

遺伝的アルゴリズムの各種問題への適用

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)
井上 豊 (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)

Yutaka INOUE (Communication and Computer Engineering,
Graduate School of Engineering)

Four studies on the application of genetic algorithms (GAs) to various problems are explained.

The first is on distributed parallel GAs, which consist of a supervisor and distributed GAs, referred to as Manager and Workers respectively. The Workers communicate with each other by immigrating individuals of other Workers through the Manager to improve their performances. These operations are performed in a host of distributed personal computers in parallel, programmed by the RMI of Java language.

The second is an application of GAs to a pattern search problem, in which a number of target patterns in a picture are searched simultaneously and identified. The Pareto GAs are applied to solve the problem as a multiple objective problem.

The third is to use GAs to design the arrangement of poles for electric power distribution in residential areas. By taking account of various regulations and limitations, the optimal location of poles are determined to make the number of required poles and the total length of electric wires minimum.

The last study is to stack optimally three-dimensional objects of various shapes and sizes, composed of rectangular boxes. The stacking is done so as to make the total height minimum within the fixed floor area.

Key words: *genetic algorithms, optimal problem, parallel GAs, pattern search, power distribution, optimal arrangement, optimal stacking*

1 ま え が き

遺伝的アルゴリズム（以下 GA: Genetic Algorithms）は各種の問題解決の手法として適していることから、今日まで種々の分野への適用を研究してきた。本論文では、これらの経験から GA を各種問題に適用する際に注意すべき点について述べた後、この 1 年間に行った主な 4 件の研究の成果について報告する。

第 1 のテーマは分散型並列 GA に関する研究である。本システムは複数の GA とこれらを統括する機構から成り立っている。これら複数の GA では独立に遺伝子操作を行い進化を図ると同時に、統括機構を通して相互に個体の交換を行い性能の改善を図っている。これら分散化した集団に対して複数のコンピュータによる並列演算が行われている。

第 2 のテーマは画像探索への GA の適用研究である。GA にパレート法を適用したパレート GA を採用し、画像中の複数図形の探索を多目的最適化問題として扱うことにより複数図形の同時探索が可能となった。

第 3 のテーマは配電用電柱の配置決定に対する GA の適用である。住宅地などでの配電用電柱の位置は地図を参照しつつ人手によって行われているが、本研究により、住宅地の環境から決まる設置可能場所と配電条件の両者を勘案しての配置位置決定の自動化が可能となった。

第 4 のテーマは 3 次元物体の最適立体配置問題への GA の適用である。2 次元の最適配置問題を 3 次元に発展させたもので、直方体の合成よりなる物体の最適立体配置問題を解くことができた。

2 GA による問題解決

GA とその応用に関する研究は本学情報科学研究所やその前身の言語情報工学研究所よりの研究費を得るなどして 8 年間にわたり進めてきた(文献:5-15)および表 1 参照)。これらの経験から GA を用いて各種問題を解く際に注意すべき事項について述べてみたい。

GA は汎用性の高い応用ソフトである。一般に各種問題を解く際に、それぞれの問題に適した手法を選定する必要がある。しかし、モデル化が困難な問題では適した方法が見つからない場合がある。それに反し GA は多くの問題に抵抗なく適用できる。GA の基本的な演算である選択、交叉、突然変異は標準化され、

表 1 GA の研究例

Table 1 Examples of studies on GAs

分類	方 式
切 断	二次元自由パターン切断最適化 ギロチン切断の最適化、棒材の最適切断
工 程	再構築可能な生産スケジュール 大学授業時間割作成
配 置	三次元物体最適立体配置、工場建物最適配置、配電用電柱の配置計画
系 統	電力系統最適運用計画、配送センター計画
画 像	パレート GA による図形探索
設 計	有限オートマトン自動生成、NN の構造決定
並列化	Java RMI による分散型並列 GA
ツール	Java によるツール Genova

ソフトウェアパッケージ化されており、その適用に際しては極言するとつぎの 2 点に注意をすればよい。

1. 遺伝子表現を決定すること。
2. 適合値を演算すること。

遺伝子表現に関しては生物の設計図が A,T,G,C と略称される 4 種の糖の組合せよりなる遺伝子で作られていることを考えるとき、遺伝子表現により各種問題を表現できる可能性が理解できる。GA の場合は対象が人工物であることから、より自由な遺伝子表現が可能であり、ビットストリング、実数ストリング、順序、および演算子など各種の表現が適用され、多様な問題に対処できるようになっている。

つぎに適合値については、ダーウイン主義の提唱する、環境が生物の世界における進化を主導したことと同様に、GA においても適合値を指標として最適状態に近づけることができる。適合値を求める方法は数式のみならず、各種の方式が考えられる。

一方、GA での問題としてしばしば指摘されるように、果たして最適状態に到達できるかに関しては、残念ながら GA の原理からこれを保証することはできないが、パラメータを種々変えてシミュレーションすることにより、ほぼ最適状態に到達することは可能であり、実用面からは十分な結果が得られる場合が多い。

