

複数チャネルの配線を考慮したセルモデルに対する 3層セル上チャネル配線手法

槌家 正博 小出 哲士 若林 真一 吉田 典可
広島大学 工学部
〒724 東広島市鏡山一丁目4番1号

本稿では、セル上配線の自由度が高いセルモデルと、それに対する3層セル上チャネル配線手法を提案する。提案セルモデルは、セル上の第2層の端子の位置を任意に設定可能であり、かつ、電源配線をセル上の第1層で行なうことによりセル上配線、チャネル配線に対する配線自由度を高めている。提案手法では、0-1整数計画法を用いることにより、端子位置を考慮した概略配線経路を求め、その結果に対して、セル上にビアを挿入して第2,3層を用いてセル上II-V配線を行なっている。また、提案手法と提案モデルの有効性を示すために行なったMCNCベンチマークデータに対する計算機実験の結果についても報告する。

A Three-Layer Over-the-Cell Channel Routing Method for a New Cell Model Considering Multi-Channel Routing

Masahiro Tsuchiya, Tetsushi Koide, Shin'ichi Wakabayashi and Noriyoshi Yoshida
Faculty of Engineering, Hiroshima University
4-1, Kagamiyama 1-chome Higashi-Hiroshima 724, JAPAN

In this paper, we propose a three-layer over-the-cell channel routing method for a new cell model that allows many over-the-cell routing patterns. In the proposed cell model, terminals can be placed arbitrarily in the second layer of over-the-cell region, and Vdd/Gnd are placed in the first layer of over-the-cell region. As a result, there are many over-the-cell routing patterns. The proposed channel routing method first performs global routing considering terminal positions with 0-1 integer programming. Next, the proposed method performs II-V routing in second and third layers by using vias. We present experimental results for MCNC benchmarks and show effectiveness of the proposed method and the new cell model.

1 まえがき

半導体技術の進歩に伴い、VLSIに要求される機能が大変複雑になりVLSIチップが大規模化している。スタンダードセル方式レイアウト設計において、チップ面積の最小化を図るために、これまでに様々な配置・配線アルゴリズムの研究がなされている[12]。特に、近年では、チップ面積を更に減少させるために、セル上領域を配線に利用するセル上チャネル配線手法が注目されている[1, 6, 8, 13]。セル上チャネル配線手法は、従来はチャネル内で配線されていたネットを配線に利用されていないセル上領域で配線し、チャネル内のトラック数を減少させることにより、チップ面積の削減を図る手法である。スタンダードセル方式の代表的なセルモデルとして、BTM(Boundary Terminal Model)[4]、MTM(Middle Terminal Model)[13]、CTM(Center Terminal Model)[1]などが挙げられる。

BTMは2層メタル配線を仮定し、端子はセルの第2層の境界に整列して配置され、電源配線(Vdd/Gnd)は第2層の中央部で行なっている。文献[4]では、BTMに対するセル上配線手法として、動的計画法から求められる平面配線可能なネット集合をセル上配線ネットとする手法が提案されている。CTMでは端子はセルの第2層の中央部で1列に整列して配置され、電源配線は第1層のセルの境界で行なっている。CTMに対しては、リバールーティングから求められる平面配線ネットと、サークルグラフから最大2部部分グラフを求めることから得られる平面配線ネットを組み合わせることでセル上配線ネットを求めている[1]。また、MTMでは、端子はセルの第2層の中央付近で2列に整列して配置され、電源配線は第1層のセルの境界で行なっている。MTMにおけるセル上配線では、セルの中央付近においてはサークルグラフから最大独立点集合を求めることによりセル上配線ネットを求め、セルの境界付近においてはCTMと同様の手法で求めている[13]。しかし、どのセルモデルを利用するにしても、セルの第1層において、端子を整列させるために余分な配線領域を必要としている。

一方、近年の半導体技術の進歩より、3層以上の多層構造を持つセルが実現可能となった。これに伴い多層配線の技術も向上し、多層のセルに対するセル上配線手法も研究されている[5]。これまでに多くのセル上配線手法[3, 7, 10]が提案されているが、これらの手法の多くは、前述したように端子がセル上第2層の境界に整列して並べてあるBTMに対して提案されている。しかし、この従来のセルモデルでは、端子をセルの境界に並べて配置するため、端子をセルの境界に引き出すまでにセル内の第1層に余分な配線領域を必要とし、多層配線には向いていない。また、近年のVLSIチップの高性能・高密度化により、機能ブロックの配線が非常に複雑になり、多くの配線領域が必要となっている。そのため、機能ブロック

