

最適問題への遺伝的アルゴリズムの適用

小野 俊彦 (情報工学科)
渡辺 玄 (大学院情報工学専攻)
池田 武史 (大学院情報工学専攻)
土性 雅史 (大学院情報工学専攻)

Application of Genetic Algorithms to Optimal Problems

Toshihiko ONO (Department of Communication and Computer Engineering)
Gen WATANABE (Graduate School of Engineering, Communication and
Computer Engineering)
Takeshi IKEDA (Graduate School of Engineering, Communication and
Computer Engineering)
Masafumi DOSYOU (Graduate School of Engineering, Communication and
Computer Engineering)

Three examples of genetic algorithms (GAs) applied to optimal problems are explained. The first example is the optimal allocation of various patterns on a sheet to achieve the minimum length of the sheet. By applying an order type GAs in cooperation with layout determining algorithms, the required calculation time is reduced considerably. The second one is also the reduction of calculation time by the division and integration of a search space. The method is applied to a salesman travelling problem. After the region of cities is divided into four regions, the GAs find the optimal route for each region, and the final optimal route is gained by applying also the GAs to the route made by integration of the above routes. The last is to determine the construction of neural networks by GAs. At neural networks of multi-layer type, the numbers of units and connections are determined by the GAs to make the required numbers of units and connections minimum.

Key words: *genetic algorithms, optimizing problem, optimal cutting control, travelling salesman problem, neural networks*

1. ま え が き

前報¹⁰⁾では遺伝的アルゴリズム(以下GA)の最適問題への応用として二つの分野即ち棒材の最適切断への応用と配送センター選定問題への適用について報告した。本報告ではその後の研究結果として以下の三つの分野への応用について述べる。

第1のテーマは二次元材の最適配置へのGAの適用である。二次元材であるシートより各種多角形状の部品を切り出す際に必要とするシート長を最小にする部品の配置をGAで求める問題である。これは前報

の棒材の最適切断即ち一次元の最適配置問題を二次元に拡大したものである。順序型GAに新たに研究した配置決定アルゴリズムを組み合わせることで、短時間に解を得ることができた。

第2のテーマは大規模GAの分割・統合による高速化の実現である。セールスマン巡回問題(TSP)を例に領域の分割・統合の効果を述べている。まず都市群を分割したものにGAを適用し、そこで得た各巡回ルートを一つに統合する。その際にも再びGAを適用する。この方法により収束時間を大幅に短縮できた。

第3のテーマはニューラルネットワーク (NN) の構造決定に GA を適用した研究である。多層型 NN において、目的とするパターンを認識し、かつノード数およびコネクション数が最小な NN を GA にて求める。認識するパターンとして排他的論理和を選び、目的とする NN が GA により実現できた。

2. 最適問題への GA の適用

GA 一般については前報(10) に記したもので省略し、ここでは最適問題への適用に際して考慮した事項について述べる。

GA 自体は比較的簡単なアルゴリズムであり、基本操作として選択、交叉、突然変異の3つのオペレータを持ち、各々に対していくつかの方式があり、それらを組み合わせて問題の解決を行なっている。GA には大別して数値型と順序型がある。前者は問題を適当な数(整数, 実数, あるいは一般的にはビットストリング)を表わす遺伝子により表現したものであり、関数最適化をはじめ多くの最適問題に用いられている。一方、後者の順序型では遺伝子は順序数の列で表わされ、その順序により問題を表現するものであり、TSP やスケジュール問題で用いられている。この二つの型は交叉の方法に大きな差があり、特に後者の方法に適した交叉方式が各種研究されている(5)。それぞれの方式ごとに多くのプログラムパッケージが作られており利用できる。まずいずれの型で問題を表わすかを定めることになる。

GA のプログラムの作成ではつぎの点が大事になる。

1. 問題をどのように遺伝子で表わすか。
2. 遺伝子からどのように適合値を計算するか。
3. 他のシステムあるいはアルゴリズムとの組み合わせ。

これらの点を考えながら今回の研究で考慮した点を述べる。

1) GA の型の変更による効率化: 二次元材の最適切断は二次元シート上に所与数の各種形状部品を、必要なシート長が最小になるように配置する問題である。この解決法としては各部品の二次元座標を遺伝子として表現する方法が一般的であるが、この方法を用いると後述のように膨大な探索空間となる。これを部品配置の順序問題とすればそれは大幅に減少する。新しく部品配置プログラムを導入することによりこれが実現できた。

2) 探索領域の分割と統合: TSP に代表される順

序問題ではその要素の数の増加に伴い探索空間が増大する。従って大規模問題ではできるだけ領域を狭める方が有利である。しかしこれが可能であるためには局所仮説(後述)が成立する必要がある。幸い TSP においてはこれが成立すると思われるので、領域を分割し、各々で最適ルートを求め、それを統合する方策を採用することにより探索時間の大幅な短縮が実現できた。もちろん領域の境界付近では問題があるので、実際は重複部分を設け、この部分の連結に多様性を持たせて統合化したものに再び GA で最適化する方法を用いている。

3) 問題に適した遺伝子表現法: GA においては問題をどのように遺伝子により表現するかが重要となる。たとえば NN の構造決定を GA で行う場合に、ネットワークの遺伝子表現として2つの方法が考えられる。一つが NN の各要素間の接続状況により表わす方法(コネクション表現)であり、いま一つが入口要素から出口要素までの連結状況により表わす方法(パス表現)である。今回はこの両者を試み性能の比較を行なった。

3. 二次元材の最適切断

3.1 問題設定

ここで取り扱う最適配置問題は図1に示すような一定幅のシート材からグレーの部分で表す各種多角形(凹型多角形も含む)の部品を切り出す際に、次の二点を満足する部品の配置を決定することである。

1. 部品を重なりなく配置する。
2. シート材の必要長を最小にする。

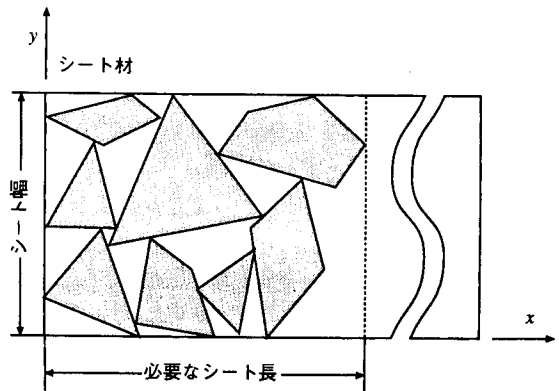


図1 シート材の切断例

Fig. 1 Example of cutting pattern

配置が決定した際のシート材の必要長は図1に示すように、最大の x 座標値をもつ頂点より求めることができる。

この種の問題は洋裁における型紙の配置やガラス、鉄板、木材板を切断する際に生じる問題である。その解法として動的計画法やツリーサーチプログラムによる研究2)が行われている。ここではGAの特徴を生かした本問題に対する適用方法を提案する。

