

C-007

GAを用いたフロアプラン専用エンジンの設計と試作 Genetic Algorithm Engine for VLSI Floorplanning

前田 哲広† 吉川 雅弥‡ 寺井 秀一†
Akihiro Maeda Masaya Yoshikawa Hidekazu Terai

1. はじめに

VLSI のレイアウト設計における重要な設計工程のひとつに、論理機能を有するいくつかのモジュールを、配線遅延などの制約条件を満たしつつ、面積が最小となるようにチップ上に配置するフロアプランがある。システム LSI や SoC など、大規模な LSI において特に重要となるこの工程はこれまで数々のヒューリスティック手法が提案されてきたが、中でもシーケンスペアを解表現に用いた遺伝的アルゴリズム (Genetic Algorithm:GA) が良好な結果を得ている[1]。

GA は生物の進化の過程を工学的にモデル化したアルゴリズムであり、選択・交叉・突然変異の3つの遺伝オペレーションを適用し、集団を進化させて解を導く。GA は多点探索を行うアルゴリズムであるために、探索範囲が広く局所解に陥りにくい。一方、多点探索アルゴリズムから来る処理時間の問題が内在する。そのため GA を実用的なアプリケーションに適用することを考えた場合、専用ハードウェアによるアプローチが重要になる。

GA をハードウェアで実現することを目的とした GA ハードウェアの設計事例には、Scott[2], Graham[3]等があるが、その報告例は小規模な問題に適用したもので、実用的なアプリケーションに適用した報告例は、これまで見当たらない。

著者らはこれまで、進化戦略に基づく高速なフロアプランを実現する専用エンジンアーキテクチャを提案し、評価実験により提案アーキテクチャの有効性を確認してきた[4]。本研究ではこれまでに提案してきたアーキテクチャにテスト回路を組み込み、LSI として試作を行う。

2. 基本アーキテクチャ

2.1 個体表現

提案アーキテクチャでは、解表現方法にシーケンスペアを用いているため、個体表現を図のように行う。各ブロックの回転 (θ) は $0^\circ, 90^\circ, 180^\circ, 270^\circ$ を 2 bit で、アスペクト比は 6 bit で表現する。シーケンスペアで表されたブロック相対位置は $8 \text{ bit} \times 2 \text{ n}$ で表現する。

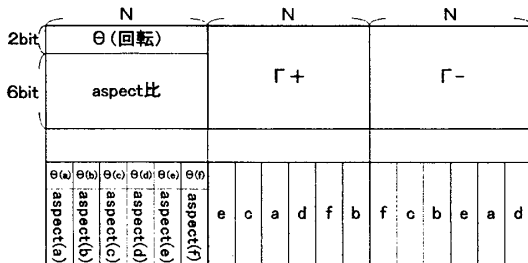


図1. 個体表現

2.2 交叉

ランダムに連続した四角形の部分領域 (以後、窓領域とする) を作成し、窓に含まれたモジュールを対象として交叉を行う。

STEP1: 親個体 1, 親個体 2 に対してそれぞれランダムな窓領域を生成する。

STEP2: 親個体 1, 親個体 2 の窓領域外の遺伝子をそのまま子個体 1, 子個体 2 の遺伝子にコピーする。

STEP3: 親個体 1, 親個体 2 の窓領域内の遺伝子座を他方の出現順序で子個体 1, 子個体 2 の遺伝子にコピーする。

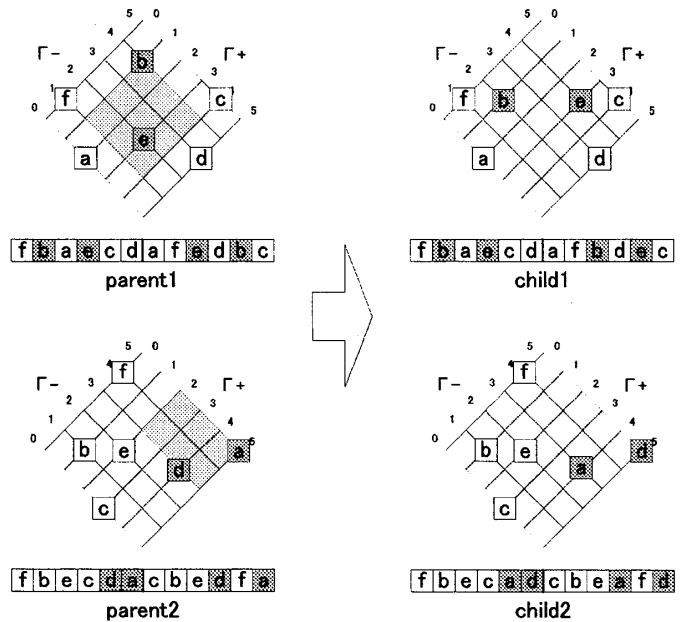


図2. 交叉手法

2.3 突然変異

突然変異は、パラメータで与えられた選択確率で次のうち1つの方法を選択し、行う。

- アスペクト比一点突然変異
1つのブロックをランダムで選択し、アスペクト比を変異させる一点突然変異法を用いる。アスペクト比はあらかじめ定められた最大値と最小値の間の値となるよう変異させる。
- 回転 (θ) 一点突然変異
1つのブロックをランダムで選択し、 θ を変異させる一点突然変異法を用いる。
- $\Gamma+$ 突然変異
1つのブロックをランダムで選択し、 $\Gamma+$ のみを変異させる。変異後に重複するブロックについては、重複しないように突然変異対象ブロックの変異前の座標に変異させる。

