アルゴリズム 34－4
（1993．8．19）

配線問題へのＧＡの適用法とその並列処理

岡 圭司　佐野 雅彦　タノマル ジュリオ　高橋 義造
徳島大学工学部知能情報工学科

本論文では、遺伝アルゴリズム（ＧＡ）の進化の原理に基づいた自動配線アルゴリズムを提案する。ＧＡには個体間で高い並列性が内在しているので、これを配線問題に適用する。提案する手法は、一般的なＧＡと同様に解の集団や進化を模倣するオペレータを使用する。しかし、染色体表現が可変で、変形オペレータはビット列の操作ではない専用のものが必要となる。これらのオペレータと集団を定義し、この方法を用いたワークステーション上の実験による評価について述べる。

EVOLUTIONARY WIRE-ROUTING ALGORITHM AND ITS PARALLEL PROCESSING

Keiji Oka　Masahiko Sano　Julio Tanomaru　Yoshizo Takahashi

Department of Information Science and Intelligent Systems, Faculty of Engineering,
The University of Tokushima, 2-1 Minimi-Josanjima, Tokushima 770, Japan

This paper presents a new Automatic Wire-Routing(AWR) algorithm based on evolutionary principles. Although the populations approach and the use of stochastic operators for simulating evolution resemble conventional genetic algorithms operation, the AWR problem requires much more advanced representation and sophisticated operators whose action takes place at the wiring level, and not at the string level. Results showing the effectiveness of the proposed algorithm in a serial implementation are presented.

1．はじめに

これまでに報告された並列配線処理方法[1-4]では、経路探索並列化を行なうのに効率のよい各種処理方法が提案されているが、複雑な場合でも完全に交差がなくなるような一般的な方法はない。

各ネットを並列処理し交差解消する場合、次の2つの問題点がある。

1）最適に交差解消する手順は複雑で、初期状態によってかなり変わる。

2）繰り返し配線して交差が少なくなると、同時に処理するネット数が減少する。

他方Genetic Algorithm(GA)では、複雑な最適化問題を解くために、可能な解の中からオペレータを用いて新しい候補集団をつくり、これを組み合わせてより評価の高い集団を生成するようになっている[7]。

我々は、ＧＡでのcrossoverやmutationオペレータにならって、配線問題向きのオペレータを定義し、１つのネットに対して複数の個体を用意して、これらの間で生存競争をさせる方式を研究した。本稿ではこれらを用いた、上記の問題点を解決する方法を提案する。

2．並列配線問題

2．1．配線問題

配線問題では、領域上に複数ネットの経路を作り、これらの経路間が重ならないようにした上で、なるべく各経路長を短く、折れ曲がり数を少なくすることが基本的な目的となる。ここでは、配線領域が1層で各ネットが2端子間の配線である場合を考える。一層では空間的にネット間の交差が起こり易いが、結果を人間が判断し易い。また1層では配線できなかった問題でも多層にすると簡単に解ける場合が多い。ネット数3本の場合の配線問題例を図1に示す。

領域上の2端子間の経路探索法には、迷路法や線分探索法[5]などの方法が知られている。しかし、ネットの最適な配線順序を見つけること

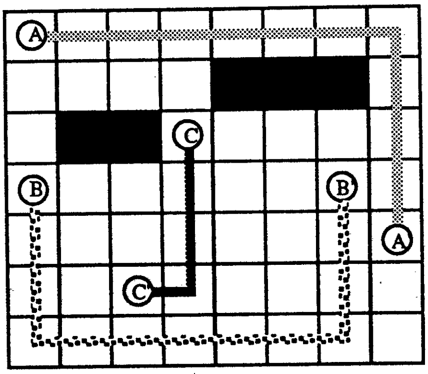

図1　ネット数3本の配線問題

は困難とされている。即ち、迷路法などで求めたその時点での各ネットの最短経路が、全体から見ると良い結果を得ることができるとは限らない。

配線順序の組み合わせ数と、各時点で交差数などの条件を満たす経路の数から探索空間ができているとする。この場合、配線するネット数が増えてゆくと、探索空間は増えてＮＰ困難になる。