3 分散型並列 GA

3.1 はじめに

GA は応用範囲の広い最適化手法であるが計算負荷が高いこと、ローカルミニマムに陥りやすいなどの問題がある。これらの問題に対し GA を分散並列化することで解決しようとする研究が数多く報告されている¹⁾。その方式の 1 つに離島モデル型の分散型並列 GA (Distributed Parallel GA) がある。個体集合を複数の部分個体集合に分け、それぞれの部分個体集合ごとに通常の遺伝的操を並列的に実行し計算時間の短縮、全体的な解の多様性の維持を図っている。

本研究¹⁴⁾では離島モデルの分散型並列 GA¹³⁾を参考にして個体の移住先の自由度を高めた。そして GA のパラメータを自動的に決定する機構をもつ GA の並列分散処理¹³⁾を取り入れることで、パラメータ設定の自動化、計算時間の短縮、解の品質向上を図った。また複数の個体集団を複数コンピュータに分散させることにより大型並列計算機を必要としないのも本システムの特徴である。

3.2 システム構成

本システムは、部分個体集合に対し遺伝的操をするアプリケーション Worker と、交換個体を蓄積する記憶データベース (記憶 DB) の管理と Worker のパラメータ設定を行うアプリケーション Manager の 2 種類のプログラムより構成されている。図 1 にその全体像を示す。

本システムの開発は Java (Java 2 SDK, Standard Edi-

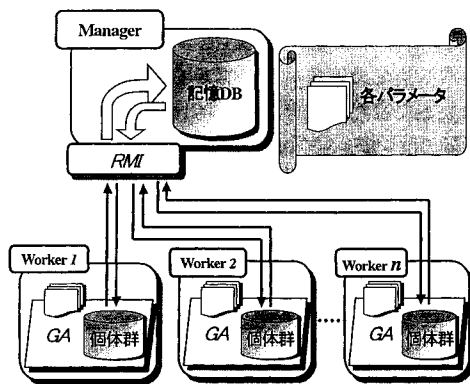


図 1 分散型並列 GA の構成

Fig. 1 Schematic diagram of distributed parallel GAs

tion V1.2.2) を用いて行った。その特徴は Java の実行可能な環境であれば OS やハードウェアに依存しないことである。ネットワークを介して個体の交換や Worker の制御を行う仕組みは、Java 2 のコア API に含まれ Java の分散オブジェクトモデルである RMI (Remote Method Invocation) を利用した。これにより TCP/IP ネットワーク上にある別の Java VM のメソッドを実行でき Manager および Worker 群を別のコンピュータに分離させることができた。

3.3 各プログラムの動作

3.3.1 Manager

図 2 に Manager のフローチャートを示す。Manager は各 Worker および記憶 DB の管理を行う。記憶 DB は可変長の配列であり、Worker から送られてきた個体をシミュレーションの期間中、記憶しておくのに使用する。Worker に返す個体もここからランク方式により選択する。このように各 Worker が記憶 DB を通じて個体の交換を行うことにより、解が収束してしまっても世代数に関係なく再探索が可能となる。また各 Worker のパラメータ値の設定や遺伝子の表現方法をグレイコードとするか否かの選択もこの Manager の役割である。なお、パラメータ値の設定は規定の範囲内でランダムに行われる。

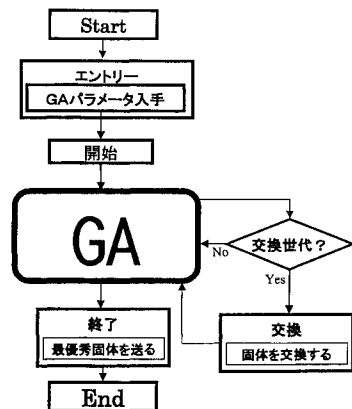


図 2 Manager のフローチャート

Fig. 2 Flow chart of Manager

3.3.2 Worker

図 3 に Worker のフローチャートを示す。Worker は通常の GA の演算および Manager を通じて他の Worker との間で個体の交換を行う。Worker ではプロ

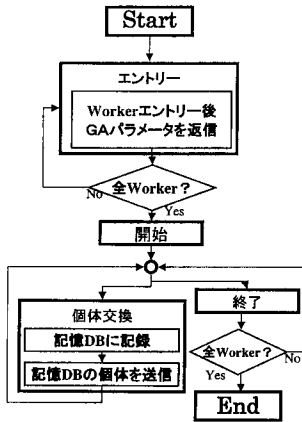


図3 Workerのフローチャート
Fig. 3 Flow chart of Workers

グラム開始時に Manager より GA のパラメータを受け取ったのち、通常の GA の処理を行い、一定世代ごとに Manager と非同期に一定個体数の個体交換を行う。この交換個体の選択はランク方式により行う。各 Worker には異なるパラメータをランダムに割り当てることにより、各 Worker がそれぞれユニークな進化を行い、全体として解の多様性が維持できるようにしている。

