

## VLSI 回路の階層設計をサポートする階層化 BSG フロアプラン

呉 中林<sup>†a)</sup> 中武 繁寿<sup>††</sup> 高橋 篤司<sup>†</sup> 梶谷 洋司<sup>†</sup>

Hierarchical BSG Floorplan for Hierarchical VLSI Circuit Design

Zhonglin WU<sup>†a)</sup>, Shigetoshi NAKATAKE<sup>††</sup>, Atsushi TAKAHASHI<sup>†</sup>,  
and Yoji KAJITANI<sup>†</sup>

あらまし 大規模集積回路の階層設計に対応できる階層レイアウト設計方式として、一般構造配置方式の一つ、BSG 方式を階層化し、配線も考慮に入れたレイアウト方式に発展させる。回路設計における階層を忠実に表現し、レイアウトの観点から評価し、配置を再構築して最適化し、その結果を回路設計修正の指針としてフィードバックする設計システムの構築が目的である。BSG の階層化は、小さい BSG (要素 BSG) を根つき木で組み合わせて実現する。要素 BSG のサイズ、個数及び階層木は設計者が指定できる。数千のセルを組み合わせて階層的回路設計環境を人工的に設定し、階層化効果を示す実験を行った。結果の面積及び配線長についても、階層化しない場合に劣らぬ性能が観察された。なお、配置アルゴリズムとしては、扱える方形数が一けた上昇し、数万個の方形が十分な品質で配置できるようになった。

キーワード BSG, 階層化 BSG, レイアウト, 協調設計, フロアプラン

### 1. ま え が き

大規模集積回路のチップ設計を回路設計とレイアウト設計とに別けて考えると、前者は、高性能化、大規模化(例えばシステム LSI)に伴って設計は階層化に向かっている。これからの大きい流れになるようになっている IP 利用設計方式は、一段とこの勢いを強めるであろう。後者、レイアウト設計は回路設計に従属する。したがって、ここも回路設計の階層との柔軟な対応がとれる階層設計が強く求められている。最近の設計の困難の多くはレイアウト後に判明する遅延や面積削減の限界に関する見積もりを速やかに回路設計へフィードバックできないことに帰結されるからである。

回路設計に対応するレイアウトの階層設計で現在自動化まで進んでいるフロアプランはスライスライン構造に基づく方式(以下 SL 方式と呼ぶ)だけであると言ってよい。これは、平面を直線で二つの半平面に分け、それぞれのフロアプランを実行し、結果を合わせ

る、という過程を再帰的に実行する。二つの部分回路の合成を繰り返して実現する単純な回路設計方式とはうまく整合し、重宝されている。しかし、一樣な平面を直線で 2 分する方式がレイアウトの一般性を欠くことは、これでは表せない 4 畳半構造をあげるまでも無く、明らかである。この欠陥を補おうとして登場したフロアプランの一つが本論文で拡張の対象とする BSG 構造に基づく方式(以下 BSG 方式と呼ぶ)である(文献 [1]~[3] 参照 [3] は解説であり、関連文献があげられている)。これは非階層であることにより構造の一般性を獲得している。

ところが、BSG 方式は、配置アルゴリズムとしては実用上の需要を追い抜く性能を発揮しているにもかかわらず、回路のレイアウト設計においては、現在までの進展を見る限り、実用的なツールにはなっていない。例えば坂主ら [4] では、構造を著しく複雑化してようやく配線領域を導入しているし、山内ら [5] ではクラスタ化とボトムアップを賢く実行すれば、SL 方式でも BSG 方式に勝る性能が得られると主張している。このように BSG の利点が回路レイアウトにおいて認められないのは、ひとえに回路設計に対応するよう階層化がなされていないからと言える。

本論文では、BSG の階層化により、回路の階層設計を助ける階層レイアウト方式の確立を目的とする。

<sup>†</sup> 東京工業大学大学院理工学研究科電気電子工学専攻, 東京都  
Electric & Electronic of Engineering, Tokyo Institute of  
Technology, Tokyo, 152-8552 Japan