の配線を第1層だけで行なうとセル自体の面積が大きくなり、チップ面積の最小化を妨げる一要因となっている。

そこで我々は、セル内配線の一部をセルの第2層で行なうことによりセル自体の面積を小さくし、更に、端子をセルの境界まで引き出さない、すなわち、セル上の任意の位置に端子を設定することができるセルモデルを新しく提案する。しかし、提案するセルモデルでは、端子が1列に整列していないため、従来のセル上配線アルゴリズムを適用しても、セル上領域を有効に利用することは非常に困難であると考えられる。そこで、提案するセルモデルを有効に利用できるセル上配線アルゴリズムを新しく提案する。提案アルゴリズムでは、提案セルモデルに対して複数のチャネルを考慮しながら有効にセル上領域を利用し、II-V配線を基にしてセル上配線を行う。この提案アルゴリズムを計算機上に実現し、MCNCベンチマークデータに対してシミュレーション実験を行なった結果、ネットを部分的にセル上配線しない従来手法と比べて、最大3トラック減少し、提案セル上配線アルゴリズムの有効性を確認した。また、BTMと提案セルモデルを比較した結果、最大2トラック減少し、提案セルモデルの有効性も確認した。

以降では、2節で提案するセルモデルの説明を行ない、3節で提案するセル上配線アルゴリズムについて述べる。4節でシミュレーション実験の結果を考察し、最後に5節で本稿のまとめと今後の課題について述べる。

2 提案セルモデル

配線は3層メタル配線(M1, M2, M3)を仮定する。各配線層には配線格子を仮定し、水平方向をトラック、垂直方向をカラムと呼ぶ。また、配線はすべてこの配線格子上で行なうものとする。隣接する層間の配線はビアを用いて行ない、ビアは配線格子が交差する点に置かれるものとする。図1に提案するセルモデルを示す。配線モデルは、チャネル内はII-V-II配線、セル上はV-II配線とし、ビアを用いることができるものとする。電源配線(Vdd/Gnd)はM1の境界に置き、フィードスルーは第2層に置かれるものとする。機能ブロックの配線にはM1の領域全部とM2の一部分を使用し、M2で使用する機能ブロックの配線は、矩形領域で表現する。これはセル上配線の際に障害物Oとなる。端子は原則としてM2の任意の場所に置くことができるものとする。ただし、端子は1カラムに高々1端子とし、そのカラム上で、少なくともいずれか一方のセルの境界まで端子を引き出すことができるものと仮定する。また、提案セルモデルでは、従来セルモデルであるBTMのようにセル上第2層の中央部で電源配線を行わないため、複数のチャネルに跨りながらネットを配線することが可能となる。その結果、より多くのセル上領域を有効に利用しながらセル上配線をする

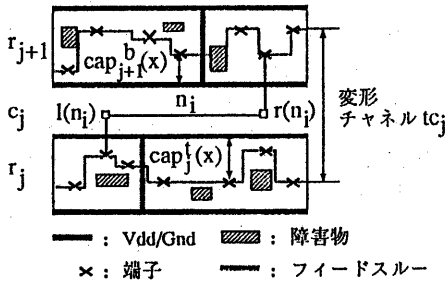


図1 提案セルモデル

ことができる。

次に本稿で使用する記号を定義する。セルはチップ内に列状に配置されているものとし、セル列集合 $R = \{r_1, r_2, \dots, r_k\}$ は、下から順に番号付けされているものとする。同様にチャネル集合 $C = \{c_0, c_2, \dots, c_k\}$ をセル列の間の領域と定義する。セル列 r_j 内のトラックは、下から順に1から H まで番号付けされており、トラック集合を $T_j = \{t_1, t_2, \dots, t_H\}$ で表すものとする。また、セル列 r_j のカラムは左から順に1, 2, ..., L まで番号づけされている。ここで L はセル列の幅の最大値とする。このとき端子 p_k の位置をカラムとトラックの2項組 (x_k, t_k) で表す。ネット n_i は概略配線によってあらかじめ2端子分解されているものとし、 $n_i = (p_{i1}, p_{i2})$ で表す。このときのネットリストを $N = \{n_1, n_2, \dots, n_{|N|}\}$ で表す。ネットは2端子分解によって、従来のチャネル配線手法で配線する場合の配線チャネルが決まっており、チャネル c_j に配線されるネット集合を $N_j = \{n_1, n_2, \dots, n_{q_j}\}$ で表すものとする。このとき、チャネル c_j の領域とネット集合 N_j の端子位置と障害物により構成される凹凸状の配線領域(図1)を変形チャネル $tc_j \in TC$ と定義する。変形チャネル tc_j において、ネット n_i の左の端子のカラム番号を $l(n_i)$ 、右の端子のカラム番号を $r(n_i)$ で表すとき、ネット n_i の水平方向の長さを $x_length(n_i) = r(n_i) - l(n_i)$ とし、この水平方向の領域のことをセグメント $s_i \in S$ と定義する。また $cap_j^b(x)(cap_j^t(x))$ は、セル列 r_j のカラム x におけるセルの上辺(下辺)の境界から最も近い端子、または障害物までのトラック数を表す。もし、カラム x に端子、障害物が共に存在しないなら、セルの高さの半分(すなわち $H/2$)とする。このとき、変形チャネル tc_j にあるセル列 r_j のカラム x に対する局所変形チャネル密度 $d_j(x)$ を $d_j(x) = through_net(x) - (cap_j^t(x) + cap_{j+1}^b(x))$ と定義する。ここで、 $through_net(x)$ はカラム x を通過するセグメント数を示す。これを変形チャネル tc_j 内にある全てのカラム x に対して求め、変形チャネル密度 $D_j = \max_{1 \leq x \leq L} d_j(x)$ として定義する。また、チャネル c_j に対してはチャネル密度 $ND_j = \max_{1 \leq j \leq L} through_net(x)$