3.4 シミュレーションと考察

本方式の有効性を検証するために、テスト問題として Rastrigin 関数、Griewank 関数とナップザック問題の3種についてシミュレーションを実施し、シンプル GA のシミュレーション結果との比較を行った。ここではその内の Rastrigin 関数についてのシミュレーション結果について述べる。

3.4.1 Rastrigin 関数

テスト関数には次式に示す Rastrigin 関数⁴⁾を使用し、その最小化を対象に本方式の性能の検証を行った。この関数は次式で表され、GA の解の探索能力を測る代表的なテスト関数として広く使用されている。

$$F = 10N + \sum_{i=1}^N [x_i^2 - 10 \cos(2\pi x_i)] \quad (1)$$

ここで N は変数の数を表す。この関数は変数 x_i の値が全て 0 の時に最小値 0 を取り、その周辺に格子状に複数の準最適解を持っている。図4に2変数のとき

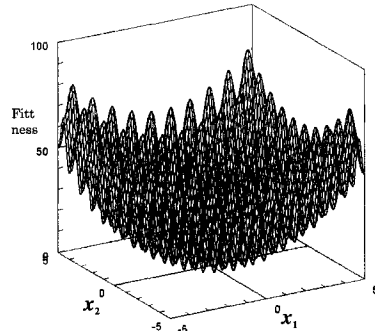


図4 Rastrigin 関数のグラフ
Fig. 4 Rastrigin function

の Rastrigin 関数のグラフを示す。シミュレーションでは変数の数 N を 10 とし、各変数 x_i はその範囲を $-5.12 \sim 5.12$ とし、それを 20bit のバイナリで表した。したがって、染色体の全長は 200bit となる。

3.4.2 シミュレーション条件

比較に使用したシンプル GA は集団の個体数を 1000 個体、交叉は 1 点交叉で交叉確率 1.0、突然変異確率は 0.005 とした。これらのパラメータ値は数十回の予備的シミュレーションを行った結果、最も成績の良かったものから採用した。

シミュレーションは Worker の数が 10, 25, 50 の 3 つの場合に対して行った。これは Worker の数が解の品質と演算時間にどのように影響するかを調査する目的からである。GA 部の計算量をシンプル GA と同じ条件にするため、全 Worker の個体数の合計がシンプル GA と等しい 1000 個体となるように設定した。したがって Worker の数が 10 のとき各 Worker の個体数は 100 個体とした。交叉点数は 1, 2, 4, 8 の中からランダムに選択し、交叉確率はシンプル GA と同じ値の 1.0 とした。また突然変異確率は 0.001 から 0.005 まで 0.001 の間隔でランダムに決定した。個体の交換は 50 世代ごとに行い、交換個体数は各 Worker の個体数の 10% とした。

3.4.3 結果と考察

図5に Worker の数を 10, 25, 50 としたときの適合値の世代による変化を示す。比較のためシンプル GA についても示す。これらの値は 30 回の実行結果の平均値である。Worker の数を増やすことにより性能が向上していることが分かる。これは各 Worker にそれぞれ

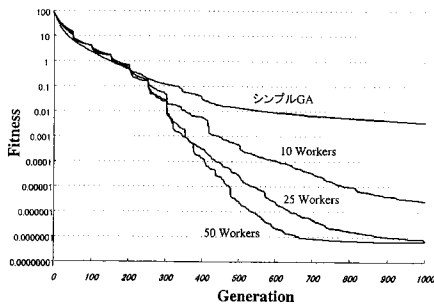


図5 適合値の推移
Fig.5 Convergence characteristics

表2 実行時間の比較

Table 2 Comparison of calculating tim

	計算時間 (単位: 秒)			
	PC 1台	PC 2台	PC 3台	PC 10台
シンプル GA	240			
10 Workers	190	98	70	38
25 Workers	192	101	73	40
50 Workers	195	102	75	42

異なるパラメータを与えることで、Worker 固有の進化を促して解全体の多様性を維持し、解の探索能力が飛躍的に向上した結果と考えられる。

また複数 PC への分散化による効果を調べるため、2, 3, 10 台の 4 種について計算時間の比較テストを行った。その結果を表 2 に示す。研究設備の関係で使用した PC がシミュレーションによって異なっており、3 台までのシミュレーションでは CPU: Pentium III, 800 MHz, OS: Windows2000 Professional のマシンを、10 台のシミュレーションでは CPU: Celeron, 466MHz, OS: Windows98 のマシンをそれぞれ使用した。表より台数の増加に応じて計算時間が減少していることが分かる。また 1 台の PC で複数の Worker を実行した本方式の方がシンプル GA よりも実行時間が短い結果が得られた。この違いは GA のソート時間の差によるものと思われる。GA においては適合値に応じて選択を行うため、次世代に移るとき全個体を適合値に関してソートしており、個体数の減少に応じてソート時間は急速に減少する。