そこで、これらの問題点を解決する上で並列処理を用いることで、最適解に近いものが得られる方法を考える。

2．2．配線問題の並列性

配線問題に内在する並列性には、ネットの並列性、ネット間の並列性と交差を求めるときの並列性がある。

ネットの並列性とは、1本のネットの経路探索を複数のプロセッサを用いて処理できることを意味する。この方法を用いた、並列配線処理方式やハードウェアルータは、幾つか発表されている [2,3]。

ネット間の並列性は、各ネットの経路探索を平行して処理することができることを指す。つまり領域上で相互関係しないネットは、独立に配線できるということである。しかし、それを予測するのには限界がある。ネット間の並列性の抽出に既略配線処理を用いて、予め同時処理可能なネットを検出しておく方法を使ったもの

には、PROTON[2]がある。また、予め相互関係を求めない方法では、プロセッサ競合方式[1] やMulti-PSI 上での並列オブジェクトに基づく方法[4] などがある。本提案では、ネット間の並列性を使う。

交差評価時の並列性は、各ネットを並列に求めて、交差があるかを調べなければならない時に各ネット毎に並列に交差を求めることができることを指す。交差の求め方を各ネット間の経路比較で行う場合は、ネット数m本とすると、最悪 m（m－1）／2回の比較を行わなければならないが、各比較は互いに独立で並列に処理できる。

3．遺伝アルゴリズム

遺伝アルゴリズム(genetic algorithm GA)は、1960年代、理論生物学的な適応理論の遺伝システムをコンピュータシミュレーションによって研究したことに端を発し、ミシガン大学のHolland らが体系的にまとめた探索手法である。従来の探索手法と比較して、複数の探索点が相互に協調または競合することで局所解を避けるという探索特徴をもつ[6-8]。

最近の研究において、GAや他の新しい探索手続きは古典的なものに比べて、より難しい探索問題に対して効果を発揮している。また、GAの限界についても指摘されており、従来の単純GAでは最適化しにくい問題の存在が知られている。

基本的なGAは、探索空間を符号化した多数のストリングによる集団を使い、交配および突然変異を起こしながら集団内の協調や競合を通してすべてのストリングの評価値を向上させることで探索を行う。GAは、あくまでも近似解法であり、大域的最適解を保証するものではない。また、単一集団の場合、遺伝子操作の繰り返しにより遺伝的多様性を減少し、次第に探索範囲を狭めてゆくという特徴を持つ。

これらのことから、遺伝アルゴリズムは評価値に基づいた探索法で、幾つかの個体間の生存競争であり並列探索としても有効といえる[9-11]。以下に遺伝アルゴリズムの概要を示す。

```
アルゴリズムの概要：
  初期集団生成、評価。
  while (満足な解がない)   {
      次の世代を作る{淘汰&複製、変形}。
      各個体を評価する。
  }
```

GAを計算機上で実行するためにはまず染色体となるものの符号化方法を決定する必要がある。配線問題にGAを適用する場合、より複雑な探索空間の表現と集団の構成、そしてそれに見合った高度なオペレータが必要とされる。

4．進化に基づく配線アルゴリズム（ＥＷＲＡ）

提案する進化に基づく配線アルゴリズム（evolutionary wire-routing algorithm EWRA) について述べる。

GAを基にして、集団、個体、各オペレータを定義する。まず、経路データを染色体として定義する。具体的には、経路のベンド点の並びを１つの染色体と考える。このとき、同一遺伝子座であるかの判定は、領域上の同一座標にあるかどうかによって行われる。領域上の経路部分を１それ以外を０とした全領域データを染色体とした場合、固定長の染色体となるが、領域上の１の部分がネットとして必要となる。このことから定義した染色体は、この部分だけを経路データで表して、データ量を減らしたものだと考えることができる。この経路データの染色体を１個体として考える。