なお、今回は特に洋裁などを対象とし、織目や模様などの点から、配置に際して各部品は回転させないことにした。

3.2 GAの適用における問題点

遺伝子の表現方法はGAの性能に大きな影響を与えるため、表現方法を中心にGAの適用を検討する。

遺伝子の部品配置の直接的な表現方法としてシート材上における各部品の位置を表す方法が挙げられる。つまり各部品の x, y 座標の位置(2進表示)を順に並べそれを遺伝子とする方法である。しかしこの表現方法ではシート材が大きくなるに従い、遺伝子長が長くなりその探索空間は広がる。遺伝子長は部品の数を N 個、シート材の幅を W 、長さを L とすると

$$N(\log_2 W + \log_2 L) = N \log_2(WL) \text{ となり、}$$

$2^{N \log_2(WL)} = W^N L^N$ 個もの配置の組み合わせより最適解を探索しなければならない。またこの表現方法では部品同士が互いに重なり合い切断が不可能な配置が生ずるため、これを致死遺伝子として取り扱うなどの対策が必要である。いずれにしてもこの表現方法での最適解の探索は困難である。

そこで二次元配置問題を一次元問題である順序問題として解くことにより以上の問題点を解決を図った。それを実現するために、遺伝子を各部品の配置を決定する順序として表現し、遺伝子の表す配置決定順序に従い隙間なくシート材上に各部品を並べる配置決定アルゴリズム(Layout Determining Algorithms: LDA)をGAに導入した。遺伝子は配置決定順序のみを表せばよく、遺伝子長は部品数 N と等しくなり、シート材の大きさに関係しない。探索空間は N 個の順序ゆえ $N!$ となり、前者の表現方法に比べ格段に縮小され、かつ部品同士の重なりの問題も解決した。

3.3 システムの構成

図2の系統図に示すように、GAでは遺伝的操作を終え、新たに生成した遺伝子の表す配置決定順序のデータをLDAに送り、LDAにては各部品を隙間なく配置して得た、シート材の必要最小長を目的関数値とし

てGAに返し、適合値を決定する。以下、遺伝子の構成、遺伝的操作(選択、交叉、突然変異)の順に説明する。

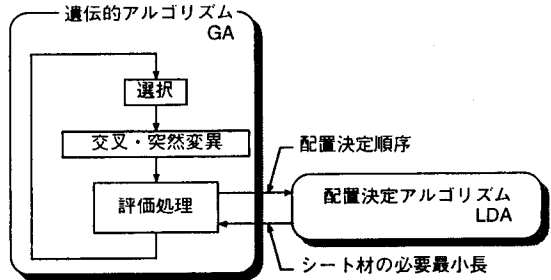


図2 システム
Fig. 2 Schematic diagram

遺伝子の構成

順序問題に対する遺伝子の表現方法としては様々な方法がある5)が、本研究では順序を直接遺伝子として表現するパス表現法を用いた。たとえば順序2→3→5→1→4を遺伝子では23514と表す方法である。

選択操作

適合値により順序づけし、順序に比例した確率にて選択を行うランク方式5)を用いた。

交叉操作

パス表現に対する交叉法は数多く研究されている5)が、本研究では代表的な方法である部分写像交叉(PMX: Partially-mapped crossover)、順序交叉(OX: Order crossover)、循環交叉(CX: Cyclic crossover)の三種についてシミュレーションを行ったが、性能に大きな差は見られなかった。

突然変異操作

ランダムに選んだ2つの遺伝子要素を交換する方式を採用している。図3に示す例では2番目の要素と4番目の要素が交換されている。

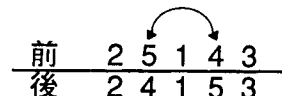


図3 突然変異の例
Fig. 3 Example of mutation

なおGAの処理部にはGENITOR5)を利用した。

GENITOR には遺伝的操作として基本的な選択と交叉が用意されており、これに上で述べた突然変異操作および LDA など必要なプログラムを付加しシステムを完成させた。

3.4 配置決定アルゴリズム (LDA)

LDA は各部品を遺伝子の示す配置決定順序に従い隙間なくシート材上に配置を行うアルゴリズムである。

GA より渡される配置決定順序情報に従い部品群より順次一つづつ取り出し、図 4 のように部品の配置可能な位置を探索する。グレーの部分はすでに配置済みの部品を、白い部分は配置可能な位置を探索中の部品 i をそれぞれ表している。なお部品 i のシート材上の位置は部品の外接長方形の左下頂点 (基準点 O_i : (X_i, Y_i)) の位置により表わす。

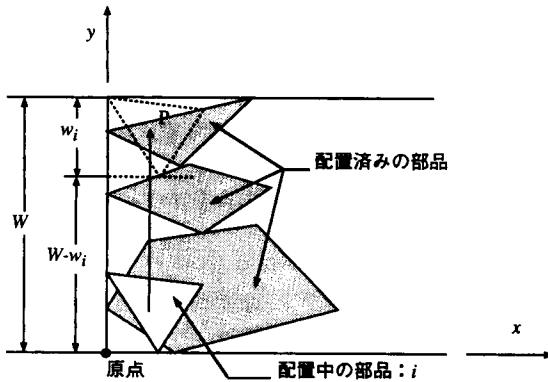


図 4 各部品の位置探索方法
Fig. 4 Searching position to put a part

図 5 に示すフローチャートをもとに、ある一つの部品 i について、探索方法を説明する。始めに部品 i の基準点 O_i をシート材の原点に重ね、まず y 軸方向に、続いて x 軸方向にずらすという操作を繰り返しながら配置可能な位置を探索する。この際に計算時間の短縮を図るため、次の点について探索の合理化を行った。

1. 部品同士の重なり判定
2. 重なりがあった場合の移動方法

最初にシート材の幅方向すなわち上端に関するチェックを行う。部品 i の基準点 O_i の y 成分 Y_i に部品の最大幅 w_i を加えたものがシート材の上端すなわち y 軸方向の最大値 W を越えると上端より部品 i がはみ出すことになるので、下端 $y = 0$ に戻し x 軸方向に 1 ドット進める。図 4 での P は上端に達した状態を示す。