3.5 まとめ

分散型並列システムでは Worker ごとにパラメータ

を自動的に互いに異なる値に設定することにより GA パラメータに各種の組み合わせが可能となり、解の多様性を維持することができ、探索能力の改善が図れた。また分散化により演算時間も短縮できることをシミュレーションの結果から検証できた。本方式では通常の GA と異なり、交叉確率や突然変異確率などのパラメータ値の設定が自動的に行われるので、その設定に思い悩む必要がないことも利点である。

4 パレート GA による図形探索

4.1 はじめに

近年、急速に発展してきた技術に画像処理がある。図形は文字や音などに比べて多くの情報を持っており、人はそれを一瞬のうちに取り込み、解析し、理解することができる。しかし人間の画像処理能力には限界があるので、それを支える必要があり、画像処理技術が求められてきた。本研究では画像処理のひとつである図形探索を GA を用いて行った。探索図形が 1 個の場合の研究⁶⁾を進展させ、探索する目標が複数パターンであり、それらが探索領域内に多数存在する場合を対象に、GA と多目的最適化手法を組み合わせたパレート GA (Pareto GA)²⁾により並列探索する方式を研究した。

4.2 多目的最適化とパレート GA

多目的最適化問題では、複数の目的関数を同時に最適にすることを目指す。一般にそれらの値を同時に改善できないというトレードオフの関係にあり、それをいかにバランスさせるかが重要となる。GA では通常、複数の評価項目がある場合、1 つの目的関数に変換して取り扱うが、多目的最適化に適した方式にするのが望ましい。これらを考慮し、本研究では複数パターンの図形を含む画像の中から目的の図形を探索する問題を多目的最適化問題として扱い、その手法のひとつであるパレート最適化方式と GA を組み合わせた方式パレート GA²⁾を適用した。

4.3 システム構成

システム構成を図 6 に示す。本システムは遺伝的アルゴリズムを多目的最適化用に改良したパレート GA 部 (PGA) と、目的図形のパターンマッチと数値変換、ボカシ処理、および局所探索の処理を行うパターン探索部 (PSP) で構成している。PGA 部より与えられた

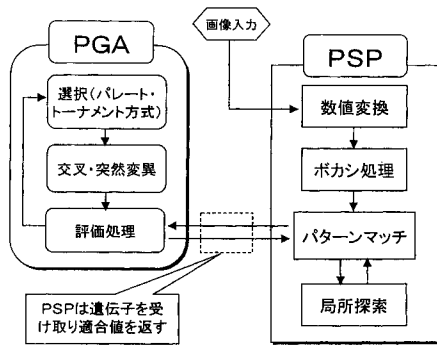
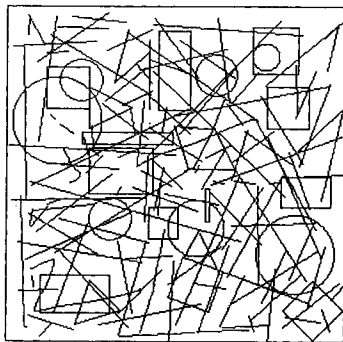


図6 パレート GA のシステム構成
 Fig. 6 Schematic diagram of Pareto GAs



原画像



モデル図形1
 (三角モデル)



モデル図形2
 (四角モデル)

図7 探索に用いる画像例
 Fig. 7 An example of picture

遺伝子をもとに PSP 部でモデル図形との類似度を計算し評価値 (適合値) を求め、これを PGA 部に返す。PGA 部ではこの評価値に従って選択・交叉などの遺伝的操作を行ない、新しく作り出した遺伝子を PSP 部に渡す。この処理を繰り返すことにより、各目的図形の位置情報を求める。

図7に示すように、図形探索の対象となる画像は複数の図形やノイズを含んだ画像であり、これを原画像と呼び、探索領域となる。また、原画像中から探し出す任意の図形をモデル図形、モデル図形のみを含んだ

画像をモデル画像とそれぞれ称している。

4.4 パレート GA の適用

4.4.1 遺伝子表現

遺伝子には探索画像の X, Y 座標, 回転角度, およびモデル情報の 4 つを記述することにし, これをビットストリングにて表現した。ここでモデル情報は複数パターンの中のどのモデル図形で探索を行うかを示すのに用いる。

4.4.2 選択

複数の目的関数がトレードオフの関係にあることに留意し, パレート最適解を求めるためにパレートトーナメント法を使用している。これはトーナメント選択を基本として, これに適応度のシェアリングを組み合わせたものであり, シェアリングは GA により収束して偏ってしまう遺伝子を分散させる役割を果たしている。

4.4.3 交叉・突然変異

交叉はランダムに選んだ 2 点間で行う 2 点交叉を使用している。突然変異はその確率を動的に変化させる可変突然変異とし, 初期状態では確率を大きく, 世代が進み解が収束するに従って小さくなるようにしている。

