

巡回セールスマン問題を遺伝的アルゴリズムにより解く場合の 致死遺伝子対策へのCプログラミングによる 漸進的機能改善施策

出 貝 賢 一* 高 橋 良 英**

An Evolutionary C Programming of Genetic Algorithms for Solving the Traveling Salesman Problem to Improve the Functionality of the Crossover Operation

Kenichi DEGAI* and Ryoei TAKAHASHI**

Abstract

One of the problems when solving the traveling salesman problem through genetic algorithms (GA) is to get an approximation of the optimum and stable solution that is obtained with the least number of crossover operations without yielding lethal genes. Our experiments show that on early stage of generations an improved EX-operator that selects the nearest town from the current visiting town successively is available for selecting the last path with the minimum required distance, and that after generations the SXX-operator that generates individuals in the next generation by performing a crossover operation on a pair of the selected parents that have the local optimum sub-paths is available. In this paper, a changing crossover-operations method (CXO) which can flexibly substitute the current crossover operator for another suitable crossover operator at any time according to fitness values of individual strings that comprise the population is proposed. With this method we can obtain the optimum solution more efficiently rather than with another single operator. Validity of the CXO-method, which changes crossover operators from the improved-EX to the SXX in our experiment, is experimentally verified through C-programming by using data of 15 cities.

Key words: Traveling salesman problem, genetic algorithms, lethal genes, improved-EX, SXX, a method of changing crossover operators (CXO), C language

1. はじめに

巡回セールスマン問題 (TSP) を遺伝的アルゴリズムで解く時の課題は、遺伝子交叉により意味のない致死遺伝子 (lethal genes) が生じさせないような対策を講じつつ、できるだけ少ない遺伝子交叉回数で安定した最適解の近似を得

ることである [1] [2] [7]。遺伝子交叉回数は、致死遺伝子対策に依存する。致死遺伝子 [1] とは、解候補である巡回路の遺伝子型を巡回する順番で並べられた都市として表現した時に、遺伝子交叉によって生じる同一都市を二回以上巡回するような意味のない死んだ遺伝子のことであり、巡回セールスマン問題を遺伝的アルゴリズムで解こうとした場合に必ず生じる問題である。致死遺伝子対策として、(1) 訪れる順番に並べた n 個の都市を表現する遺伝子列 (染色体) を n 個の文字の順列と考え、順列に対する互換

平成 16 年 12 月 17 日受理

* 大学院工学研究科電気電子工学専攻博士前期
課程・1 年

** システム情報工学科・教授

や表現方法に対して工夫をこらすことで致死遺伝子が生じないような遺伝子交叉（置換）を実現した Grefenstette 法 [4] [18], PMX 法 [7] [8], CX 法 [7] [9], OX 法 [9] [10], (2) 隣接都市間の所要距離等枝のつながりに着目した EX 法 [11], SXX 法 [3] [12] [13], EXX 法 [19] が発見されている。これまでの我々の実験結果によれば、比較的初期の世代では、所要距離の最も短い都市を親の隣接都市リストの中から次々に選択していく改良 EX 法が適応度の高い個体を生成する傾向があり有効であること、世代が経るにつれて、ある都市からある都市まで部分的に最適な経路となっている親の部分枝と部分枝を遺伝子交叉させて次世代の個体を生成する SXX 法が集団の適用度を向上させるのに有効であることがわかっている。このため、本研究では、都市の空間的な位置関係によって遺伝子交叉方法と交替時期を柔軟に選択可能な、遺伝的アルゴリズムの実現方法について検討することとした。当方式の有効性を、15 都市を対象とした小規模 C プログラミング実験により検証したので報告する。

2. 巡回セールスマン問題

あるセールスマンが、 N 個の都市を 1 回ずつ巡回して、出発都市に戻ってくる経路のうち所要距離（または所要費用、所要時間）が最小となる経路を求める問題を巡回セールスマン問題という。但しどの 2 都市間も道で結ばれており、その間の所要距離はわかっている。逆順を同一経路と見なして全経路数は $(N-1)!/2$ 通りと計算できる。 $N!$ は Stirling の公式により、 $\sqrt{2\pi} \exp(-N) * N^{(N+1/2)}$ と近似されるが、この式は経路の選択に指数オーダー以上の時間がかかることを示している。表 1 は、1 秒間に約 100 万命令 (2^{20}) 実行するコンピュータが、全ての巡回経路について所要時間を調べ、それを横並びに比較することで巡回セールスマン問題の最適解を得

