

# リーフセル合成のための一配線手法

四 宮 典 子<sup>†</sup> 濱 脇 浩 二<sup>††</sup> 福 井 正 博<sup>†</sup>

リーフセルのための有効な配線モデルと配線最適化手法を提案する。高さ方向のサイズが固定されているが、配線領域内の配置要素の上下移動が可能であるレイアウトモデルを前提とする。格子列全体と、格子列を配置要素により分割した区分配線領域の両方に対し、通過可能な配線本数である容量を定義し、これらの容量を管理することにより高さ制約を満足させる。また容量違反と設計規則違反の両方を、コスト関数を用いて管理し、引きはがし再配線により配線経路の最適化を行う。実験により、容量違反と設計規則違反の両方が改善されていく度合いを示し、結果として配線の最適化が行われることを示す。本手法は、実際のセル開発で使用され、実用的効果を上げている。

## A Routing Method for Leaf Cell Design

NORIKO SHINOMIYA,<sup>†</sup> KOJI HAMAWAKI<sup>††</sup> and MASAHIRO FUKUI<sup>†</sup>

This paper presents an efficient routing model and a routing algorithm for leaf cell design. The height of the routing region is fixed. Placement objects in it can be moved vertically, with keeping the relative positions. Capacity is the maximum number of wires passing a region. Grid columns of a cell have the capacity. And the routing regions partitioned by placement objects have the capacity, too. Routing optimization is processed by rip-up and rerouting method, managing both capacity violation and design rule violation with a penalty function. Experimental results show the process of reducing the number of both violations, then finally, optimizing the routing results.

### 1. はじめに

従来のライブラリ開発においては、ライブラリのライフサイクルが約3年以上と長かったため、人手を十分にかけて最適なセル開発を行う方法がとられていた。しかしながら、近年、より短期間に多種類のライブラリを同時開発することが要求され、10倍以上の開発効率向上が課題となっている。我々は、このような状況に対処すべく、ライブラリセルのレイアウト設計の自動化を進めるため、回路のトランジスタ幅の最適化、トランジスタ配置、トランジスタ配線、コンパクション等の最適化システムの開発を行ってきた<sup>1),2)</sup>。本文では、トランジスタレベル配線に適したレイアウトモデルおよび、配線手法について議論する。

配線手法に関しては、従来、標準セル方式 LSI や、ゲートアレー方式、ビルディングブロック方式に対する議論が行われてきた。その配線領域のモデル化の主

なものとしては、チャンネル配線で用いられるように1方向に伸縮可能な配線モデル、あるいは、スイッチボックスや領域配線で用いられているサイズ固定の配線モデルである。チャンネル配線モデルを用いる場合は、チャンネル幅を適切なものにできれば、未結線を生じる心配がないので、グローバル配線と詳細配線に問題を区分し、グローバル配線においては各ネットの概略配線経路を最適化し、詳細配線においては再配線等を用いず、順次、チャンネルごとに配線を完成させていく方法が用いられる。一方、サイズ固定の配線モデルにおいては、配線の処理順等の影響により、処理順の早い配線が後に処理される配線の障害となるような状況が発生し、未結線が生じる場合が多い。そのため、引きはがし再配線による解法が広く用いられている。たとえば、文献3)においては、配線どうしの衝突等を Penalty 関数によって管理し、引きはがし再配線を繰り返すことによって、配線処理順の影響を減らし良好な配線結果を得る方法が提案されている。

ライブラリセルのための配線モデルを考えた場合、配線領域の特徴としては、セルの高さ方向の距離が固定である。配置要素は上下左右に移動が可能である。セルの幅方向の距離は可変である等の特徴を有する。

<sup>†</sup> 松下電器産業株式会社半導体開発本部半導体先行開発センター  
Advanced LSI Technology Development Center, Matsushita Electric Industrial Co., Ltd.

<sup>††</sup> 株式会社松下ソフトリサーチ

Matsushita Soft-Research, Inc.

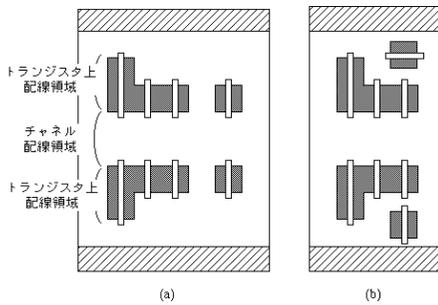


図1 トランジスタ配置配線モデル

Fig. 1 Models for transistor placement and routing.

このような配線モデルに対しては，従来の配線モデルに対する引きはがし再配線手法<sup>3)~9)</sup>をそのままでは適用することが困難である。

従来のセル設計用のトランジスタ間配線技術では文献10)が着目に値する．文献10)は，P型トランジスタ列とN型トランジスタ列を平行に配置するトランジスタ1次元配置モデルを前提とし，同一トランジスタ列内のソース/ドレイン間の接続はトランジスタ上配線領域を用いて行う．トランジスタ列上に収容しきれなかったネットと残りのネットに対しセル中央部においてチャンネル配線を行う(図1(a))．これに対して，我々が前提とする2次元配置モデルは人手設計並みの自由度を持ち，トランジスタの縦積みや，横向きトランジスタを扱うことができる2次元配置モデルを扱う<sup>1),2)</sup>(図1(b))．

