

FPGA 配線問題に対する貪欲法と ニューラルネットワークを併用した 3 段階アルゴリズムの提案

江川 晋爾 黒田 映史 北道 淳司 船曳 信生

大阪大学大学院基礎工学研究科情報数理系専攻

{egawa,e-kuroda,kitamiti,funabiki}@ics.es.osaka-u.ac.jp

あらまし

本稿では、二次元配列型 FPGA の配線問題に対して、貪欲法とニューラルネットワーク (NN) 解法を併用した 3 段階アルゴリズムを提案する。提案するアルゴリズムは、経路候補抽出アルゴリズム、Greedy 配線アルゴリズム、NN アルゴリズムの 3 段階で構成される。まず、経路候補抽出アルゴリズムが各配線の有効な経路候補を抽出する。次に、Greedy 配線アルゴリズムが経路候補を用いてヒューリスティックに初期配線を行う。最後に、NN アルゴリズムが初期配線のうち長い配線結果を固定した上で、残りの配線の改良を行なう。性能評価として、9 種類の例題を作成し、シミュレーションを行う。その結果、3 段階アルゴリズムにより Greedy, NN 各アルゴリズム単体で用いる場合よりも高精度の解を求めることができることを示す。

キーワード

FPGA, 配線問題, 貪欲法, ニューラルネットワーク, 三段階アルゴリズム

A proposal of a three-stage greedy neural-network algorithm for FPGA routing problems

SHINJI EGAWA, EIJI KURODA, JUNJI KITAMICHI,
AND NOBUO FUNABIKI

Department of Informatics and Mathematical Science,
Graduate School of Engineering Science, Osaka University

{egawa,e-kuroda,kitamiti,funabiki}@ics.es.osaka-u.ac.jp

Abstract

In this paper, we propose a three-stage routing algorithm for two-dimensional Field-Programmable Gate Arrays (FPGAs) by combining the greedy algorithm and the neural network algorithm. The three-stage algorithm consists of the route-candidate extraction stage, the greedy routing stage, and the neural network routing stage to achieve the high routability. The first stage extracts necessary route-candidates composed of a sequence of routing segments for every net. The second stage selects one of the candidates and assigns one track for each net so as to avoid short circuits. The final stage rearranges locations of shorter nets by the second stage to reduce the necessary tracks. We evaluate the performance of our algorithm by solving nine randomly generated instances, where the simulation results show that our three-stage algorithm achieves the high performance by used together.

key words

FPGA, Routing, Greedy, NeuralNetwork, Three-stage Algorithm

1 はじめに

FPGA (Field Programmable Gate Array) は、規則正しく整列配置された論理ブロックと、各論理ブロックの入出力端子間の信号配線のための配線領域から構成される、ユーザによるプログラム可能な VLSI である。論理ブロックは、小規模な論理機能を RAM によって実現される。また、論理ブロックの入出力端子間の接続は、チャンネルセグメントと呼ばれる配線領域中にあらかじめ用意された複数の信号線を用いて実現される。同一チャンネル内の各信号線はトラックと呼ばれている。各信号線の端はスイッチにより絶縁されており、スイッチの ON/OFF によって、論理ブロックの入出力端子と信号線間や、絶縁された信号線間の接続/断線がなされる。FPGA は、この柔軟性と製作の低コストのため、ASIC において非常に注目され、活用されてきている。本稿では、FPGA において与えられた配線要求に対する経路及びトラックを決定する配線問題を対象としている。

FPGA 配線問題に対する従来の研究では、その NP 完全性が証明されており、いくつかの近似解法が提案されている [1]。既存の配線手法の多くは、概略配線問題 [2] と詳細配線問題 [3] に分割し、二段階で配線を行っている。まず、概略配線問題でチャンネル密度が均等になるように配線経路を決定し、その結果を用いて、詳細配線問題で適切なトラックを割り当てるといふものである。この手法では概略配線を適切に行えたとしても、詳細配線で良い配線結果が得られることが保証されないという問題がある。そこで近年、配線を一段階で行う手法もいくつか提案されている [4][5]。