上記のチェックに合格すると、つぎはすでに配置済みの部品と重なっていないかのチェックを行う。この重なりチェックには計算時間を短縮するため、頂点チェックおよび境界チェックの二種を段階的に適用する。これら両チェックに合格すると部品 i をシート材上に配置する。

一方頂点チェックまたは境界チェックに失敗すると直ちに重なりがあると判定し、後述の方法にて y 軸方向の移動量を計算し、その分移動させる。

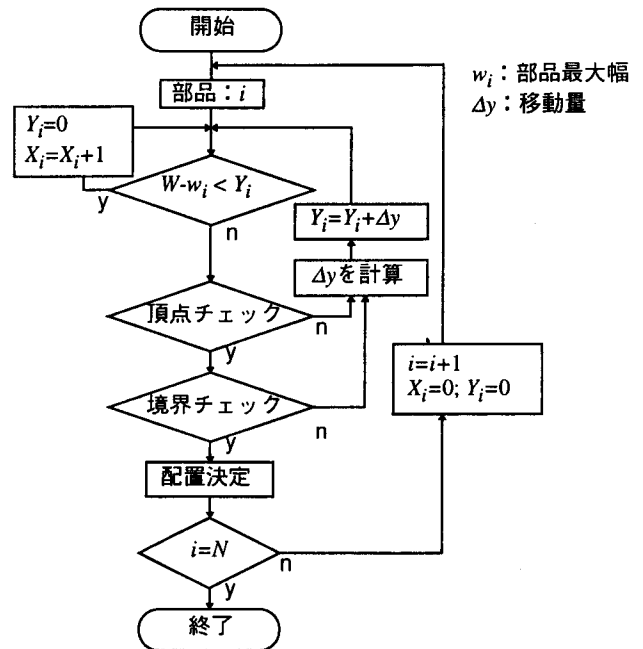


図 5 LDA のフローチャート
Fig. 5 Flow chart of LDA

この一連の操作により、GA より与えられた配置順序に従ってすべての部品を配置すると、隙間や重なりなく部品を配置でき、この配置順序でのシート材の必要最小長が求まる。次に二種の重なりチェックと移動量の計算方法について説明する。

頂点チェック

図 6 に示すように適正位置を探索中の部品 A が配置済みの部品 B と重ならないためには、少なくとも部品 A のすべての頂点が部品 B の内部にないことが条件となる。従って部品 A のすべての頂点でこのチェックを行ない、すべてが合格すれば頂点チェックは合格となる。

境界チェック

図7に示すように部品Aの頂点が配置済みの部品Bの外にあり、各頂点を結ぶ境界線の部分が部品Bの内部にある場合には、頂点チェックでは重なりを検出することができないので、頂点チェックに合格すると、つぎに境界チェックを行う。境界チェックでは境界線上のすべての点に対して、頂点チェックと同様の方法にて行う。

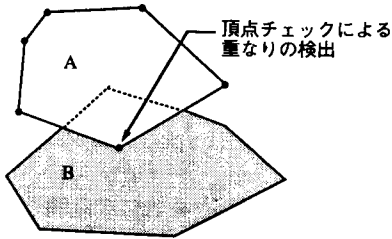


図6 配置状況-1
Fig. 6 Relation between parts-1

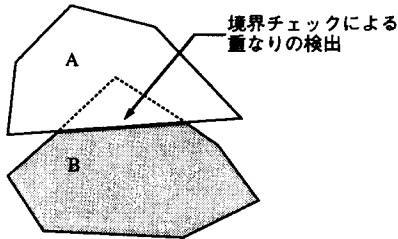


図7 配置状況-2
Fig. 7 Relation between parts-2

以上部品同士の重なり判定には二つのチェックを用いる。切断部品の形状によって差はあるが、一般に境界線上の点の数は頂点の数の数倍から数十倍になるため、はじめに頂点チェックを行うことにより、重なりチェックのための計算時間を大幅に短縮できる。

移動量の計算

配置中の切断部品の頂点、境界線上の点のいずれかが重なっている場合には、その点よりy軸正方向に配置済みの部品の外部にできるまで垂直に走査することにより、部品の移動量を決定する。

3.5 シミュレーションの結果と検討

本方式の特性を検証するため、つぎの二つのシミュレーションを行った。

1. あらかじめ最適解が判明している多角形部品の最適配置 (シミュレーション-1)

2. 自由多角形部品の最適配置

(シミュレーション-2)

シミュレーション-1

最適配置の判明している14個の多角形部品についてシミュレーションを行った。なお集団の個体数は50個体とし交叉方法にはCX法を採用した。シミュレーションの結果を図8に示す。これら14個の多角形部品を最適に配置したとき、シート材の必要最小長は110ドット (ドットは1単位の長さを表す) であることが分かっており、この最適値に達したのは147世代であった。0世代ではランダムに個体を生成するため部品の配置に隙間が多い、しかし世代が進むに従いその隙間は減少し最適解に到達している。なお、ここでは1回の計算を1世代としているので、一般に用いられている世代数の表示では147/50すなわち3世代となる。



図8 シミュレーション-1の結果
Fig. 8 Result of simulation-1

シミュレーション-2

36個の自由多角形の最適配置を個体数を50個体、交叉法をCX法としシミュレーションを行った。図9に示すように497世代 (497/50=10世代) でシート材の必要最小長は81ドットとなった。シミュレーション-1と同じく初期世代では比較的隙間が多いが、最終世代では高い密度で配置されている。0世代では配置決定順序を乱数で与えているにもかかわらず、かなり良

い配置が得られているのは LDA の働きによるものである。これは LDA が全部品についてシート材全面の探索を行い、部品の配置できる隙間を発見次第そこに配置するためであり、このことは最適解を得る上にも効果的に働いている。

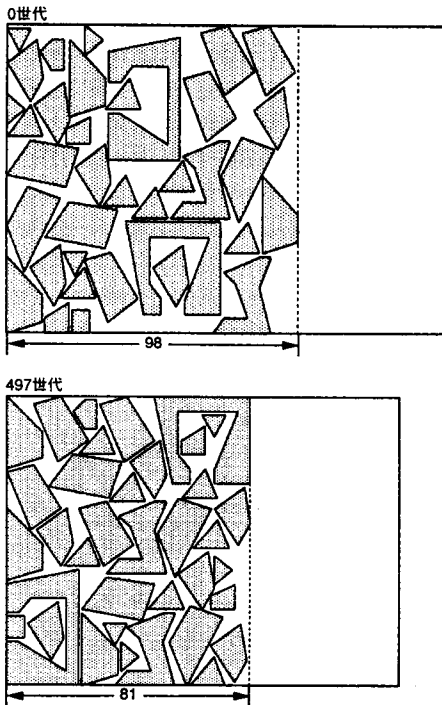


図9 シミュレーション-2の結果
Fig. 9 Result of simulation- 2

考 察

あらかじめ最適解の判明している部品を本システムを用いて最適配置を行った結果、最適解を得ることが確認でき、また自由多角形についても同時に密な配置を得ることができた。またこの最適解を短時間で得ることができた。これは遺伝子表現を配置決定順序にして次元順序問題として扱ったこと、LDA の導入により効果的な配置決定を可能としたことなどが貢献したものと考えられる。

