

## ULSI 多層配線に対するバッファ挿入と配線幅を考慮した タイミングドリブン階層概略配線手法

出口貴浩 小出哲士 若林真一

広島大学工学部

〒739-8527 東広島市鏡山一丁目4番1号

E-mail:{degu, koide, wakaba}@ecs.hiroshima-u.ac.jp

あらまし

本稿では配線層が6層のディープサブミクロンVLSIに対するレイアウト設計において、与えられたタイミング制約を満たす概略配線経路を数理計画法を用いて階層的に求める手法を提案する。提案手法では配線遅延モデルとしてElmore遅延モデルを仮定し、セル間のすべての配線は層ごとに配線幅の異なる6層の配線層で行われるものとする。提案概略配線手法においては配線を階層的に詳細化することにより、配線が局所的に混雑することを避ける。各階層における配線では、数理計画法を用いて複数のネットの配線経路を同時に決定する。その際に与えられたタイミング制約を満たすために、配線経路の決定とバッファの挿入、および配線幅の決定を同時に行う。

キーワード 概略配線, 階層配線, 多層配線, バッファ挿入, 配線幅調整

## A Timing-Driven Hierarchical Global Routing Method with Buffer-Insertion and Wire-Sizing for Multi-Layer ULSI

Takahiro DEGUCHI, Tetsushi KOIDE and Shin'ichi WAKABAYASHI

Faculty of Engineering, Hiroshima University

4-1, Kagamiyama 1 chome, Higashi-Hiroshima 739-8527 JAPAN

Abstract

In this paper we present a timing-driven hierarchical global routing method with wire-sizing and buffer-insertion for multi-layer ULSI. The proposed algorithm adopts the Elmore delay as the interconnection delay model. All nets are routed on each layer which has a different wire width. In our routing method, we adopt a hierarchical routing approach to avoid making locally congested regions. In each hierarchical level, the routes of nets are determined by solving a linear programming considering wire-sizing and buffer-insertion under timing constraints.

key words global routing, hierarchical routing, multi-layer, buffer-insertion, wire-sizing

## 1 まえがき

近年の半導体技術の急速な発展により、回路はますます大規模化する傾向にあり、数百万ゲートを搭載したVLSIが実現されるようになってきている。さらに、微細化が進むに伴い、回路遅延に対する配線遅延の割合が大きくなってきている。回路全体の遅延に対する配線による遅延の割合は0.18 $\mu\text{m}$ ルールでは80%以上に達すると言われている。配線遅延はレイアウト設計の結果で決まるため、配線遅延を考慮した配線手法が近年盛んに研究されている[1,2]。また、回路の大規模化により配線するネット数が大きくなると配線の品質も問題となってくる。例えば、配線が密になりすぎると隣接する配線との相互影響によりクロストークノイズによるパフォーマンスの低下などが起きてくる[4,5]。従って、大規模回路の配線システムにおいてはタイミング制約と共に配線混雑度とクロストークのトレードオフの考慮も重要となってくる。また、近年の半導体技術の進歩で、配線層数が6層以上の多層配線が一般的になりつつあり、そのような多層配線層を持つVLSIに対する配線手法の開発も望まれている。

VLSIレイアウト設計の概略配線手法としてはこれまでも多くの手法が提案されている。文献[6]ではネットの配線を重み付きスタイン木を用いて逐次配線していく手法が提案されている。しかし、この手法は多層配線における層割り当て問題を扱っておらず、また配線が混雑した場合の再配線処理に計算時間が非常にかかるため大規模回路への適応は困難である。また、文献[9]では配線長と配線混雑度を同時に考慮した概略配線手法が提案されているが、配線遅延は考慮されていない。

一方、タイミング制約を考慮した配線手法もこれまでに提案されている。配線遅延を減少させる手法として、配線幅調整[7]、バッファ挿入[8]といった手法が研究されているが、これらの手法は与えられた配線トポロジにおいて配線幅調整やバッファ挿入により遅延を減少させる手法であり、多数のネットをチップ上で実際に配線する場合に生じる配線混雑などの物理的制約が考慮されていない。

そこで本稿では数十万ゲート以上の超大規模集積回路(ULSI)を対象としたタイミングドリブン概略配線手法を提案する。本手法においては配線遅延モデルとしてElmore遅延モデルを仮定し、すべての配線は6層の配線層で行われるものとする。提案概略配線手法においては配線を階層的に詳細化することにより、配線が局所的に混雑することを避ける。また、与えられ