トランジスタ配置処理が配線処理に与えるデータは，配置要素(トランジスタ，端子)の配置情報である．ここに端子とは，トランジスタ端子(ゲート端子，ソース端子，ドレイン端子)および，セルの入出力端子のことである．またトランジスタ配置処理は，配置要素間を通過する配線本数を見積もって，それら配置要素の間隔を決定する．しかしながら配置要素間隔を正確に見積もることは困難であり，人手設計では，配線を行いながら配置要素間隔を調整するといった試行錯誤が繰り返される．通過配線本数を少なく見積もった場合，配置要素間隔が狭くなり，格子不足による未結線を生じる．文献4)では，これを解決するため，新たな格子を挿入してレイアウト領域を広げる方法を提案している．たとえば，スタンダードセルのレイアウト設計はセル高さが制約として与えられるが，指定のセル高さ(トラックN本分相当)でトランジスタを配置した場合に，セル中央付近で格子不足による未結線を生じる例を図2(a)に示す．この場合に文献4)の手法は図2(b)のように，格子を1行挿入して配線を完了する．レイアウト領域の拡大により，このままでは

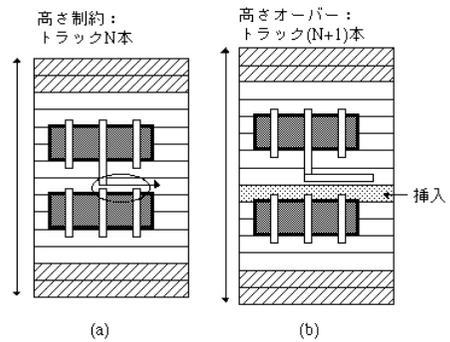


図2 従来法による配線領域確保の方法

Fig. 2 A method to get space for routing.

セル高さ違反を生じるので，高さ制約を満足するために，別の場所の格子を1行削除する等の対策を行う．しかしながらこのような方法では，セル全体にとって最適な格子挿入位置を求めることは難しく，また格子の挿入削除により処理が複雑になる．

本文で示す配線手法は，配線処理に輸入される配置要素間隔に影響されず，高い配線率を保ちながら，一方で，セルの高さ制約をも満足させることができる手法である．まず配線率向上のために，デザインルールよりも小さな格子間隔の設定を許して全体の格子数を増やす．これは配置要素間隔が狭い場合でも配線経路を見つけやすくするためである．このような設定により生じるデザインルール・エラーは，後のコンパクション処理で解消することができる．また，格子数が多ければ，そのままでは，セル高さ制約を満足できない恐れがあるが，本配線システムは2章で詳しく述べるように，セル高さ制約を守るために，セル高さ方向の格子の集合である格子列に割り当てることができる配線本数である「格子列容量」を定義し，これを用いてセル高さを管理する．さらに，格子列を配置要素(トランジスタ，端子)によって分割した区分配線領域モデルを導入し，分割された領域である区分配線領域を通過することができる配線本数を「区分配線領域容量」と定義する．特定の区分配線領域に配線が集中すると「格子列容量」によって，セル高さを管理しているにもかかわらず，コンパクション後にセル高さオーバーを引き起こす場合があるので，これを防ぐために「格子列容量」に加えて「区分配線領域容量」も管理する．本手法においても，従来のサイズ固定の配線モデルと同様に，引きはがし再配線手法を用いるが，設計規則違反に加えて，新たに「格子列容量違反」および「区分配線領域容量違反」との両方をペナルティ関数によって管理し，引きはがし再配線手法を繰り返すことにより配線経路の最適化を行う．また，設計違

反が解消されても、コスト最小化のための配線改善を引き続き行い、最終結果を得る。本文では、2章で、配線のためのモデルについて示す。3章では、配線のアルゴリズムを述べる。次に、4章では、本手法の有効性を示すために行った実験結果を示す。従来のサイズ固定モデルに対して、提案する配線モデルによると、配線率が向上できることを示す。また、既設計セルを用いた実験により、人手配線との性能比較を行う。また、本システムを用いた配置、配線、コンパクションの一連の処理の適用結果から実用的な結果が得られることを示す。最後に5章でまとめを行う。本システムは実際のセルライブラリ開発に用いられており、開発工数の削減に貢献している。

## 2. 配線モデル

### 2.1 セル・アーキテクチャ

セル高さはあらかじめ与えられる。セルの横幅は自由である。信号配線はポリシリコン層 (Poly) と第1金属層 (M1), および、より上層を使用し、電源配線はM1でセルの上下端に置く。各層における配線方向に制約はない。ソース/ドレイン端子は、コンタクトを拡散領域に最大限設ける形状 (非サリサイドプロセス) と、1コンタクト形状 (サリサイドプロセス) があり、プロセスにより選択する。外部端子には、セル間配置配線に適した配置制約 (一列, 交互等) を与えることができる。なお、電源位置の変更や配線層 (拡散層や第2金属層等) の追加等のセル・アーキテクチャ変更への対応も可能である。

### 2.2 格子モデル

本配線システムは、自動配置処理後の配線処理だけでなく、たとえば人手で設計された既設計のセルを入力として配線処理を行うことができる。このような既設計セルは、配線のための格子情報を持っていない。さらに本自動配置処理も配置処理用の格子を用いているが、そのままでは配線に適用できない。そこで、本配線システムにおいて、まず配線領域を格子に分割することにより、配線経路探索用の格子を作成する。配線領域は矩形であり、N層の配線層に対応可能である。配置要素 (トランジスタ, 端子) は、大きさや、マスクレイヤ, ネット情報等の属性を持つ矩形の集合で表現される。また、配置要素に対応した配線禁止領域が定義される。配線層間の接続はコンタクトを介して行う。配線用格子間隔は均一である必要はないが、配線層乗り換えのためのコンタクトの位置を上下に隣り合う配線層で一致させるために、配線層面に垂直な方向から見たとき、各層の格子を一致させる。続いて、格