これまでは多角形部品の最適配置について述べてきたが、この方式は容易に一般の曲線部品配置に拡張できる。図10に示すように曲線図形に内接する仮想的な多角形を考え、頂点チェックはこの多角形の頂点について行い、境界チェックは多角形の辺の代わりに曲線図形の境界線について行うことにより LDA をそのま

ま適用できる。したがって、本論文に述べた方式はこのような多少の変更により自由図形に対しても使用できる。

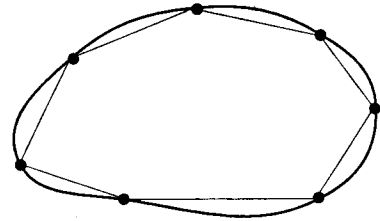


図10 曲線図形とその内接多角形
Fig. 10 Curve pattern and in-polygon

4. 大規模 GA の分割・統合による高速化

4.1 問題設定

本研究はセールスマン巡回問題 (TSP) を例とし大規模な順序問題への GA の適用に際し、領域の分割とその統合を行うことによって最適解を得るまでの時間を短縮しようとするものである。TSP とはある規模で配置された都市を最短経路で巡回するというものであり、GA の適用に関して最もよく研究されてきたものの一つである。

4.2 高速化の手法

TSP において都市の数を n とするとその探索空間は $\frac{1}{2}(n-1)!$ となり、 n の増加とともに急速に拡大していく。従って解を得るまでの時間も加速的に増大する。もし、都市数を4分の1の領域4つに分割できれば各探索空間は $\frac{1}{2}(\frac{1}{4}n-1)!$ となり大幅に減少し、4領域合計でも計算時間は大きく減少する。しかし、

このような分割により最適解が得られるためには「局所領域で得られた最適解が全体での最適解につながる」という局所仮説が成立する必要がある。この仮説が成立するならば、各領域で得られた最適経路を統合することにより、全領域での最適経路を得ることができる。

TSP においては分割領域の境界線付近を除いて、この仮説が成立するものと考えられる。何故なら TSP においては隣接都市が互いに距離的に大きく離れているとは考え難く、分割領域の境界線からはなれている都市群では、この仮説が成立すると考えられる。しかし境界線付近の都市群では隣接都市が異なった分割領域にある可能性があり、単純に各領域を統合する

だけでは最適経路は得られない。この対策として本研究では領域の分割に際して境界線付近では重複領域を設け、この重複領域内の都市は隣接の両領域に重複させた。そしてこれら各隣接領域で得られた巡回経路の統合の際には、重複領域の都市および近傍都市との間の各種隣接状態を作り、これを基に重複都市を取り除き、経路を統合した。こうして得た巡回経路からなる集団を初期集団としてGAを適用することにより最終的な最適経路を求めた。

4.3 システム構成

以上述べた方式を実現する具体的な方法を述べる。

1) 全領域を図11で示すように、適当な重複領域を設けて4つの領域に分割する。重複領域の大きさは設計パラメーターである。

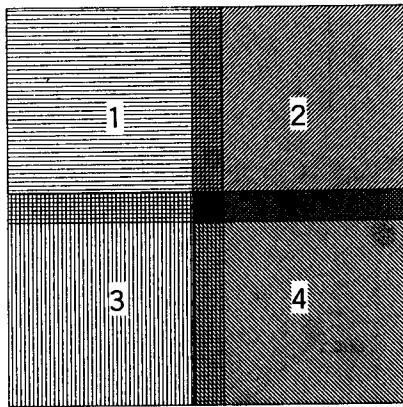


図11 領域分割法
Fig. 11 Division of region

2) 各領域に対して標準のTSP用GAを適用し、各領域それぞれでの最適経路を求める。各領域でのGAの適用は並列で処理する。

3) こうして得られた各領域での最適経路を領域統合アルゴリズムによって一つの巡回経路にする。統合はNo 1-2領域の統合、No 3-4領域の統合、No 12-34領域の統合の順で行う。統合の際に重複した都市を取り除くとともに、重複する領域とその近傍での都市群との関係を考えつつ、かつ統合に多様性を持たせるために、乱数によって各種の統合経路を作成する。

4) こうして得た各種統合経路をもつ個体の集団を初期集団としてGAを適用する。初期集団は分割領域でのGAにより最適化されており、全領域GAでも早い段階で最適解に到達することが期待される。以上で述べた構成を図12に示す。

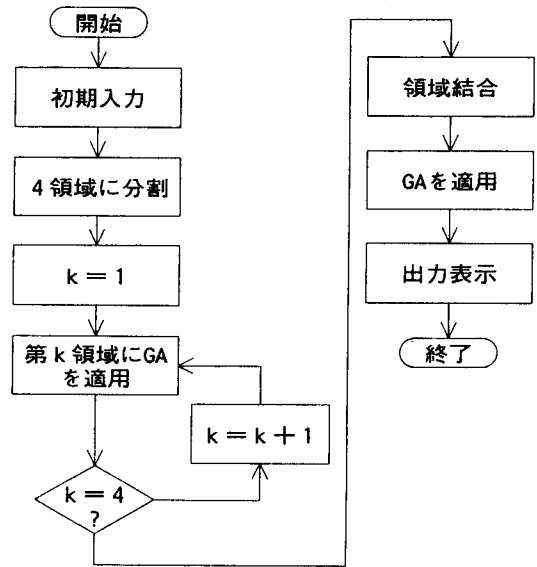


図12 分割・統合GAのフローチャート
Fig. 12 Flow chart of GA by division and integration

4.4 統合化アルゴリズム

統合化アルゴリズムでは重複した都市を取り除くとともに経路に多様性を持たせることを目的とする。統合するにあたっては重複部の状況を考慮する必要がある。まず、図13のように重複する都市が1つの場合を考えてみると、そのときの接続パターンは図14のように6パターンが存在する。このとき多少距離が長くなるような接続パターンでも使用するようにした。つぎに連続して重複する都市が2つの場合を考えてみると、このときの接続パターンは24種考えられ、その一例を図15に示す。連続して重複する都市が3つ以上のときは上記の重複都市が1, 2のときの接続パターンを組み合わせて求めるようにした。

