

遺伝的アルゴリズムの応用

小野 俊彦 (情報工学部情報工学科)
横井 智幸 (大学院工学研究科情報工学専攻)
山田 正和 (大学院工学研究科情報工学専攻)
河村 将史 (大学院工学研究科情報工学専攻)

Application of Genetic Algorithms to Various Problems

Toshihiko ONO (Department of Computer Science and Engineering,
Faculty of Information Engineering)
Tomoyuki YOKOI (Communication and Computer Engineering,
Graduate School of Engineering)
Masakazu YAMADA (Communication and Computer Engineering,
Graduate School of Engineering)
Masashi KAWAMURA (Communication and Computer Engineering,
Graduate School of Engineering)

Four studies on the application of genetic algorithms (GAs) are explained. The first is to develop java language-based GAs workbenches, which are applicable to various problems of both a binary bit string representation and an integer ordinal one of genes. By making use of network adaptability and a graphic user interface, various simulations become possible from remote computers. The second is to develop the system to build up an automaton by GAs from the requirements of acceptance and rejection of input signals. This can be used as a design automation tool. The third is to solve a multiple traveling salesman problem (TSP) by GAs, where a visiting is shared by many persons different from the ordinary TSP. The last is a pattern search by GAs, that is, to detect a target pattern in a picture containing various patterns and smudged by noises.

Key words: *genetic algorithms, optimal problem, java, automaton, traveling salesman problem, pattern search*

1. ま え が き

今日まで遺伝的アルゴリズム (以下 GA: Genetic Algorithms) の各種応用について、特に最適問題への適用を中心に研究を進め、随時その成果を報告してきた⁷⁻¹¹⁾ 本報告ではそれらに続く以下の4テーマについての研究成果について述べている。

第1のテーマはGAのワークベンチに関する研究である。今日までGAを実現するためにC言語によるワークベンチを主として使用してきたが、本研究はこれらをJava言語により実現することである。

Java言語化による利点には種々あるが、その中でも最大のものはネットワーク機能であり、研究成果を広く学内外に公表し、遠隔教育などに利用する場合に効果がある。今回の研究により、遺伝子表現にビットストリングまたは順序数を使用する各種用途に適用することが可能となった。

第2のテーマはGAによるオートマトンの自動生成に関する研究である。本研究では入力信号の受理および却下の条件を与えて、GAにより自動的にオートマトンを生成することを目的としている。オートマトンは言語や自動機械などの機能の表現として広く利用

されており、本研究は設計自動化に関する研究でもある。

第3のテーマは複数セールスマン巡回問題（以下複数 TSP: Traveling Salesman Problem）への GA の適用である。従来の TSP では一人のセールスマンによる都市巡回であるが、本問題では複数のセールスマンにより分担して都市巡回する場合の最適経路を求める問題に拡張している。

第4のテーマはパターン探索への GA の適用である。雑音を含む多数のパターンよりなる画像の中から所定のパターンを探索する際、GA の持っている並列探索の性質を生かし、効率的に探索する方法の研究である。

以上のテーマについて以下、順次説明する。

2. Java 言語による遺伝的アルゴリズム

2.1 開発の経緯

GA は多くの特徴を持っているが、その中でも特に各種問題に適用する際に融通性が高く、数式モデルが得られない場合にも使用できることが挙げられる。対象とするシステムや問題をビットストリングや順序数などからなる遺伝子で表現し、適合値関数を定義できれば、後は標準のプログラムで解くことができる。そこで使用される選択、交叉、突然変異などの遺伝子演算子は多くの場合、標準のものまたは多少の修正で利用できる。このような目的で作成したプログラムをここでは GA ワークベンチと呼ぶことにする。今日まで GA の研究には主として、C 言語で設計された Genesis^{1), 2), 3), 5), 6)} や Genitor^{1), 3), 5), 6)} などを使用してきたが、本研究ではこの GA ワークベンチを Java 言語で実現することを目的にしている。

Java 言語の採用による利点は種々あるが、本研究では特に以下の事項に注目した。

1. ネットワーク対応であること：研究成果の外部への発表、社会人教育用教材の提供、実験など各種の用途において大きな効果を発揮する。ホームページを通してプログラムを提供し、ネットワークを通して遠隔地で WWW ブラウザーにより簡単にシミュレーションできることは他の言語にない特長である。
2. GUI 機能に優れていること：GUI (Graphical User Interface) により画面上で容易にパラメータやモードを変えて実行できることは、GA を実行する際に各種の条件での性能の比較対照が可能とな