4.5 パターン探索

4.5.1 前処理

パターン探索のための前処理として, 画像情報の数値変換と画像に濃淡を付けるボカシ処理を行っている。数値変換では画像ファイルを探査しやすい形式にするために二次元配列へのデータの変換・格納を行う。ボカシ処理は, 原画像全体に濃淡を与えることにより適合値に幅を持たせ, 探索の効率化を図っている。

4.5.2 パターン比較

パターン比較は, 数値変換によって配列に格納した原画像データとモデル画像データを, PGA の遺伝子より与えた座標を基準として, 重ね合わせて比較することにより行い, その類似度を求めて適合値としている。具体的にはボカシ処理で使用するボカシの階層 (深さ) m , モデル図形を形成するドットの総数 n , 画像同士の重ね合わせた部分の類似度 (探索点における値

の総和) $\sum a_i$ を用いて次式により求める.

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

4.5.3 局所探索

GA では交叉を中心として新しい探索点を生成するため、解の近傍での細かな探索に適さない。そこで遺伝的操作により所定誤差内の解が得られたら、一時的に GA による探索から線形探索に切り替えて、その座標の上下左右を局所的に探索することで最適解を素早く探し出すようにしている。

4.6 シミュレーション

先に示した図7を用いてパターン探索のシミュレーションを行った。原画像の領域は256×256ドット、モデル画像は64×64の大きさとした。なお、モデル図形として三角と四角の2種類を用意し、原画像中にランダムに三角モデルを3個、四角モデルを5個配置した。個体数を300とし、交叉確率を0.6、突然変異率を0.7から0.1への可変とした。シミュレーションの結果、600世代で図8に示す結果が得られた。探索結果は図のようにモデルごとに分けて表示されており、正しく探索が行われている。

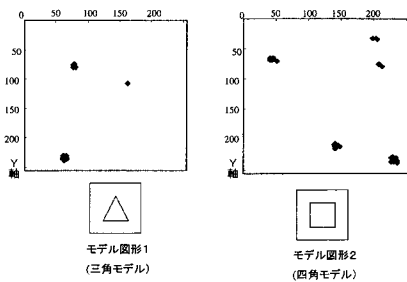


図8 パターン探索のシミュレーション結果

Fig. 8 Result of simulation for pattern search

4.7 ま と め

本研究ではパレート GA を用いての図形探索を行った。シミュレーションでは2種類のモデルパターンを同時に探索し、原画像中からすべて見つけ出すことができた。今後の課題として、より実用的な画像(カラー・濃淡画像)の図形探索への適用である。

5 遺伝的アルゴリズムによる配電用電柱の配置決定

5.1 研究目的

従来、住居や工場などへの電力供給用の配電用電柱の配置は、その業務に精通した作業者が該当地域の地図などより経験的に決定しているため、作業者によって判断が異なり統一的な作業が行い難く、業務経験の少ない作業者には判断が難しいなどの問題があった。また、1本ずつ電柱の配置を決める作業は効率が良いとは言いがたく、作業者のレベルによっては余分な電柱を配置する可能性もある。

本研究⁹⁾は電柱の配置を GA によって求めることにより作業者の負担を軽減させ無駄の少ない配置を行うことを目的としている。

5.2 研究のアプローチ

本研究は配電用変電所から高压需要家または低压需要家に電力を供給するために必要な電柱の位置を対象にした。ここで、需要家とは電気を使用する建物あるいは一般家庭のことを指す。

まず、電気事業法に基づく省令である電気設備技術基準によって定められている規定を考慮し、下記の条件を設定した。

- ・ 高压配電、低压配電の区別は行わない。
- ・ 引込線は最寄りの電柱から引く。
- ・ 電柱の種類と経間における規定は考慮しない。

それとは別に、電柱設置工事の観点から以下の制約条件を設定した。

- (1) 需要家の周囲に少なくとも1本の電柱を配置する。
- (2) 既に設置してある電柱はそのまま利用する。
- (3) 電柱は設置可能な場所に配置する。
- (4) 全ての需要家へ電力を供給する。
- (5) 電線路は施設可能な箇所を通る。
- (6) 電線路が交差および曲がる箇所には必ず電柱を設置する。

GA を適用するに際して、解空間をメッシュで表しそのメッシュ上に住居や道路などが存在するものと考えた。この1つのメッシュの大きさは住居1軒程度とし、1つのメッシュに最大で1本の電柱が設置できるものとした。

5.3 システムの実現

まず、以下の点を考慮して解空間となる地図データを作成する。

- ・各メッシュは正方形とする。
- ・需要家の位置と電柱および電線路の設置可能場所を地図上に設定する。
- ・既設の電柱がある場合は、その位置を地図上に設定する。

こうして作成した地図データを基に以下の3ステップにより、制約条件を満たし、かつ電柱の設置本数が最も少ない配置を求める。

Step 1：電柱配置アルゴリズムの適用

ここでの目的は制約条件(1)~(4)を満たし、より少ない電柱を配置することであり、アルゴリズムはGAが中心となる。地図データを基に各需要家に電力を供給する電柱の設置場所を遺伝子で表現し、選択・交叉・突然変異等の遺伝子操作を行う。遺伝子情報に従って電柱の配置を評価し、収束した解より最適な電柱の配置を求める。