<sup>††</sup> 北九州大学 国際環境工学部, 北九州市  
Faculty of International Environmental Engineering,  
Kitakyushu University, Kitakyushu-shi, 802-0001 Japan

a) E-mail: wzl@ss.titech.ac.jp



る点から右のそれへ向かう有向枝を付加する．枝の重みは，対応する 2 セグの間の部屋に割り当てられたモジュールの幅とする．割当てが無ければ重み 0 とする (図 (C))．大ソース，大シンクと呼ばれる点  $s_v, t_v$  を設け， $s_v$  から最も左にあるセグに対応するすべての点に重み 0 の枝を付加する．最も右にあるセグに対応するすべての点から  $t_v$  に至る重み 0 の枝を付加する．

$G_h$  は水平セグについて同様に定義される (図 (D))．

Step 3.  $G_v$  の  $s_v$  から各点に至る最長パスの長さを求める．原点を決め，その長さをセグの  $x$  座標とすれば，各部屋はモジュールを収めるに十分な幅を有する．

水平制約グラフ  $G_h$  を用いて同様に水平セグの位置を決める．すると，各部屋は割り当てられたモジュールを収める十分なサイズとなり，配置が得られる (図 (E))．□

Si-BSG-Pack の計算の主要部は，ソースから各点に至る最長パス計算である．これは，枝数のオーダで可能である．枝数は図から明らかに部屋数のオーダ  $O(p \times q)$  であり，これが計算量である．

文献 [1] 等の実用化されている配置アルゴリズムでは，モジュール割当てを次々と取り替えてより良い解を探索する．部屋数  $n \times n$  の有限な BSG を用意すれば ( $n$  はモジュール数)，どのような配置に対しても，それより高さも幅も大きくないフロアプランを与える割り当て方が存在する，という事実が知られている [2]．つまり，この有限 BSG の上へのあらゆる割当てについて配置を求めれば，最小面積配置を得ることができる．BSG のサイズを  $n \times n$  とすれば，最適配置は全探索で有限時間内に求まる．しかし，これは計算量上実際的であるとは言えない．論文 [1] では，BSG のサイズを正方に  $p \times p \sim 4n$  あたりに定めれば，SA (Simulated Annealing) による探索手法で実用的には十分な密度が得られると述べている．

### 3. 階層化 BSG

階層化 BSG は根付き多分木 (階層木) で表される．各点はサイズが  $p \times p$  である正方 BSG を表す．これを要素 BSG と呼ぶ (図 3 を参照されたい)．

木の各点を  $b_i (i = 1, 2, \dots, \ell)$  と表す．これは対応する要素 BSG をも表す記号とする．

枝は下 (葉) 側から上 (根側) に向かう有向枝とする．枝  $(b_i, b_j)$  が存在する，すなわち， $b_i$  が  $b_j$  の直接の子であるとき，枝はデータ構造で使うポインタのような意味をもち，その先は  $b_j$  の一つの部屋を指し，

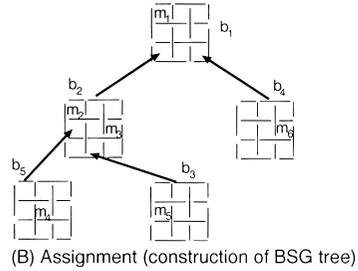
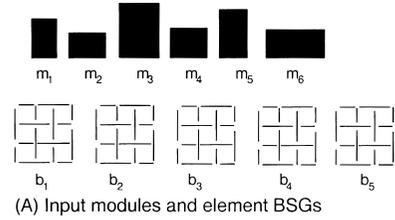


図 3 Hi-BSG-Pack の階層木:  $n = 6, \ell = 5, p = 3$   
Fig. 3 The hierarchical tree in Hi-BSG-Pack:  $n = 6, \ell = 5, p = 3$ .

$b_i$  がその内容である．階層化 BSG による配置は，先の Si-BSG-Pack を使って以下のようにまとめられる (Hi は Hierarchical の略である)．

Procedure:Hi-BSG-Pack

要素 BSG の全部屋は  $\ell p^2$  個存在し，そのうち  $\ell - 1$  個はポインタによって指されている．したがって，モジュールを割り当てることのできる部屋数は全部で  $\ell p^2 - \ell + 1$  個である． $\ell p^2 - \ell + 1 \geq n$  であるように  $p, \ell$  を決め，全モジュールを，各部屋にたかだか 1 個となるように割り当てる．

以下のように，ボトムアップに配置を計算する．

葉の要素 BSG の配置を決定する．ここには，モジュールしか割り当てられていないから，Si-BSG-Pack を適用し，結果の配置を囲む最小正方形を 1 個のモジュールとして扱う．

