

## VLSI自動設計システムに於けるランダムイズド クラスタリングアルゴリズム

島津 一寿, 白石 洋一  
群馬大学 工学部 情報工学科

VLSI自動レイアウト設計に於ける初期配置問題は、仮想配線長合計最小化等を目的関数とした組合せ最適化問題として定式化できる。しかし、大規模な初期配置問題に対しては、実用時間内にその最適解を得ることは不可能である。したがって、従来より様々な発見的手法が用いられてきた。ここでは、局所最適解に陥らずにいかにして大域的最適解に到達するかが問題となる。本稿では、初期配置問題を可能な限り独立になるように分割する、クラスタ分割問題にランダムイズドアルゴリズムを適用した手法について論じる。

### A Randomized Clustering Algorithm in a Design Automation System for VLSI's

Kazuhisa Shimazu, Yoichi Shiraishi

Department of Computer Science, Gunma University  
E-mail: {kazu, siraisi}@keim.cs.gunma-u.ac.jp

An initial placement problem of components in a design automation system for VLSI's can be formulated as a combinatorial optimization problem whose objective functions are minimization of the total pseudo-wire length, etc. However, it is actually impossible to obtain the optimum solution for very large scale initial placement problems. Therefore, various heuristic algorithms have been suggested and utilized for years. In this heuristic approach, the difficulty is how to obtain the global optimum solution not to be trapped in the local ones. This paper discusses an experimental method based on the randomized algorithm approach and its application to the clustering problem which partitions the original initial placement problem into subproblems so as to minimize the number of pseudo-wires between subproblems.

# 1 背景

VLSI(Very Large Scale Integrated circuit) チップの集積度(単位面積当りの素子数)は年率 1.3 ~ 1.5 倍の割合で増大し、現在では 100 万トランジスタ規模の VLSI チップが出現してきている。かつて集積度は IIC(Integrated Circuit) 当り数ゲートにすぎなかったが、最近では計算機の CPU(Central Processing Unit) 全体が 1 チップに集積化されるようになってきた。それに伴い、VLSI の機能はより一層複雑化し、さらに少量多品種、設計・製造期間の短縮が重要な ASIC(Application Specific Integrated Circuit) と呼ぶ VLSI チップが出現して、その設計は工数・期間ともに最早人手の能力の限界を越えたといえる。このため電子計算機を用いて VLSI チップの設計を支援、または自動化する、CAD(Computer Aided Design)、または DA(Design Automation) が提案され、研究されてきた。その歴史は古く、LSI の集積度は未だ低かったにもかかわらず、1961 年には既に VLSI チップ上の部品間を配線する基本的な配線アルゴリズムが報告されている。当時より研究されてきた自動配置配線アルゴリズムは、CAD、または DA システムに組み込まれ、1980 年代のその急激なニーズの高まりと共に実用レベルに達してきた。そこでは各種 VLSI チップモデル、各レイアウト設計フェーズに対して自動レイアウトシステムが構築されて、設計工数削減、期間短縮に大きく貢献している。しかし現在では、超大型計算機に使用される VLSI チップ、アナログ LSI チップのレイアウトなど、単に素子を配置し素子間を配線するだけでなく、回路の電気的特性についても同時に最適化を図る、との高度な要求が高まっている。そのため、より広い回路形式とより広い範囲の VLSI チップモデルを対象とし、かつ、高品質なレイアウトを生成する配置配線アルゴリズムの研究が活発に行なわれている。

# 2 配置問題

## 2.1 VLSI の設計

VLSI の設計は、論理設計と実装設計に分けられ、さらに実装設計は回路設計とレイアウト設計に分けることができる。VLSI におけるレイアウト設計では、論理設計結果の素子の結線関係と、回路設計結果の部品情報から実際の部品をチップ上に表示する。レイアウト設計は配置処理、配線処理の順で行なう。

## 2.2 配置処理の概要

### 2.2.1 階層設計方式

配置領域に配置する対象は、論理回路を実現する素子である。配置処理時には、複数の論理ゲートをまとめたセルを最小単位として扱う。VLSI の設計方式は複数存在し、重要な方式の一つに階層設計方式がある。この方式は、配置対象をセル、