たタイミング制約を満たすために、配線経路の決定とバッファの挿入、および配線幅の決定を同時に行う。

## 2 準備

### 2.1 配線遅延モデル

本稿では配線遅延モデルとして以下に示すElmore遅延モデル[3]を用いる。ネット $n$ について、 $c_{n_v}, r_{n_v}$ はそれぞれノード $v$ に入ってくるネットセグメントの配線容量と配線抵抗とし、 $l_{n_v}$ を $v$ に入ってくるセグメントの配線長、 $w_n$ をネットの配線幅とする。また、 $c(T_v)$ をノード $v$ を根とする配線木の配線容量の総和とし、 $D_{n_v}$ はネット $n$ の根からノード $v$ までの遅延である。 $path\{0, v\}$ をノード0から $v$ までのパス上のノード集合とする。ノード $v$ までのElmore遅延 $D_{n_v}$ は、

$$c_{n_v} = \alpha l_{n_v} \cdot w_n + cf \cdot l_{n_v} \quad (1)$$

$$r_{n_v} = \beta l_{n_v} / w_n \quad (2)$$

$$d_{n_v} = r_{n_v} \left( \frac{c_{n_v}}{2} + c(T_v) \right) \quad (3)$$

$$D_{n_v} = \sum_{j \in path\{0, v\}} d_{n_j} \quad (4)$$

となり $c_{n_v}, r_{n_v}$ は $l_{n_v}, w_n$ の関数となっている。

ディープサブミクロンVLSIの設計ルールでは、従来は無視できていたフリンジング容量の割合が大きくなってきているため、遅延モデルにおいてこれを考慮することも重要である[5]。上記の遅延モデルにおいては単位長当りのフリンジング容量を $cf$ とすることによりフリンジング容量を考慮している。

### 2.2 レイアウトモデル

本稿で仮定するレイアウトモデルはセルベースモデルとし、各セル列間には配線領域は存在せず(チャンネルレス)、セル間の配線はすべて専用の配線層で行われるものと仮定する。また、任意のセルの高さ、幅を持つマクロセルを含む場合も考慮する(図1)。

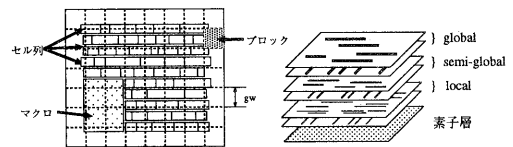


図1 レイアウトモデル

配線層は6層を仮定し、2層ずつX方向配線層、Y方向配線層の対になっているものとする。以下では配線

層  $i$  とは  $i$  番目の配線層対を意味するものとする ( $i = 1, 2, 3$ ).  $i$  番目の配線層対の配線幅を  $w_i$ , 配線間隔を  $s_i$  とし, 配線ルールを  $g_i = w_i + s_i$  とする. 3つの配線層対の配線ルールは  $g_1 < g_2 < g_3$  を満足するものとし, それぞれの配線層対を local, semi-global, global 配線層と呼ぶ.

隣接する配線層間はビアで接続するものとし, ビアによる遅延を  $vd_{ij}$  とする. 配線遅延を改善するためのバッファは他のセルと同様にセル列に置かれ, 配線時にバッファを挿入する場合は既配置セルを左右にシフトすることにより挿入されるものとする. ただし, バッファ挿入によるチップ面積の大幅な増加を防ぐため, 各ブロックに挿入できる最大バッファ数を  $B_{max}$  とする. ここで, ブロックとはチップの配置領域を矩形に分割した場合の各矩形を表す. ブロックの詳細は次節で説明する. さらに, 1セル列に挿入可能な最大バッファ数を  $RB_{max}$  とする.

バッファ挿入後のセルと端子位置の座標の変更はごく僅かであり, 次節で定義する概略配線をモデル化したグラフ上においては端子位置, 配線経路への影響はないものとする.

### 2.3 グラフモデル

提案手法では階層的に概略配線を行うため, チップの配置領域全体を再帰的に  $4 \times 4$  の領域に分割する (図2). 初期分割をレベル1とし, 以下レベル  $n$  まで分割する. 各レベルにおいて分割されてできる部分領域をブロックと呼ぶ.