り、優れた機能を提供する。

3. 今後の主要コンピュータ言語となる可能性が大きいこと：学生の教育の面でもこのような将来性のある言語を学ぶ機会は貴重であり、その面でも適している。

GA の実現に関してまず考慮すべき事項は遺伝子の表現方法である。これまでの適用経験によると、殆どの対象に対してつぎの何れかの遺伝子表現で実現できる。

- バイナリビットストリング表現：バイナリビットストリングを適当に区分し、対象とする問題に応じた遺伝子表現型に変換する。
- 順序数表現：セールスマン巡回問題に代表される順序問題に使用される表現方法であり、1 から始まる順序数で表現される。

これらの両者では交叉、突然変異の方法が異なるので、無理に一つプログラムにせずに別々にした方がよいとの判断のもとで、今回はこの両者に対する GA ワークベンチを開発した。しかし、これら両者は遺伝子表現や遺伝子操作を除いては同じ考えのもとに作成されているので、統一システム名として Genova (Genetic Algorithms by Java) と呼ぶことにする。以下、この両者について説明する。

2.2 ビットストリング表現 GA

GA ワークベンチを開発するに際して以下の考え方に基づいた。

1. 各種の対象に使用されることを考え、共通部と対象に固有のものに分け、後者をできるだけ少ないものにする。Java 言語のオブジェクト機能を活用し、対象固有の処理、適合値の計算などは一つの対象固有のオブジェクトにまとめた。
2. 各種の条件でのシミュレーションが可能のようにモード選択ができ、かつパラメータ値を種々変化させて実行できることを考慮した。Java 言語は GUI 機能に優れており、この面では有効に活用できた。
3. グラフィック機能を活用する。最適問題の収束性を検証するための適合値の世代に対する変化と、得られた対象の状態の変化の両者を図示するようにした。後者は対象により異なるので、その都度設計する必要があるが、前者は対象には関係ないので、共通に利用できる。

以上の考えのもとに作成した一例を図 1 に示す。ここでは多数の家庭 (図では 100 軒) よりなる地域に複数のストア (図では 4 軒) を設置する場合の最適な

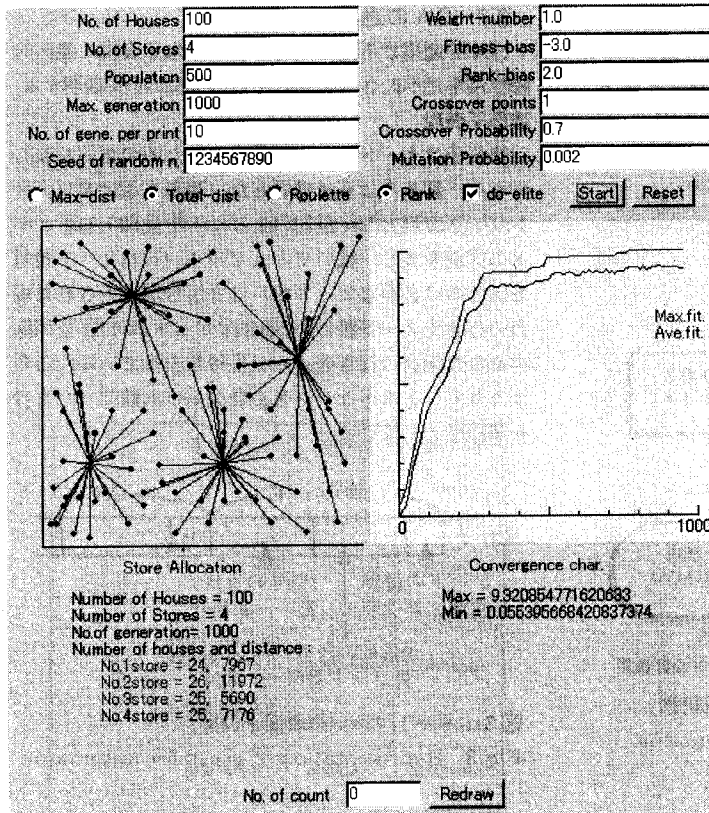


図1 実行結果の表示画面
Fig.1 Display of results

配置をGAを用いて決定する問題を解いている。

図の上部は各種数値パラメータの設定部になっており、ここで数値を入力したり、変えたりすることができる。図の例ではまず家庭の数、ストアの数、続いてGA自体に関するものとして、個体数、最大世代数、出力間隔、乱数のシード、適合値関係のパラメータ3種、交叉点数、交叉確率、突然変異確率などが設定できる。