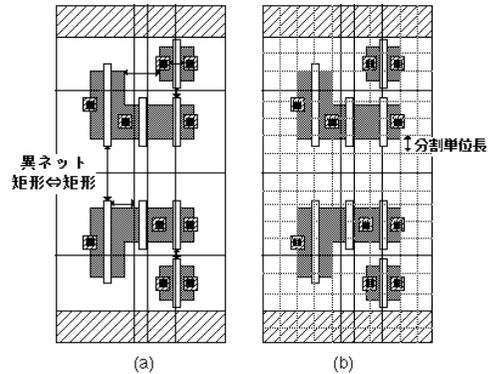


図3 格子分割

Fig.3 Grid partitioning.



図4 トラック間隔

Fig.4 Track width.

子の作成手順を以下に示す。

- (1) 異ネット矩形間の格子分割 (図3(a))
- (2) 分割単位長よりも間隔が広い格子間を分割単位長によって細分割 (図3(b))

(1) は、異ネット矩形間の格子分割処理であり、配置要素を表現するすべての矩形に関して、全体として互いに異なるネットの矩形どうしが少なくとも1つの格子で分割される状態を得るために最低限必要な格子を作成する。このとき生成される格子の間隔は、一般に不均一となる。

(2) は、(1) で作成した格子に関し、分割単位長よりも間隔が広い格子間を分割単位長により細分割する。分割単位長は可変であり、これを制御して配線領域の格子数を変化させることができる。格子不足による未結線が生じた場合、分割単位長を短くして格子数を増やし配線をやり直す。格子間隔が図4に示すトラック間隔より狭いとき、隣り合う配線 (あるいはコンタクト) 間でデザイン・ルール違反が生じるが、本システムはコンパクションによりこれを解消することができることを前提とする。

### 2.3 格子列容量および格子列容量違反

セル高さ方向の格子の集合である格子列において、配線を割り当てることができるトラックの本数を「格子列容量」と定義する。セル高さを図4に示すトラック間隔で割ると、セル高さの範囲に配線を割り当てることができるトラックの本数 (セル高さのトラック換算本数) が求まる。さらに格子列に配線禁止領域が存

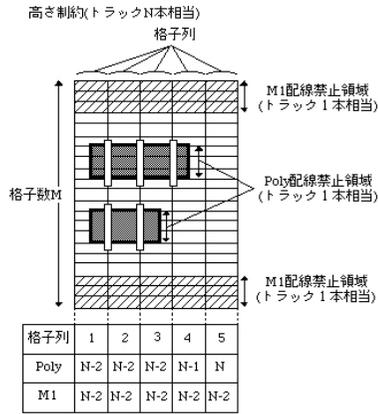


図5 格子列の容量  
Fig. 5 Capacity of grid columns.

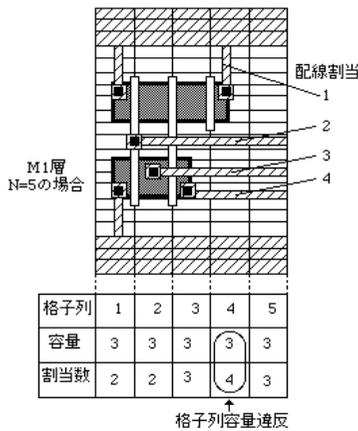


図6 容量違反  
Fig. 6 A capacity overflow example.

在する場合は、セル高さのトラック換算本数から配線禁止領域高さのトラック換算本数を引いた値が「格子列容量」である。

図5には5つの格子列が存在する。セル高さのトラック換算本数をすべての配線層においてN本であるとする。2.2節で説明したように、未結線をなくするために、格子の分割単位長を、トラック間隔よりも小さい値に設定することを許している。このようなときは、格子列に含まれる格子数Mは、セル高さのトラック換算本数Nよりも多くなっている。次に、右から2番目の格子列に着目すると、ポリシリコン層(Poly)に関する配線禁止領域がトラック換算で1本存在し、第1金属配線層(M1)に関する配線禁止領域が合計でトラック換算で2本存在する。よって、右から2番目の格子列における「格子列容量」は、Polyの(N-1)、M1の(N-2)となる。

ここで、格子列への割り当て配線本数が「格子列容量」より大きい状態を「格子列容量違反」と呼ぶこととする。格子列容量違反はセル高さオーバーを意味する。図6は、図5について、N=5とした場合の図である。第1金属配線層(M1)に着目すると、右から2番目の格子列に関する「格子列容量」は、図5の表より(N-2)であったから、この場合「格子列容量」は3である。ところが、図6において、配線が4本割り当てられている状態が示されており、右から2番目の格子列において「格子列容量違反」が生じている。