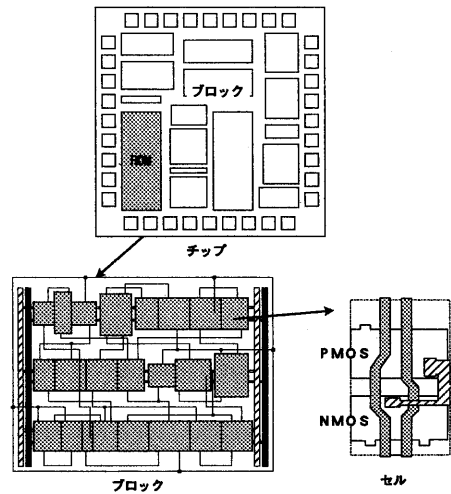


図 1: VLSI チップの階層設計方式

複数のセルで構成するブロック、複数のブロックで構成するチップの3階層に分けて、再帰的に設計する。扱う最大規模はブロック数で数百、各ブロックを構成するセル数で数万~数十万程度である。図1は階層設計方式によって構成されたVLSIの例である。本稿の配置処理では階層設計方式を対象としている。以下ではセルおよびブロックをまとめて部品と呼ぶ[1]。

## 2.2.2 配置問題の概念的定義

配置問題は、通常、論理情報から配置結果を得る初期配置と、その結果をさらに改善する配置改善に分けることができる。VLSIのレイアウト設計における初期配置問題は、配置領域の面積最小化を目的関数とした組合せ最適化問題として解くことができる。配置領域の面積最小化を行なうためには実際の配線長合計最小化による配線領域の面積最小化が必要であるが、初期配置処理の段階で配線長合計を求めることは事実上不可能である。よって、初期配置処理では配線長合計を仮想的に決定した配線径路長である仮想配線長合計に近似し、配線領域の面積最小化を行なうものとする。仮想配線長はネットを囲む最小矩形の半周長で求められる。ある端子が配線によって別の端子と結ばれる状態にあるとき、その端子は配線要求を持つと定義すると、ネットとは、配線要求を持つ部品の端子の集合である[2]。

## 3 ランダムイズドアルゴリズムと初期配置問題

### 3.1 初期配置処理の概要

初期配置処理は、VLSI全体を配置対象領域としてスロットと呼ばれる部分領域に分割するスロット分割処理、ある制約条件あるいは評価指標にしたがって部品全体をクラスタという部分集合に分割するクラスタ分割処理(クラスタリング)、

仮想配線長合計が最小化されるようにスロットにクラスタを配置するクラスタ配置処理に分けることができる。1つの階層レベルでの配置を終えると、各スロットをさらに細分化して次のレベルで同様の処理を行ない、最終的にVLSI全体の部品の配置結果が得られる。あるレベル以降の処理を1プロセッサで実行することによって、並列処理を行なうことが可能である。

### 3.2 ランダムイズドアルゴリズム

ランダムイズドアルゴリズムとは、すなわち、乱数を利用したアルゴリズムである。ランダムイズドアルゴリズムはソーティングやハッシュなどのリスト処理、組合せ問題、グラフ問題、線形・整数計画法、計算幾何学、素数判定等の数論アルゴリズム、分散並列処理など、ほとんどすべての計算分野で使われている。一般にランダムイズドアルゴリズムは乱数を使わない場合に比べて単純で、プログラムするのに簡単であり、計算効率が良い(すなわち、速い)ため、実用的である。その反面、アルゴリズムの解析(計算効率の評価)には、確率論の知識が不可欠であり、しばしば数学的に深い手法を用いる。近年は計算理論の主要学会でも非常に多く取り上げられ、多くの興味深い理論が展開されている。

ランダムサンプリングが統計やデータ解析を必要とする様々な分野で活躍している理由と同じく、手軽さ、簡単さがランダムイズドアルゴリズムの最大の優位性である[3][4]。

## 4 クラスタ分割問題

部品のクラスタ分割問題は、部品を頂点、部品間の結線関係を辺で表した無向グラフにおいて、目的関数を最適化するようにその頂点をグループ化する問題として表される。

入力: 部品の集合, 部品間の結線, 分割問題数

出力：部品の部分集合

目的関数：クラスタとクラスタの間を渡る結線数  
(カット数)最小化, クラスタサイズ均等化

目的関数に配線長制御による電気的特定最適化を追加することは標準的であるが、本稿ではまずカット数最小化を考える。

#### 4.1 従来手法

クラスタ分割問題で現在、最も使用されている手法に、クラスタ生長法がある。クラスタ生長法は、始めに、“種”と呼ばれる他のセルとの結合が最も強く、かつ互いに遠距離にあるセルを、目的とするクラスタの数だけ求め、次に、各々の“種”をクラスタの核として、残りのセルを最も結合の強いクラスタへ順次割り付けていく。しかし、この手法は、各割り付け段階では最適解を採っているものの、終了段階での解は、局所解に陥ってしまう可能性がある。