以下再帰的に，子 BSG の配置がすべて決まっている要素 BSG に Si-BSG-Pack を適用し，結果の配置を囲む最小正方形を 1 個のモジュールとする．

Hi-BSG-Pack の計算量もオーダとして，やはり枝の数 ( $\sim$  部屋の数) であるから， $O(\ell p^2)$  である．

配置が確定した要素 BSG を考える．各部屋はモジュール割当てがあるか，ポインタで指されているか，あるいはどちらでもない 3 通りのいずれかである．ポ

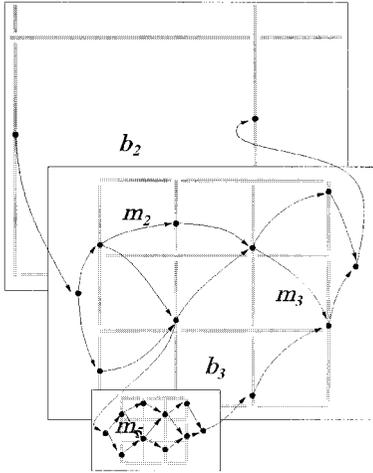


図4 要素BSGの内部配置  
Fig. 4 Inner structure of an Element BSG.

インタで指されている部屋には内部構造がある．その詳細を知るには，ポインタを逆にたどればよい．図4にはその様子を示す．

#### 4. 階層化BSGによるレイアウトの解空間

Hi-BSG-Packは、階層木とモジュール割当てが決まれば、一意にレイアウトを決める．したがってその多様性は、階層木の多様性と割当ての多様性の積である．

一つのレイアウトを解と呼び、Hi-BSG-Packが出力する可能性のある解の集合を解空間と呼ぶ．応用においては、この中で最も良い評価を与える解を探索する．その際、この解空間の大きさ（要素数）は探索の収束性を示す指標である．

1層 ( $\ell = 1$ ) の場合、この空間に最適解が含まれるための一般必要十分条件は  $p = n$  であった（一般必要十分条件とは、条件が満たされれば入力によらず最適解を与える割り当てが可能であり、そうでなければ最適解が含まれないような入力が存在する、という意味）．多層 ( $\ell > 1$ ) 化した場合、必要十分条件は未解決であるが、次の事実が成り立つ．

[定理1] 条件：

$$\ell p \geq n \quad (1)$$

$$p^2 \geq n \quad (2)$$

が最適解が含まれるための必要条件である．

証明 最適解が横1列あるいはその転置に限られる例が存在する．それに対応する割当てを得るためには

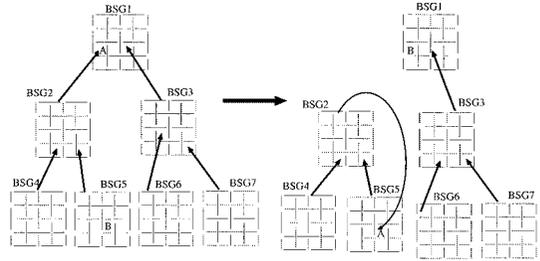


図5 非子孫制約を満たさぬ2部屋AとBとの交換  
Fig. 5 An illegal exchange result of rooms A and B.

式(1)が必要である．また、Poly-spiralと呼ばれる階層をもたない構造[6]を実現するためには式(2)が必要である．□

1層のときでも、実験を通じての経験から、 $p$ は  $O(\sqrt{n})$  にとって、実用上十分であるとしている[1]．我々も実験ではこの考察に頼る．

我々が採用した解空間探索における解の変換操作 (move) は、「二つの部屋の中身の交換」あるいは「部屋の中身の90度回転」である．

前者の場合、以下のような制約が課せられる．

非子孫制約： 中身が交換される二部屋について、一方が他方の子孫BSGの部屋であってはならない．

これが守られなければ、変換の結果、階層木が分離してしまうからである（図5参照）．

変換の結果、新しいレイアウトを得るには、変換によってサイズが変わった要素BSGを計算し直せばよい．その範囲は中身が交換される二つの部屋が属す要素BSGから根までの要素BSGのすべてである．階層化方式では、データ構造上このような部分的更新計算を適用できるので、計算効率の向上に反映するはずである．

