

## ゲートアレイ/CBIC自動レイアウトシステム GALET (3)

5L-6

## - 自動配置手法 -

多和田 茂芳 水牧 俊博† 田形 充†  
 袖 美樹子 水沼 貞幸† 石森 彰雄†

日本電気(株) †NECソフトウェア北陸(株)

## 1 はじめに

LSIの製造プロセスの微細化に伴い、設計回路が大規模化し信号遅延に占める配線遅延の割合が相対的に増大してきている。

そのような状況のもとで配線遅延を制御するための自動配置での種々の方法が提案されている。クリティカルパス上のネットに重み付けして配置する方法[1]は、結果についての保証が無く繰り返し実行になる。ネットの配線長制限を満足させる配置手法[2]は、パスの遅延時間制限をネットに割り振るので配置改良の自由度が低下し配線収容性改善を必要以上に妨げる。パスとしての遅延制限値を扱い配置改良毎に遅延解析結果を更新しながらタイミングを保証していく方法[3]は、一般的に処理時間がかかるが遅延時間制限を満足して更に配線収容性の高い良質な配置結果を得ることができる。

本論文で提案する配置手法は、上述の3番目の方法の一つで、配置処理中のミニカット、2次元改良という二つのフェーズでタイミングを考慮しそれぞれに高速化を図って実用化している。

## 2 タイミング制約

本論文で対象とするタイミング制約はパスの最大遅延時間の制限として与えられ信号のRISE/FALLを考慮することができる。FF間、外部入力端子-FF間、FF-外部出力端子間のパスについてタイミング制約を与えることができる。パスの始終点のみをタイミング制約で指定した場合には始終点間の全経路のタイミング制約を考慮する。

## 3 タイミング保証自動配置手法

## 3.1 基本アルゴリズム

GALETの配置の基本アルゴリズムは階層クラスタリング法を用いたミニカット手法[4]を用いている。素子の接続関係に基づいて階層的にクラスタを生成し、クラスタの階層を崩しながら各階層レベルでカットラインを跨いだクラスタの交換を行う。これにより交換個数の削減による高速化とクラスタサイズの均一化による交換容易化を図っている。

A Gate Array/CBIC Automatic Layout System GALET(3) - Automatic Placement Method -. Sigeyoshi TAWADA, Toshihiro MIZUMAKI†, Mitsuru TAGATA†, Mikiko SODE, Sadayuki MIZUNUMA†, Akio ISHIMORI†  
 NEC Corporation, †NEC Software Hokuriku, Ltd

1次元交換によるミニカットの配置改良はカットラインの生成順序に対する解の依存性が高い。そこで、配線性を考慮してカットラインを横切る配線数を2次元的に平坦化して解を補正するために2次元改良配置によるクラスタの多段反復交換処理を行う。2次元改良配置はミニカットの任意の時点で実行可能だが処理時間と効果を考慮して4×4分割、16×16分割…等適当なタイミングで行う。

ミニカット、2次元改良によりカット領域に割り当てられた素子の詳細な配置位置は2次計画法の手法(SCG法)を用いたマッピング処理により決定する。

## 3.2 ミニカットにおけるタイミング保証

ミニカット配置でのタイミング保証は[5]で既に報告済みである。カットライン毎にネットの配線形状を見積り配線遅延を再計算して遅延解析結果を更新し、スラック値[6]を求めてスラックが負であるネットについて以下のような考慮を行う。

- スラックが負であるネットにつながる素子を優先的に階層クラスタリングする。
- ペア交換時にスラックが負であるネットの配線長が短くなるような交換を優先する。

## 3.3 2次元改良におけるタイミング保証

2次元改良でのタイミング保証はパスの遅延時間制限を保証した上で、パス遅延改善による配線性の悪化を最小限にとどめることができる。

(タイミング保証2次元改良アルゴリズム)