を定義する。そして、チャネル c_j で配線するネット集合に対してチャネル配線を行なった結果、全てのネットを配線するのに必要なチャネル c_j 内のトラック数を at_j とすると、全てのチャネルに対してトラック総数 $AT = \sum_{j=0}^k at_j$ を定義する。

以下に本稿で取り扱うセル上チャネル配線問題の定式化を示す。

【セル上チャネル配線問題】

入力: (1) ネット集合 N
(2) 障害物集合 O
出力: チャネルとセル上の配線パターン
制約: 障害物と交差しない
目的関数: トラック総数 AT の最小化 □

目的関数はトラック総数 AT の最小化であるが、これを直接求めることは困難であるため、以降では、トラック数と関係深い変形チャネル密度の総和 $\sum_{j=0}^k D_j$ の最小化を目的関数として扱うものとする。

3 セル上チャネル配線アルゴリズム

提案手法では、セル上チャネル配線問題を大きく分けて3つのフェーズに分解して取り扱う。

ここでは3つのフェーズの概要について説明する。

【セル上チャネル配線問題】

フェーズ1: ネット割当て

端子位置を考慮したネットの変形チャネルへの割当てを0-1整数計画法を用いて求め、変形チャネル tc_j ごとに配線を行なうネット集合 N_j を求める。

フェーズ2: セル上配線

変形チャネル tc_j ごとに変形チャネル密度 D_j が減少するようなセル上配線ネットを求める。

フェーズ3: チャネル配線

フェーズ2よりセル上配線ネットとして選択されなかったネット集合 $N - \bigcup_{j=0}^k N_j$ に対して端子をセルの境界まで引き出し、BTMと同様のモデルにみなす。そして、従来の3層チャネル配線手法[2]を適用してチャネル配線を行なう。

以降では、各フェーズごとに詳細を説明する。

3.1 フェーズ1: ネット割り当てアルゴリズム

フェーズ1では、端子位置を考慮したネットの変形チャネルへの割当てを0-1整数計画法を適用して求め、変形チャネル tc_j の外周を構成するネット集合 $N_j = \{n_1, n_2, \dots, n_{q_j}\}$ を新たに決定する。以下に、フェーズ1のネット割当て問題に対する問題の定式化を示す。

【ネット割当て問題】

入力:	(1) ネット集合 N (2) 障害物集合 O
出力:	ネット n_i の変形チャネル tc_j への割当て $A: N \rightarrow TC$
制約:	(1) 障害物と交差しない (2) 同じネット n_i を異なる2つの 変形チャネルに同時に割当てない
目的関数:	$\sum_{0 \leq j \leq k} D_j \rightarrow \min$ □

フェーズ1のネット割当てアルゴリズムは、上に示した問題を0-1整数計画問題に定式化して求めるが、問題で使用する変数や制約式の数小さくするため、文献[14]で用いられているゾーン表現を利用する。

提案セルモデルを利用するとき、セル上第2層にある障害物が配線を妨げない場合、ネットのセグメントは図2に示すように複数の変形チャネル tc_j の中から、いずれか1つを利用して配線することが可能となる。図2(a)の場合では、最大3つの変形チャネル tc_j を考慮し、図2(b)の場合では、最大4つの変形チャネル tc_j を考慮しながらフェーズ1のネット割当てを実行するものとする。対象となる変形チャネル tc_j の範囲を限定する理由は、ネットの配線長が長くなりすぎるのを防ぐためである。もし、対象となるネットが障害物と交差する場合は、配線不可能なので、そのネットのセグメントは考慮に入れないものとする。

次にネット割当て問題を0-1整数計画法で解くために問題の定式化を行なう。ネット割当て問題の目的関数は、変形チャネルの密度の総和の最小化だから、

$$\sum_{j=0}^k D_j \rightarrow \min.$$

となる。

制約式には、(1) セグメント制約、(2) カラム制約、(3)