つぎはモード選択に関するもので、左の2つは適合値の計算に関するものとして最大距離か合計距離かの選択ができる。続いての2つは選択方法に関するものであり、ルーレットホイール選択かランク選択かの何れかが選べる。つぎはエリート選択の有無の指定である。最後の2つは実行に関するものであり、実行開始ボタンとパラメータなどを変化させた場合に押しリセットボタンがある。

つぎの図示部分では左側に各ストアと各家庭との

関係を直線で示しており、これはGAによって得られて結果の図示である。右側は世代により適合値がどのように変化しているかを示しており、これより収束状態を知ることができる。なお図の下部にあるデータは図に關係する数値を示している。

最下段はデータの入力部と押しボタンよりなり、最終計算結果が得られた段階で途中世代の状況呼び出して表示し、世代による状態の変化を知ることができる。

なお、図では白黒の濃淡表示になっているが、実際はカラー表示を採用し、特に図示部分では線その他を色で分け、理解し易いものになっている。

2.3 順序数表現

遺伝子表現として今一つよく使用されるのが順序数表現であり、TSPを始め、スケジューリング問題、最適切断問題など多くの用途に利用されている。この目的のGAワークベンチも基本的には前項と同じであるが、遺伝子表現が異なることにより多少違ったものとなっている。例えば、交叉の方式が異なり多くの方法があるので、辺再組合せ交叉(ER)、順序交叉(OX)、

循環交叉(CX)、部分写像交叉(PMX)などを選択して実行し、比較対照できるようになっている。

3. GAによる有限オートマトンの自動生成

3.1 問題設定

本研究の目的は任意の与えられた受理条件を満足させる有限オートマトンをGAを用いて自動生成することである。このようなシステムを構成するに際してつぎの2点を考慮した。

- 与えられた受理・却下理信号をすべて満足する。
- 構成ノード数を最小にする。受理条件を満足するオートマトンは多数考えられるが、最適化の観点から最小のノードをもつオートマトンを生成するようにする。

なお、入力信号は1,0よりなるバイナリ信号とした。

3.2 システム構成

本システムの構成を図2に示す。図のようにノード数制御部、GA、オートマトン構成アルゴリズム(ACA: Automaton Construction Algorithm)および、受理・却下信号データファイルで構成し、主な動作は以下のようになっている。

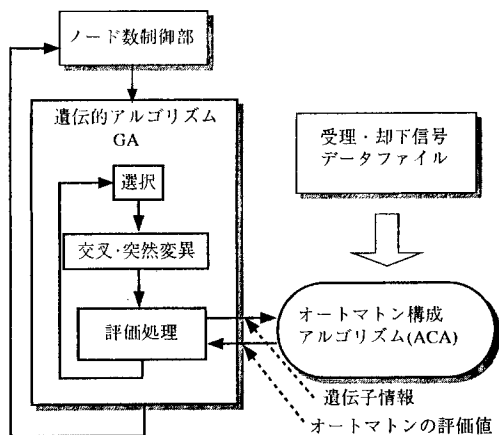


図2 オートマトン自動生成システム系統図
Fig.2 Schematic diagram for automaton creation

まずノード数制御部ではノード数を決定する。ノード数は最小のノード数(2ノード)より開始し、後述のように順次増やして行く。GAでは選択、交叉、突然変異等の遺伝的操作を行い、それによって生成された遺伝子情報をもとにACAによってオートマトンを作成し、受理・却下データファイルを参照してこのオートマトンの評価を行い、その評価値をGAに返す。受理・却下データファイルには受理および却下する入力信号列を記述しており、これにより両条件を定義している。GAは評価値が最適になるように遺伝的操作を繰り返して行い、所定の世代数までに両条件を満たすオートマトンが求まらない場合には、ノード数を増加させて継続する。以上の操作により、最小ノード数のオートマトンを求めることができる。

GAのワークベンチにはGenesisを使用し、これに必要な機能を追加して、システムを実現した。また、その他に受理・却下データを生成するプログラムやオートマトンを描画するプログラムを作成している。

