph $\text{SIG}$ by the following two steps:

1. Let $H$ be a subgraph of $\text{SIG}$, and $H$ have vertices in $S_{\text{max}}$ and corresponding edges in $\text{SIG}$.

2. The graph $\text{NIG}$ is built by merging two or more vertices in $H$ when they are in the same net-segment.

Fig. 7 is the two steps of constructing of $\text{NIG}$ from the graph $\text{SIG}$ of Fig. 6.

By the definition of $\text{NIG}$, the vertices corresponding to the feasible net-segments and the removal of net-segments in $\text{NIG}$ are useful to reduce maximum channel densities at some columns in the channel. For example, the net-segment $h_{41}^l$ can reduce the local densities $d(8)$ and $d(9)$ when $h_{41}^l$ is selected to be routed over the cell, but it cannot reduce the local density $d(7)$ because $h_{41}^l$ is a member of dummy net-segment $(h_{41}^l, h_{4}^d)$ at column 7 of Fig. 3. However, the net segment $(h_{31}^l, h_{31}^b), (h_{52}^l, h_{52}^b)$, or $h_{61}^b$ is feasible to reduce the local densities $d(7), d(8)$ and $d(9)$. Based on the above statements, the removal of net-segment $(h_{31}^l, h_{31}^b), (h_{52}^l, h_{52}^b)$, or $h_{61}^b$ is better than the net-segment $h_{41}^l$. Therefore, our problem is how to select net-segments in $\text{NIG}$ for significantly reducing the maximum channel density.

For the best selection of net-segments, some parameters are weighted at the vertices in $\text{NIG}$. These parameters at each vertex $k$ of $\text{NIG}$ are defined in the following.

\begin{align*}
\alpha_k &= \text{the number of solid edges connected to vertex } k \text{ in } \text{NIG}. \\
\beta_k &= \text{the number of zones } j \text{ with } d(j) = d_{\text{max}} \text{ and the net-segment } k \text{ in zone } j
\end{align*}
Based on the weights of all vertices in NIG, we present a heuristical approach to find all better segments which are in the same independent set of $\text{SIG}_T$ or $\text{SIG}_B$. Fig. 8(a) shows the weighted graph of Fig. 7(b).

4. The Algorithm

The determination of feasible independent sets in graph $\text{SIG}_T$ and $\text{SIG}_B$ is proposed heuristically into two steps:

(1) Select one vertex based on the weight $(\alpha, \beta)$.

(2) Update the weighted graph $\text{NIG}$ after the net-segment associated with the selected vertex is removed.

The algorithm receives the graph $\text{NIG}$ and executes two steps iterately until no feasible vertex is selected.

For step 1, vertices with minimal $\alpha$ are decided first, and then a vertex with maximal $\beta$ is selected from them. If $W_j = \alpha_j + 1/(\beta_j)$ for all vertices $j$, the vertex with minimal $W_j$ is selected. For instance of Fig. 8(a), the vertex $h^b_{11}$ is selected.

After a vertex is selected, the graph $\text{NIG}$ is updated and the weights of updated graph will be modified for the consideration of next selection. There are three operations needed to update the graph $\text{NIG}$ after the vertex $j$ is selected.

1) Mark some vertices, which are connected to vertex $j$, to be inactive in the successive selection.

2) Reconstruct a new graph $\text{NIG}$ when the maximum channel density is reduced after the segments in vertex $j$ are removed.

3) Delete all inactive vertices from the new graph $\text{NIG}$ and recalculate all weights of vertices in $\text{NIG}$.

Because all selected vertices should not be connected with each other in $\text{NIG}$ for the planar routing of segments over the cells. Therefore, all vertices connected to the selected vertex should not be considered in the next selection. These inactive vertices will be marked and neglected from the graph $\text{NIG}$. Fig.
8(c) shows that an inactive vertex \((h_{51}^t, h_{52}^t, h_{51}^b)\) is marked (by dashed line node) since \(h_{51}^b\) is connected to the selected vertex \(h_{61}^b\) shown in Fig. 8(b). Fig. 8(d) is a weighted graph NIG when the inactive vertex is neglected. A step-by-step example for the removal of net-segments is presented in Fig. 8 and Fig. 8 (g) shows the final step in which no feasible net-segment exists.

In summary, the procedure of routing over the cells can be presented as follows.

**MAIN PROCEDURE**

1. Input two lists of terminals at top and bottom sides in the channel.
2. Define zone representations and local densities.
3. Construct the graphs of \(SIG_T\), \(SIG_B\) and NIG.
4. **REPEAT**
5. Select a vertex \(j\) with minimum \(W_j\) for all vertices in NIG;
6. **FOR** all segments \(u\) in net-segment \(j\), **DO**
7. Marking the vertices, which are connected to \(u\) in \(SIG_T\) or \(SIG_B\), to be inactive.
8. **IF** vertex \(u\) corresponds to the top-segment **THEN** \(R_T = R_T \cup u\),
9. **ELSE** \(R_B = R_B \cup u\)**
10. Recalculate the local densities and \(d_{max}\) after net-segment \(j\) is removed.
11. Reconstruct a set of \(S_{max}\) and the weighted graph NIG with a neglect of the inactive vertices.
12. **UNTIL** (NIG is null or no feasible vertex is selected)
13. **RETURN** \((R_T, R_B)\)

**END PROCEDURE**

The final results of main procedure are stored in \(R_T\) and \(R_B\). All segments corresponding to the vertices in \(R_T\) and \(R_B\) are feasible to reduce the \(d_{max}\) after they are routed over the cells at the top and bottom sides respectively.