2.4 区分配線領域モデルおよび区分配線領域容量

図7(a)に示すように、配線領域においてセル高さ方向の格子列を、配置要素(配線禁止領域を成す)によって分割した区分配線領域モデルを定義する。分割

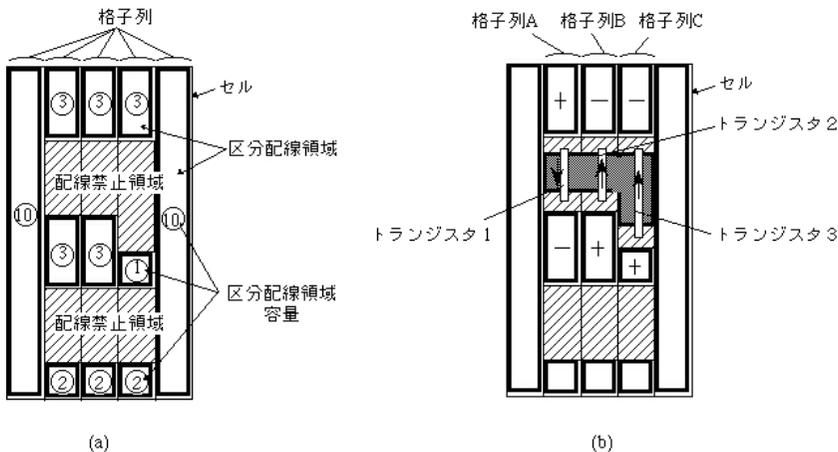


図7 区分配線領域モデル  
Fig. 7 Partitioned routing area.

された領域である区分配線領域を通過することができる配線本数を「区分配線領域容量」と定義する．図中の数字は「区分配線領域容量」を示す．「区分配線領域容量」は，特定の区分配線領域に配線が集中するのを防ぐためのものである．

特定の区分配線領域に配線が集中すると，「格子列容量」によって，セル高さを管理しているにもかかわらず，コンパクション後にセル高さオーバーを引き起こす場合がある．これは，コンパクションにおいて配置要素の上下移動が，限られた範囲でしか行われないうことによる．すなわち，元々拡散を共有しているトランジスタに対し，互いの上下移動により，拡散の分離が生じないように，上下の移動距離に制約を与えているからである．図 7 (a) に示す「区分配線領域容量」が与えられ，図 7 (b) の配線割当てが得られたとする．図 7 (b) において「区分配線領域容量」を超えた配線割当てが行われている状態の区分配線領域をプラス記号 (+) で「区分配線領域容量」以下の割当てが行われている状態の区分配線領域をマイナス記号 (-) で表現している．トランジスタ 1 の上側の区分配線領域は + で下側の区分配線領域は - であり，逆にトランジスタ 2 の上側の区分配線領域は - で，下側の区分配線領域は + である．もしもコンパクション処理において，まったく自由にトランジスタの上下移動が可能とすれば，トランジスタ 1 は下方へ，トランジスタ 2 は上方へと移動されるであろう．しかしながら，トランジスタ 1 とトランジスタ 2 は元々拡散を共有しており，このような場合本システムは拡散の共有を保持し，拡散の分離を禁止する．トランジスタの自由な上下移動を前提とすれば「格子列容量」を管理するだけでセル高さ制約を満足できる．しかしながら，拡散分離を禁止した本システムにおいてはさらに「区分配線領域容量」を極端に超えるような配線割当てが生じないように「区分配線領域容量」を管理して，セル高さ制約を満足させる．実際は，コンパクションにおいて配置要素の多少の上下移動が可能であるので，局所的に「区分配線領域容量」をオーバーするような配線経路の割当ては許している (3.2 節参照)．

### 3. 配線アルゴリズム

本手法は以下に示す特徴を有する．

- (1) セル高さ制約を満足する配線を行う．
- (2) ソース/ドレイン端子位置の最適化：サリサイド技術等を用いた場合，1 コントクトのソース/ドレイン端子が実現でき，トランジスタ上配線が可能となる．セル面積削減のため周囲の配線

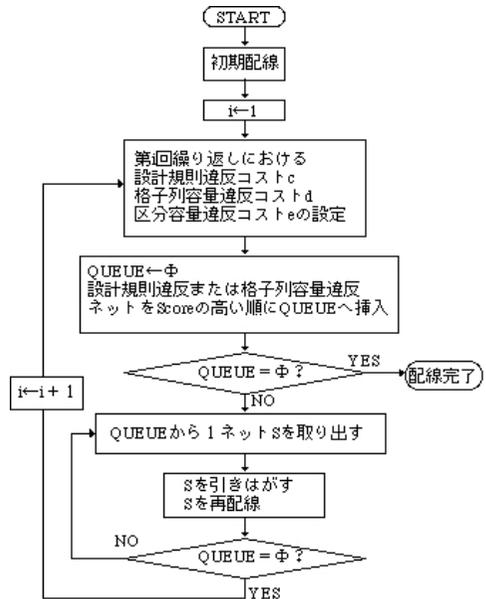


図 8 配線アルゴリズム・フロー  
Fig. 8 Routing flow chart.