ようとした場合にかかる時間を示している。この表に示すように都市数が 18 以上になると、計算機で 10 年以上かかり実効上解くことが困難な問題であることがわかる。このような意味で巡回セールスマン問題は解くアルゴリズムはあるものの、実効上計算機でもその解を解くことが困難な NP クラスに属する問題であることが知られている [14]。更に、NP クラスに属する問題の中でも最も難しい NP 完全クラスに属する問題であることが知られている [1]。NP クラスに属する問題のように、問題を解く手掛かりが見つからない場合は、その最適解の近似を得る手法として、遺伝的アルゴリズムによる手法が有効であることが知られており [7]、本検討でも遺伝的アルゴリズムにより巡回セールスマン問題を解く手法を研究することとした。

表 1. 網羅的（横型）探索により巡回セールスマン問題の解を得るための時間

| 都市数 | 13 | 15 | 17 | 19 | 21 |
|---------|---------|--------|--------|--------|-------|
| 所要時間(年) | 0.00001 | 0.0013 | 0.3147 | 96.359 | 36634 |

3. 遺伝的アルゴリズムによる巡回セールスマン問題の解法の問題点と対策

3.1 遺伝的アルゴリズムの考え方

遺伝的アルゴリズム (Genetic Algorithm: 略して GA) は、J. Holland が、1970 代に提案した、生命科学の情報処理への応用理論である [3] [4] [5] [6]。その目標は、「自然界に見られる個体の環境への適応現象は、染色体（遺伝子列）の「選択」と、「交叉」、「突然変異」といった遺伝的操作により説明される。このメカニズムを、コンピュータシステムに応用すること」である。弟子の DeJong 等が、標準関数の最大値を求める等、GA を、最適化問題を解く有効な手法として広めた。GA が最適化問題を解くのに有効な手法であることは、積み木仮説（「定義長が短く、オーダーが小さな、環境への適応度の高い個体は他個体と「遺伝子交叉」することにより、

適応度の高い個体を生成できる」)に基づいている。

3.2 遺伝的アルゴリズム

3.2.1 個体設計と適応度計算

巡回セールスマン問題における、「個体」の表現法と「適応度」の計算方法を以下に整理する。

(1) 個体の遺伝子型と表現型

個体は巡回セールスマン問題の解候補となる巡回路である。

●**遺伝子型**：各遺伝子は1バイトで構成され、訪れる都市の番号(1以上の整数)をその値として持つ。訪れる順番に都市番号が遺伝子に付与されている。

●**表現型**：個体は都市を巡回する経路を示す。巡回路は訪問する順番で並べられた都市列で表現される。

図1で示される遺伝子型は“1”から“a”で表現される10個の都市の場合の遺伝子型を示し、その表現型は図2に示される各都市を巡回する巡回路となっている [7]。

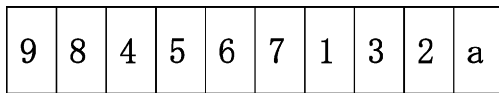


図1. 遺伝子型

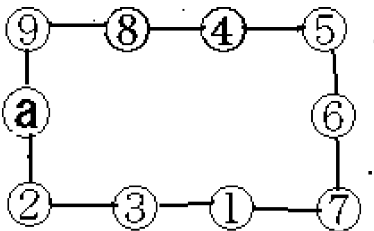


図2. 表現型

(2) 個体の適応度：

個体の適応度は、巡回路 α の相対的経路長を示す式であり、以下で定義される。

巡回路 α の適応度

= 各都市間の距離の総和 γ / 経路 α の長さ β

巡回セールスマン問題における各都市は、ユークリッド空間上で位置づけられていると仮定する。すると、上式の分子は都市と都市を結ぶ全ての辺の長さの総和 γ であり、巡回する都市の位置関係のみで決まる選択する経路に依存しない固定パラメータである。また、分母は選択した経路の経路長 β である。経路長 β は経路 α を構成する各都市 X_i と X_j 間の距離の総和である。適応度は、全距離 γ が経路長 β の何倍なのかを示しており、経路長が短い程、適応度が大きくなる。