Step 2：電線路決定アルゴリズムの適用

電柱は需要家への電力の供給点のほか、電線路の途中にも電線を支持するものが必要であるので、まず制約条件(5)を考慮して電線路を決定する。電線路の全長は出来るだけ短いのが望ましいので、Step 1で求めた電柱を節点とした木を考え、なるべく電線路が交差せず経路が短くなるような全域木を求めて電線路を決定する。それによって追加配置すべき箇所を求める。

Step 3：配置補正アルゴリズムの適用

本アルゴリズムはStep 2により決定した電線路の交差および曲がる箇所に電柱を配置することを目的とする。それにより制約条件(6)を満たすことができる。また、ここで電線路の交差箇所に電柱を配置することによって不要な電柱や電線路が発生した場合はそれらを除去する。

5.4 電柱配置アルゴリズム

GAを適用するに際して、遺伝子表現には以下のように各需要家の位置を基準とした電柱の位置を用いた。前述の制約条件(1)を考慮し、各需要家の周囲8箇所に最低1本の電柱を設置することにし、電柱の位置を図9に示すように各需要家の位置を基準にした3ビットのバイナリ・ビットストリングで表現した。もちろん、複数の需要家が1本の電柱を共用することも可能である。

(000) ₂ =(0) ₁₀	(001) ₂ =(1) ₁₀	(010) ₂ =(2) ₁₀
(011) ₂ =(3) ₁₀	需要家	(100) ₂ =(4) ₁₀
(101) ₂ =(5) ₁₀	(110) ₂ =(6) ₁₀	(111) ₂ =(7) ₁₀

図9 電柱配置を表す遺伝子

Fig. 9 Genes expressing arrangement of utility pole

適合値は電柱の数が少ないほど高くなるよう定め、選択方式はエリート保存を加えたルーレットホイール方式を採用した。交叉方式には2点交叉を採用し、交叉点は親の形質の継承を考慮して、各電柱間となるよう3ビット間隔で選定した。突然変異はビット反転により行う。

5.5 シミュレーションと考察

本システムの有効性を検証するためにシミュレーションを行った。

探索空間を900メッシュ(30×30)とし、そこに135軒の需要家が存在するものとした。従ってビットストリング長は405となり、パラメータ値は交叉確率を0.6、突然変異確率を0.001とした。

Step 2およびStep 3の各段階にてシミュレーションで得られた電柱数と電線路全長を表3に示す。Step 3の配置補正処理後に処理前と比べて電柱の本数が増えているのは、Step 3にて電線路交差箇所への電柱追加が行われたことによるものであり、電線路の長さが短くなっているのは不要な電線路が除去されたためである。

表3 電柱数と電線路長

Table 3 Number of electric poles and length of electric wires

	電柱の本数	電線路の全長
Step 2の結果	60	163
Step 3の結果	68	154

つぎに、解の収束状況と得られた電柱の配置を示す。図10に示すように約2,400世代で収束しており、高い評価値を持つ個体を得ている。得られた電柱配置を図11に示す。図中の記号の説明を図12に示す。電柱の配置はすべての制約条件を満足している。