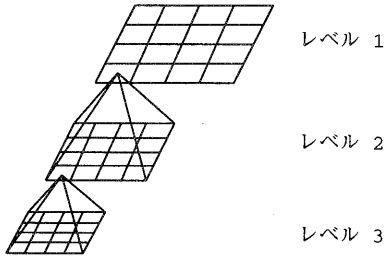


図2 階層分割

いま, レベル  $l$  におけるブロックを節点とし, 上下左右に隣接するブロック間を枝で接続することにより得られるグラフを階層グラフ  $G_l = (V_l, E_l)$  とする.

レベル  $l$  における座標  $i, j$  のブロックに対応する節点を  $v_{ij(l)} \in V_l$  とする.  $V_l$  の節点  $v_{pq(l)}$  はレベル  $l+1$  では  $4 \times 4$

の節点集合  $V_{l+1}^{vpq(l)} = \{v_{ij(l+1)} : 4p \leq i \leq 4p+3, 4q \leq j \leq 4q+3\}$  に分割され, 隣接する節点との枝により,  $v_{pq(l)}$  の分割グラフ  $DG_{l+1}^{vpq(l)} = (V_{l+1}^{vpq(l)}, E_{l+1}^{vpq(l)})$  が定義される (図3).

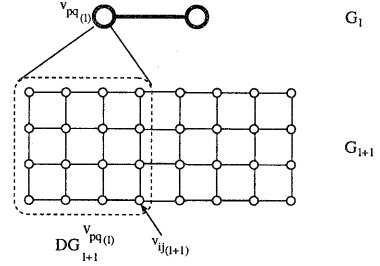


図3 分割グラフ

$G_l$  において水平方向に隣接する2つの節点  $v_{ij(l)}, v_{i+1j(l)}$  間の枝  $e_{i,i+1(l)} \in E_l$  は各配線層の配線容量を

$CAP_{i,i+1(l)} = (cap_{i,i+1(l)}^{(1)}, cap_{i,i+1(l)}^{(2)}, cap_{i,i+1(l)}^{(3)})$  の3項組として持つ.  $G_l$  の垂直方向の枝についても同様に定義する.

ここで, 各階層グラフの枝の配線容量はレベル  $l+1$  の配線容量に基づいて設定される. レベル  $n$  での  $G_l$  の配線容量はブロックの一辺の長さを  $gw_n$ ,  $h$  層の配線ルールを  $g_h$  とすると,  $cap_{i,i+1(l)}^{(h)} = \frac{gw_n}{g_h}$  とする.

マクロセル上を配線禁止とする場合, 対応する階層グラフの枝  $e_{i,i+1(l)}$  の配線層  $h$  の配線容量を  $cap_{i,i+1(l)}^{(h)} = 0$  とする.

階層グラフの枝  $e_{i,i+1(l)}$  を通る各層の配線本数は

$F_{i,i+1(l)} = (f_{i,i+1(l)}^{(1)}, f_{i,i+1(l)}^{(2)}, f_{i,i+1(l)}^{(3)})$  の3項組として持つ. ここで  $e_{i,i+1(l)}$  の  $h$  層における配線混雑度を

$$dns_{i,i+1(l)}^{(h)} = cap_{i,i+1(l)}^{(h)} - f_{i,i+1(l)}^{(h)}$$

とする. すなわち,  $dns_{i,i+1(l)}^{(h)} \geq 0$  で全てのネットが配線されなければならない.

タイミング制約については, 任意のネット  $n_j$  においてそのシンク  $s_{ij}$  に許される最大許容遅延時間  $req(s_{ij})$  が与えられ, ネットの全てのパスの遅延時間が  $req(s_{ij})$  以下であるならば, そのネットはタイミング制約を満たすものとする.