Step1. 遅延解析結果を更新する。

Step2.for (パラメータを変えた改良条件数分) |

Step2-1. 全カット領域で階層クラスタリング。

Step2-2.for (クラスタの階層を崩しながら) |

Step2-2-1. パスグループ、ネットグループを作成する。

Step2-2-2. クラスタ、グループの仮想ゲインを算出しバケットにセットする。

Step2-2-3.for (バケット内のクラスタ、グループの最大ゲインが規定値以下になるまで) |

Step2-2-3-1. バケットからクラスタまたはグループを取り出す。

Step2-2-3-2. グループの場合仮移動してクラスタ(グループクラスタ)にする。

Step2-2-3-3. クラスタの反復移動の最適パターンを探索する。

Step2-2-3-4. 最適移動パターンの移動処理を行い、同時に遅延解析結果をインクリメンタルに更新する。

Step2-2-3-5. 移動結果を基にバケット内仮想ゲインを更新する。

Step2-2-3-6. グループクラスタの場合解体してグループに戻す。

以下にアルゴリズムの詳細についての説明を行う。

ミニカット処理により各クラスタがカット領域に割り当てられ2次元改良に渡されると、Step1でカット領域の位置関係よりネットの配線形状を見積り配線遅延時間を計算し遅延解析結果が更新される。

Step2-2-1ではバス遅延エラーを改善するためのバスグループとネットグループを作成する。バスグループとは、遅延エラーを起こしているバスすなわちスラックが負となるバス上の連続したクラスタの集合(図1)で、改善効果のあるものを抽出する。改善効果についてはバスグループを通るエラーバスの前段と次段のクラスタが囲む矩形領域外にそのバスグループが存在するとき有効とする。またサイズ制約として、カット領域の許容サイズを越えないグループのみを有効にする(図2)。次に、ネットグループとは、スラックが負となるバス上ネットにつながるクラスタの集合でバスグループと同様のサイズ制約を満たして有効となる。ネットグループはエラーバス上に無いクラスタを移動することでネット配線長を小さくして配線遅延を改善する効果を得る目的で生成する(図3)。

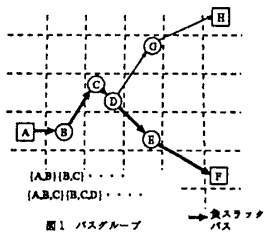


図1 バスグループ

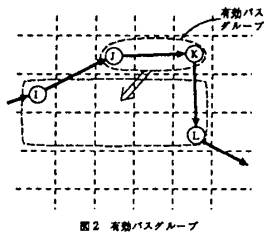


図2 有効バスグループ

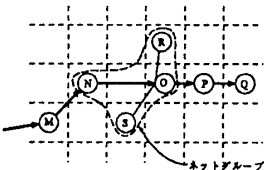


図3 ネットグループ

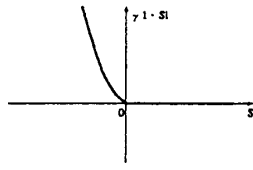


図4 最悪スラックの係数

Step2-2-2ではクラスタ、バスグループ、ネットグループの仮想ゲインを計算し、それをキーに降順にソートしたバケットを作成する。仮想ゲインは以下の式で求められ、グループについてはグループ内のクラスタのゲインの和で求められる。移動については各々最適な移動が行われた場合を想定する。

$$a \cdot \Delta Ct + \beta \cdot \Delta Sz + \gamma \cdot 1 \cdot S1$$

( $a, \beta, \gamma$  は係数)

$\Delta Ct$ : クラスタの移動によるカット数の改善度

$\Delta Sz$ : クラスタの移動によるカット領域の許容サイズオーバーの改善度

$S1$ : 最悪スラック値

$S1$ の係数 $\gamma$ はスラックが正のときは0で、負のときはこの項が他のゲインの項に比べて大きくなるように設定される(図4)。このスラックの項により遅延エラーバス上のクラスタやグループがバケットの先頭の方に現れ早い段階で改良が行われる。