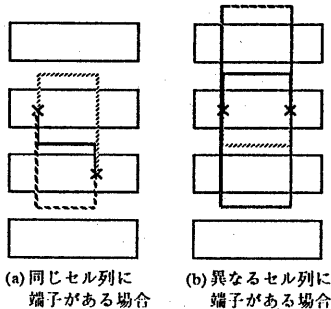


図2 対象となる変形チャネルの範囲

密度制約、の3つを考える。最初にセグメント制約について説明する。ネット n_i にはあらかじめ m_i 通りの配線パターン $PT_i = \{pt_{i1}, pt_{i2}, \dots, pt_{im_i}\}$ があるものと仮定する。このとき、ネット n_i は複数の変形チャネル tc_j に割当ててはできないので、1通りの配線パターン pt_k だけを選択する必要がある。ネット n_i の配線パターン pt_k に対応する変数を $y_{ik} (y_{ik} \in \{0, 1\})$ とする。これは、 $y_{ik} = 1$ ならば、ネット n_i がパターン pt_k で配線されることを意味する。よって、セグメント制約は

$$\sum_{k=1}^{m_i} y_{ik} = 1, \forall n_i \in N \quad (1)$$

となる。

次にカラム制約について説明する。変形チャネル tc_j のセル上のあるカラム x_j に全く端子がない場合には、カラム x_j を利用してネットを本来の端子があるセル列と接する変形チャネルだけでなく、少し離れた変形チャネルで配線することが可能となる(図2)。このとき、そのカラム x_j に対して、どのネット $n_i \in N$ の配線パターン pt_{jk} を通過を許すかどうかで制約式を設ける必要がある。これは、あるカラム x_j を通過することができるネットの配線パターンの集合 $pt_{jk} \in PT_{x_j}$ に対応する変数 $y_{jik} \in Y_{x_j}$ に対して、

$$\sum_{y_{jik} \in Y_{x_j}} y_{jik} \leq 1, \forall tc_j \forall x_j \text{ without pin} \quad (2)$$

となる。

最後に密度制約について説明する。これは変形チャネル tc_j にどのネット $n_i \in N$ を割当ててくるかを2節で述べた $cap_j^l(x), cap_j^b(x)$ を考慮しながら表現する。セル上のトラック数を H 、ゾーン z_{ji} に配線可能なネットの配線パターン集合 $pt_k \in PT_{z_{ji}}$ に対応する変数 $y_{jik} \in Y_{z_{ji}}$ 、ゾーン z_{ji} の上側のセル列 r_{j+1} の最大の $cap_{j+1}^b(x_j)$ を $Cap_t(z_{ji})$ 、下側のセル列 r_j の最大の $cap_j^b(x_j)$ を $Cap_b(z_{ji})$ とする。このときの制約式は、

$$\sum_{y_{jik} \in Y_{z_{ji}}} y_{jik} + (H - Cap_t(z_{ji})) + (H - Cap_b(z_{ji})) \leq CT_{z_{ji}}, \forall z_{ji} \in Z, \forall tc_j \in TC$$

となる。この式は、ゾーン z_{ji} を通過するネットから上下セル列 r_{j+1}, r_j のセル上配線可能なトラック数を引いたものを変形チャネル内の配線密度として表現している。

以下に0-1整数計画問題を示す。ここで得られたネット割当ての解に基づいて、フェーズ2以降を実行する。

【ネット割り当て問題】

$$\text{minimize} \quad \sum_{j=0}^k D_j$$

$$\begin{aligned}
s.t. \quad & \sum_{k=1}^{m_i} y_{jik} = 1, \forall n_i \in N \\
& \sum_{y_{jik} \in Y_{x_j}} y_{jik} \leq 1, \forall l_{c_j} \forall x_j \text{ without pin} \\
& \sum_{y_{jik} \in Y_{z_{ji}}} y_{jik} + (H - Cap_i(z_{ji})) \\
& + (H - Cap_h(z_{ji})) \leq CT_j, \forall z_{ji} \in Z, \\
& \forall l_{c_j} \in TC
\end{aligned}$$

3.2 フェーズ2：セル上配線アルゴリズム

フェーズ1では、変形チャネル l_{c_j} の外周を構成する端子のネット集合、すなわち、変形チャネル l_{c_j} 内で配線するネット集合 $N_j = \{n_1, n_2, \dots, n_{q_j}\}$ を決定した。フェーズ2では、フェーズ1で割当てたネット集合からセル上で配線するネットの決定を行なう。セル上配線アルゴリズムでは、ネットを部分的にセル上配線することを繰り返しながら、セル上配線パターンを求めている。フェーズ2を全ての変形チャネル l_{c_j} に対して順に適用した後、セル上配線は終了する。変形チャネル l_{c_j} は0～ d まで順次処理するため、以降では1つの変形チャネル l_{c_j} について着目する。以下にフェーズ2の変形チャネル l_{c_j} に対するセル上配線問題に対する定式化を示す。

【セル上配線問題】

入力： (1) 変形チャネル l_{c_j} に割当てられたネット集合: $N_j = \{n_1, n_2, \dots, n_{q_j}\}$
 (2) 隣接するセル列 r_j, r_{j+1} にある障害物集合
 出力： セル上配線パターン
 制約： 障害物と交差しない
 目的関数： 変形チャネル密度 D_j の最小化 □