3.3 GAの適用

GAを適用する際に問題となる遺伝子表現・遺伝的操作法を述べ、さらにACAの機能について説明する。

遺伝子表現

遺伝子はオートマトンを作成するための情報を表しており、図3のように構成した。表現方法はビットストリング表現とし、最初の1ビットで受理・却下ノードの区別を表し、続いてのnビット(nはノード番号を表すのに必要なビット数)は0信号が入力された場合の行き先ノード番号、次のnビットは1信号が入力された場合の行き先ノード番号を表している。このようなビットストリングを必要なノード数並べたものを遺伝子としている。

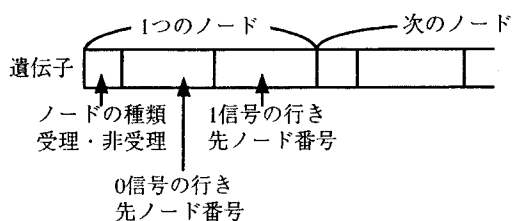


図3 オートマトン用遺伝子表現
Fig.3 Representation of genes for automaton

選択・交叉・突然変異操作及び目的関数

選択はエリート保存を加えたランク方式を、交叉は2点交叉方式を採用し、突然変異は0と1を入れ替える方法を採用した。なお、目的関数値には以下で説明するACAの評価値を用いた。

3.4 オートマトン構成アルゴリズム(ACA)

ACAはGAからの遺伝子情報を基にオートマトンを生成し、データファイルを参照してオートマトンを評価を行うアルゴリズムである。生成されたオートマトンの評価は次に示すステップで行う。

1. データファイルから1つデータを取り出す。
2. 取り出したデータをオートマトンに入力し受理・却下の結果を得る。
3. 得られた結果とデータの受理・却下条件が一致した場合に評価値に1を加算する。
4. データファイルのすべての受理・却下条件が終了するまで1~3を繰り返す。

得られた評価値とデータの総数が同じ値となれば、所定のオートマトンが求まったことになる。

3.5 シミュレーション

本システムを検証するために文献⁹⁾で紹介されている7種の受理条件でシミュレーションを行った。表1に各シミュレーションで生成するオートマトンの条件を示す。実行時のノード数は、初期ノード数を2とし、各100世代間で受理・却下条件を満足するオートマトンが得られない場合に、順次1つ増加させる方式とした。

表1 オートマトンの受理・却下条件

Table.1 Acceptance and rejection conditions of automaton

シミュレーション	受理条件	受理例	却下例
1	b*	bbbbbb	bbaba
2	(ba)*	babababa	abbaaab
3	奇数個のbのあとのaは奇数個ではない	aaabaabb	abbbaaaaa
4	aaa列が無い	bbabbbaa	abbbaaaba
5	ab列の数とba列の数の和が偶数	aabbaaaba	aabbaaab
6	bの数とaの数の差が3の倍数	abbaabbbb	bbabaaa
7	a*b*a*b*	aababbb	abbaabba

また受理・却下データファイルには、シミュレーション1, 2では長さ5ビットまでの、その他のシミュレーションでは長さ10ビットまでの、それぞれ取り得るすべてのビット列について、受理するものと却下するものに分類して定義した。従ってそれぞれ63列および2047列のビット列となっている、これらのデータは別のテストデータ作成プログラムを用いて作成した。

このほか、GA関係のパラメータとしては集団の個体数を100、交叉確率を0.7、突然変異確率を0.02にそれぞれ設定した。

3.6 シミュレーション結果と検討

7種のシミュレーションのすべてにおいて受理・却下の両条件を満足するオートマトンが得られた。その例として、シミュレーション3, 5, 6の実行結果を図4, 5, 6に示す。なお、これらの図は遺伝子表現よりオートマトン作図プログラムにより描画したものである。