#### 4.2 ランダムイズドクラスタリングアルゴリズム

各割り付け段階で最適解以外の解を採択する確率を残すことにより、その段階では望ましくない解であっても、終了段階での解が最適解であるようにクラスタ分割を行なうアルゴリズムが、ランダムイズドクラスタリングアルゴリズムである。ランダムイズドクラスタリングの処理を図2に示す。

##### 4.2.1 “種”の抽出

ここではクラスタ生長法と同様に、他のセルと最も結合度の強いセルを、分割するクラスタの数だけ求める。このとき“種”となるセルは、互いに遠距離(互いの結合がない)でなくてはならない(図3)。

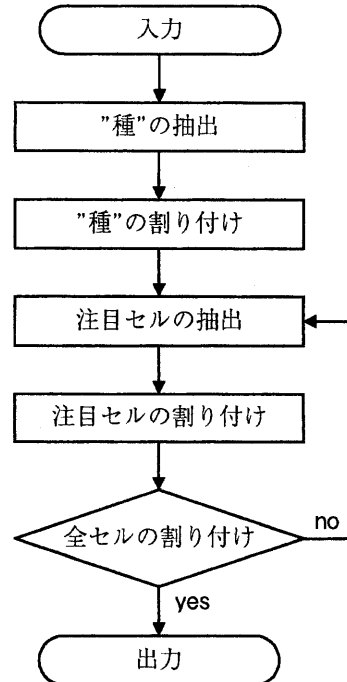


図2: ランダムイズドクラスタリングアルゴリズム

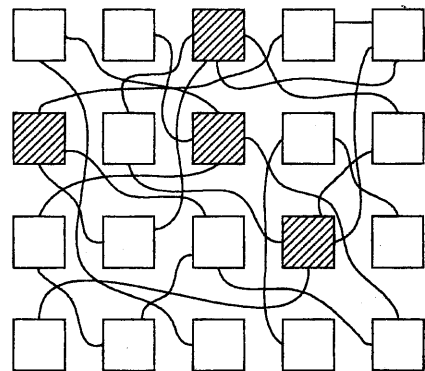


図3: “種”の抽出

##### 4.2.2 “種”のクラスタへの割り付け

“種”として抽出されたセルを、それぞれ1つ

ずつのクラスタに割り付ける (図 4)。

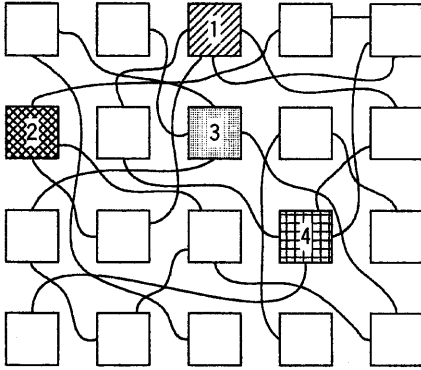


図 4: "種" のクラスタへの割り付け

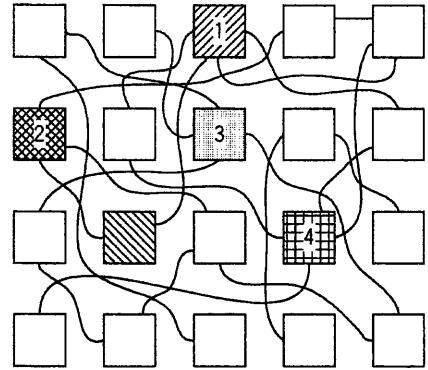


図 6: 注目セルの決定

#### 4.2.3 注目セルの抽出

クラスタ生長法では、クラスタと最も結合の強いセルを抽出しているが、本アルゴリズムでは、セルの割り付けの順序にも問題があると考えた。そのため、残ったセルとクラスタとの結合度を求め、それに基づく確率を算出する (図 5)。確率は、クラスタとの結合度の高いものほど大きくなる。また、クラスタとの結合度が 0 の場合でも、確率を 0 にしない。それから 0 ~ 1 の乱数を発生させ、注目するセルを決定する (図 6、例として乱数値 0.65 の時)。