以上に述べたようにGAの解を基本に電線路を追加

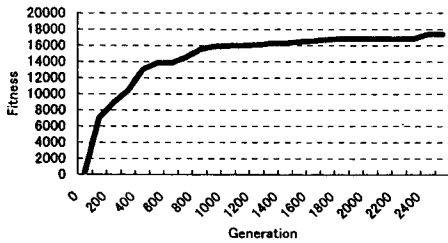


図10 評価値の収束状況

Fig. 10 Convergence characteristics

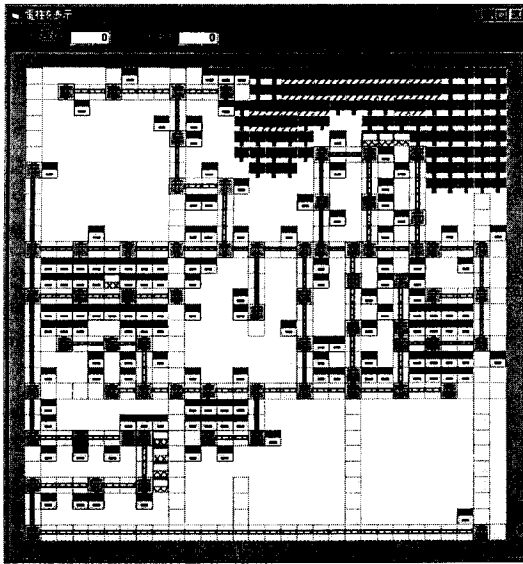


図11 シミュレーションで得られた電柱配置

Fig. 11 Arrangement of poles gained by simulation

記号	記号が示す意味
	需要家
	電柱
	電線路
	電柱設置可能個所
	電線路施設可能個所
それ以外の図形	電柱設置不可個所 電線路施設不可個所

図12 記号の説明

Fig. 12 Explanation of symbols

し最適化することにより、電柱配置作業の自動化とその最適化が可能であることが確認できた。

6 立体配置決定

6.1 問題設定

本研究の目的は図13に示すような複数の部品を任意の大きさの領域に詰める際に、できるだけ隙間のない配置を GA を用いて決定することである。

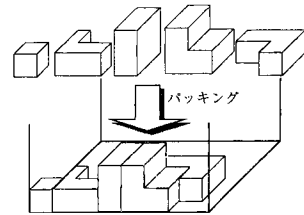


図13 立体配置問題

Fig. 13 Stacking problem

GA によりつぎの3点を考慮した部品の配置を決定する。

- ・ 部品同士を重なりなく立体的に配置する。
- ・ 部品の回転を考慮する。
- ・ 配置した全体の高さを最小にする。

なお、各部品の形態は直方体の集合とし、回転は水平軸および垂直軸に対し90度単位の計24種を行えるものとした。

6.2 システム構成

システム全体は図14に示すように GA、配置決定アルゴリズム (Parts Arranging Algorithms : PAA) および各種データファイルで構成している。その動作は、まず GA によって遺伝子集団が生成され、選択、交叉、突然変異などの遺伝的操作を行う。GA での評価処理後、PAA に遺伝子が渡され、その情報を基に部品の配置を行いその全体の高さを求める。その高さを遺伝子の評価値として GA に返し、GA はこの値が小さくなるように遺伝的操作を繰り返すことで、最小な高さの配置方法を探索する。

データファイルには詰める領域の大きさやその状態を記述した領域ファイル、部品の形状を記述した部品ファイルがある。PAA はこれらのファイルを参照しながら配置を行う。

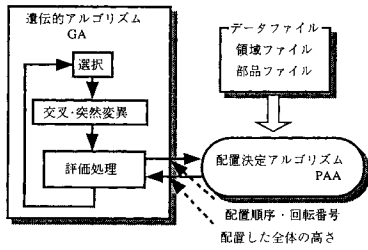


図14 立体配置システムの構成

Fig. 14 Schematic diagram of stacking system

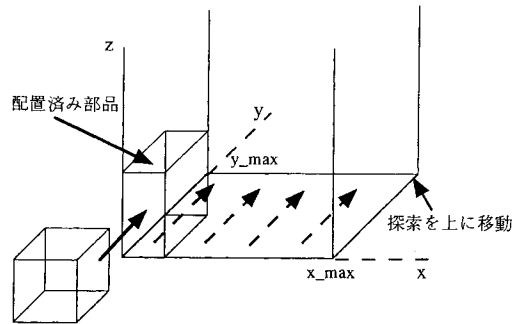


図15 PAAによる配置

Fig. 15 Arrangement by PAA

6.3 GAの適用

6.3.1 遺伝子表現

遺伝子は各部品の配置を行う順序を表す順序遺伝子と部品の回転を表す回転遺伝子との2つの部分より成り立っている。順序遺伝子は各部品に整数の通し番号をつけ、その番号を配置する順序に従って並べたものとし、順序をそのまま表すパス表現とした。回転遺伝子は、その部品を回転させた向きを表した番号とし、90度単位の24種の回転を表す0～23の整数とした。

6.3.2 選択・交叉・突然変異操作及び目的関数

選択は適合値の大きさの順位に比例して個体を選択するランク方式を、交叉はパス表現の交叉法である循環交叉(CX)法を採用した。突然変異は順序遺伝子と回転遺伝子のそれぞれに対して用意した。順序遺伝子の突然変異はランダムに2点を選びその要素を交換する位置交換法を採用し、対応する回転遺伝子の要素も同時に入れ替える。また、回転遺伝子の突然変異はランダムに選んだ要素に対し0～23の整数区間にて変化させる方法を用いた。目的関数には以下で説明するPAAの評価値を採用した。

6.4 配置決定アルゴリズム(PAA)

PAAは各部品を遺伝子情報を基になるべく隙間なく配置を行うアルゴリズムである。まずGAより渡された順序・回転遺伝子に従い部品群より順次1つずつ部品を取り出し回転を行った後、図15に示すように配置可能な位置を探索する。その動作を以下に段階を追って説明する。

1. 各部品ごとに左下端より探索を開始する。x座標およびz座標方向の移動量を記憶する変数 x_{mov_min} および z_{mov_min} を予め定めた最大値に初期化する。これらはx方向及びz方向のジャ