グループは複数のクラスタの集合体であり各々別のカット領域に存在している場合があるので、グループがバケットから選択されるとStep2-2-3-2によりグループ内のクラスタを1つのカット領域に仮移動して結合し1つのクラスタとして以後扱う。

Step2-2-3-3のクラスタの最適な移動パターンの探索は再帰的に行われ、再帰呼出の深さ(回数)と移動候補領域の選択数の制限により処理時間の増大を抑える。移動候補領域は、普通のクラスタの場合に

はそれにつながる全ネットを囲む矩形領域となる。グループの場合はその前段と次段のクラスタを囲む矩形領域となる。クラスタの移動候補領域は以下に示すクラスタ移動ゲインによって選択される。

$$a \cdot \Delta Ct + \beta \cdot \Delta Sz + \gamma \cdot 2 \cdot \Delta S1$$

( $a, \beta, \gamma$  は係数)

$\Delta S1$ : スラックの改善度

$\Delta S1$ (スラックゲイン)は、移動前の最悪スラック値を $S11$ 、移動後の最悪スラック値を $S12$ とすると、以下のように計算され、スラックが悪化するような移動は選択されないようにしている。

$$S11, S12 \geq 0 \text{ のとき, } 0$$

$$S11 \geq 0, S12 < 0 \text{ のとき, } -(S12)^2$$

$$S11 < 0, S12 \geq 0 \text{ のとき, } S11^2$$

$$S11, S12 < 0 \text{ のとき, } S11^2 - S12^2$$

最適移動パターンでの多段反復移動をバケットに現れるクラスタやグループについて繰り返すことにより配線性とバス遅延エラーが改善できる。

## 4 実験結果

下地ゲート数202KG、素子数12852、ゲート使用率49.5%のデータで、タイミング保証配置の性能を評価した。制約バス遅延値は16ns、制約バス数は928本である。

タイミング保証無し(NoTD)に対して、タイミング保証を使用した結果(2DTD)ではタイミング違反はほとんど無くなり、カット数に関しても同等であるが、配置時間は2.4倍に増加している。また、2DTDにおいて残った4つの違反は配線遅延の見積り誤差によって生じており、高ドライブ素子への置き換えと設計変更処理で対応した。

表1: 評価結果

	NoTD	2DTD
違反バス	888	4
最大遅延	19.4ns	16.1ns
最大Vカット数	960	928
最大Hカット数	862	882
ミニカット時間	6m	9m
2次元改良時間	27m	1h12m
トータルCPU時間	34m	1h22m

## 5 まとめ

GALETのタイミング保証自動配置についてアルゴリズムを紹介し、実験結果をもってその有効性を示した。今後は2次元改良配置の更なる高速化改善を行っていきたい。

## 参考文献

- A.E.Dunlop, V.D.Agrawal, D.N.Deutsch, M.F.Jukl, P.Kozak, M.Wiesel, "CHIP LAYOUT OPTIMIZATION USING CRITICAL PATH WEIGHTING," 21st DAC, pp.133-136, 1984
- M.Terai, K.Takahashi, K.Sato, "A New Mincut Placement Algorithm for Timing Assurance Layout Design Meeting Net Length Constraint," 27th DAC, pp.96-101, 1990.
- W.E.Donath, R.J.Norman, B.K.Agrawal, A.E.Bello, S.Y.Han, J.M.Kurtzberg, P.Lowy, R.I.McMillan, "TIMING DRIVEN PLACEMENT USING COMPLETE PATH DELAY," 27th DAC, pp.84-89, 1990.
- 枝廣, 吉村, "階層クラスタリング法を用いたセル列型LSIのための配線手法," VLD90-62, pp.9-15, 1990.
- M. S. Tanaka, Y. Miyazawa, H. Aizawa and M. Minowa, "Minimum Delay Placement with Influence of Nets and Hierarchical Clustering," Proc. CICC, pp.663-666, 1994.
- P.S.Hauge, R.Nair, J.Yoffa, "Circuit Placement for Predictable Performance," Proc. of ICCAD, pp.88-91, 1987.