本稿では、FPGA 配線問題に対し、経路候補抽出アルゴリズム、Greedy 配線アルゴリズム、ニューラルネットワーク配線アルゴリズムの 3 段階で構成される解法を提案する。本問題では、各配線に対して取り得る信号線上の経路数が非常に多く、それらを全て配線上の考慮に入れるのは困難である。そこで、本解法では、まず経路候補抽出アルゴリズムにより、解を得るのに必要かつ有効であると思われる経路候補のみを抽出する。以後の配線配置アルゴリズムでは、それらの経路候補のみを考慮することとしている。次に、Greedy 配線アルゴリズムを用いて各配線の配置を行う。最後に、更なる配線配置の最適化を目的として、Greedy 配線アルゴリズムでの配置結果の一部を固定した上で、

ニューラルネットワークアルゴリズムを用いて固定されていない配線の配置に対する最適化を行なう。

以降、2 章では対象とする FPGA モデルの概要について述べる。3 章では FPGA 配線問題の定義について述べる。4 章では FPGA 配線問題に対する 3 段階アルゴリズムの提案を行なう。5 章ではシミュレーションによる性能評価について述べる。最後にまとめを 6 章で行う。

2 FPGA モデル

本稿で対象とする FPGA モデルは、Xilinx 社の 4000 シリーズに準じたものとする。対象とする FPGA モデルを図 1 に示す。本モデルは、論理ブロック (L)、スイッチブロック (S)、コネクションブロック (C) の 2 次元配列と、それらの間の領域である水平および垂直の配線チャンネルから構成される。

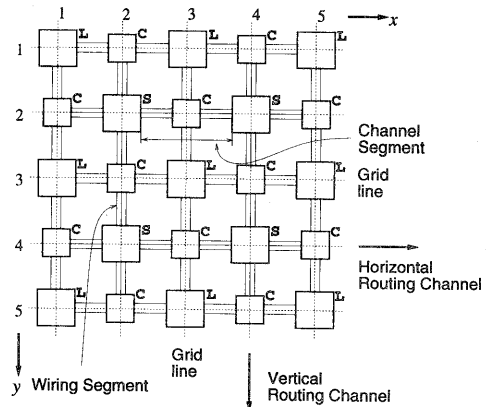


図 1: FPGA の構造

L ブロックは、プログラム可能な小規模の組合せ回路と順序回路を含み、これらによって回路の論理機能を実現する。

S ブロックは、あるチャンネルセグメントの信号線を、他のチャンネルセグメントの信号線に接続するのに用いられるスイッチボックスである。

C ブロックは、L ブロックの入出力端子をチャンネルセグメントの信号線に接続するのに使われるスイッチボックスである。

本稿で対象とする FPGA では、各チャンネルセグメントは同数の信号線を持つものとし、その数をトラック数 W と呼ぶことにする。また、C ブロックでは、図 2 に示すように $F_c = W$ であることと

する。ここで、 F_c とは、1つのLブロックの端子に接続可能な信号線の数である。また、Sブロックでは、図3に示すように $F_s = 3$ とする。ここで、 F_s とは、1つの信号線に接続可能な信号線の数である。

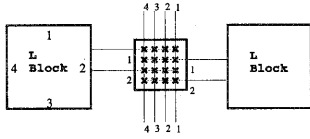


図2: Cブロック ($F_c = W$)

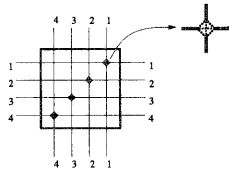


図3: Sブロック (diagonal S box)

本稿において以上のFPGAモデルを対象としたのは、 F_c が W であるなら、 F_s を3よりも大きくしてもFPGAの配線可能性の大きな向上は得られず、逆に、 F_s を3よりも小さくすると、 F_c を大きくしても配線可能性は大きく低下してしまうことが知られているからである [6][7].