状況に応じて端子位置決定を最適化する．

図 8 に配線アルゴリズムのフローを示す．初期配線と引きはがし再配線の 2 フェーズに別れる．初期配線の後，見積り配線長，設計規則違反回数，容量違反回数等の評価指標に基づいて引きはがし再配線を行い経路改善を行う．ここで，設計規則違反とは，1 つの格子に異ネットの配線やコンタクトが重複して割り当てられる状態である．配線経路探索には迷路法<sup>11)</sup>を用いる．

#### 3.1 初期配線

初期配線では，ネットの配線経路を他のネットの配線経路に無関係に求める．設計規則違反，およびセル高さオーバーを意味する容量違反を許して配線経路を求める．なお，配線禁止領域への配線経路割当ては認めない．初期配線の配線経路コストを以下に示す．

$$cost = a \times (\text{配線長}) + b \times (\text{コンタクト数})$$

ここに， $a, b$  はパラメータ (1)

まず，配線すべきネットに関するサブネットの集合を求める．サブネットとは，同電位である接続された端子と配線の集合である．部分的に配線経路が決定された部分配線以外にも，便宜上端子 1 つでもサブネットと呼ぶこととする．配線するネットに関し，サブネットが複数存在するとき，2 つのサブネット対に分解してその間を配線する 2 サブネット配線と，複数のサブネットを一度に配線する多サブネット配線の 2 つのモードを備えている．本配線システムは多サブ

ネット配線を可能な限り続ける．多サブネット配線では，まず格子数が最大のサブネットを始点とし，残るサブネットすべてを終点とする．なお，ここにサブネットの格子数とは，サブネットを構成する配線および端子が占める格子の数である．次に，始点からすべての終点を包含する最小矩形で定義するターゲット矩形までの配線経路を求める．このときすでに，始点がターゲット矩形上にあれば，2個のサブネット配線を行う．2個のサブネット配線は，マンハッタン距離が小さいサブネット対から順に配線するもので，格子数が多いサブネットを始点とする．

### 3.2 配線引きはがし

設計規則違反または格子列容量違反を生じているネットを引きはがす．ネットの得点 *Score* を以下で求め，得点の高い順に引きはがし再配線を行う．

$$Score = \alpha \times (\text{見積り配線長}) + \beta \times (\text{設計規則違反回数}) + \gamma \times (\text{格子列容量違反回数})$$

ここに， $\alpha, \beta, \gamma$  はパラメータ，  
 見積り配線長はネットの端子を囲む  
 最小矩形の 1/2 周囲長 (2)

$\alpha$  は，見積り配線長が長いほど可能な配線経路の数が多いと考えられ，引きはがしても他の配線経路を見つけやすいと考えられるためである．パラメータ  $\beta$  は設計規則違反が多いほど， $\gamma$  は格子列容量違反が多いほど引きはがされやすくするためのものである．*Score* には，「区分配線領域容量」に関する項を含めていないので，局所的に「区分配線領域容量」をオーバーする配線経路の割当てが行われる．

### 3.3 再配線

再配線における配線経路コストは，初期配線のコストに，設計規則違反コストおよび格子列容量違反コストおよび区分配線領域容量違反コストを加えたものである．

$$cost = a \times (\text{配線長}) + b \times (\text{コンタクト数}) + c \times (\text{設計規則違反回数}) + d \times (\text{格子列容量違反回数}) + e \times (\text{区分配線領域容量違反回数})$$

ここに， $a, b, c, d, e$  はパラメータ (3)

設計規則違反コストは，文献 3) と同様に設計規則違反を解消するために，引きはがし再配線の繰返しが進むにつれて  $c$  を大きくする．

また，格子列容量違反コストおよび区分配線領域容量違反コストは本システム独自のものであり，それぞれ，セル高さ制約を満たすためおよび特定の区分配線

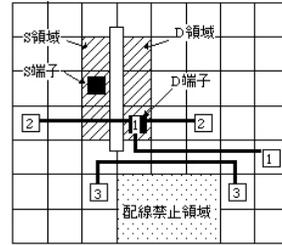


図 9 S/D 端子初期位置決定

Fig. 9 Initial position of source/drain terminals.

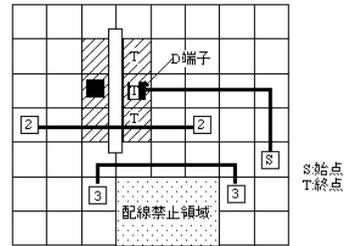


図 10 S/D 端子位置最適化

Fig. 10 Optimized source/drain terminals.

領域への配線集中を防ぐためのコストである．引きはがし再配線の繰返し数  $i$  が大きくなるにつれて  $d$  および  $e$  を大きくする．

### 3.4 ソース/ドレイン端子位置最適化

配線経路探索において，配線とソース/ドレイン端子との重なりを許すものとする．まず，初期配線において，他のネットと独立してソース/ドレイン端子の初期位置を求める (図 9)．続く引きはがし再配線において，ソース/ドレイン端子につながる配線が引きはがされた場合，ソース/ドレイン領域全体をソース/ドレイン端子の候補として再配線経路を求め，この経路とソース/ドレイン領域が接する場所にソース/ドレイン端子を移動する (図 10)．このように，ソース/ドレイン端子位置の移動も含めた引きはがし再配線を行う．

### 3.5 ゲート間配線

セル中央側にあるゲート端子からの配線経路が優先して求まるように，セル中央から遠いゲート端子にあらかじめ高いコスト (基底コストと呼ぶ) を付加し，探索を遅らせる．

図 11 において， $tp_1, tp_2, tn_1$  のゲートが同一ネットとする． $tp_1$  のゲートを始点 (S)，残りのゲートを終点 (T) とする． $tp_1$  の中央から離れたゲート端子 S に基底コストを与えることにより，配線長最小の点線で示す経路よりもセル中央の経路が選ばれる．基底コストの値としては，セル中央にあるゲートの始点集合

表 1 人手との比較  
Table 1 Comparison to manual design.