そして、生存競争させるために１つのネットについて複数の経路データ（個体）を用意する。複数ネットの配線問題を考えているので、個体には、ネット番号と、個体番号の２つの属性がある。個体間はネット番号によって、同一ネットに属する（同ネット）のか異なるネットに属する（異ネット）のかが区別される。個体番号は、同ネット内で個体を識別するためのもので

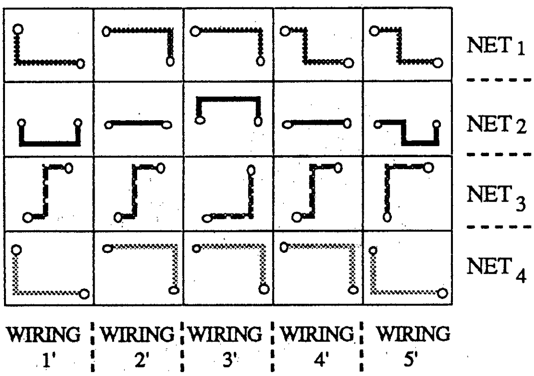

図2　集団の構成図

ある。

次に、集団について定義する。集団は、個体の集合である。１つのネットに対して、n個の個体を用意し、m個のネットについての合計n×m個の個体からなる集団を定義する。

ネット数4本、１ネットの個体数5個用いた場合の集団の構成を図2に示す。図の横列が、同ネットに属する個体群の集合を表し、それが4ネット分ある。

この集団に対して、ＧＡで一般的に使われるcrossover 等の変形オペレータを適用しても、各経路データの交差をなくすことは出来ない。また、交差をなくすのは異ネット間である。これらのことから、交差をなくす変形オペレータとオペレータ使用方法を用いることで、解を探索する。

また、個体の評価で交差を調べなければならないが、図2の縦列を交差をなくすべき評価グループとし、このグループ内での交差数を求める。図2からわかるように、グループ内には、各ネットの個体が1つずつ存在する。そして、評価グループ内の全個体の交差が0になる又は、総評価値が高いものが最適な解となる。このことから、この評価グループを１つの染色体（個体）と考えることもできる。

次に各オペレータについて定義する。

４．１．集団の初期化

幅広いランダム性のある探索のためには、集団の初期化の段階で、各ネット内で複数の異なる経路をつくることが必要となる。簡単な方法では、迷路法等で最短経路を求める時にバックトレース処理を工夫することで、同じ経路長でも異なる折れ曲がり数の経路を作ることができる。ここでは、２端子間＋αの矩形内で幾つかの点を確率的に選択し、それらを通る経路を作ることで異なる経路を複数もつ集団を用意する。

４．２．各個体の評価

各個体の評価には、経路間の交差数、経路長、折れ曲がり数を使う。交差数は先に述べた評価グループ内で求めたものである。各個体の評価関数を以下に示す。

評価関数

$$F_j(t)=\sum_{i=1}^{m} q_j^i(t)$$

$$q_j^i(t)=q[net_j^i(t)] \quad \cdots (1)$$

$$=c_j-\left[\alpha\, n_c\left(net_j^i(t)\right)+\beta\, d\left(net_j^i(t)\right)+\gamma n_b\left(net_j^i(t)\right)\right]>0$$

$\alpha,\beta,\gamma>0$.：重み

c_j：評価値が負にならない為の定数

n_c(route)：交差コスト

＝Σ（重なり量 / Const + 1）.

d (route)：経路長.

n_b(route)：折れ曲がり数.