フェーズ2では、ネット n_i の水平セグメント s_i をセル上の端子が存在しないカラム x_j を利用することにより、部分的にセル上配線を行なうことを繰り返しながら変形チャネル密度を減少させる。

提案するセル上配線アルゴリズムは、大きく分けて2つの手続きを繰り返すことにより構成されている。最初に手続1では、変形チャネル密度に大きく寄与するネット n_i に対してセグメント s_i ごとにセル上配線を行なう。次に手続2では、手続1でセル上配線ネットとして選択されなかったネット n_i に対して、端子が存在しないカラム x_j を利用することによりセグメント s_i の一部をセル上配線する。

セル上配線ネットの選択にはセル上のトラック $l_k \in T_j$ における重みつき水平制約グラフ $G(l_k) = (V, E)$ を利用

して求める。セル上のトラック l_k に対して、セル上配線ネットを求めるときの $G(l_k)$ の作成方法について説明する。 $G(l_k)$ における節点は、処理トラック l_k においてセル上II-V配線可能なネット集合 $N_{l_k} \in N_j$ から構成される。また、枝は、 $G(l_k)$ の節点 $v_{n_i}, v_{n_m} \in V$ として選択されたネット n_i, n_m 間に重なりがなく、かつ $r(n_i) < l(n_m)$ の場合、すなわち、ネット n_i からネット n_m に有向枝 $e = (v_{n_i}, v_{n_m}) \in E$ を挿入する。図3に $G(l_k)$ の例を示す。例中の $G(l_k)$ は上側のセル列 r_{j+1} の処理トラック l_k に対して作成されている。節点 V は処理トラック l_k でセル上配線可能なネットであり、ここではネット $n_1 \sim n_6$ が選択されている。有向枝に関しては、例えば、 n_1 と n_2 は互いにセグメントが交差しているため、有向枝は挿入されていない。これに対して、 n_1 と n_3 では互いのセグメントが交差しており、かつ $r(n_1) < l(n_3)$ なので、左側にある n_1 から右側にある n_3 へ有向枝を挿入することになる。こうして作成した $G(l_k)$ の節点に対して重み付けを以下に行なう。まず、トラック l_k の未配線区間をトラック l_k で配線に使用されていない区間と定義する(図4)。提案するセル上配線アルゴリズムでは上述したように大きく分けて2つの手続きがあり、それぞれの手続きにおいて $G(l_k)$ は利用される。そして、そのときの手続1、手続2に用いる重み付けを以下のようにそれぞれ $w_1(n_i)$ 、 $w_2(n_i)$ と定義する。

$$\begin{aligned}
w_1(n_i) &= \max_{1 \leq x \leq l} \frac{d_j(x)}{D_j} + \alpha \times x_length(n_i) \\
w_2(n_i) &= Sd(n_i) + \beta \times Sl(n_i)
\end{aligned}$$

ここで、 α, β は定数で、 $Sd(n_i)$ は、ネット n_i の未配線区間内における最大局所変形チャネル密度を示し、 $Sl(n_i)$ は、未配線区間内でネット n_i から最も遠い端子が存在しないカラムまでの水平方向の長さを表すものとする。 $w_1(n_i), w_2(n_i)$ は、変形チャネル密度に寄与する度合の高いネットほど大きな値を持ち、セル上配線ネットとして選択されやすくなる。セル上配線アルゴリズムでは $G(l_k)$ から節点の重みの総和が最大のパス $p(l_k)$ を求めることによりセル上配線ネットを決定する。このとき、パス $p(l_k)$ 上の節点の重みの総和を手続1のときは $p w_1(p(l_k)) = \sum_{v_i \in p(l_k)} w_1(v_i)$ 、手続2のときは $p w_2(p(l_k)) = \sum_{v_i \in p(l_k)} w_2(v_i)$ 、で表すものとする。 $G(l_k)$ の重みの総和が最大のパスを求めることは、変形チャネル密度に大きく寄与するネットで、かつ、トラック l_k において互いに交差しないネットの集合を求めることになる。 $w_1(n_i)$ は、セグメント全体を考慮するときを使用し、 $w_2(n_i)$ は、セグメントを部分的に考慮するとき、すなわち、ネットを部分的にセル上配線を行なうときに使用する。セル上配線アルゴリズムは常に隣接する2つセル列 r_j, r_{j+1} を同時に考慮しながら処理を行ない、セ