セル名	# of Trs	# of contacts (人手)	# of contacts (自動)	total length (人手)	total length (自動)	CPU time (sec)
INV1	2	1	1	16	15	0.9
AND2	6	3	4	46	28	2.9
BUF8	8	4	6	62	48	11.2
OR4	12	5	6	87	74	10.4
NOR6	20	11	9	159	134	32.3
LATCH1	22	14	14	192	142	51.7
LATCH2	28	15	18	186	166	70.8
FF1	34	25	26	299	218	93.3
FA	40	25	32	385	252	215.1
FF2	46	40	42	438	348	281.3

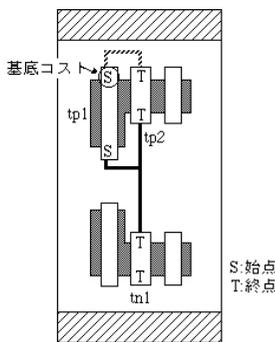


図 11 基底コスト  
Fig. 11 Base cost.

と終点集合との間の最小マンハッタン距離を用いる。

4. 評価実験

人手配線結果と本手法による自動配線結果を比較するために、人手で設計(配置,配線およびコンパクションすべて人手により実施)された既存のセルを用いて実験を行った。実験の手順を説明する。

(1) 格子の作成

人手設計セルを入力し、本配線システムにより、2.2 節で説明した手順で格子の作成を行う。これにより、人手配線は、格子を用いていったん抽象化される。このときの配線が占める格子の総数を人手配線における総配線長と定義する。

(2) 人手配線の消去

本配線システムの配線引きはがし機能を使用して、いったんすべての人手配線を引きはがす。

(3) 配線処理

全ネットに対し本配線システムによる自動配線処理を行う。このとき配線が占める格子の総数を自動配線における総配線長と定義する。

人手配線結果と自動配線結果をコンタクト数と総配

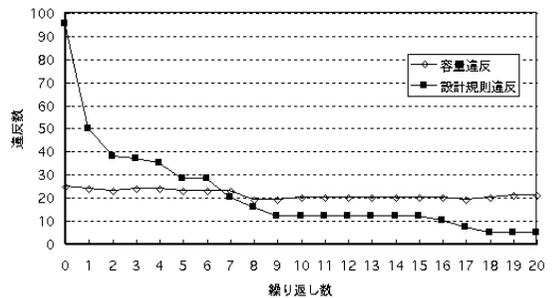


図 12 格子分割単位長をトラック間隔に設定した場合の設計規則違反個数の変化  
Fig. 12 A change of design rule violation with large grid distance.

線長に関し比較した結果を表 1 に示す。コンタクト数はほぼ人手並みである。配線長に関しては、自動配線結果はコンパクション前の結果であり、コンパクション後には、デザインルールを満足させるために、配線経路に凹凸が発生するため最終的にはやや表の値よりも増加すると思われるが、ほぼ人手並みの結果が得られたといえる。

次に、本手法の配線率の性能を確認するために、格子分割処理における分割単位長をトラック間隔に等しく設定した配線実験をセル FA を用いて行った。引きはがし再配線におけるコストとしては、設計規則違反コストを有効にし、格子列容量違反コストおよび区分配線領域容量違反コストを無効にした(すなわち、式(2)において  $\gamma = 0$ 、式(3)において  $d = e = 0$  とした)。引きはがし再配線の繰返しにともなう設計規則違反個数の変化の様子を図 12 に示す(参考に格子列容量違反個数の変化も示す)。最終的に 5 個の設計規則違反が残った。これは、配置要素間隔に比べて格子間隔が大きすぎたためであると思われる。実験に用いた人手設計セルは非常に高密度な設計となっており、

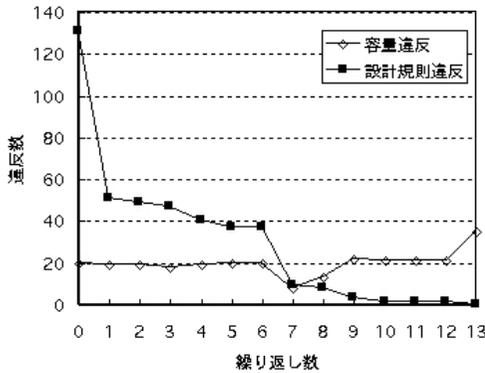


図 13 容量違反コストを導入しない場合の設計規則違反回数と容量違反回数の変化  
 Fig. 13 A change of design rule violation and capacity violation (without the penalty of capacity violation).

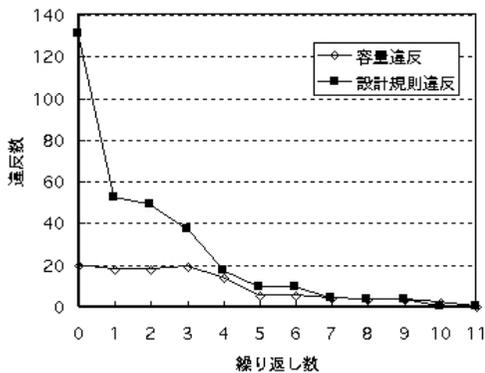


図 14 容量違反コストを導入した場合の設計規則違反回数と容量違反回数の変化  
 Fig. 14 A change of design rule violation and capacity violation (with the penalty of capacity violation).