ンプ量の決定に用いる。

2. 配置済み部品との重なりの有無をチェックする。重なりがある場合はその部品との重なりをなくすために必要な移動量 x_{mov} , y_{mov} および z_{mov} を計算しステップ3に進む。重なりがない場合はそこに配置できるのでステップ7に進む。
3. $x_{mov_min} = \min(x_{mov_min}, x_{mov})$, $z_{mov_min} = \min(z_{mov_min}, z_{mov})$ を計算する。ここで $\min(*, *)$ は最小値を求める演算を示す。
4. y方向に y_{mov} だけ移動しステップ2に戻る。この時、y座標値が y_{max} に到達した場合はステップ5に進む。
5. x方向に x_{mov_min} だけ移動し、 x_{mov_min} を最大値に初期化してステップ2に戻る。この時、x座標が x_{max} に到達した場合はステップ6に進む。
6. z方向に z_{mov_min} 移動し、 z_{mov_min} を最大値に初期化してステップ2に戻る。
7. この位置に部品を配置する。

以上の動作を全部品が配置されるまで繰り返し、全体の高さを求め、その値をGAに評価値として返す。

6.5 シミュレーション

本システムを検証するために、すでに最適解の判っている16個の部品を使ってシミュレーションを行った。個体数を1000、交叉確率を1.0、順序遺伝子の突然変異確率を0.15、回転遺伝子の突然変異確率を0.15とした。シミュレーションで得られた世代による配置の変化を図16に示す。初期世代では隙間が多いが、世代が進むにつれて高い密度で配置されていく様子が確認できる。

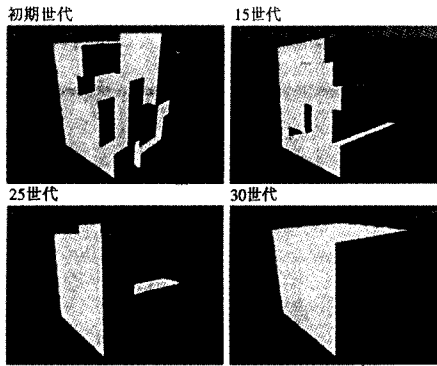


図16 立体配置のシミュレーション結果
Fig. 16 Results of simulation of stacking operation

6.6 ま と め

立体配置問題にGAを用いた本方式はシミュレーションにより有効であることが確認できた。本システムの応用としてプラント・レイアウトの設計などの実用問題への適用が考えられる。

7 あとがき

本論文では4分野への遺伝的アルゴリズムの適用研究の成果について報告した。第1の分散型並列GAでは収束性能の改善と演算性能の向上のために行った研究について述べた。これによりPCにより十分に高速でかつ収束性のよい方式が得られた。第2のパレートGAによる図形探索は複数図形探索を多目的最適化問題として扱いパレート法を取り入れたGAによりこれが実現できることが検証できた。第3の配電用電柱の配置決定に対するGAの適用の研究では従来、人手に行っていた配電用電柱の配置の決定がコンピュータにより可能であることが判明した。第4の3次元の物体の立体配置へのGA適用の研究は機器の3次元の最適配置決定などへの適用の可能性を見いだすことができた。

なお、本研究は本学情報科学研究所短期研究として実施したものである。

参 考 文 献

- 1) Cantu-Paz,E: A Survey of Parallel Genetic Algorithms, Revised version of IlliGAL Report No.97003, 28p (1997).
- 2) Horn,J and Nafpliotis, N: Multiobjective Optimization Using The Niched Pareto Genetic Algorithm, IlliGAL Report No.93005, p.32 (1993).
- 3) Tanese,R.: Distributed genetic algorithms, Proc.of Third International Conference on Genetic Algorithms, pp.434-439 (1989).
- 4) Whitley D. et al.: Building better test functions, Proc.of Sixth International Conference on Genetic Algorithms, pp.239-246, (1995).
- 5) 井上, 小野: 遺伝的アルゴリズムによる配電用電柱の配置決定, 火の国情報シンポジウム2001, 情報処理学会, pp.345-352 (2001).
- 6) 河村, 小野: 遺伝的アルゴリズムを用いた図形探索, SICE九州支部学術講演会予稿集, pp.335-336, (1999).
- 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, 同上, 第7巻, pp.57-70(1996).
- 9) 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, 同上, 第8巻, pp.19-29(1997).
- 10) 小野, 田代, 横井: 伝統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所報, 第9巻, pp.11-22 (1998).
- 11) 小野, 横井, 山田, 河村: 遺伝的アルゴリズムの応用, 同上, 第10巻, pp.49-58 (1999).
- 12) 電気技術研究会: 電気設備技術基準ハンドブック, 電気書院, 153p (1998).
- 13) 三木ほか: 環境分散型並列アルゴリズム, 電子情報通信学会信学技法, pp.87-94 (1999).
- 14) 山田, 小野: ネットワーク機能を用いた分散型並列遺伝的アルゴリズム, SICE学術講演会, 309A-4, (2000).
- 15) 横井, 小野: 遺伝的アルゴリズムによる有限オートマトン設計の自動化, 電気学会論文誌C, 121巻, 6号, pp.1096-1101 (2001).