#### 5. 回路の階層設計との協調レイアウト設計

我々のフロアプランナの inputs は、回路設計の階層データである．1例を図6(A)に示す．これは機能に関する再帰構造であり、最下層は最小機能回路でセルと呼ぶ．セルを複数集めてまとめた回路をモジュールと呼ぶ．一般に、セル及びモジュールが集まって構成される回路をモジュールという．

フロアプランナが受け取る回路設計のデータはレイアウトを考慮しないで設計されている．したがって、レイアウトしてはじめてその欠陥、ときには特性改善の策が判明する．我々が提案する階層化BSGに基づく

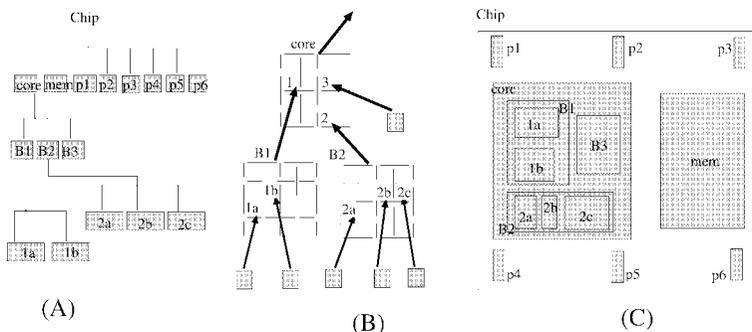


図 6 回路階層木 (A), BSG 階層木 (B) とレイアウト (C)  
 Fig. 6 Circuit hierarchy tree (A), BSG hierarchy tree (B), and layout (C).

フロアプランは、このような情報を取り入れた回路設計を助けるツールになり得る。我々の提案するフロアプランの使い方について言及しておく。実験はこれに対応して実行される。

図 6 の (A) のような回路データ (回路階層木と呼ぶ) を得れば、それを (B) のように BSG 階層木に投影する。この際、要素 BSG のサイズ  $p$  及びポイントの行き先に自由度がある。それらをどのように決めても、一意にレイアウトが決まる。こうして、とりあえず回路が実現でき、回路性能が評価できる。その評価が期待する性能を下回ったときには、レイアウトの修正を試みる。しかしそれは回路設計者が容認できる範囲でなければならない。例えば、階層内のいくつかのモジュールをまとめて近くに配置したいという要求もあれば、設計の都合上特定の回路階層をまったく変えたくない要求も考えられる。本システムでは、回路設計者はそのような要求を回路のプロパティとして記述し、レイアウト設計に回す。プロパティの設定は回路階層木の上で、例えば次のように記述されるであろう。

Free: 制限なし。モジュールやセルは制限なしで移動できる。

Lock: 一つの要素 BSG を指定し、その BSG 内及びその子孫のモジュールやセルは親の要素 BSG から出ることを禁止する。ただし、それ以外の要素 BSG のモジュールやセルが入ることは許す。モジュールを近くにまとめたいとき、このように設定する。この制限によって生ずるかもしれない無駄な空間には外からのモジュールを受け入れることができる。

ほかにも場合ごとにそれにふさわしいプロパティを設定することができる。

こうして、レイアウトの観点から評価の良い解が得られる。それは回路設計者から与えられた要求を遵守する範囲での最適解であり、回路設計者はその採用を考慮することになる。

## 6. 階層化 BSG の実装及び実験

Hi-BSG-Pack を構築し、プロパティの枠内での move による SA で探索するフロアプランを実装した。我々のフロアプランは回路の階層設計をサポートするものであり、プロパティを忠実に守ることに意味をもつ。6.1 はプロパティに関する実験である。

6.2 は配線なしのレイアウト、すなわち配置アルゴリズムとしての性能を実験する。

最近では高位言語でモジュールを設計する機会が多い。このような場合、合成されるモジュールのサイズを制御できず、分散が大きい。一方、スタンダードセル方式のようにレイアウトの便宜上モジュール形状をそろえる場合もある。6.2.1 ではそれぞれの場合に、我々のフロアプランはどのように振舞うかを実験する。

アルゴリズムの階層化は必然的に扱える問題の規模を拡大する。6.2.2 は規模に対する挑戦である。

### 6.1 回路設計をサポートする性能

ここで提案するシステムはレイアウト設計のサポートが主目的であるが、実際の環境におけるフロアプランとしての性能を見るため、スタンダードアレー用の実問題を参考にし、配線つき階層回路を以下のように用意した。

$N_0$ : セル数は 1863, ネット数は 2071, 端子数平均は 3. 階層数  $\ell$  は 441,  $p^2\ell \sim 4n$  である。

我々の問題はこれを配置配線することである。最適

表 1 Hi-BSG-Pack の Si-BSG-Pack に対する性能評価

Table 1 Comparison of performances of Hi-BSG-Pack to Si-BSG-Pack.