配置要素間隔が狭く、分割単位長を十分小さく設定しなければ設計規則違反が残ることが分かる。本システムは、後のコンパクションでデザインルールエラーが解消されることを前提とするので、分割単位長を配線トラック間隔以下の値も含めて自由に設定できるので、後の実験結果図 13 および図 14 から分かるように、確実に設計規則違反をなくすることができる。よって、高い配線率が得られることが分かる。

続いてセル高さ制約が与えられたときの、本配線手法の有効性を示す実験について説明する。今回の実験は、トランジスタ配置も本システムで行った。セル FA に関し、格子列容量違反コストと区分配線領域容量違反コストを導入した場合と、導入しない(すなわち、式 (2) において  $\gamma = 0$ 、式 (3) において  $d = e = 0$  とした)場合の両方を比較する実験を行った。

まず、格子列容量違反コストと区分配線領域容量違反コストを導入しない場合の、引きはがし再配線の繰返しにともなう設計規則違反回数および格子列容量違反回数の変化の様子を図 13 に示す。設計規則違反回数が 0 になり配線処理を終えた時点でも、まだ格子列容量違反が残っている。これは高さ制約を満足できなかったことを示している。

次に、格子列容量違反コストおよび区分配線領域容量違反コストを導入した実験では、図 14 に示すように 11 回の繰返しで、設計規則違反および格子列容量違反を 0 にすることができた。

このときの配線結果を図 15 に、またこれをコンパクションした結果を図 16 に示す。コンパクション後は、指定のセル高さ制約を満たすことができた。

また、セル LATCH1 の配線結果と、コンパクシヨ

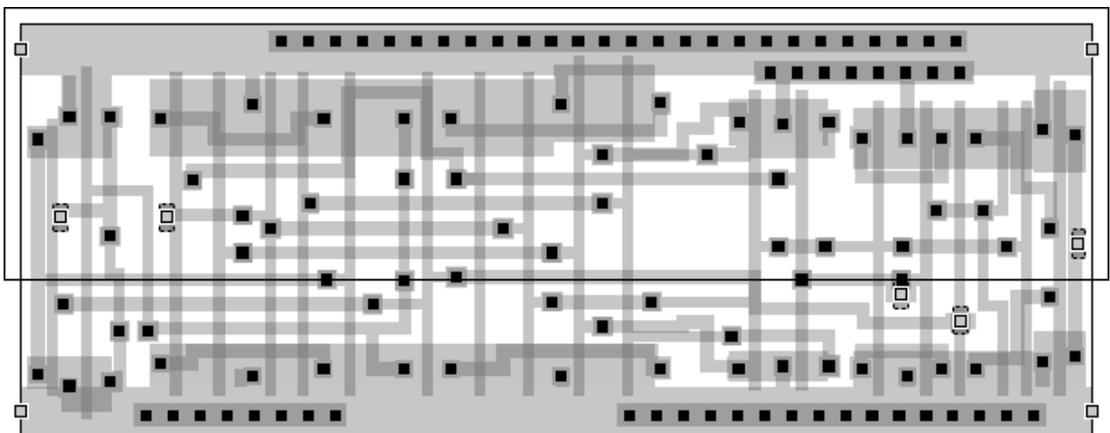


図 15 配線結果 (FA)  
 Fig. 15 A routing result (FA).

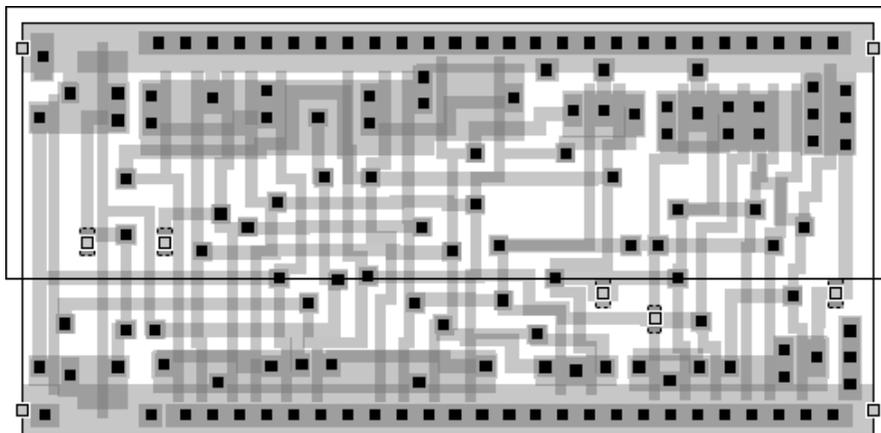


図 16 コンパクション結果 (FA)  
 Fig. 16 A compaction result (FA).

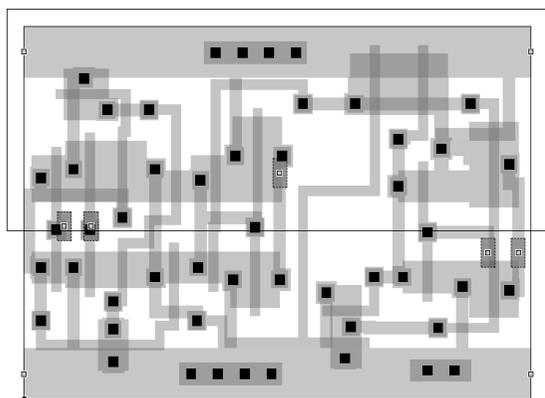


図 17 配線結果 (LATCH1)  
 Fig. 17 A routing result (LATCH1).