The time complexity of algorithm is analyzed as follows. Each time by scan-
ning all columns in the channel from left to right, local densities and $S_{\text{max}}$ are obtained and graphs SIG_T and SIG_B are constructed. Therefore, it is easy to show that the complexity of Steps 1-3 takes $O(M)$, where $M$ is the number of columns in the channel. The graph NIG is defined once at each iteration in REPEAT-UNTIL loop of the algorithm based on the members in $S_{\text{max}}$. For each iteration of REPEAT-UNTIL loop, the algorithm takes $O(M + |S_{\text{max}}|)$, since the updates of both local densities and $S_{\text{max}}$ in Step 10 and Step 11 are done in $O(M)$. In the worst case, the complexity of algorithm is $O(KM + N)$, because of $O(\Sigma KS_{\text{max}}) = O(N)$ while $N$ is defined to be the number of segments including top segments and bottom segments and $K$ is the number of segments routed over the cells (or number of iterations in REPEAT-UNTIL loop). However, it is clear that $O(N) = O(\Sigma n_i) = O(M)$, where $n_i$ is the number of terminals in net $i$. It concludes that the complexity of algorithm is $O(M)$ since $K << M$ based on our experiments.

5. Results and Conclusions

The algorithms of routing over the cells were coded in C language and implemented on the SUN 4 SPARC II workstation. The algorithms are efficient and have been evaluated on some examples including examples 1, 3a, 3b, 3c, 4b, 5, and Deutsch's Difficult example from [7]. Table I compares our results with [7]'s and [8]'s results. For the number of tracks used in the channel or over the cells, our results are satisfactory as [8] does and better than [7] does. However, taking CPU time in account, our algorithm is much faster than the previous approaches [7] or [8].

In this paper, we have proposed a new fast algorithm for routing over the cells. We heuristically choose a "critical" set of segments to significantly reduce the maximum channel density. The proposed algorithm improved the drawback of computational inefficiency in [8, 9] and obtained satisfactory results.
REFERENCES

### Table 1 Experimental Results

<table>
<thead>
<tr>
<th>Examples</th>
<th>Original Density</th>
<th>Density over the lower cells</th>
<th>Density over the upper cells</th>
<th>Final density</th>
<th>CPU-time(sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3a</td>
<td>15</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3b</td>
<td>17</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3c</td>
<td>18</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4b</td>
<td>17</td>
<td>4</td>
<td>3</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Deff</td>
<td>19</td>
<td>7</td>
<td>3</td>
<td>3</td>
<td>8</td>
</tr>
</tbody>
</table>
Fig. 1 The example of over-the-cell routing.

Fig. 2 Definitions of segments shown in Fig. 1.

<table>
<thead>
<tr>
<th>Column i</th>
<th>Zone representation z(i)</th>
<th>Local density d(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$h_{11}^t h_{22}^d$</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>$(h_{11}^t h_{11}^b) (h_{22}^d h_{21}^t)$</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>$(h_{11}^t h_{11}^b) h_{21}^t h_{21}^d$</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>$h_{11}^b h_{21}^t h_{31}^d h_{51}^t$</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>$h_{21}^t h_{31}^d (h_{51}^t h_{51}^b)$</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>$h_{41}^t (h_{51}^t h_{52}^b h_{51}^b) h_{31}^b$</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>$(h_{41}^t h_{42}^t) (h_{52}^t h_{51}^b h_{51}^b) h_{31}^b h_{61}^b$</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>$h_{41}^t (h_{52}^t h_{51}^b) (h_{31}^t h_{31}^b) h_{61}^b$</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>$h_{41}^t h_{52} (h_{31}^t h_{31}^b) h_{61}^b$</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>$h_{52}^t (h_{31}^t h_{31}^b h_{32}^b)$</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>$h_{52}^t h_{32}^d$</td>
<td>2</td>
</tr>
</tbody>
</table>

Fig. 3 Zone representation.
Fig. 4 The layout with $d_{\text{max}} = 2$ after routing over-the-cell

(a) Top-segment intersect graph

(b) Bottom-segment intersected graph

Fig. 5 Segment-intersected graph of Fig. 2.
Fig. 6 Segment-intersected graph of Fig. 2.

Fig. 7 The steps for contracting NIG (d_{max} = 4) of SIG shown in Fig. 6
Fig. 8 A step-by-step example for the removal of net-segments, and the final outputs are $h_{11}^b$ (after $h_{21}^b$ is removed and it cannot further to remove net-segments).
以過晶元佈線方式減少通道密度

張國恩

國立台灣師範大學資訊教育研究所

摘要

本文提出一個線性時間之演算法以完成過晶元佈線，該佈線可以有效地減少最大通道密度。此演算法首先以新的通道表示方式描述問題，並建立二個交叉圖形模式。然後針對每一個交叉圖形各找出獨立集合以代表可以有效地完成過晶元佈線之部份網列連線。本文中之演算法已完成程式設計，並以著名的實例完成測試評估。和以前之研究成果比較，本演算法可以在較短之執行時間內得到良好的結果。