経路長、ベンド数は、各個体毎に並列に求めることが出来る。また交差数は、各グループ間で互いに並列に求めることが出来る。

４．３．各ネットの複製

ネット複製処理は、集団配列の横列である同一ネットの個体間で行われる。個体は式（１）から評価値が与えられ、それぞれのネット毎に部分集団として複製され、淘汰される。淘汰方法は一般的なＧＡと同じ方法が使える。この淘汰処理は、ネット数と同じ回数行われるが、ネット間で並列に淘汰が行える。

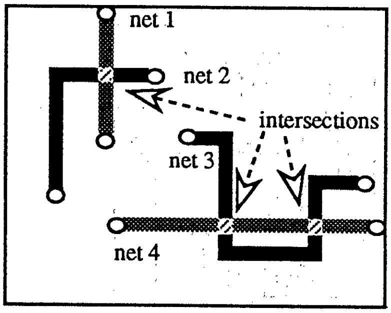

図３　交差数３の４本ネットの例

ネット毎に各個体は、以下の複製率

$$r_j^i(t) = q_k^i(t)\left[\sum_{j=1}^{m} q_j^i(t)\right]^{-1} \qquad (2)$$

が計算され、相対的に大きな値をもつ個体が多く複製される。

４．４．経路変形オペレータ

各個体の複製、淘汰が終った後、その集団の縦列毎に、変形オペレータを適用する。各個体間に交差がある場合には、それを解消する変形オペレータが必要になる。以下では図３に示す、４本ネットの配線処理途中で３点の交差がある場合を想定し、各オペレータについて説明し、使用後に交差が解消されてゆくのを示す。

４．４．１．交換オペレータ(Exchange operator)

交換オペレータではネット間で交差する点が複数ある時に交差解消する。図３のネット１と２、ネット３と４に交換オペレータを使ったときの結果を図４に示す。この方法では交点間の経路データを交換することで交差解消し、タッチ状態に変える。ただし交点が奇数個ある場合は、１交点は交差解消されない。図４の例では、ネット３、４間の交差２点がタッチに変っており、ネット１、２間は変化しない。

４．４．２．タッチ回避オペレータ (Touch clearing operator)

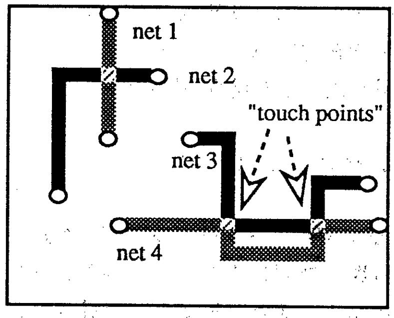

図４　交換オペレータ使用後

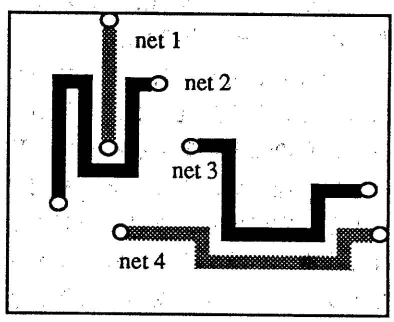

図５　タッチ解消オペレータ使用後

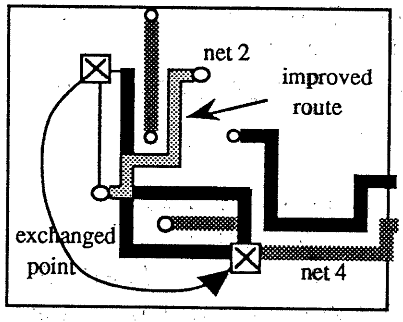

図６　ベンド点交換オペレータ使用後

交換オペレータでは、交差していた点がタッチに変わったことになる。また交差する点が１つの時や奇数個ある場合には、上記のオペレータだけでは、交差を完全に解消できない。これらの重なりを解消するのが、タッチ回避オペレータである。この方法はタッチした部分を平行移動させたり、どちらかのネットをを障害領域としてその周りに線分を発生させ経路を探索する。このオペレータを使った結果を図５に示す。全ての重なりが解消されている。ネット１、２間では、ネット２の方がネット１の周りを通るよ