3 FPGA 配線問題の定式化

まず、本問題において入力として与えられるものは、FPGAの配線構造と、ネットと呼ばれる配線要求の集合である。FPGAの配線構造とは、FPGAの規模を表すLブロック数、SブロックとCブロックの構造、L、C、S各ブロック間の配線構造の情報である。また、ネットは、配線要求のあるLブロックの入出力端子の集合で与えられる。そこで、FPGA配線問題とは、FPGAの配線構造とネット集合が与えられた時、各ネットを配線構造に矛盾しないように配線することである。

次に、ネットについてであるが一般には多端子ネットである。しかしながら、多端子ネットは複数の2端子ネットに分割することが可能であるため、本稿でのネットは2端子ネットのみであるとする。

以上をまとめると、本研究の対象とするFPGAの配線問題は、同一信号線に2ネット以上割り当てないように、各ネットに対して以下の2つを決

定する問題である。

- (1) ネットの通過する経路 (チャンネルセグメント系列)
- (2) ネットを配置するトラックドメイン (トラック番号)

4 三段階アルゴリズム

提案する3段階アルゴリズムは、逐次のヒューリスティック解法と並列的ニューラルネットワーク解法を併用して、FPGAの配線問題の準最適解の探索を行なうものである。本アルゴリズムは、以下の3つのアルゴリズムで構成されている。

- (1) 経路候補抽出アルゴリズム
各ネットについて、解を得るために必要であると考えられる経路候補群を抽出する。
- (2) Greedy 配線アルゴリズム
文献で提案されているヒューリスティック手法 [8] を用いて逐次的に各ネットを配置し、必要トラック数 W の上限を求める。
- (3) ニューラルネットワークアルゴリズム
Greedy 配線アルゴリズムによる配線結果の一部を固定し、配線の固定したネットを除く残りの各ネットに対しニューラルネットワーク解法 [9][10] を用いて並列的に配線を行なう。

以上の3つのアルゴリズムを順次行なうことで、FPGAの配線問題の高精度の解を探索する。以降、各アルゴリズムの詳細を記述する。

4.1 経路候補抽出アルゴリズム

まず、各ネットの経路候補として、解を得るのに必要である可能性が高いと思われるものから順に、抽出することを考える。小さなトラック数 W で解を得るためには、各チャンネルセグメントに毎に、経路候補の通過数をなるべく均等しておくのが望ましい。これは、経路候補の集中するチャンネルセグメントが存在すると、その部分でトラック数を多く必要とするからである。そのため、そのような部分が全体の配線の成否に大きく影響すると考えられる。

以上を考慮し、経路 j における候補抽出の評価基準として、 $cost(j)$ を適用する。その評価基準となる用語を以下に定義する。

[定義 4.1]

端子CS混雑度 (n) とは、チャンネルセグメント n に

Lブロックの入出力端子が接続されるネットの総数である。

[定義 4.2]

CS 混雑度 (n) とは、チャンネルセグメント n を通過する経路候補を持ち、かつ n に入出力端子を持たないネットの総数と端子 CS 混雑度の和である。

[定義 4.3]

ネット端子密集係数は、以下の式で与えられる。

$$1 + 10 \times \left(\frac{\text{端子 CS 混雑度}(n)}{\text{最大端子 CS 混雑度}} \right)^4 \quad (1)$$

[定義 4.4]

$cost(j)$ とは、経路 j の通過する各チャンネルセグメントにおける、ネット端子密集係数と CS 混雑度の積を全て足し合わせたものである。

また、一般に長いネットほど異なる最短経路数が多い。従って、長いネットほど数多くの経路候補を抽出するのが自然である。そこで各ネット i の経路候補数はネット i のネット長に比例して、以下の式で与える。ただし、ネット長とは、各ネットの最短経路長 (チャンネルセグメント数) である。

$$\text{経路候補数} = \left\lfloor \frac{\text{ネット } i \text{ のネット長} + 1}{2} \right\rfloor \quad (2)$$

ここで $\lfloor x \rfloor$ は x を超えない最大の整数である。

さらに、経路候補を抽出するネットの順番については、ネット長の長いネットほど多くの経路が存在して選択の自由度が高く、ネット長の短いネットの経路を先に抽出しても良い経路が残されると考えられるため、ネット長の昇順に経路候補抽出を行なうこととした。

上述のように経路候補は、ネット長の昇順にネットを選択し、そのネットの経路候補を選出しつつ各経路候補毎に $cost(j)$ を計算し、その昇順に上述で定義した数だけ候補を抽出する。