各レベルにおいてネット  $n_j$  の端子は階層グラフ  $G_l$  上の節点に置かれ, 配線経路はネットのソースを根とし,  $V_l' \subset V_l$  とソースからの有向枝  $E_l' \subset E_l$  とで与えられる

スタイナ木  $T_l(n_j) = (V_l', E_l', L)$  で表す。ここで  $T(n_j)$  の枝  $e_{(i)}$  が使用する配線層は関数  $L(e_{(i)})$  で与える。求める概略配線経路は最下位レベル  $n$  での  $T_n(n_j)$  である。

階層分割において、レベル  $l+1$  に分割する際、レベル  $l$  での配線は分割領域の境界に仮想端子を設定することでレベル  $l+1$  の各領域における配線として詳細化される。

## 2.4 概略配線問題の定式化

入力： ネットリスト  $(C, N)$

( $c_i \in C$  : セル集合,  $n_j \in N$  : ネットリスト)

タイミング制約  $(T)$

バッファ集合  $(B)$

セル情報 (位置, 大きさ, 端子位置)

各配線層の配線幅  $w_k$ , 配線間隔  $s_k$

最大階層レベル  $n$

出力： 各ネットの配線経路, 配線層

バッファ挿入位置

目的： 最大混雑度の最小化

制約： タイミング制約  $d_{s_{ij}} < req_{s_{ij}}$

配線混雑度  $dns_{ij} \geq 0$

最大バッファ挿入数  $B_{max}, RB_{max}$

## 3 提案手法

本稿ではセル配置が与えられた数十～百万ゲート規模のVLSI(ULSI)を対象とし、与えられたタイミング制約を満たす概略配線経路を求める手法を提案する。大規模回路の配線においては配線混雑度を考えずに配線しようとする、配線容量制約を満たさない配線が生じた場合に引き剥し再配線を行ったとしても、局所的に混雑している場合は再配線の繰り返し回数が増加し、非常に効率が悪い場合がある。また、再配線で経路を変更した場合、タイミング制約が満たされなくなる場合も生じる。配線層が多層であると引き剥し再配線はさらに複雑な問題となる。このような問題に対処するために、本手法では階層的に概略配線を行う。まず、上位レベルにおいて配線混雑度を考慮した粗い配線経路を決定する。この配線経路を段階的に詳細化していき、各階層レベルにおいて配線混雑度を均一化することにより、引き剥し再配線の適用回数を減少でき、効率的に配線を行うことができる。また、各レベルでバッファ挿入と異なる配線幅を配線経路として与えることにより、タイミング制約を考慮した概略配線が可能となる。

### 3.1 手法の概要

初期分割レベル1より、ブロックをトップダウンにレベル  $n$  まで階層的に  $4 \times 4$  の領域に分割することにより、配線を詳細化して行く。レベル  $l$  で求められた配線経路は経路が通る各ブロックにおいて仮想端子を与えることにより、複数の経路に分割される。分割された配線はレベル  $l+1$  の  $4 \times 4$  の領域で詳細化される。このとき、バッファがレベル  $l$  のブロックに挿入されている場合はバッファを端子としてレベル  $l+1$  のブロックに置かれるものとする。

レベル  $l$  での配線は階層分割された  $4 \times 4$  の領域ごとに行う。ここでは混雑度を考慮しながら複数のネットの同時配線を行うために、数理計画法を用いて配線対象となるネットリスト  $N'$  の配線経路を求める。このとき、配線対象となるすべてのネットに対して一度に配線層の選択、バッファ挿入を考慮した配線経路を求めるのは困難なため、ネット数  $|N''| = \lambda (N'' \subset N')$  の部分ネットリストを入力とし、数理計画法を  $m$  回 ( $|N''| \leq m|N''|$ ) 繰り返すことによりネットリスト  $N'$  の配線経路を求める。

レベル  $n$  まで配線の詳細化を行うと、最下位レベルでの各ブロックの配線混雑度が求まる。この配線混雑度を考慮してボトムアップに、タイミング制約、配線容量を違反しているネットに対して配線経路を変更する。

フェーズ 1： 配置領域を階層的に  $4 \times 4$  のブロックに分割。  $l \leftarrow 1$ 。

フェーズ 2： レベル  $l$  の各  $4 \times 4$  の部分領域において数理計画法により配線経路を決定。

フェーズ 3： 各ブロックの階層分割を行い、レベル  $l$  での配線経路をレベル  $l+1$  の各  $4 \times 4$  の部分領域に分割。

フェーズ 4：  $l \leftarrow l+1$  とし、フェーズ 2, 3 をレベル  $n$  まで繰り返す、その後タイミング違反しているネットに対してボトムアップに再配線。

### 3.2 フェーズ1(初期分割)

フェーズ1ではチップの配置領域を階層的に領域分割し、レベル  $n$  までの階層グラフ  $G_l$  を作成する。そしてレベル  $n$  での配線容量、配線禁止領域を設定することにより、ボトムアップに各階層における配線容量の