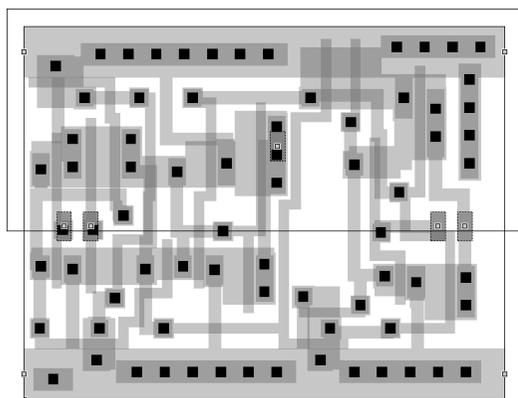


図 18 コンパクション結果 (LATCH1)  
 Fig. 18 A compaction result (LATCH1).

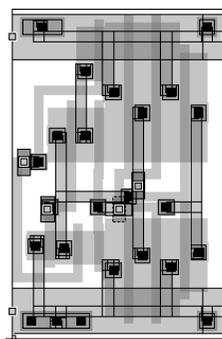


図 19 端子位置最適化なし  
 Fig. 19 No optimization for terminal positions.

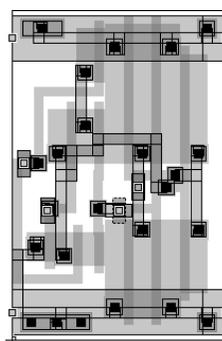


図 20 端子位置最適化あり  
 Fig. 20 Optimization for terminal positions.

配線長が短くなる位置に端子が移動し、トランジスタ上を金属配線が通過するようになった。

### 5. 結 論

本文では、リーフェルのための配線モデルと、配線最適化手法としてセル高さ制約に関するコストを評価指標とする引きはがし再配線手法を提案した。また本

ン結果をそれぞれ図 17 と図 18 に示す。  
 S/D 端子位置決定を最適化しない場合を図 19 に、最適化した場合を図 20 に示す。最適化した場合は、

手法は、ソース/ドレイン端子位置決定を最適化することができる。本文中で提案した配線モデルが、配線率を向上することを実験により示した。また、既設計セルを用いた実験により人手配線並みの結果が得られることを確認できた。また提案した格子列容量および区分配線容量の管理により、セル高さ制約を満たすことができることを実験により確認した。本配線システムは実際のセル開発に適用され、実用的効果を上げている。

### 参 考 文 献

- 1) Fukui, M., Shinomiya, N. and Akino, T.: A new layout synthesis for leaf cell design, *Proc. Asia and South Pacific Design Automation Conference*, pp.259-264 (1995).
- 2) Saika, S., Fukui, M., Shinomiya, N. and Akino, T.: A two-dimensional placement for cell synthesis, *Proc. Asia and South Pacific Design Automation Conference*, pp.557-562 (1997).
- 3) 羽根, 油井, 島本, 白川, 西口: 引き剥し再配線手法を用いた分散処理型多層 vlsi 配線システム, *信学技報*, CAS-91-18, pp.29-36 (1991).
- 4) Poirier, C.J.: Excellerator: Custom cmos leaf cell layout generator, *Trans. Computer-Aided Design of Integrated Circuits And Systems*, Vol.8, No.7, pp.744-755 (1989).
- 5) Shin, H. and Vincentelli, A.S.: Mighty: A rip-up and reroute detailed router, *Proc. ICCAD*, pp.2-5 (1986).
- 6) Rosenberg, E.: A new iterative supply/demand router with rip-up capability for printed circuit boards, *Proc. 24th Design Automation Conference*, pp.721-726 (1987).
- 7) Lin, Y.L., Hsu, U.C. and Tsai, F.S.: Silk: A simulated evolution router, *Proc. 24th Design Automation Conference*, Vol.8, No.10, pp.1108-1114 (1989).
- 8) Tzeng, P.S. and Sequin, C.H.: Codar: A congestion-directed general area router, *Proc. International Conference on Computer Aided Design*, pp.30-33 (1988).
- 9) Kawamura, K., Shindo, T., Shibuya, T., Miwatari, H. and Ohki, Y.: Touch and cross router, *Proc. International Conference on Computer Aided Design*, pp.56-59 (1990).
- 10) Ong, C.L., Li, J.T. and Lo, C.Y.: Genac: An automatic cell synthesis tool, *Proc. 26th Design Automation Conference*, pp.239-244 (1989).
- 11) Rubin, F.: The lee path connection algorithm, *IEEE Trans. Comput.*, Vol.23, No.9, pp.907-914 (1974).

(平成 11 年 9 月 22 日受付)

(平成 12 年 2 月 4 日採録)



四宮 典子

昭和 63 年大阪府立大学工学部電気工学科卒業。同年、松下電器産業(株)入社。以来、レイアウトアーキテクチャ、コンパクション、自動配線等、半導体 CAD の研究開発に従事。電子情報通信学会会員。



濱脇 浩二

平成 4 年近畿大学理工学部経営工学科卒業。同年(株)松下ソフトリサーチ入社。以来、自動配線、セル合成等、LSI 設計自動化の研究開発に従事。



福井 正博(正会員)

昭和 58 年大阪大学大学院修士課程修了。同年、松下電器産業(株)入社。以来、自動配置配線、モジュール合成、セル合成等半導体 CAD の研究開発に従事。平成元年～平成 3 年 U.C.Berkeley にて客員研究員。電子情報通信学会、IEEE 各会員。工学博士。