4.2 Greedy 配線アルゴリズム

本アルゴリズムでは、既に提案されている Greedy 手法である GBP (Greedy Bin Packing) 配線アルゴリズム [8] を用いて、抽出した経路の中から各ネットの経路選択を逐次的に行なう。

GBP アルゴリズムでは、各トラックドメイン毎に、表 1 の 5 つの Pass を順に行なって逐次的にネットを配置していく。全てのネットを配置し終えたら

終了する。配置されていないネットをこれ以上そのトラックドメインに配置できなければ、次のトラックドメインに対して同様に配置を行っていく。各 Pass では、各未配線ネットはネット長の降順に走査され、ネットを 1 つ配置すると、再び未配線の全ネットをネット長最大のものから走査する。

表 1: GBP 配線アルゴリズムの構成

Pass	ネットの制約	経路候補選択の制約	
		経路長	通過セグメント
1	mcc に端子有	最短	mcc を不通過
2	なし	最短	なし
3	mcc に端子有	最短+1 以下	mcc を不通過
4	なし	最短+1 以下	なし
5	なし	なし	なし

ここで、mcc とは、未配線ネットの両端の端子が結線されるチャンネルセグメントの中で結線数が最大のものを指す。

4.3 NN 配線改良アルゴリズム

4.3.1 ニューロン表現とニューロン関数

本稿では、ネット数 N 、ネット i ($i = 1, \dots, N$) の経路候補数 M_i 、トラック数 W に対応して、 $\sum_{i=1}^N M_i \times W$ の 3 次元ニューラルネットワークを用いる。ijk 番目のニューロンの出力 V_{ijk} は、

- (1) $V_{ijk} = 1$: ネット i の経路候補 j をトラック k に割り当てるべき
- (2) $V_{ijk} = 0$: ネット i の経路候補 j をトラック k に割り当てるべきでない

ことを表す。出力 V_{ijk} は、以下に示すヒステリシス・バイナリニューロン関数を用いて、入力 U_{ijk} 更新される。

$$V_{ijk} = \begin{cases} 1 & \text{if } U_{ijk} > UTP \\ 0 & \text{if } U_{ijk} < LTP \\ V_{ijk} & \text{otherwise} \end{cases}$$

ここで、 UTP 、 LTP は定数である。

4.3.2 エネルギー関数

FPGA 配線問題の全ての制約条件を充足した場合のみに値が 0 となり、それ以外の場合には正の値をとる 2 次形式関数をエネルギー関数として採

用する。本稿で対象とする FPGA の配線問題における制約条件は以下の 2 つである。

- (1) 各ネットを 1 つの経路の 1 つのトラックに割り当てる。
- (2) 2 ネット以上を同一チャネルセグメントの同一トラックに割り当てない。

この 2 つの制約条件に対応して、次式のエネルギー関数を定義する。

$$E = \frac{A}{2} \sum_{i=1}^N \left(\sum_{j=1}^{M_i} \sum_{k=1}^W V_{ijk} - 1 \right)^2 + \frac{B}{2} \sum_{i=1}^N \sum_{j=1}^{M_i} \sum_{k=1}^W \sum_{p=1, p \neq i}^N \sum_{q=1}^{M_p} d_{ijpq} V_{ijk} V_{pqk} \quad (3)$$

ここで、 A 、 B は係数である、また d_{ijpq} は定数でその値は、

- (1) $d_{ijpq} = 1$: ネット i の経路候補 j と ネット p の経路候補 q が同一チャネルセグメントを通過するとき。
 - (2) $d_{ijpq} = 0$: ネット i の経路候補 j と ネット p の経路候補 q が同一チャネルセグメントを通過しないとき。
- となるように定める。