設定を行う。

レベル  $l$  の水平方向  $e_{i,i+1}$  の枝の配線容量は両端の節点に対応するレベル  $l+1$  の節点間を結ぶ水平方向の枝の容量の平均を  $cap_{h_{l+1}}$  とする ( $h = 1, \dots, 4$ ) と,  $cap_{i,i+1(l)} = \sum_h cap_{h_{l+1}}$  により配線容量を見積もる。これにより各階層の配線容量を設定する。

### 3.3 フェーズ 2 (4×4 配線)

フェーズ 2 ではレベル  $l$  における各  $4 \times 4$  の部分領域での配線経路を数理計画法を用いて決定する。各  $4 \times 4$  の部分領域において配線対象となるネット集合を  $N'$  とする。

- ステップ 1:  $N'$  の各ネットの遅延スラックを計算。
- ステップ 2:  $N'$  より  $|N''| = \lambda$  のネットを遅延スラックの順に選択し,  $N''$  の各ネットの配線パターンを生成。
- ステップ 3: ステップ 2 で作成された  $N''$  の配線パターンを入力とし, 数理計画法により配線経路を決定。
- ステップ 4: ステップ 2, 3 を  $m$  回繰り返す。

#### 3.3.1 ステップ 1 (遅延スラック計算)

配線対象となる  $N'$  の各ネットに対し, シンクにおける遅延スラックを計算する。ここで, 配線対象となるネットは階層分割によって設定される仮想端子 (仮想ソース, 仮想シンク) を含む場合がある。ネットの配線遅延は階層分割により設定された仮想ソースを根として各端子間の半周近似により見積もる。ブロック内に複数端子が含まれる場合は各シンクの端子位置の中心を根とする部分木として見積もる。

各シンク  $s_{ij}$  の遅延スラックを

$$Sl(n_j, s_{ij}) = req(s_{ij}) - d(s_{ij})$$

とする。  $Sl(n_j, s_{ij}) \leq d_{th}$  の場合,  $s_{ij}$  が含まれるシンク, 又は  $s_{ij}$  のパス上の仮想シンクをクリティカルシンクとし, ネット  $n_j$  はクリティカルである呼ぶ。ここで  $d_{th}$  はユーザが与える定数である。

ネット  $n_j$  のスラックは各シンクの遅延スラックの最小値とする。

$$Sl(n_j) = \min\{Sl(n_j, s_{ij}) | \forall\}$$

#### 3.3.2 ステップ 2 (配線パターン)

$N'$  のネット集合から, ネットのスラックの小さい順, つまりタイミングのクリティカルなシンクを含むネットの順に  $\lambda = |N''|$  個のネットを選択する。選択された各ネット  $n_j \in N''$  の配線パターン  $p_{ji}$  を生成する。生成されたネット  $n_j$  の配線パターン集合  $P_j$  はステップ 3

の数理計画法の入力となる。  $4 \times 4$  の領域において, 配線パターンとして 3 種類の配線層, バッファ挿入を行う場合をすべて生成すると, 数理計画法の入力変数の総数が飛躍的に増加するため, 各ネット  $n_j$  についてタイミング制約を考慮したパターン数を最大  $P_{max}$  に制約する。

配線対象となる  $N'$  のネットの端子数は階層分割の性質から, 2 端子ネットとなる場合がほとんどである。  $4 \times 4$  の領域における任意の 2 端子間の配線パターンの列挙は簡単であるため, 配線パターンの生成は 2 端子ネットとそれ以外の場合に場合分けして行う。

タイミングのクリティカルなネットについてはタイミング制約を満足するように, 配線長の短いパターンが生成されることが望ましい。さらに, 配線幅調整やバッファ挿入を行うことにより配線遅延の減少を計ることも必要である。一方, クリティカルでないネットについては配線容量制約を満足するように最短経路での配線以外にも迂回経路による配線パターンを生成する。

タイミング制約については, 階層分割により設定された各仮想端子に接続される部分木と生成される配線パターンに基づいてネットのソース, シンク間の遅延を計算し, ネットの各シンクにおいてタイミング制約を満足する配線パターンを生成する (図 4)。