う各経路を作っている。実際にはネット２が回避できる幾何学的形態は２パターンあるが、ここでは経路長が短い方の経路が作られる。

４．４．３．ベンド点交換オペレータ

上記のタッチ解消オペレータは、相手の経路を回り込み経路が複雑になり、不必要な折れ曲がり経路が出来る。また交差がない経路間は、全く変化させないので、探索範囲が限定されすぎる。これらを補うために、経路上の折れ曲がり点を交換するオペレータを定義する。

このオペレータでは、各経路毎に折れ曲がり点を１つ選択し、それらを交換する。各経路は選択された点が、別の位置に置き変るので、配線条件を満たさない経路が出来る。それを配線条件を満たすように整形することで、上記のオペレータでは出来ない経路を作る。図６にネット２の１点をネット４の１点と交換した例を示す。交換後に経路を縦横配線に整形した場合、黒い部分のループを含む不必要な経路が出来る。しかしその中の最短経路を見つけることで、結果として経路が改良される。

実際はネット４の経路も変化するのだが、この図では示していない。

４．４．４．変形オペレータの適用法と相手個体の選択法

以上のように変形オペレータ使用時に各個体は、同一評価グループの個体群から１つを相手個体として選択する。この選択条件には、まず交差があるかどうかが上げられ、交差があるネットを相手とする。交差するネットが複数ある場合には、それらの中から確率を用いて選択し、交差がなければ評価グループ内の全個体中から適当に選ぶ。この相手選択は、各個体毎に独立に行われ、１つの個体が複数の個体から相手として選択されることがある。