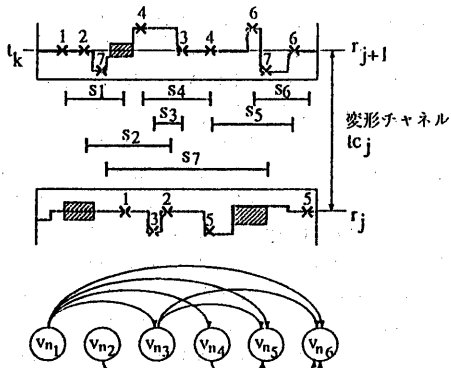


図3 水平制約グラフ $G(l_k)$

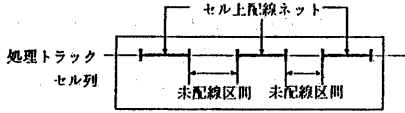


図4 未配線区間

セル列 r_j に対してはトラック II からトラック I へ、セル列 r_{j+1} に対してはトラック I からトラック II へ処理を行なう。これは、変形チャネルから遠いトラックから近いトラックに向かって処理を行なった方が、その逆に処理を行なうより、垂直制約が緩和されるからである。

以下にフェーズ2のセル上配線アルゴリズムを示す。

【フェーズ2：セル上配線アルゴリズム】

(手続1)

Step1: セル列 r_j における処理開始トラック; $bt_j = II - \max_{1 \leq x_j \leq Lcap_j^l(x_j)}$, セル列 r_{j+1} における処理開始トラック; $lt_{j+1} = \max_{1 \leq x_{j+1} \leq Lcap_{j+1}^h(x_{j+1})}$;

Step2: トラック bt_j , lt_{j+1} にセル上配線可能なネット n_i に対して重み $w_1(n_i)$ を計算する;

Step3: $G(bt_j)$, $G(lt_{j+1})$ を作成し、重みの総和が最大のパス $p(bt_j)$, $p(lt_{j+1})$ をそれぞれ求める;

Step4: $p(bt_j)$ と $p(lt_{j+1})$ 上の節点の重みの総和を比較し、もし $pw_1(p(bt_j)) > pw_2(p(lt_{j+1}))$ ならば、 $p(bt_j)$ 上のネットをトラック bt_j に割り当てる; そうでなければ、 $p(lt_{j+1})$ 上のネットをトラック lt_{j+1} に割り当てる;

(手続2)

Step5: 選択したセル列のトラック $st = \{bt_j \text{ or } lt_{j+1}\}$

上において左側にある未配線区間から右側にある未配線区間に向かって順に以下の処理を行なう;

Step5.1: 未配線区間に端子が存在するネット n_i に対して重み $w_2(n_i)$ を計算する;

Step5.2: Step3と同様に $G(st)$ を作成し、重み最大のパス $p(st)$ を求める;

Step5.3: パス $p(st)$ 上のネットをセル上配線する; このときネットは未配線区間でセル上配線可能な区間内でセグメントの一部分をセル上で配線する;

Step5.4: トラック st 上の全ての未配線区間を処理したらStep6へ; そうでなければ、次の未配線区間に移してStep5.1へ;

Step6: $bt_j = H$, $lt_{j+1} = 0$ なら終了; そうでなければ、Step4で選択したセル列が r_j ならば、 $bt_j \leftarrow bt_j - 1$ としてStep2へ; r_{j+1} ならば、 $lt_{j+1} \leftarrow lt_{j+1} - 1$ としてStep2へ;

図5にセル上配線アルゴリズムの配線例を示す。ここではセル列 r_j だけを考慮したときのセル上配線パターンを求めるものとする。また、図5のセル列の下に示すセグメントは、チャネル配線セグメントを表しているものとする。図5(b)に示すように、チャネルから最も近いトラック4からアルゴリズムを適用する。図5(b)では、トラック上にある n_2 と n_3 がセル上配線ネットの候補となる。この例では、 n_2 をセル上配線ネットとして選択し、第3層を利用してセル上配線を行なっている。更に n_3 に関しては、右側の端子が未配線区間にあるので、部分的なセル上配線を行なうことが可能である。ここでは、未配線区間にある右側の端子から左側の端子の方向へ向けて、セル上配線ネットである n_2 と交差しない端子がないカラムまでセル上配線を行なっている。そして、その端子がないカラムの位置を新たに n_3 の右側の端子の位置とする。この操作により、 n_3 の水平方向の長さ $x_length(3)$ は、 $x_length(3) = 5$ から $x_length(3) = 2$ となる。次の図5(c)でも同様の操作を行なう。このとき注意することは、セル上配線ネットの候補は、トラック上にある n_1 と n_4 だけでなく、このトラックでセル上配線可能な n_3 も候補になることである。以下、同様にして図5(d), (e)のように順次処理を行ない、全てのセル上トラックを処理したとき、セル上配線は終了する。

提案セルモデルでは、任意の位置に端子を持つことができる。そのため、端子が整列している従来モデル[4]に比べ、利用可能な空き端子の数が多くなる。提案セル上配線アルゴリズムは、この空き端子を有効に利用し、チャネル配線で使用するセグメントをなるべく短くすることにより、チャネル密度の減少を図っている。