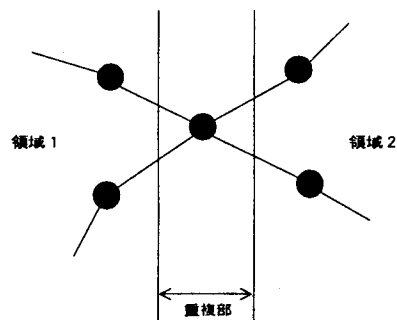


図13 重複部の状況
Fig. 13 Situation of overlap

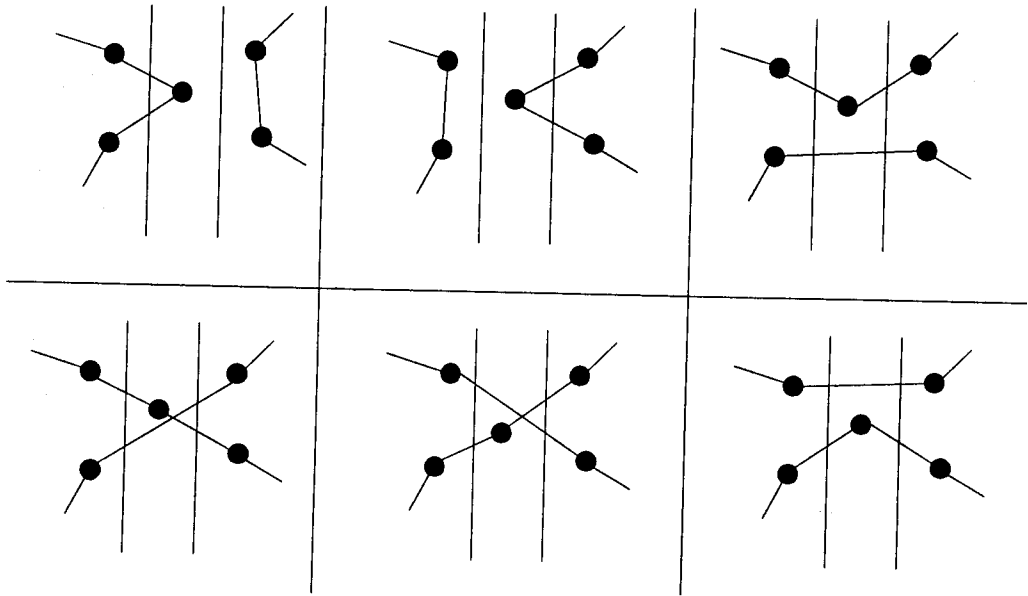


図14 接続パターン-1
Fig. 14 Connection pattern-1

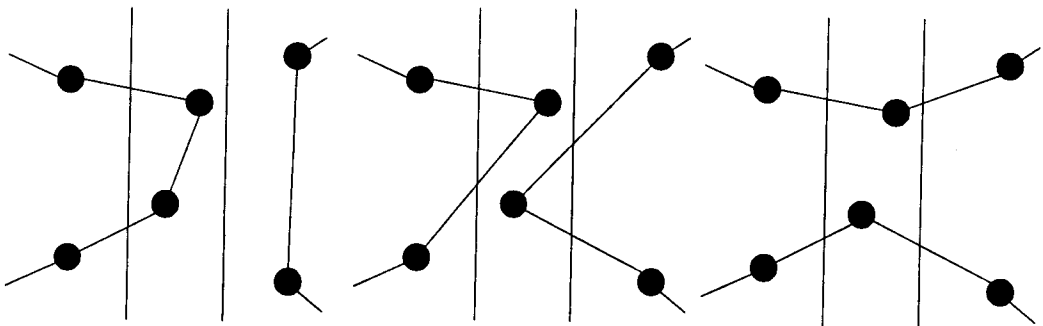
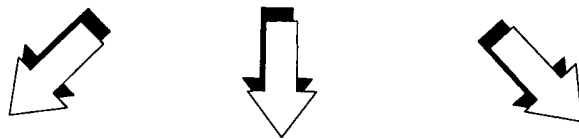
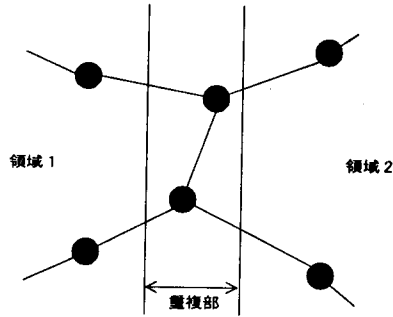


図15 接続パターン-2
Fig. 15 Connection pattern-2

統合に際しては双方向循環リストを用い、各経路をこのリストで表わす。今、統合する二つの経路を表わすリストをリスト1、リスト2とする。統合経路の作成は一つの都市から出発してこれらリストを進行しつつ行なう。その進行状況には以下の4つの場合がある。

- 1) リスト1を正順に進んでいる
- 2) リスト1を逆順に進んでいる
- 3) リスト2を正順に進んでいる
- 4) リスト2を逆順に進んでいる

今、1)の状況にあるとすると、重複都市に到達するまではそのままリスト1を進行する。重複した都市に到達すると、つぎの進行は、1), 3), 4)の3つ場合(そのまま進むか、もう片方のリストに移り正順か逆順に進む)から1つをランダムに選び、そちらに

進行状況を移す。また使用した都市はその都度各リストより取り除く。このような処理を、2つのリストのいずれもが空リストになるまで、繰り返すことにより統合経路が完成する。

図16の例を用いて具体的に説明する。まず、リスト1で都市1を出発点として1)の状況すなわち正順に進み、重複している都市3に到達したとする。つぎは1), 3), 4)の中からランダムに一つを選ぶので、4)が選ばれたとすると、リスト2の都市4に移り、ここを逆順に進むことになる。以後、同様の処理を繰り返しながら進んで行き、2つのリストが空リストとなったとき一つの閉経路が求まる。こうして統合された経路が図17である。