No.	実験内容	面積	配線長	評価 $f$	時間 (s)
実験 1	Si-BSG-Pack	$1.00 = S_1$	$1.00 = L_1$	1.00	28079
実験 2	Hi-BSG(all lock)	0.97	1.29	1.04	6030
実験 3	Hi-BSG(free)	0.95	1.01	0.96	7044
実験 4	Hi-BSG(lock)	0.95	1.01	0.96	6744

化の評価関数を  $f = \alpha \times S + \beta \times L \times W$  とする．ここで、 $S$  は配線領域を考慮しない配置面積であり、 $W$  は配線幅、 $L$  は配線すべき端子を囲む方形の半周長の全ネットについての総和である． $\alpha = 1$ 、 $\beta = 1$ 、 $W = 0.05$  と設定した．

以下のように 4 通りの実験を行った．

実験 1:  $N_0$  から階層を外し、従来の Si-BSG-Pack を適用する．この結果のレイアウトを  $N_1$  とし、その面積と配線総長を比較の基準とする．ほかの実験はすべてこれらとの比で表され、小さいほど良いと理解する．なお、計算資源をいくらでも使えば、Si-BSG-Pack が最高品質を出力する．

実験 2:  $N_0$  の全要素 BSG を Lock して Hi-BSG-Pack による SA を適用する．得られた結果を  $N_2$  とする．

実験 3:  $N_2$  にプロパティ Free を設定し、Hi-BSG-Pack による SA を適用する．

実験 4:  $N_2$  に対応する階層木において、根の子となる 1 要素 BSG を指定し、それを Lock して Hi-BSG-Pack による SA を適用する．

表 1 には以上の実験結果がまとめてある．図 7 には実験 2 のレイアウト  $N_2$ 、図 8 には実験 3 のレイアウトそれぞれが示してある．

SA なので計算時間の比較に意味をもたせるため、試した交換の回数が等しくなるようにスケジュールを設定した．したがって、計算時間の差は 1 回当りの計算速度に比例する．

これらの実験の範囲では以下のように観察される．階層化によって約 4 倍の高速化が達成されている．実験 2 の配線長を除き、Si-BSG-Pack の結果と同程度の出力を得ている．

$\ell$ 、 $\alpha$ 、 $\beta$  の影響を見るために、これら及び初期解を様々に取り替えて実験を行った．ここでは載せなかったが、 $\alpha$ 、 $\beta$  の比が 100 程度を越せば、配線長あるいは面積に予想される影響が観測された．そうでないときには、面積も配線長もほぼ一定であった．表 1 はこの場合のデータで、 $\alpha$ 、 $\beta$  の比は 1 である．

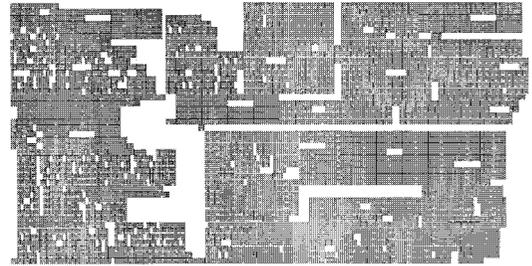


図 7 実験 2 のレイアウト  
Fig. 7 Layout  $N_2$  by Experiment 2.

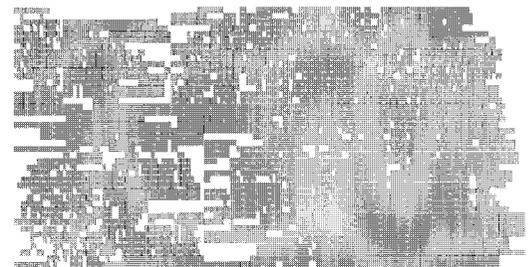


図 8 実験 3 のレイアウト  
Fig. 8 Layout of Experiment 3.