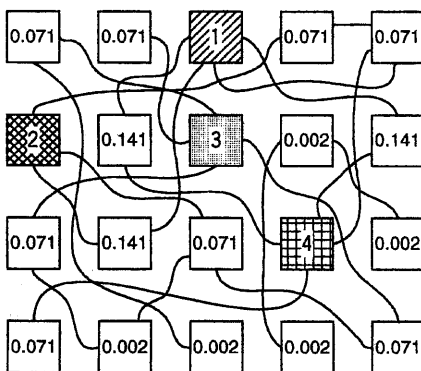


図 5: 結合度に基づく確率の算出

#### 4.2.4 注目セルのクラスタへの割り付け

クラスタ生長法では、前項で抽出されたセルを最も結合度の高いクラスタに割り付けているが、本アルゴリズムでは、その決定が後の処理で悪影響を与える可能性があると考えた。そのため、解が悪い場合でもその採択確率を残し、セルを割り付けるクラスタを確率的に決定する。ここでは、各クラスタとの結合度を求め、それに基づく確率を算出する (表 1)。確率は、クラスタとの結合度の高いものほど大きくなる。また、クラスタとの結合度が 0 の場合でも、割り付け確率を 0 にしない。それから、0 ~ 1 の乱数を発生させ、割り付けるクラスタを決定する。例えば表 1 のような時、乱数値 0.55 を得たならば、注目セルをクラスタ 2 へ割り付ける。

表 1: 結合度に基づく確率の算出

クラスタ	1	2	3	4
結合度	1	1	0	0
確率	0.445	0.445	0.005	0.005

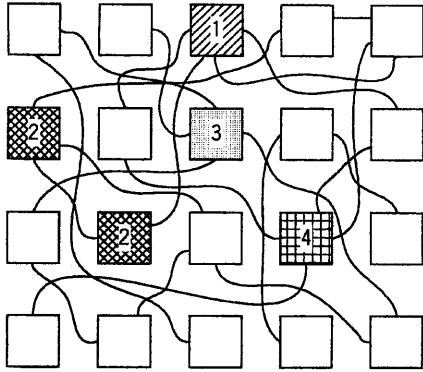


図 7: 注目セルの割り付け

## 5 実験

従来手法であるクラスタ生長法と、提案する手法により、分割数 4([5]) にてカット数の比較を行った。実験は、MCNC(Microelectronics Center of North Carolina) ベンチマークから標準セル 2 品種 (fract, industry1) を選んで行った。また、本手法はランダムアルゴリズムに基づくものであるため、データの採取を fract では 3000 回、industry1 では 1000 回行った。ベンチマークの諸元を表 2 に示す。

表 3 には、クラスタリング結果のカット数の比較を示す。これから分かるように、本手法は、クラスタ生長法に比べ最良値で、fract では約 46%、より規模の大きい industry1 では約 61% のカット数削減がなされ、セル数が多くなるほど効果が見られている。

なお、処理時間は UltraSPARC 上で、fract データ 3000 回で 819 秒、industry1 データ 1000 回で 88331 秒であった。

表 2: ベンチマーク諸元

品種	セル数	ネット数
fract	125	128
industry1	2271	2479

表 3: カット数の比較

品種	本手法			クラスタ生長法
	最良値	平均値	最悪値	
fract	14	31.017	76	26
industry1	56	157.881	306	142

## 6 まとめ

- ランダムアルゴリズムに基づくクラスタリング手法を提案した。
- MCNC ベンチマークデータによる実験から、クラスタ生長法と比べ、カット数が約 46 ~ 61% 削減された。

## 7 今後の課題

- 電気的特性を考慮するためのアルゴリズムの拡張
- 大規模データに対する処理速度低下の削減

## 参考文献

- [1] 白石洋一:「VLSI レイアウト自動設計アルゴリズムに於ける諸問題」, 情報処理学会, アルゴリズム研究会資料, アルゴリズム 19-6, 1 月, 1991
- [2] 小澤時典, 石井建基, 三浦地平, 寺井秀一: 「標準セル方式 VLSI の配置手法とその評価」, 電子情報通信学会論文誌, vol.J67-D, No.10, 10 月, 1984
- [3] 徳山豪:「ランダムアルゴリズムの話題から」, 電子情報通信学会誌, vol.77, No.9, 9 月, 1994
- [4] Motowani R. and Raghavan P.: "Randomized Algorithms", Cambridge University Press, U.K.(1994)
- [5] 荒川浩一, 小川泰:「4 分割初期配置手法」, 設計自動化, 1992