†立命館大学大学院 理工学研究科 Ritumeikan University

‡名城大学 理工学部 情報工学科 Meijo University

- 「-突然変異
1つのブロックをランダムで選択し、「-」のみを変異させる。変異後に重複するブロックについては、重複しないように突然変異対象ブロックの変異前の座標に変異させる。
- 「+, 「-突然変異
1つのブロックをランダムで選択し、「+」と「-」を変異させる。変異後に重複するブロックについては、重複しないように突然変異対象ブロックの変異前の座標に変異させる。
- ペア交換
2つのブロックをランダムで選択し、「+」、「-」それぞれの値を入れ替える。

3. テスト回路

動作テスト用回路は、外部からテストモードの入力信号を受けると、テスト用パターンとして2つの親個体データを生成し、同時にブロック数、個体数、窓領域設定などのパラメータ値をランダムに生成する。次に、交叉回路からの窓領域を受信し、期待値を生成する。また、突然変異回路から突然変異方式・突然変異箇所を受信し、期待値を生成する。その後、それぞれの期待値と処理データを比較し、比較結果をチップ外に送信する。この結果、チップの動作確認をすることができる。

テスト用個体はランダムな個体データを使用するのが理想であるが、交叉オペレーションはランダムな窓領域を用いるため、ランダムな個体データをテスト用に与えると期待値生成が困難である。また、回路規模も期待値生成のための回路が交叉回路と同等以上になる。そこで、図4に示すような単純な個体データを親個体データとして用いる。

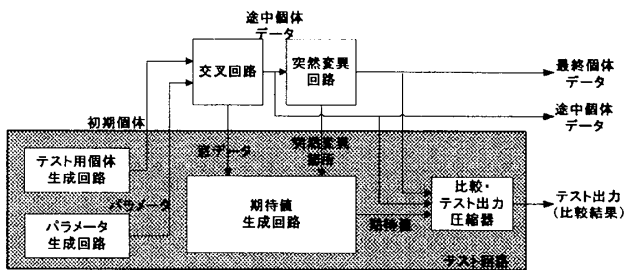


図3. DFT回路

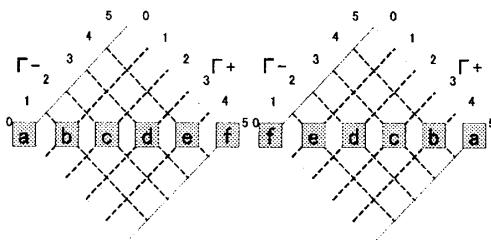


図4. テスト用親個体

4. 評価実験

今回作成したフロアプランシステムの有効性は Gigascale Silicon Research Center のベンチマークにより確認した。システムの ASIC 化にあたっては 0.18μm CMOS プロセスライブラリ、Cadence 社の NC-Verilog, Physical Knowledgeable Synthesis, Silicon Ensemble, Virtuoso, Dracula, Synopsys 社の Design Compiler, Prime Time を用いた。チップの仕様を表1に示す。

また、ASIC と FPGA との遺伝オペレーション1ステップについての処理速度の比較を表2に示す。ただし、ASICとのデータの送受信はPCIバスで行うものとする。

表1. チップの仕様

チップサイズ	6.71 mm×6.81 mm
I/O ピン数	133pin
動作周波数	33.3 MHz
ゲート数	27420
セル数	8159
Dual-Port RAM	256word×8bit : 10 512word×8bit : 5

表2. PCIバスを用いた場合のFPGAとの速度比

ブロック数	FPGA(μs)	ASIC+PCIバス(μs)	FPGA:ASIC 速度比
10	6.90	4.599	1.500
30	18.90	12.715	1.486
50	30.90	20.831	1.483
100	60.90	41.122	1.481
200	120.90	81.704	1.480

5. まとめ

解探索手法に GA を用いたフロアプランシステムの更なる高速化を目指したチップの試作を行うことができた。このことにより、フロアプランシステムの更なる高速化が期待できる。今後の展望としては、チップの評価、フロアプランシステム全体の ASIC 化、SA などとのハイブリッド化による局所解からの脱出の容易化などが挙げられる。

6. 参考文献

[1]Koichi Hatta, Shin'ichi Wakabayashi, Tetsushi Koide, "Solving the rectangular packing problem by an adaptive GA based on sequence pair," Proc. Asia-South Pacific Design Automation Conference, pp.181-184(1999).

[2]S.D.Scott, A.Samal, and S.Seth: "HGA: A Hardware-Based Genetic Algorithm", Int. Symposium on Field-Programmable Gate Array, pp.53-59,1995.

[3]P.Graham and B.Nelson:"Genetic Algorithm in Software and in Hardware A performance Analysis of Workstation and custom Computing Machine Implementations", FPGAs for Custom Computing Machines, pp.216-225,1996.

[4] 金光亮輔, 尾藤彰訓, 吉川雅弥, 寺井秀一「シーケンスペアに基づく交叉専用アーキテクチャの設計」 電子情報通信学会技術研究報告 Vol.104 No.589 2005 pp.69-74