## 6.2 配置アルゴリズムとしての性能

### 6.2.1 分散と move

先の実験では move は非子孫制約さえ満たせば無条件であったが、部屋は階層化されているので、交換される対がどのような階層から選択されるかによって異なる効果が予想される．これを見るために、move 操作を以下のように 3 通り定義して SA を実行する：

- local move: 各要素 BSG の部屋の中身に関する move.
  - filiation move: 一つの要素 BSG の部屋と、その要素 BSG の親 BSG の部屋に関する move.
  - global move: 任意の 2 部屋に関する move.
- 回路を配置するにあたって、回路を特徴づけるのはモジュール形状の分散であると考えられる．そこで、こ

れが著しく異なる二つの回路 A, B を図 9 に示すように人工的に作成した. 共通にモジュール数  $n = 3000$ , 階層数  $\ell = 300$  である.

A, B に対する結果をそれぞれ表 2, 表 3 に示す. A の場合, local move でよい結果を得られる. B の場合, 階層を超えた filiation move や global move などがよい結果が得られる.

これは, 回路 A ではモジュールの形状分散が小さく, 同じ階層内にソフトブロックのようにうまく詰められたからと思われる. これは更に SA でなく, greedy なアルゴリズムでもよい結果が得られるのではないかと推測する. これに対し, 回路 B はモジュールの形状分散が大きく, 同じ階層だと面積的にどうしても無駄なスペースができてしまい, BSG 内 move よりも階層を超える filiation move か global move の方が効率が良くなる. 図 10, 11 は回路 A, B それぞれのレイアウト

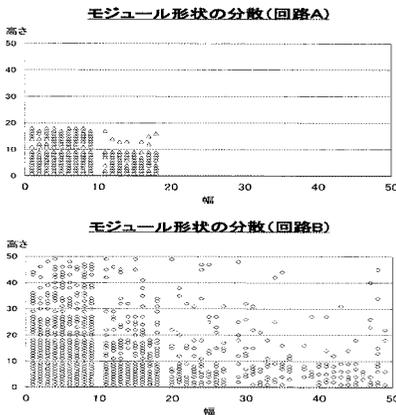


図 9 入力回路 A: モジュール面積分散小, B: 分散大  
Fig. 9 Input circuits A: uniform, B: diverse.

表 2 回路 A に対する SA の結果  
Table 2 Results of SA on circuit A.

No.	操作	面積	時間	時間 (s)
1	Si-BSG-Pack	1.00	1.00	66157
2	local move	0.83	0.09	6019
3	filiation move	1.04	0.33	21660
4	global move	1.03	0.29	19225

表 3 回路 B に対する SA の結果  
Table 3 Results of SA on circuit B.

No.	操作	面積	時間	時間 (s)
1	Si-BSG-Pack	1.00	1.00	73209
2	local move	1.17	0.10	7408
3	filiation move	1.02	0.27	19507
4	global move	1.02	0.23	17162

ト例を示す.

### 6.2.2 スケラビリティ

階層化のもう一つのメリットは入力規模の拡大である. Si-BSG-Pack ではたかだか千個が扱える上限であったのに対し, Hi-BSG-Pack では以下のように数万個のデータを処理できる.

入力: ランダムに作成した  $n = 28000$  個のモジュール  
Hi-BSG:  $p = 15, \ell = 300, \ell p^2 = 67,500$

SA: 標準的な温度スケジュール

図 12 に, Pentium 90 MHz で 41 時間を要した結

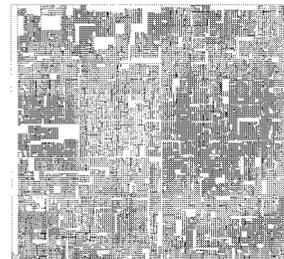


図 10 Local move だけを適用した回路 A のレイアウト  
Fig. 10 Layout of circuit A by localmove operation only.

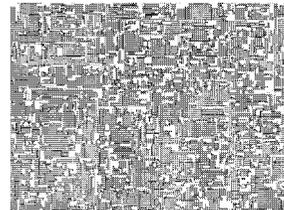


図 11 Global move を適用した回路 B のレイアウト  
Fig. 11 Layout of circuit B by global move operation only.