以上の結果より、有限オートマトンの自動生成にGAを適用することの有効性が確認できた。

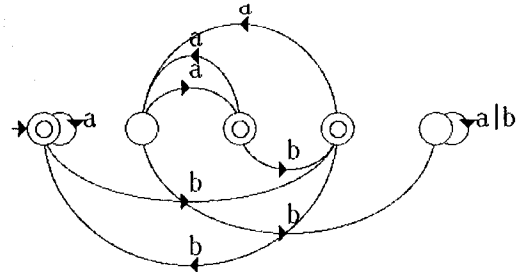


図4 生成オートマトン-1
Fig.4 Created automaton-1

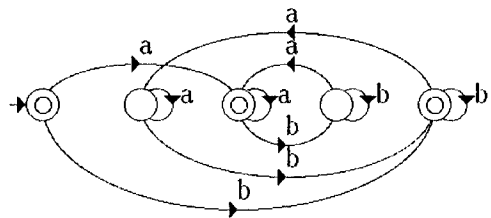


図5 生成オートマトン-2
Fig.5 Created automaton-2

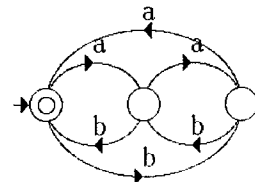


図6 生成オートマトン-3
Fig.6 Created automaton-3

現在、本システムに適した交叉・突然変異方法の検討、ノード数を増加させる世代数の決定法、GP(遺伝的プログラミング)を使用した場合との比較などについて研究中であり、詳細は今後報告する予定である。

4. 複数セールスマン巡回問題

4.1 研究内容

GAの典型的な問題としてセールスマン巡回問題がある。この問題は都市数が増えるに従って計算量が階乗的に増大し、解を求めるのが難しくなるNP困難な問題として有名であり、多くの研究がある。ここで取り上げた複数セールスマン巡回問題では、これを拡張し複数人のセールスマンで分担する方法に対して効

率よく解くことを目的にしている。シミュレーションでは例題として最適解の分かっている40都市を4人で10都市づつ巡回するものと最適解の分からない100都市を4人で25都市づつ巡回するものを解くことができ、この種の問題の解法が確立できた。

4.2 遺伝子表現と評価方法

遺伝子表現は40都市を4人で巡回する場合、都市番号を表す整数型の長さ40の配列としその一つ一つにあらかじめ位置の分かっている都市の都市番号を割り当てる。この配列を図7で示すように4人の各セールスマンにつき10個ずつに区切り、全セールスマンの巡回距離の合計を最短にする最適解を求めた。このように今回はセールスマン各個人の巡回距離を最短にする方式でないため、都市の配置によっては巡回距離の個人差は生じるが、適合値の計算方式を変えることによりこの解決を含む各種の最適モードを作ることが可能である。

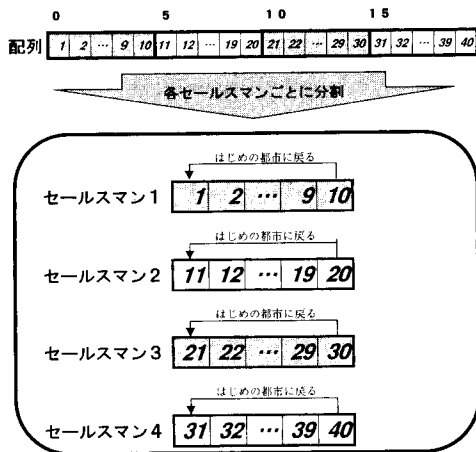


図7 複数 TSP 用遺伝子の表現と評価法
Fig.7 Representation and evaluation of genes for multiple TSP

4.3 交叉法

交叉法には図8で示すように従来の辺再組み合わせ交叉 (Edge Recombination Crossover)³⁾ を複数 TSP 用に改良を加えたものを使用した。主な改良点はつぎの2点である。1つは辺リストの作成時に各セールスマンごとに最後の都市と最初の都市を隣り合う都市として処理したこと、もう1つは各セールスマンが最初に訪問する都市を個別にランダムに選べるよう

にしたこと、即ちあるセールスマンの最初の都市は前のセールスマンの最後の都市の辺リストに無関係に決定できるようにしたことである。

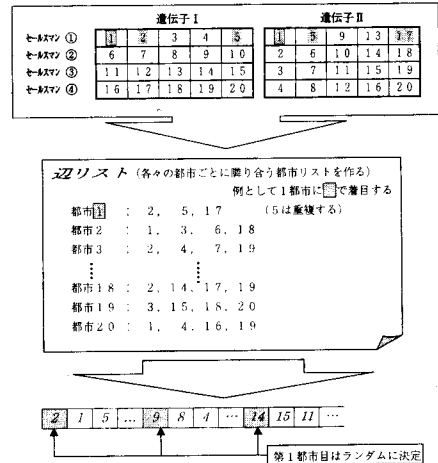


図8 複数 TSP 用辺再組み合わせ交叉
Fig.8 Edge recombination crossover for multiple TSP

4.4 突然変異法

突然変異法には任意の2つの遺伝子を入れかえる SWAP法を採用した。さらに初期収束、即ち全個体で多様性が失われたときの対策として、通常時の突然変異確率に加え全個体数の半数以上が最良の適合値と同一になったときに、多様性を回復するために突然変異確率を強制的に高くする方法を採用し解の収束を改善した。