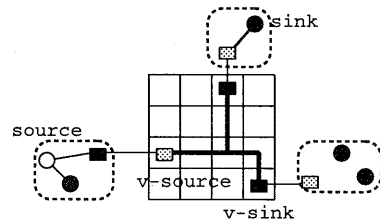


図 4 配線パターンの遅延

#### ● 2 端子ネット

1. 2 端子間の最短経路  $P_{jk}$  を local 層で作成。
2.  $P_{jk}$  クリティカルならば配線幅調整, バッファ挿入。
3.  $P_{jk}$  に含まれる枝以外を使い, 各層の配線混雑度を最小にするパターン  $P'_{jk}$  を作成。
4.  $P'_{jk}$  のパターンでタイミング違反するものは経路上の遅延最小となる位置にバッファを挿入。

#### ● 2 端子ネット以外のネット

1. クリティカルでない場合, 最短木  $T_{jk}$  を生成。

2. クリティカルな場合、クリティカルシンク以外を最短木で生成し、ソース、クリティカルシンク間を最短経路で接続 (CS-Tree [10]).
3. シンクのキャパシタンスの小さい順の組み合わせで、ボトムアップにスタイナ木を生成。このとき、端子間は配線混雑度を重みとした迷路法により接続する。
4. 3. で生成したパターンでタイミング違反するものは配線幅調整、バッファ挿入を行う。

### 3.3.3 ステップ 3

ステップ 3 では、ステップ 2 で生成した配線パターンを入力とし、 $N'$  に含まれる各ネットをどの配線パターンで配線するかを数理計画法で決定する。0-1 変数  $x_{ij}$  を用意し、 $x_{ij} = 1$  はネット  $n_i$  をパターン  $p_{ij} \in P_i$  で配線することを表す。 $k$  層で  $e_{uv(i)} \in E'_i$  を通る配線パターン集合を  $A(u, v, k)$  とする。ネット  $n_i$  のパターン  $p_{ij}$  が  $v_{pq(i)} \in V'_i$  で使用するバッファ数を  $B(i, j, p, q)$  とする。ここで、 $v_{pq(i)}$  のバッファ容量を  $S(v_{pq(i)})$  とする。

$k$  層での配線数スラック  $X(i, j, k)$  の最小値  $Y$  の最大化を目的関数とする数理計画法を  $m$  回繰り返すことにより  $N'$  の配線経路を求める。 $N'' \subseteq N'$  に対して配線パターンを決定する整数線形計画法の定式化を以下に示す。

$$\begin{aligned}
 & \text{maximize} && Y \\
 & \text{subject to} && \\
 & \sum_{1 \leq j \leq |P_i|} x_{ij} = 1 && i = 1, \dots, \lambda \\
 & X(i, j, k) + \sum_{p_{ij} \in A(u, v, k)} x_{ij} = \delta \cdot \text{cap}_{uv(i)}^{(k)} && \forall i, j, k \\
 & \sum_{v_{pq(i)} \in N''} \sum_{p_{ij} \in P_i} B(i, j, p, q) x_{ij} \leq S(v_{pq(i)}) && \forall v_{pq(i)} \in V'_i \\
 & Y \leq X(i, j, k) && \forall i, j, k \\
 & x_{ij} \in \{0, 1\} &&
 \end{aligned}$$

ここで、制約となる各ブロック間の配線容量は領域分割で設定される容量  $\text{cap}_{uv(i)}^{(k)}$  の  $\delta$  倍 ( $0 \leq \delta \leq 1$ ) とする。整数計画法を適用して各ネット  $n_i \in N''$  の配線経路を決定するが、整数計画法では計算負荷が大きすぎる場合は線形計画法として解いた後、得られた実数解を整数化することにより許容解を得る。

### 3.4 フェーズ 3 (階層分割)

フェーズ 3 ではレベル  $l$  のブロックをレベル  $l+1$  に分割し、レベル  $l+1$  の境界となるブロック間に仮想端子を設定する。レベル  $l+1$  の境界上のブロックペアをスロットと呼ぶ。レベル  $l$  の  $e_{i, i+1(i)}$  を通るネットに対

し、レベル  $l+1$  の分割グラフ  $DG_{l+1}^{v_{ij(i)}}$ ,  $DG_{l+1}^{v_{i+1j(i)}}$  の境界上に仮想端子を設定する。スロットに割り当てられる仮想端子はソース側を仮想ソース (v-source)、シンク側を仮想シンク (v-sink) と呼ぶ (図 5)。