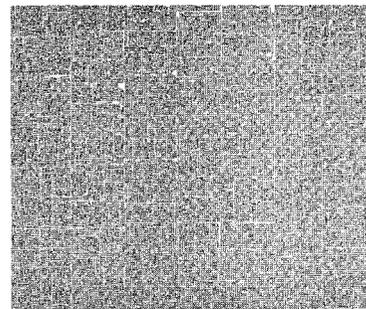


図 12 モジュール 28,000 個の配置: 配置面積はモジュール面積総和の 1.15 倍  
Fig. 12 An example of packing of 28,000 modules with area ratio 1.15.

果を示す．階層構造が観察できる．正解への近さはわからないが，モジュール面積総和に対する配置面積の比を見ると 1.15 であるので，実行時間内で扱える入力規模の壁を大幅に広げたとと言える．

## 7. む す び

回路の設計階層に忠実かつ建設的に対応するフロアプランは今後の不可欠ツールである．更に，回路設計において，早い段階で出来上がったときの全体を見積もることが重要である．このような認識から BSG を階層化することによる一つの方式を提案した．

回路設計方式を考慮して，形状の分散が異なる回路に対し本方式の適用実験を試みた．階層化のもう一つのメリットである実用的な計算資源で扱える入力規模の上限拡大を実験で示した．

本方式は多数のパラメータを含む．それらを最適に決めるために，実用上の適用経験を積むことが今後の課題である．

## 文 献

- [1] S. Nakatake, H. Murata, K. Fujiyoshi, and Y. Kajitani, "Module placement on BSG-structure and IC layout applications," International Conf. on Computer Aided Design, pp.484-491, San Jose, US, Nov. 1996.
- [2] S. Nakatake, H. Murata, K. Fujiyoshi, and Y. Kajitani, "Module packing based on the BSG structure and IC layout applications," IEEE Trans. CAD, vol.17, no.6, p.519, June 1998.
- [3] 梶谷洋司, "配置の数理: 多数の長方形を最小面積に埋め込む," 信学技報 (招待), VLD98-38, 1998.
- [4] S. Nakatake, K. Sakanushi, Y. Kajitani, and M. Kawakita, "The channeled-BSG: A universal floorplan for simultaneous place/route with IC applications," International Conf. on Computer Aided Design, pp.418-425, San Jose, US, Nov. 1998.
- [5] T. Yamanouchi, K. Tamakashi, and T. Kambe, "Hybrid floorplanning based on partial clustering and module restructuring," International Conf. on Computer Aided Design, pp.478-483, San Jose, US, Nov. 1996.
- [6] T. Izumi, A. Takahashi, and Y. Kajitani, "Air pressure model and fast algorithm for zero-wasted-area layout of general floorplan," IEICE Trans. Fundamentals, vol.E81-A, no.5, May, 1998.

(平成 11 年 9 月 24 日受付, 12 年 3 月 10 日再受付)

## 吳 中 林



平 8 東工大・工・電気・電子卒．平 10 同大学院理工学研究科電気・電子工学専攻修士課程了．現在，同大学院博士課程．VLSI レイアウト設計に関心をもつ．

## 中武 繁 寿



平 4 東工大・電気・電子卒．平 6 北陸先端大修士課程了．平 8 東工大助手．平 10 同工学博士．平 11 北九州大学 国際環境工学部 講師．VLSI レイアウト設計に関する研究に従事．

## 高橋 篤 司 (正員)



平 1 東工大・工・電気・電子卒．平 3 同大学院理工学研究科電気・電子工学専攻修士課程了．同年東京工業大学工学部助手．平 9 より同助教授．グラフ理論, VLSI レイアウト設計自動化に関する研究に従事．平成 9 年度電子情報通信学会学術奨励賞受賞．博士 (工学)．情報処理学会, IEEE 各会員．

## 梶谷 洋 司 (正員)



昭 44 東工大大学院博士課程了．同年工学博士．同年東工大助手．同大学助教授を経て教授, 現在に至る．92 年から 4 年間北陸先端大教授併任．東工大研究体 CAD21 研究長．グラフ理論．組合せ理論を経て現在は VLSI のレイアウト設計研究に関心が向かっている．昭 46, 48, 60 論文賞などいずれも本会から受けた．1992 年 IEEE Fellow．