4.5 シミュレーション結果

以下の2つのシミュレーションを実施した。

4.5.1 40都市星型配置の場合

まず本方式の収束性、即ち最適解が求まることを検証するためにあらかじめ最適解の判明している都市配置に対してシミュレーションを行った。その条件を以下に示す。なお GA ワークベンチとして Genitor の Java 移植版を使用した。

- セールスマンの人数：4人
 - 都市数：40都市
 - 遺伝子の個体数：100個体
 - 交叉：辺再組み合わせ交叉 (改良版)
 - 突然変異：SWAP法 (変異確率：0.1%)
- 標準的なパソコン (OS: WindowsNT4 CPU:

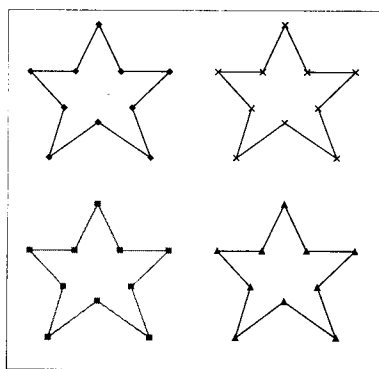


図9 40都市星型配置の解
Fig.9 Result of simulation for 40 cities arranged in star

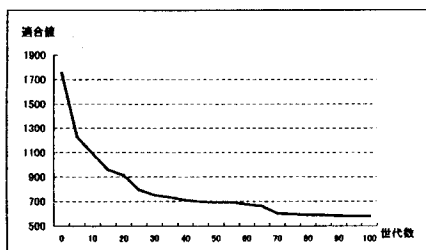


図10 40都市星型配置の解の収束特性
Fig.10 Convergency characteristics of 40 cities

Pentium II 350MHz) で10秒で116世代目に図9に示す最適解が得られた。その時の収束特性を図10に示す。

4.5.2 100都市ランダム都市配置の場合

ここではランダムに配置した100都市に対してシミュレーションを行った。前述の40都市星型の問題と異なる点は、都市数が100都市に増えたことと最適解が不明であることであり、その他のパラメータの値は同一である。2万世代での巡回経路と収束特性をそれぞれ図11および図12に示す。最適解は不明ではあるが前述の40都市のシミュレーション結果からかなり最適解に近いものが得られたと思われる。

このように複数 TSP を解く方法として GA の適用が有効であることが確認できた。また、遺伝子全体の多様性が失われたとき突然変異確率を強制的に高くすることで解の収束もかなり改善することも確認された。

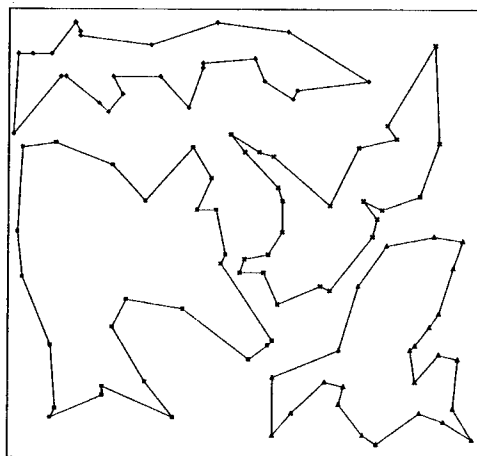


図11 ランダム100都市の解
Fig.11 Result of simulation for 100 cities

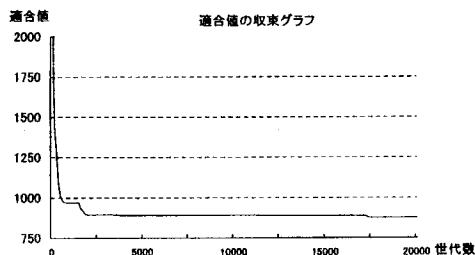


図12 ランダム100都市の解の収束特性
Fig.12 Convergency characteristics of 100 cities

5. GA を用いたパターン探索

5.1 研究目的

本研究で対象にするパターン探索は複雑な図形の中から特定の形を見つけだす図形探索である。この方法として通常の方法を用いた探索領域の全面探索方式では、多量の処理時間を必要とし実用的ではない。そこでランダム探索を応用した GA を用いることにより、より短時間で効率の良いパターン探索を可能にすることを目的としている。