用いる変形オペレータは複数あり、それをどの様に使うかには、幾つかの方法があるが、今回は変形処理時に各オペレータの１つを確率的に１つ選択し使用する。

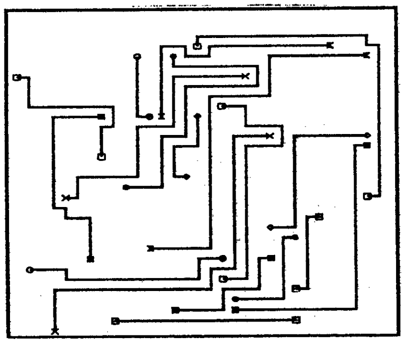

図７　第５１世代目のベスト

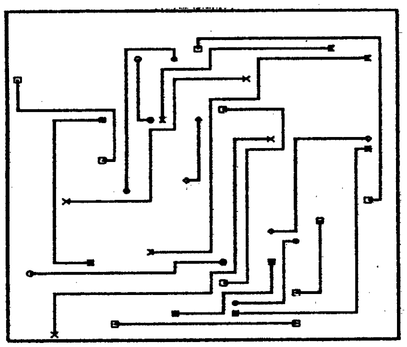

図８　第６１７世代目のベスト

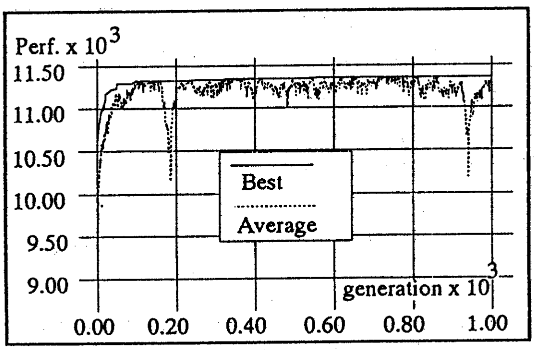

図９　評価値の変化

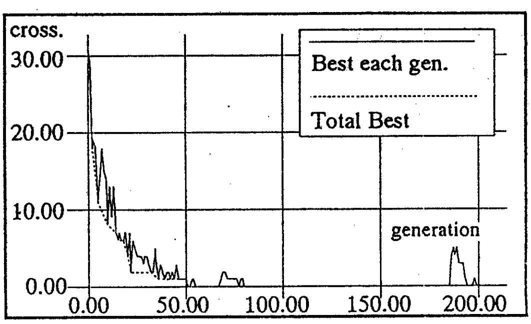

図１０　交差コストの変化

5．逐次計算機上の評価

ＥＷＲＡをワークステーション上に実装して行なった簡単な実験の結果を示す。

配線領域が１層の３２×３２グリッドで、ネット数ｍが１８本の場合の実験結果の１例を図７－１０に示す。初期データとして与えたネットの端子は、乱数を用いて作られたものである。

１つのネットに用意した個体数ｎは１０で、各パラメータは、$\alpha=10$、$\beta=1$、$\gamma=2$である。

第５１世代目のベスト配線結果を図７に示す。この世代ではじめて交差数がない配線結果が得られた。しかし、交差は無くなってはいるが、多くの必要のない折れ曲がり経路が出来ている。その後の第６１７世代目のベストを図８に示す。この頃になると、１００％の配線率で、不必要な折れ曲がりがほとんどない最適な配線結果がえられた。

各世代毎の平均値とベスト値についての評価値の変化を図９に、ベスト値についての交差コストの変化を図１０に示す。交差コストは５１世代目で０になっている。

この結果は集団の数が小さく、領域の大きさも小さい場合であり、実際の配線ではもっと多くのネット数を配線するので、全て配線できるとは限らない。

6．並列計算機への実装

ＥＷＲＡを並列計算機に実装する上で配線問題と定義したオペレータの並列性を以下に示す。

○配線問題に内在する並列性
- ネットの並列性（使わない）
- ネット間の並列性
- 交差検証時の並列性

○オペレータの並列性
- 初期化時の経路探索は、個体毎に並列処理できる。
- 評価時の各評価グループ毎の並列処理とグループ内の各個体の交差数評価、経路長、折れ曲がり数を並列に求めることが出来る。

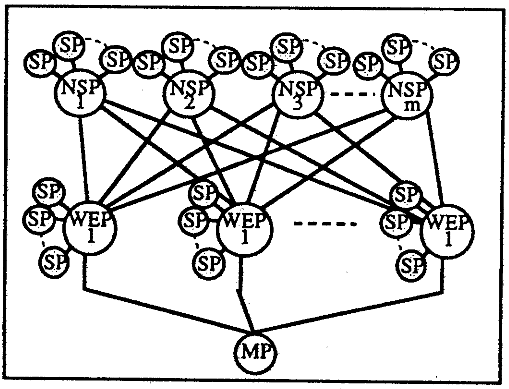

図１１　Parallel architecture idealized for the proposed EWRA, consisting of 1MP(master processor), n WEPs (wiring evaluation processors), m NSPs(net selecton processors) and a number of SPs(slave processors).

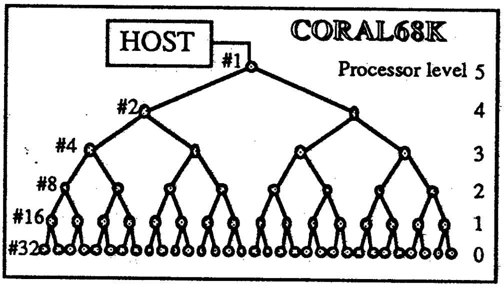

図１２　並列計算機CORAL68Kの構成

- 変形時（評価時と同じ）に各評価グループで並列に処理できる。グループ内で並列処理できる。
- 淘汰時の各ネット毎に並列に処理できる。

評価と変形処理は同じ並列性を持ち、評価グループ間で並列に処理できる。淘汰は全く違う並列性をもち、ネット間で並列に処理できる。

１台のマスタープロセッサと、ｎ台の経路評価プロセッサ、ｍ台のネット淘汰&変形プロセッサ、そして幾つかのスレーブプロセッサからなる並列処理モデルを図１１に示す。

○Coral68k：

CORAL68Kは、MIMD型で63台のプロセッサを使った２進木構造で構成されている並列計算機

である。各プロセッサは、それぞれCPUとしてモトローラのMC68000と512kbyteのメモリをもつ。共有メモリは持たない。各プロセッサは＃1から＃63の番号が割り振られている。＃1のルートプロセッサが外部のホストコンピュータとつながっている。図12にその構成を示す。