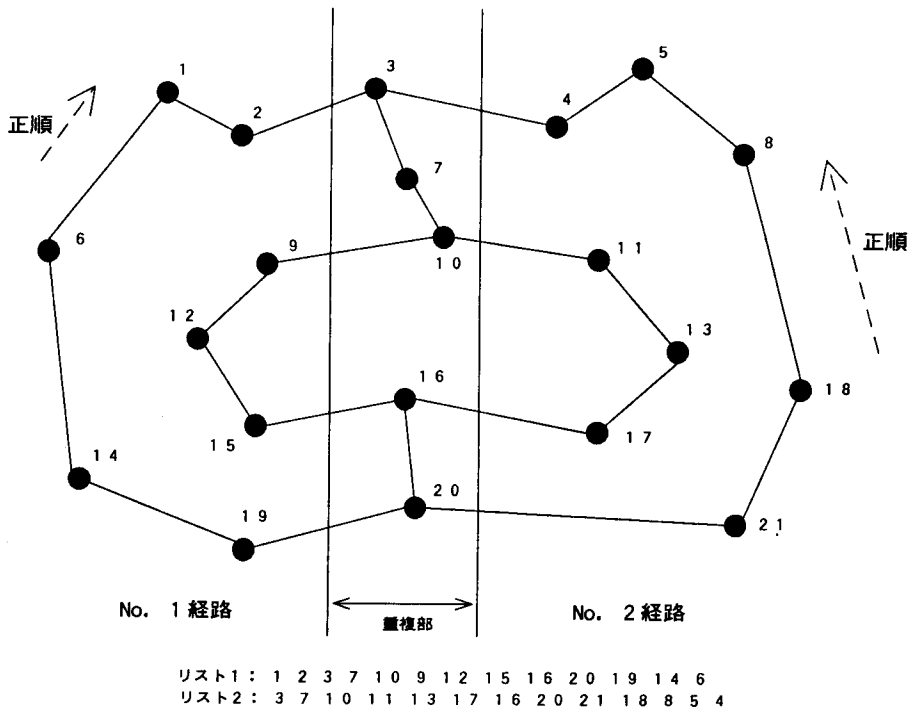


図16 結合前の経路
 Fig. 16 Root before integration

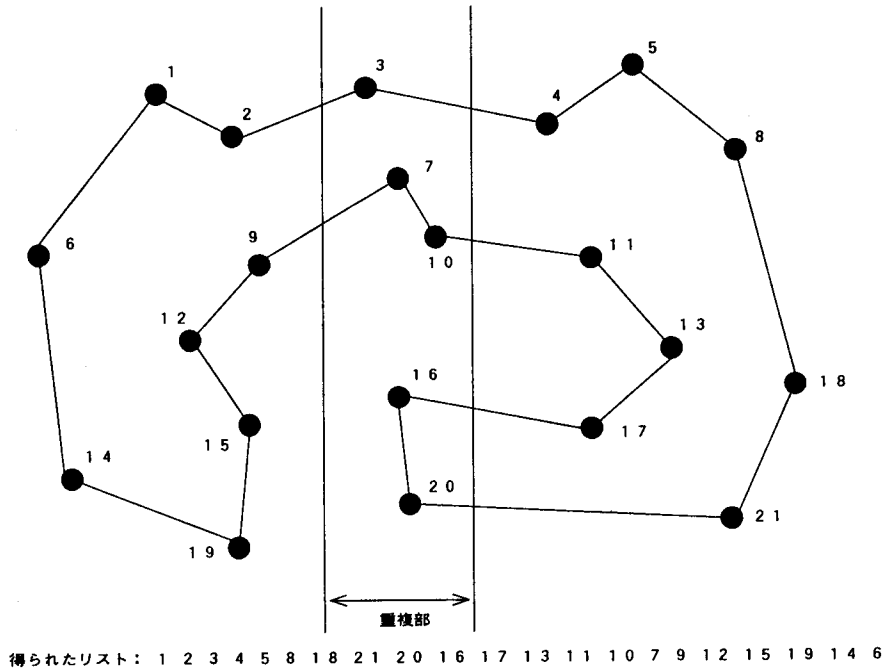


図17 結合後の経路
 Fig. 17 Root after integration

4.5 シミュレーション結果と考察

本方式を検証するにあたって101都市の TSP を用いた, GA のワークベンチとして GENITOR 5) を使用し, 交叉法には辺再組合せ交叉 (ERX), 突然変異には挿入法を用い, 個体数は500とした. 図18に本方式を適用して最終的に得られた経路を示す. 図19は従来の方式即ち分割せずに GA を適用した方式と本方式の統合後の各世代の適合値の最良値を比較したものである. このグラフでは世代数が2万世代からの表示となっているのは本方式と従来の方式との比較の際, 不公平にならないようにするためである. 本シミュレーションでは分割時の世代数も2万世代であるので(4分割領域は並列演算), 統合後の世代は2万世代より開始するとして表示している. なおここでは1回の計算を1世代としているので, 通常の世界表示ではこれらの値の $\frac{1}{500}$ となる. 図より分かるように初期段階から良い結果が得られており, このことから局所仮説が成り立っていると考えられる. また, 従来の方式では7万世代でも本方式の初期値に到達していない. 以上述べたように本方式の採用により高速化が達成できたことが分かる.

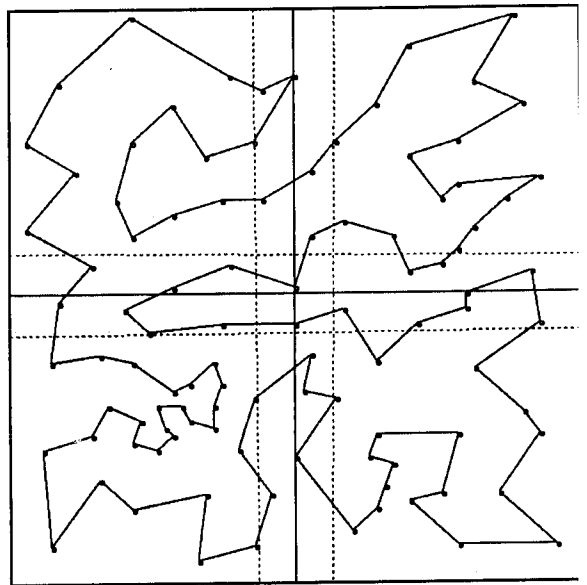


図18 シミュレーションの結果
 Fig. 18 Result of simulation

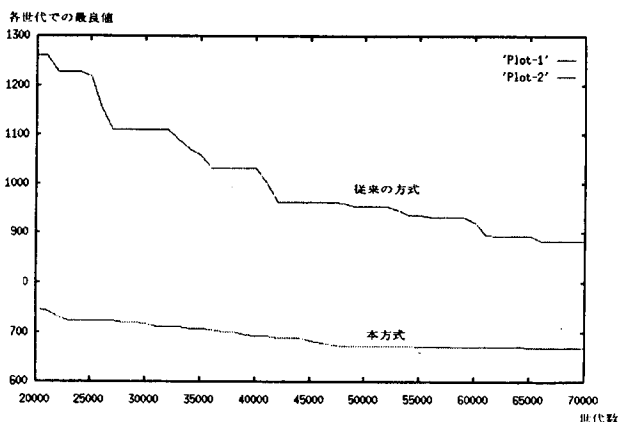


図19 最良値の変化の比較
Fig. 19 Comparison of convergence

本報告では分割領域に対してGAを適用し、その結果を統合したものに再びGAを適用することにより、GAによる最適問題の解の高速化を図る研究について述べた。領域の分割方式はこの述べたものの外に、2分割して統合する方式を基本とし、これを再帰的に適用して4分割、16分割など行なう方式も考えられる。また重複部の統合方式についても検討の余地があるが、本研究によりこのような分割方式の有効性が検証できた。