4.3.3 動作方程式とその改良

エネルギー関数を 0 とするニューロン出力の探索のために、動作方程式は、最急降下法に基づきエネルギー関数 E を V_{ijk} で偏微分することにより与えられる。さらに、局所解収束を避けるため、動作方程式に対してヒルクライミング項 (C 項) とオメガ関数 (B 項) を導入する。採用した動作方程式を以下に示す。

if ($t \bmod T$) < ω then

$$\begin{aligned} \Delta U_{ijk}(t) = & -A \sum_{q=1}^{M_i} \sum_{r=1}^W (V_{iqr} - 1) \\ & - B \left(\sum_{p=1, p \neq i}^N \sum_{q=1}^{M_p} d_{ijpq} V_{pqk} \right) V_{ijk} \\ & + Ch \left(\sum_{q=1}^{M_i} \sum_{r=1}^W V_{iqr} \right) \end{aligned} \quad (4)$$

else

$$\begin{aligned} \Delta U_{ijk}(t) = & -A \sum_{q=1}^{M_i} \sum_{r=1}^W (V_{iqr} - 1) \\ & - B \left(\sum_{p=1, p \neq i}^N \sum_{q=1}^{M_p} d_{ijpq} V_{pqk} \right) \end{aligned}$$

$$+ Ch \left(\sum_{q=1}^{M_i} \sum_{r=1}^W V_{iqr} \right) \quad (5)$$

ここで、 t は更新回数、 T と ω はオメガ関数の正の定数である。また、 C はヒルクライミング項の係数で、 $h(x)$ は以下に示す関数である。

$$h(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{if } x \neq 0 \end{cases}$$

4.3.4 配線配置の部分固定

本 3 段階アルゴリズムでは、Greedy 配線アルゴリズムを用いた各配線の配置後の更なる最適化を目的として、Greedy 配線アルゴリズムの配置結果の一部を固定した上で、ニューラルネットワークアルゴリズムを用いて、固定されていない配線の配置に対する最適化を行なう。そこで、ニューラルネットワークの更新計算前に、Greedy 配線アルゴリズムでの一部のネットの配置結果に対応したニューロン出力を固定し、それらのニューロンの更新計算をスキップする。ネットの配置を固定するか否かの判定基準にはネット長を用い、以下に定義する配置固定基準長より長いネット長のネットの配置を固定する。これは、長いネットほど配置が困難であると考えられ、それらのネットの配置を固定することによってニューラルネットワークの収束性能を改善できるからである。

$$\begin{aligned} \text{配置固定基準長} = & 200 \times \frac{(\text{Lブロック幅})}{(\text{ネット数})} \\ & + \frac{1}{2} \times (\text{全ネットのネット長の平均値}) \end{aligned} \quad (6)$$

5 シミュレーションによる性能評価

5.1 シミュレーション対象例題

提案アルゴリズムの評価のために、シミュレーション用の例題を作成し、それらの例題に対してシミュレーションを行なった。シミュレーションは PentiumII300MHz 128MB 上で行なった。以下に作成した 9 つの例題をまとめる。

表 2: 9 種類の例題のサイズ

例題 No.	FPGA サイズ	ネット数	密度
1	4 × 4	80	6
2	7 × 7	200	7
3	8 × 8	250	7
4	9 × 9	300	8
5	11 × 11	450	8
6	12 × 12	500	9
7	13 × 13	550	9
8	14 × 14	700	9
9	16 × 16	850	10

上記の表において、FPGA サイズとは縦横のロジックブロック数、ネット数とは配線要求のあるネットの数、密度とは1チャンネルセグメントに接続される端子数の最大値である。

5.2 シミュレーション結果

表 2 の例題を用いて次の 3 つのアルゴリズムのシミュレーションを行い、提案する 3 段階アルゴリズムの有効性を評価する。

- (1) 提案する 3 段階アルゴリズム
- (2) 経路候補抽出アルゴリズムと Greedy 配線アルゴリズムのみを行なうアルゴリズム
- (3) 経路候補抽出アルゴリズムと NN アルゴリズムのみを行なうアルゴリズム

表 3 に表 2 の例題に対する上記の 3 つのアルゴリズムに対するシミュレーション結果をまとめる。表中、上記の 3 つのアルゴリズムをそれぞれ 3 段階、Greedy、NN と表記している。W は必要トラック数、時間は実行時間 (秒) である。

表 3 より、必要トラック数 W に関しては、3 段階アルゴリズムが Greedy アルゴリズム、NN アルゴリズムよりも優れていることが分かる。これにより、Greedy 配線アルゴリズム後の、NN 配線アルゴリズムが効果的に再配線を行っていると言える。また、計算時間に関しては、Greedy アルゴリズムが格段に早い。これは、アルゴリズムに繰り返し計算がないからである。また、3 段階アルゴリズムと NN アルゴリズムを比較した場合、回路規模が小さいと NN アルゴリズムの方が計算時間は短い。回路規模・ネット数の増大につれ、3 段階アルゴリズムの計算時間は NN アルゴリズム程は増加せず、より高速に配線できていることが分かる。