○CORALへの実装

このモデルをCORALへ実装する場合には、使用できるプロセッサ台数が少ない。

淘汰、評価、変形の処理をするプロセッサを別々にするのではなく、各処理を63台（n台）のプロセッサを使うように割り振る。現在CORAL68Kに実装中である。

7．まとめ

配線問題の並列度を上げるために、1ネットについて複数の個体を用意し生存競争をさせ、交差解消用の変形オペレータを使う方法を提案した。1ネットについて同時に複数の探索をすることが多く、並列度が上がると思われる。

提案した方法の利点は、オペレータ等を考えることでより効率のよい探索が出来るのと、そのルールとして他の配線問題で使われている方法を実装しやすいことである。そして、遺伝アルゴリズムが持つ探索特性と並列性が使える。

実験結果は、20本程度のネット数についての簡単なものであるが、実際にはネット数は数千本と多く、配線領域も大きく多層である。この様な問題も実時間で解けるようになるには、多くのプロセッサ台数と、改良されたオペレータが必要となる。

ここで述べた手法は、1集団を使った場合であり、大きな問題を解くためには、部分集団を使った方法が必要になるだろう。また使用オペレータについても改良が必要で、回り込みを必要とする問題[3]を解くことは出来ない。その他に、一度交差数が無くなったネットの個体群は収束し易く、そのネットが新たに交差する経路を作ってもそれが生き残る確率は低くなる。このために交差が連鎖的に起こるような場合には、途中で収束する可能性が高い。

この様に多くの改良を必要とするが、配線問題に遺伝アルゴリズムの特性を用いることで、その並列性と探索可能性が高いことを示すことが出来たと思う。

参考文献

[1]佐野雅彦: 分散メモリ型と共有メモリ型マルチプロセッサによる並列配線処理の性能比較,並列処理シンポジウムJSPP'91, pp.197-204 (1991).

[2]山内宗:MIMD型並列計算機のLSIルータPROTON,並列処理シンポジウムJSPP'92, pp.445-452(1992).

[3]河村 薫:超並列マシンMAPLE-RP,並列処理シンポジウムJSPP'91, pp.373-379 (1991).

[4]伊達 博:並列オブジェクトモデルに基づくLSI配線プログラム,並列処理シンポジウムJSPP'91, pp.381-388 (1991).

[5]Ohtsuki, T. : maze-running and line-search algorithms, in T. Ohtsuki(Ed.), *Layout Design and Verification*, Elsevier, pp.99-131 (1986).

[6]竹内勝:遺伝的アルゴリズムと機械学習,コンピュータソフトウェア.vol.8,No.5,pp.16-24 (1991).

[7]Goldberg, D.E.: *Genetic Algorithms in search Optimization & Machine Learning* ,Addison-Wesley (1989).

[8]Davis, L.: *Handbook of Genetic Algorithms*,Nostrand (1991).

[9]Petty,C.B.,M.R.Leuze,and J.J.Grefenstte."A Parallel Genetic Algorithm",in J.J.Grenfenstette(Ed.), Proc.ICGA'87,Lawrence Erlbaum Associates, Hillsdale,NJ,pp.155-161(1987).

[10]Reiko Tanese."Distributed Genetic Algorithms", Proc.3rd ICGA'89,pp.434-439(1989).

[11]H. Muhlenbein."Parallel Genetic Algorithms, Population Genetics and Combination Optimization", Proc. ICGA'89,pp.416-421(1989).