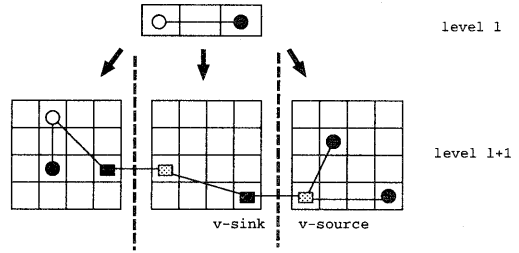


図 5 仮想端子

ネット  $n_j$  をスロット  $u_i$  へ割り当てるコスト関数を  $\text{cost}(n_j, u_i)$  と定義し、スロットへの割り当て問題を線形割り当てにより求める。

コスト  $\text{cost}(n_j, u_i)$  は配線トポロジにより以下の 3 つの場合に分けられる (図 6)。

- case 1 :  $\text{cost}(n_j, u_i) = \alpha \cdot \text{length}(\text{term}, s)$  信号端子を接続。
- case 2 :  $\text{cost}(n_j, u_i) = \beta$  仮想端子を接続。
- case 3 :  $\text{cost}(n_j, u_i) = \gamma \cdot \text{length}(\text{cut}, s)$  仮想端子が配線セグメントの端点。

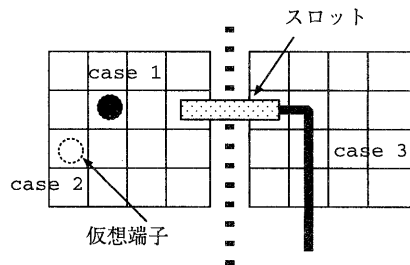


図 6 コスト

ここで、 $\text{length}(\text{term}, s)$  は端子とスロットの矩形距離、 $\text{length}(\text{cut}, s)$  は配線セグメントが通過する境界とスロットとの距離である。これによりネット  $n_j$  のスロ

ト  $u_i$  への割り当てコストを境界の両面から求まる以下の式で与える.

$$cost(n_j, u_i) = cost_1(n_j, u_i) + cost_2(n_j, u_i)$$

仮想端子位置を設定した後、各ネットは  $4 \times 4$  の部分領域に分割され、各境界の仮想ソースの抵抗、仮想シンクにおける要求遅延時間を設定する。ここで仮想ソースを部分木の根とする配線トポロジにより、レベル  $l+1$  の配線遅延を Elmore 遅延モデルにより計算する。

### 3.5 ボトムアップ再配線

最下位のレベルでの配線において、タイミング制約を満足しないネットに対し、配線経路の変更を行う。このとき、経路の探索範囲は対象となるネットを囲む最下位レベルでの階層グラフとし、経路の変更はボトムアップに行い、配線混雑度を緩和する。

## 4 実験の評価

提案手法を C 言語を用いて Ultra COMP Station model 170 上に実現し、シミュレーション実験を行った。実験に用いた各種パラメータを表 1 に示す。入力に用いた配置データは TimberWolf [11] で作成したものである。アルゴリズム中の数値計画法による配線では数値計画パッケージ LOQO [12] を用いて整数計画問題を解いた。各層の配線幅は local, semi-global, global 層をそれぞれ  $0.18\mu\text{m}$ ,  $0.25\mu\text{m}$ ,  $0.5\mu\text{m}$  としている。テストデータ (表 2) に対する実験結果を表 3 に示す。階層レベルによって概略配線の細かさが決定されるが、階層分割によるブロックの大きさがセルの平均の大きさとなるようにし、実験では最大階層レベルを  $n=5$  としている。表中の配線混雑率は各配線層の総配線容量に対する総配線本数の割合  $\sum_{i,j} \frac{cap_{i,j}}{f_{i,j}^k}$  を示している。配線率はフェーズ 3 終了時の評価であり、提案手法により、ほとんどのネットが制約を満足して配線されることがわかる。残りの制約違反のネットの配線はフェーズ 4 の再配線で処理され、制約を満足した 100% 配線が可能である。

## 5 あとがき

本稿では 6 層の配線層を持つ VLSI に対し、バッファ挿入と配線幅調整を考慮した階層概略配線手法を提案した。今後の課題としては、効率的な配線パターン生成アルゴリズムの開発、クロストーク制約の考慮、提案手法の並列化などが挙げられる。また、大規模データに対する実験の評価に基づく提案手法の更なる改良を行う予定である。