表 3: 表 2 の例題に対するシミュレーション結果

例題 No.	3 段階		Greedy		NN	
	W	時間	W	時間	W	時間
1	7	19.7	7	0.2	7	18.6
2	9	64.0	9	1.6	9	74.2
3	9	96.4	10	2.6	9	94.4
4	10	172.5	11	4.3	10	160.7
5	11	242.6	12	12.3	11	260.8
6	11	342.4	13	14.5	12	361.9
7	12	360.4	13	19.9	12	492.5
8	14	443.4	14	36.7	14	816.9
9	14	567.8	15	60.7	14	1839.0

これにより、Greedy 配線アルゴリズムによる初期配線及び配置結果の部分固定の効果がうかがえる。総合的に見て、計算時間は Greedy アルゴリズムには劣るが、必要トラック数の少ない解が得られる 3 段階アルゴリズムは非常に有効であると言える。

6 あとがき

本稿では、二次元配列型 FPGA の配線問題に対し 2 端子間ネットのみを対象として、ヒューリスティック解法とニューラルネットワーク解法を併用した 3 段階アルゴリズムの提案を行なった。また、性能評価として、9 種類の例題を作成し、シミュレーションを行なった。その結果、3 段階アルゴリズムからニューラルネットワークアルゴリズムを除いた場合や、3 段階アルゴリズムから Greedy アルゴリズムを除いた場合と比較し、3 段階アルゴリズムによって高精度の解が求まることを示した。

現在、多端子間ネットに関する配線アルゴリズムを考案し、その実装及び改良中である。本アルゴリズムと今回提案したアルゴリズムを組み合わせることで、一般的な配線問題を解くことが可能となる。そこで、これらを含めてベンチマーク問題を用いて評価を行う予定である。

参考文献

- [1] Y.-L. Wu and D. Chang, "On the NP-completeness of Regular 2D FPGA Routing Architectures and A Novel Solution," in Proc. ACM, vol. 11, pp.362-366, 1994.
- [2] Y.-W.Chang, S.Thakur, K.Zhu, and D.F.Wong, "A new global routing algorithm for FPGA's," in Proc. ACM, val.11, pp.356-361, 1994.
- [3] S.Brown, J.Rose, and Z.G.Vranesic, "A detailed router for field-programmable gate ar-

- rays," IEEE Trans. Computer-Aided Design, vol.11, pp.620-628, May. 1992.
- [4] M. J.Alexander and G. Robins, "New Performance - Driven FPGA Routing Algorithms," in Proc. 32th DAC., vol33, no.3, pp562-567, June, 1995.
 - [5] Y.-S.Lee and A.C.H.Wu, "A performance and routability-driven router for FPGA's considering path delays," IEEE Trans. Computer-Aided Design, vol.16, no.2, pp.179-185, Feb. 1997.
 - [6] Y.-L.Wu, S.Tsukiyama, and M.M.Sadowska, "Graph based analysis of 2-D FPGA routing," IEEE Trans. Computer-Aided Design, vol.15, no.5, pp.33-44, Jan. 1996.
 - [7] J.Rose and S.Brown, "The effect of switch box flexibility on routability of field programmable gate arrays," in Proc. CICC, 1990, pp.27.5.1-27.5.4.
 - [8] Y.-L.Wu and M.M.Sadowska, "Routing for array-type FPGA's," IEEE Trans. Computer-Aided Design, vol.16, no.5, pp.506-518, May. 1997.
 - [9] N.Funabiki, J.Kitamichi, and S.Nishikawa, "A digital neural network for multilayer channel routing with crosstalk minimization," IEICE Trans. Fundamentals, vol.E80-A, No.9, Sep. 1997.
 - [10] N.Funabiki, J.Kitamichi, and S.Nishikawa, "A massive digital neural network for total coloring problems," IEICE Trans. Fundamentals, vol.E80-A, No.9, Sep. 1997.