5. ニューラルネットワーク 構造決定問題への応用

5.1 問題設定

ニューラルネットワーク (NN) は生物の神経系の特徴的な機能に着目してそのモデル化を行ったものであり、最近では文字認識や音声認識などのパターン認識などに用いられている。このNNの開発には問題の設定、構造の決定、学習、評価という過程があり、ここで問題になるのはNNの構造決定である。これはネットワークの構造を直接評価する方法がないからである。そのため現在では経験的に構造決定を行うしかない。しかしこの方法では大規模なネットワークになるとその構造決定は困難になる。

ここで取り扱うNNワーク構造決定問題は、任意のNNモデルを必要最低限のユニット数およびコネクション数で実現しようとする最適化問題として位置づけする。

つまり最初に任意の大きさを持ったネットワークを与えておき、その構造からノードおよびコネクションを

削除していくことで解を求める。

ここで問題になるのはNNの構造をどのように評価するかである。本研究では、一定回数だけバックプロパゲーション法 (BP法) により学習させ、その結果を用いて評価する方法を採用した。またNNとしてはフィードフォワード型の多層NNを対象にした。

5.2 システム構成

5.2.1 遺伝子表現

本問題においては以下の2つの遺伝子表現を考えた。

1. コネクション表現 図20に示すようにネットワークのユニット間のコネクションの有無を用いる表現であり、細かな構造表現が可能である反面、致死遺伝子 (評価対象外の遺伝子) が発生する確率が高くなる。

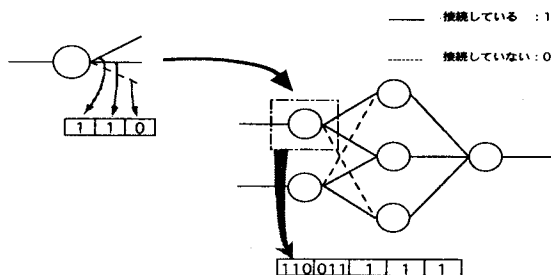


図20 遺伝子の表現
Fig. 20 Composition of genes (connection representation)

2. パス表現 図21に示すように、入力層から出力層までのパスの有無を用いる表現である。この表現では致死遺伝子は発生し難い特長がある。

これらの遺伝子表現において必要なビット数は以下のようなになる。

1. コネクション表現

$$bit = \sum_{i=1}^{l-1} L(i)L(i+1) \quad \dots (1)$$

2. パス表現

$$bit = \prod_{i=1}^l L(i) \quad \dots (2)$$

ここで bit 必要ビット数、 $L(i)$ は i 番目の層のノード数、 l は層数である。

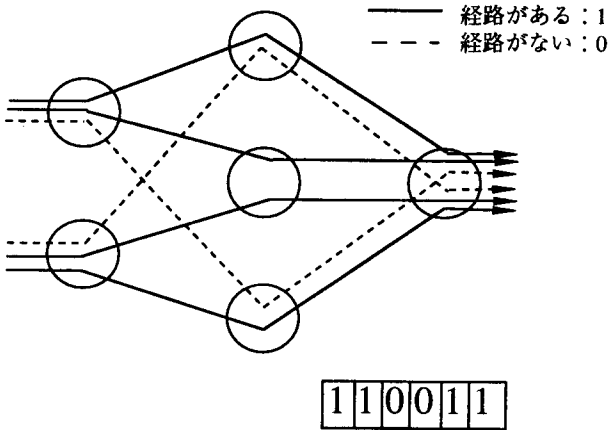


図21 遺伝子の構成
Fig. 21 Composition of genes (path representation)

5.2.2 適合値

構造決定問題では以下の条件を満たす必要がある。

1. ニューラルネットワークとして実用に耐えるだけの収束率を示すこと。
2. 可能な限り少ないノード数およびコネクション数であること。

この条件を満たすために以下のような評価式を用いて適合値を求めた。

$$Perf = Unit \times Connect \times \max_T \sum_i^I (Y_i - o_i)^2 \dots\dots(3)$$

ここで *Perf* は適合値、*Unit* はユニット数、*Connect* はコネクション数、*T* は教師パターン数、*I* は出力層のノード数、*Y_i* および *o_i* はそれぞれ出力層 *i* 番目の教師出力と実出力である。また、この適合値を最小にするようにしている。

5.2.3 遺伝的操作

遺伝的操作には以下の方式を採用した。

1. 個体の選択方式はルーレットホイール方式とする。また常に最良構造が次世代に残るようにエリート選択方式とする。
2. 交叉は2点交叉とする。
3. 突然変異は一定の確率で任意のビットを反転させる。

図22に今回実施したシミュレーションのフローチャートを示す。GAによって発生させた個体に対してそれぞれBP法による学習を適用し、その結果を適合値としてGAに戻し、新しい個体を生成する。こうして最良構造を導く。

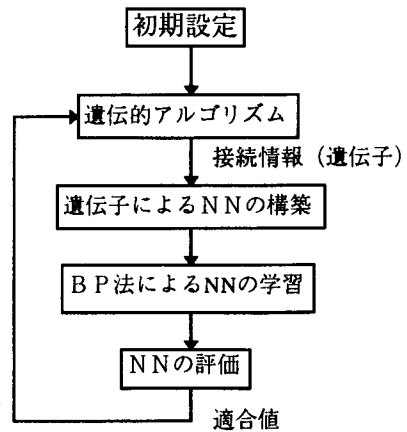


図22 NNの構造決定GAのフローチャート
Fig. 22 Flow chart of structure-determining GA of NN

5.3 シミュレーション結果と検討

本システムの検証のために実施したシミュレーションの例とその検討結果について以下に述べる。

このシミュレーションでは、排他的論理和 (XOR) を実現する多層型 NN の最良構造を GA で求めることを行った。入力層のノード数2と出力層のノード数1は既知であるが、中間層は今回は1層と固定しノード数を未知としてGAで最適化を行った。ノード数の最大値は10とした。また各ノードのバイアス値および各コネクションの重みの初期値は各個体で同じ乱数系列を用いて設定した。