5.2 研究内容

5.2.1 遺伝子の適用

遺伝子の表現方法としては、ビットストリング表現を採用し、探索位置を図形の左上を起点として、X、Y 座標をそれぞれ 8 ビットの計 16 ビットで表現した。

5.2.2 画像処理

本研究で使用する画像および図形は、探索する図形

も含む複数の図形とノイズからなる原画像，原画像の中からパターン探索により見つけたモデル図形，モデル図形単体を含んだモデル画像から成り立っている。これら原画像及びモデル画像の作成や入力，Xウィンドウ上の画面操作ライブラリ関数 Xlib を用いて作成したプログラムにより行う。同様に，結果の出力も視覚的に確認できるような同じく X ライブラリを用いて CRT モニタ上に表している。

原画像とモデル画像はビットマップ形式により作成し，2次元配列として保存する。なお画像は線画像としているので各要素はバイナリビット(0, 1)となっている。

5.2.3 ボカシ処理

GA によりパターン探索を行う際の効率を高めるために前処理として，ボカシ処理を原画像に施して，白黒2値画像を濃淡階調画像に変換する。これを使用して，モデル図形が原画像上の目的の図形にどれだけ接近しているかを適合度で定め，GA により探索を行っている。

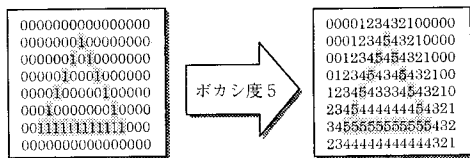


図13 ボカシ処理の方法
Fig.13 gradation method

ボカシ処理の方法は図13に示すようにボカシ度を n (n は整数) を設定し，原画像の配列で値が1の要素には n を格納する (図では $n = 5$)。 n が格納された要素の前後左右の配列には $n - 1$ を格納する。同様に $n - 1$ の前後左右の配列には $n - 2$ を格納するというように， $n = 0$ になるまで繰り返していくことにより画像全体にボカシを加えていく。またボカシが重なった場合にはボカシ度の大きい方を採用する。こうすることにより数字が大きいほど濃い点となり目的の図形と近いことになる。

5.2.4 パターン探索

パターン探索はモデル画像を格納した配列と原画像を格納した配列とを比較して行う。原画像上をランダムに探索して行き，GA によって導き出した X, Y 座標を基準としてその場所の適合度をモデル画像を用いて求め，GA によりこれを最大にする位置を求める。

適合度は各点の濃淡に対応した値を加算して行くことにより得られる。各点の濃淡値を a_i ，モデル画像の配列内にある1の総数を m とすると，適合度 x の計算式は

$$x = \frac{\sum a_i}{m} \{i = 1, 2, 3 \dots\} \quad (1)$$

となる。ここで i はモデル画像の画素の数である。

5.2.5 線形探索

本研究では GA により目的の図形の近傍に到達したら，以降は線形探索に切り替える方法を採用した。それは GA の探索は基本的にランダム探索なので目的図形の近傍への到達は効率がよいが完全に適合度を一致させるには不向きであるためである。切り替えるタイミングは，しきい値を定め適合度がこれ以上になるとその場所より上下左右に数ドット分を線形的に探索を行う。

5.2.6 システム構成

以上の動作に対するシステム構成を図14に示す。

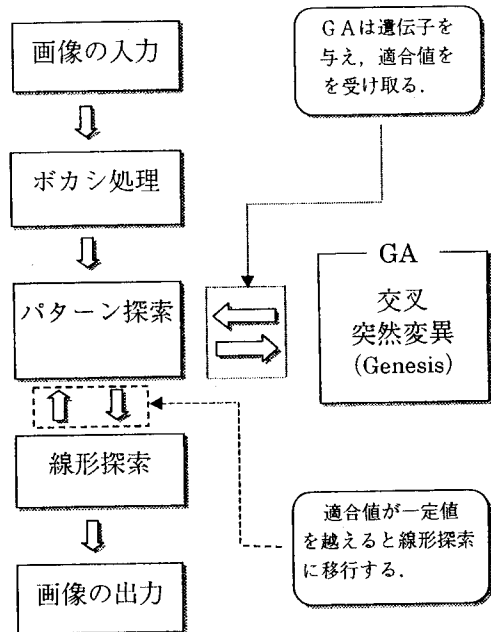


図14 図形探索システム構成図
Fig.14 Schematic diagram for pattern search

5.3 シミュレーションと考察

本方式の有効性を実証するために以下の条件でシミュレーションを行った。

- 原画像及びモデル画像は正方形の領域を考える。
- 探索するパターンは三角形・四角形または円とする。
- 探索する図形は原画像上に1つしか存在しないものとする。
- 原画像上には雑音（ノイズ）も含む。
- すべての図形のパターンは線画とする。
- モデル図形の原画像上における、回転及び拡大縮小は考えない。