表 1 遅延パラメータ (  $0.18\mu\text{m}$  CMOS モデル )

パラメータ	値
最小ドライバ抵抗 ( $\Omega$ )	280
最小ゲートキャパシタンス ( $fF$ )	0.097
単位配線抵抗 $\alpha$ ( $\Omega/\mu$ )	0.076
単位配線キャパシタンス $\beta$ ( $fF/\mu\text{m}$ )	0.044
単位フリンジキャパシタンス $cf$ ( $fF/\mu\text{m}$ )	0.055
バッファのゲート遅延 ( $nsec$ )	0.125

表 2 実験データ

データ	セル数	ネット数	IOパッド数	制約数
C6	1338	1215	301	334
s38584	8622	9002	342	2729
s35932	11838	11873	355	2823
avq_large	25114	25384	64	3422

表 3 実験結果

データ	総配線長 [ $\mu\text{m}$ ]	配線率 [%]	最大遅延 [psec]	バッファ 総数	CPU 時間 [s]
C6	1489291	100	102	41	95
s38584	11757728	96.4	238	89	951
s38932	18281656	97.2	464	128	1295
avq_large	48932138	95.3	939	381	4830

表 4 各配線層の配線長と混雑度

データ	配線層	配線長 [ $\mu\text{m}$ ]	配線混雑率 [%]
C6	global	235287	40.0
	semi-g	385462	41.5
	local	868542	66.3
s38584	global	2861461	58.2
	semi-g	3861461	48.1
	local	5236982	55.7
s38932	global	5236982	62.7
	semi-g	3861461	69.2
	local	3861461	68.6
avq_large	global	14608026	51.7
	semi-g	12458634	59.3
	local	21865478	61.7

## 参考文献

- [1] X. Hong, T. Xue, J. Huang and C.-K. Cheng: "TIGER: An efficient timing-driven global router for gate array and standard cell layout design," *IEEE Trans.CAD*, Vol.16, No. 11, pp. 1323-1331 (1997).
- [2] J. Lillis, C.-K. Cheng and T.-T. Y. Lin: "Optimal wire sizing and buffer insertion for low power and a generalized delay model," *Proc. of International Conference on Computer-Aided Design*, pp. 138-143 (1995).
- [3] E. S. Kuh and M. Shih: "Recent advances in timing-driven physical design," *Proc. of Asia-Pacific Conference on Circuits and Systems*, pp. 23-28 (1992).
- [4] T. Gao and C.L.Liu: "Minimum crosstalk channel routing," *Proc. ACM/IEEE Int. Conf. Computer-Aided Design*, pp. 692-696 (1993).
- [5] M. Lee: "A fringing and coupling interconnect line capacitance model for VLSI on-chip wiring delay and crosstalk," *Proc. of International Symposium on Circuits and Systems*, pp. 233-236 (1996).
- [6] B. S. Ting and B. M. Tien: "Routing techniques for gate array," *IEEE Trans.CAD*, Vol.CAD-2, No. 4, pp. 301-312 (1983).
- [7] T. D. Hodes, B. A. McCoy and G. Robins: "Dynamically-wiresized Elmore-based routing constructions," *Proc. of International Symposium on Circuits and Systems*, pp. 457-460 (1994).
- [8] M. Kang and W. W.-M. Dai: "Delay bounded buffered tree construction for timing driven floor-planning," *Proc. of International Conference on Computer-Aided Design*, pp. 707-717 (1997).
- [9] C.Chiang, C.K.Wong and M.Sarrafzadeh: "A weighted Steiner tree-based global router with simultaneous length and density minimization," *IEEE Trans. Comput.-Aided Design of Integrated Circuits & Syst.*, Vol. 13, No. 12, pp. 1461-1469 (1994).
- [10] K. D. Boese, A. B. Kahng and G. Robins: "High-performance routing trees with identified critical sinks," *Proc. of 30th Design Automation Conference*, pp. 182-187 (1993).
- [11] "TimberWolf 6.0 Macro/Standard Cell Floor-planning, Placement and Routing Package," Yale University (1994).
- [12] "LOQO : A system for solving linear and/or quadratic programing problems," Princeton University (1992).