トラック

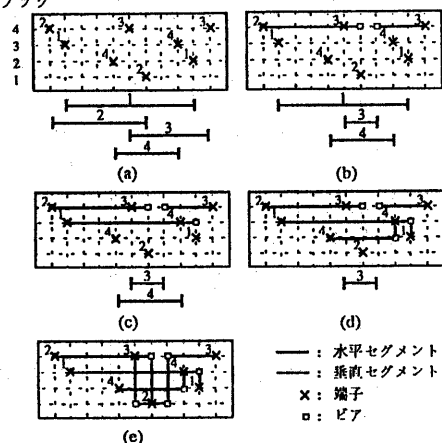


図5 セル上配線アルゴリズム

3.3 フェーズ3：チャネル配線

フェーズ3では、フェーズ2でセル上配線ネットとして選択されなかったネットに対してチャネル内で配線を行ない、全てのネットを100%配線することを目的とする。フェーズ3のチャネル配線手法は、従来から用いられている文献[2]の3層グリーディ・チャネルルータを使用する。この手法は端子がセルの境界に並んでいる従来のセルモデルに対して適用可能であるため、提案セルモデルを従来のセルモデルの形にまで変形する必要がある。これは単純に端子をセル上第2層を利用してセルの境界まで引き出すことにより変形を行なうものとする。こうして各チャネル c_j の上下のセル列 r_{j+1}, r_j のセルの境界に端子を並べたモデルに対して文献[2]の手法を適用し、セル上チャネル配線アルゴリズムは終了する。

4 実験結果

提案アルゴリズムをサン・マイクロシステムズ社のSPARCserver1000上(135MIPS)でC言語を用いて実現した。実験は、表1に示すMCNCベンチマークデータを入力データとして使用している。MCNCベンチマークデータは配置をTimberWolfSC4.2c[11]で行ない、概略配線を本研究室で開発したスタンダードセル概略配線システムGLORIA[9]を用いて行なっている。MCNCベンチマークデータのセルライブラリでは、端子はセルの上下の境界に存在し、更に等電位端子を持つため、提案セルモデルと異なる。そこで、今回の実験では上下のセル境界にある等電位端子を1つにし、それをセル上のトラックにランダムに配置したものを用いた。ここで、端子位置のトラック番号は1~MaxTrackのトラック番号までをランダムに割り当てた。表1に実験データを示す。入力データのprimSC1に対して、端子がセルの境界に並ぶ

表1 入力データ

Data	#Nets	#Cols	#Rows	MazTrack
primSC1	1266	447	27	12
mult16	264	2086	2	6
struct	116	1270	2	6

従来のセルモデル(BTM)にセル上II-V配線を行なう従来の手法と提案するセル上配線アルゴリズムとをチャネル内のトラック数の総和で比較してみた。このとき、従来の手法におけるセル上配線容量は、セルの上下の境界から均一に6トラックとした。言い換えれば、セルの高さは12トラックである。

表2中の T_{before} はセル上配線前のトラック総数、 T_{after} はセル上配線後のトラック総数を表す。また、con.は従来の手法を適用したときのトラック総数、pro.は提案手法を適用したときのトラック総数をそれぞれ示す。表2より、従来の手法と比較して、55%のトラック数を削減し、提案手法の有効性を確認することができる。

次に入力データ中のmult16, structに対して、提案セル上配線アルゴリズムと従来のセル上配線アルゴリズムとの比較実験、提案セルモデルと従来のセルモデルとの比較実験を行なった。ここで、従来のセル上配線アルゴリズムとはネットの部分的な配線を行なわないII-V配線のことを示し、従来のセルモデルとはBTMのことを示している。セル上配線アルゴリズムの比較実験の結果を表3に、セルモデルの比較実験の結果を表4に示す。表3中のalg_con., alg_pro.はそれぞれ従来のセル上配線アルゴリズムの適用結果、提案したセル上配線アルゴリズムの適用結果を示し、表4中のmod_con., mod_pro.は従来のセルモデルでの実験結果、提案セルモデルでの実験結果を示している。表3,4の実験結果より、提案セル上配線アルゴリズム、提案セルモデルの有効性の確認をすることが出来る。提案セル上配線アルゴリズムが有効な理由として、端子がセルの境界にないためにセル上でのドックレッグ配線を利用してトラック数の増加につながるサイクル制約の発生を回避することができるからである。提案セルモデルが有効な理由として、端子がセルの境界から離れて配置されているため、従来のセルモデルよりセル上配線領域が広くなることや、配線するときネット間に生じる垂直制約をセル上領域で処理することができることなどが考えられる。

5 あとがき

本研究では、端子の位置が任意に設定可能であり、かつセル上第2層に障害物を持つセルモデルを新しく導入し、それに対する3層セル上チャネル配線アルゴリズムを提案した。提案手法では、提案セルモデルに対し、部分的にセル上配線を行なうことにより、より多くのネッ