また GA のパラメータは、個体数を100、交叉確率を0.6、突然変異確率を0.001、とした。図形関係では原画像の領域を256×256、モデル画像の領域を64×64、ボカシ度を5、線形探索のしきい値を0.7とした。

画像は図15を用いて探索のシュミレーションを行った。その結果、短時間で目的の図形を見つけることができた。また画像をより複雑に変えたシュミレーションでも同じく短時間でパターン探索することができた。このことより画像領域の大きさや画像の複雑さによってボカシ度や線形探索のしきい値などの設定を調節することによってあらゆる画像からのパターン探索が可能となった。

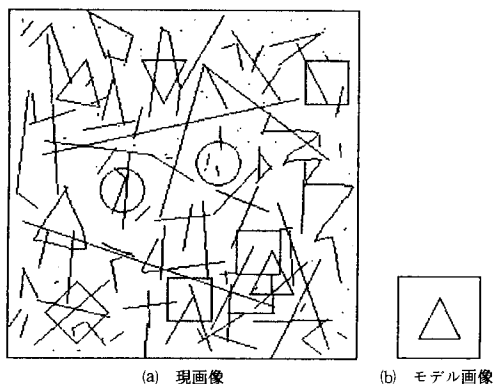


図15 図形探索シュミレーション画像
Fig.15 Pattern used in simulation

6. あとがき

本論文では遺伝的アルゴリズムに関する4つの研究について報告した。第1のJava言語によるプログラム開発ではビットストリング表現および順序数の順列表現を対象にした汎用的なツールであるGAワークベンチを設計し、各種の応用に適用できる手段を確立した。第2のオートマトンの自動生成への応用ではオートマトンの受理および却下の条件を与えて自動的にオー

トマトンを生成する方式の研究であり、自動システムの設計の機械化を目的にしたものである。第3の複数セールスマン巡回問題への適用では通常の単一セールスマン問題を拡張し、複数のセールスマンによる分担方式での最適巡回経路を求める問題への適用である。最後のパターン探索では雑音を含む信号から目的のパターンを探索するものであり、ロボットなどの物体認識などにも適用できると考えられる。これらの研究を通じて遺伝的アルゴリズムの適用に関する応用技術を獲得することができた。

なお、本研究は本学情報科学研究所の1998年度特定研究（重点課題）として実施したものである。

参 考 文 献

- 1) L. Davis (Ed): Handbook of Genetic Algorithms, Van Nostrand Reinhold, 385p (1991).
- 2) D.E. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Pub. Co., 412p (1989).
- 3) Z. Michalewicz: Genetic Algorithms + Data Structures = Evolution Programs (Second, Extended Edition), Springer-Verlag, 340p (1994).
- 4) S. Brave: Evolving Deterministic Finite Automata Using Cellular Encoding, Proc. of The First Annual Conference Genetic Programming, pp.39-44 (1996).
- 5) 北野編：遺伝的アルゴリズム, 産業図書, 328p (1993).
- 6) 北野編：遺伝的アルゴリズム2, 産業図書, 335p (1995).
- 7) 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).
- 8) 小野, 渡辺, 池田, 土性：最適問題への遺伝的アルゴリズムの適用, 福岡工業大学言語情報工学研究所彙報, 第7巻, pp.57-70 (1996).
- 9) 渡辺, 小野：遺伝的アルゴリズムによる二次元材切断方法の決定, 電気学会論文誌D, 117巻, 3号, pp.356-363 (1997).
- 10) 小野, 池田, 土性, 田代：遺伝的アルゴリズムによる最適問題の解法, 福岡工業大学言語情報工学研

究所彙報, 第8巻, pp.19-29 (1997).

11) 小野, 田代, 横井: 遺伝的アルゴリズムと最適問

題, 福岡工業大学情報科学研究所報, 第9巻,
pp.11-22 (1998).



(平成11年5月28日 受付)