以上の条件のもとで2表現についてシミュレーションを行った。シミュレーションにあたり個体数は50、交叉確率は0.6、突然変異率は0.01とし、BP法による学習回数は100とした。学習回数は通常のNNの回数に比べて少ないが構造決定には十分であった。また、遺伝子の必要ビット数は、式1および式2より、コネクション表現で30ビット、パス表現で20ビットである。

シミュレーションにより得られた各表現での適合値の推移のグラフを図23に示す。このグラフから分かるように、いずれの表現においても十分な収束を得ることができた。グラフでパス表現が約200世代で途切れているのは、全個体が収束したので計算を終了したためである。パス表現の方が比較的安定した収束を示しているのは、コネクション表現に比べて発生する致死遺伝子が少ないからだと思われる。最終的には、どちらの表現も0.3545という同じ最良適合値を得ることができた。このときの遺伝子を表現型になおすと、いず

れも図24に示す構造になっている。この構造のNNは表1にその動作を示すように、排他的論理和の特性を満足している。かつ中間層のノード数は3、コネクション数は7と、この特性を持たせるに必要な最小のNNとなっている。計算時間については、プログラム上はバス表現の方がコネクション表現に比べて時間がかかるが、少ないビット数で表現できるため、全体では大きな差はなかった。

以上よりニューラルネットワークの構造決定問題へのGAの有効性を確認できた。今後は、より大規模なニューラルネットワークモデルへの適用が課題になる。

表1 得られた最良構造の動作特性

Table 1 Characteristic of NN gained by GA

教師入力		教師出力	実出力
X_1	X_2	Y	o
0	0	0	0.0851
0	1	1	0.8857
1	0	1	0.8935
1	1	0	0.1255

6. あとがき

本論文では最適問題に対する遺伝的アルゴリズムの適用の3つの分野、すなわち二次元材の最適切断、領域分割・統合によるセールスマン巡回問題解法の高速度化、ニューラルネットワークの構造決定についての研究結果について報告した。遺伝的アルゴリズムの適用に際しては遺伝的操作方法もさることながら、遺伝子の表現方法、他のシステムとの協調などが重要である。これら3つの例はいずれもそれぞれ特徴があり、今回得られた結果は今後の同種の問題に適用する際に多くの示唆を与えている。これらの成果をもとに引き続いて遺伝的アルゴリズムの応用について研究を進める予定である。

参考文献

- 1) J. H. Holland: Adaptation in Natural and Artificial Systems, The University of Michigan Press, 211p, 1975
- 2) N. Christofides and C. Whitlock: An Algorithm for Two-Dimensional Cutting Problems, Operation Research, Vol. 35, No. 1, pp. 30/44, 1977
- 3) L. Davis (Ed): Handbook of Genetic Algorithms, Van Nostrand Reinhold, 385 p, 1991
- 4) D. E. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Pub. Co., 412p, 1989
- 5) Z. Michalewicz: Genetic Algorithms + Data Structures = Evolution Programs (Second, Extended Edition), Springer-Verlag, 340 p, 1994
- 6) T. Ono & G. Watanabe: Application of Genetic Algorithms to Optimal Selection of

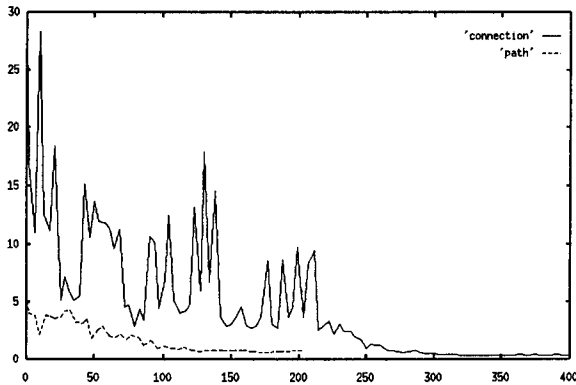


図23 適合値の推移
Fig. 23 Variation of fitness values

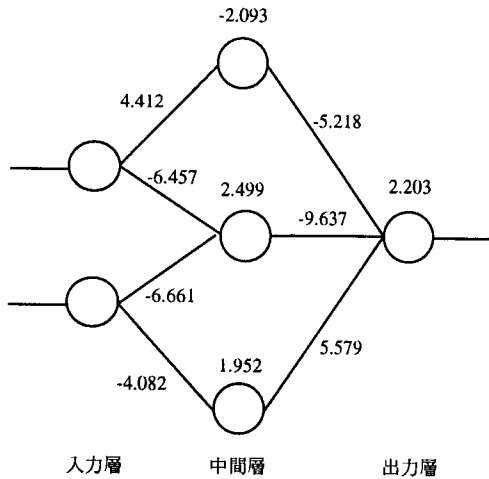


図24 得られた最良構造
Fig. 24 Best structure gained by GA

- Cutting Length of Bars, Proceedings of Fifth FIT-Ajou University Joint Seminar, pp. 24-31, 1995
- 7) 北野編：遺伝的アルゴリズム，産業図書，328p, 1993
- 8) 小野，渡辺：遺伝的アルゴリズムによる最適切断組合せの決定，SICE九州支部学術講演会予稿集，pp. 173-174, 1994
- 9) 小野，渡辺：遺伝的アルゴリズムを用いた棒材の最適切断組み合わせ，電子情報通信学会総合大会論文集—情報・システム 1, pp. 240, 1995
- 10) 小野，渡辺：遺伝的アルゴリズムの最適問題への応用，福岡工業大学言語情報工学研究所彙報，第6巻，pp. 89-96, 1995
- 11) 石丸，小野：遺伝的アルゴリズムによる電力系統の最適運用計画作成に関する研究，情報処理学会全国大会，pp. 2-111-112, 1996
- 12) 渡辺，小野：二次元最適配置への遺伝的アルゴリズムの応用，情報処理学会全国大会，pp. 2-113-114, 1996
- 13) 渡辺，小野：遺伝的アルゴリズムを用いた二次元最適切断，第40回システム制御情報学会研究発表講演会論文集，pp. 45-46, 1996

~~~~~

(平成8年5月29日 受付)