トをセル上配線ネットとして選択している。また、シミュレーション実験により、提案手法の有効性と提案セルモデルの有効性を確認した。今後の課題として、他のセル上配線アルゴリズムとの比較、更にチャンネル密度を減少させるためのセル上再配線による改良などが挙げられる。

謝辞

本研究を行なうにあたり、プログラムの作成にご協力頂いた本学学部生 根来靖典君に感謝致します。本研究の一部は文部省科学研究補助金試験研究 (B)(2) (課題番号 06558042) による。

参考文献

- [1] S. Bhingarde, A. Panyam and N. A. Sherwani: "Efficient OTC routing algorithm for via-less middle terminal models," Proc. of the European Conference on Design Automation, pp. 127-132 (1993).
- [2] P. Bruell and P. Sun: "A greedy three layer channel router," Proc. of International Symp. on Circuits & Syst., pp. 298-300 (1985).
- [3] J.-S. Cherng, S.-J. Chen and C. Changfan: "An efficient over-the-cell channel router," Proc. of the European Conference on Design Automation, pp. 133-137 (1993).
- [4] J. Cong, B. Preas and C. L. Liu: "General models and algorithms for over-the-cell routing in standard cell design," Proc. of 27th ACM/IEEE Design Automation Conference, pp. 709-715 (1990).
- [5] J. Cong, M. Hossain and N. A. Sherwani: "A provably good multilayer topological planar routing algorithm in IC layout design," IEEE Trans. Comput.-Aided Design of Integrated Circuits & Syst., 12, 1, pp. 70-78 (1993).
- [6] J. Cong and C. L. Liu: "Over-the-cell channel routing," IEEE Trans. Comput.-Aided Design of Integrated Circuits & Syst., 9, 4, pp. 408-418 (1990).
- [7] N. D. Holmes and N. A. Sarrafzadeh: "Utilization of vacant terminals for improved over-the-cell channel routing," Technical Report, Western Michigan University (1991).
- [8] N. D. Holmes and N. A. Sarrafzadeh: "Utilization of vacant terminals for improved over-the-cell channel routing," IEEE Trans. Comput.-Aided Design of Integrated Circuits & Syst., 12, 6, pp. 780-792 (1993).
- [9] T. Koide, S. Wakabayashi and N. Yoshida: "A new global routing algorithm for over-the-cell routing in standard cell layouts," Proc. of the European Conference on Design Automation with EURO-VHDL'93, pp. 116-121 (1993).
- [10] M.-S. Lin, H.-W. Perng, C.-Y. Hwang and Y.-L. Lin: "Channel density reduction by routing over the cell," Proc. of 28th ACM/IEEE Design Automation Conference, pp. 120-125 (1991).
- [11] C. Sechen and K. Lee: "An improved simulated annealing algorithm for row-based placement," Proc. of International Conference on Computer-Aided Design, pp. 478-481 (1987).
- [12] N. A. Sherwani: "Algorithms for VLSI Physical Design Automation," Kluwer Academic Publishers (1993).
- [13] B. Wu, N. A. Sherwani, N. D. Holmes and M. Sarrafzadeh: "Over-the-cell routers for new cell model," Proc. of 29th ACM/IEEE Design Automation Conference, pp. 604-607 (1992).
- [14] T. Yoshimura and E. Kuh: "Efficient algorithms for channel routing," IEEE Trans. Comput.-Aided Design of Integrated Circuits & Syst., CAD-1, 1, pp. 25-35 (1982).

表 2 提案手法と従来手法との比較

Data	T_{before}	T_{after}		
		con.	pro.	diff
primSC1.1	8	5	1	4
primSC1.2	5	2	0	2
primSC1.3	5	3	0	3
primSC1.4	6	3	1	2
primSC1.5	5	2	1	1
primSC1.6	6	3	2	1
primSC1.7	6	2	1	1
primSC1.8	7	3	2	1
primSC1.9	9	6	2	4
primSC1.10	8	3	2	1
primSC1.11	12	8	3	5
primSC1.12	10	5	3	2
primSC1.13	10	5	3	2
primSC1.14	11	5	4	1
primSC1.15	11	6	2	4
primSC1.16	9	5	3	2
primSC1.17	9	5	2	3
primSC1.18	11	7	3	4
primSC1.19	8	4	2	2
primSC1.20	10	5	3	2
primSC1.21	11	5	2	3
primSC1.22	7	3	2	1
primSC1.23	9	4	1	3
primSC1.24	7	3	1	2
primSC1.25	7	2	1	1
primSC1.26	9	3	1	2
primSC1.27	5	0	0	0
primSC1.28	0	0	0	0
total	221	107	48	59

表 3 セル上配線アルゴリズムの比較

Data	T_{before}	T_{after}		
		alg.con.	alg.pro.	diff
mult16	9	4	2	2
struct	14	10	7	3

表 4 セルモデルの比較

Data	T_{before}	T_{after}		
		mod.con.	mod.pro.	diff
mult16	9	3	2	1
struct	14	9	7	2