3.2.2 遺伝的アルゴリズムの概要

以下は、本巡回セールスマン問題を解く遺伝的アルゴリズム(GA)の概要である [1] [6] [7]。

(1) **初期設定**・充分な個体を遺伝子プールに確率的に発生させる。一括型世代交代方式では、次期世代生成用に予備遺伝子プールを準備しておく。即時世代交代方式では1プールのみ用意する。

(2) **親の選択**・遺伝子プールの中から、ある選択基準で親を2個体決める。遺伝子プールの個体の適応度を順番に足し合わせていき、適応度総和に対してある割合を満たした時の個体を親として選択している。これはルーレット方式である。この割合は試行の度に变化する乱数で決める。

(3) **遺伝的操作**・選択した親(染色体(個体の遺伝子型))の間で遺伝子交叉(Crossover)や突然変異(Mutation)といった遺伝的操作を行って子供2個体を生成する。ただし、これらの操作を行う場合、それぞれ適当な確率で行うようにする。

① **遺伝子交叉**：遺伝子の部分的な交換を行う。交叉点が1点である一点交叉、交叉点が2点である二点交叉がある。順列の互換に着目する方法では2点交叉を前提としている。

② **突然変異**：ある遺伝子の on, off 状態を

反転させる。巡回セールスマン問題では、ある遺伝子のみの値を変更させる突然変異の考え方は TSP では致死遺伝子を生じさせるので、本実験では突然変異を実現しないこととした。

(4) 一括型世代交代方式では、新たに産まれた個体を予備遺伝子プールへ格納する。この予備遺伝子プールが、親の個体数と同じになるまで、(2) から繰り返す。同じになったら (5) へ行く。即時型世代交代方式では、新たに産まれた個体を両親と即時に入れ替え (2) から繰り返す。新たに産まれた個体数が親の個体数と同じになったら (6) へ行く。

(5) 予備遺伝子プールを現遺伝子プールとする。このとき旧世代はすべて、新世代に置き代わる。

(6) 終了処理 (集団の評価) ・ ・ 終了条件を調べて、終了していなければ、(2) から繰り返す。終了条件は、通常、最大世代数等で与える。

3.3 致死遺伝子問題と対策

3.3.1 致死遺伝子

遺伝的アルゴリズムを適用して巡回セールスマン問題を解く場合、解候補である巡回路の遺伝子型を巡回する順番で並べられた都市として表現した時に、遺伝子交叉によって同一都市を二回巡回するような遺伝子が生じる。こうして生じた遺伝子を致死遺伝子と呼ぶ。図3の遺伝

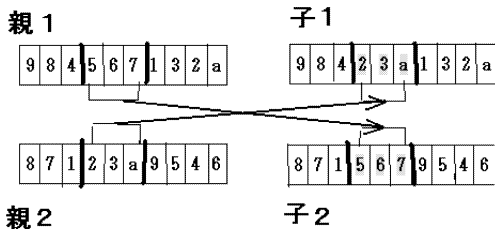


図3. 遺伝子交叉による致死遺伝子

子交叉では、子1では“2”，“3”，“a”，子2では“5”，“6”，“7”の遺伝子が重複している。

3.3.2 対策

致死遺伝子が生じないように、二点交叉法における中間の交叉部分のみを部分的に入れ替え残りの部分については矛盾のないようにもう一方の親の対応する都市で入れ替える PMX 法や、親の巡回路における都市の訪問順番はできるだけ一致させるよう、訪問する都市集合が部分的に一致する部分巡回路を見つけ、正と逆の両方向交換を行う SXX 法等様々な手法が提案されている。致死遺伝子対策 [2] を表2にまとめる。表2の括弧 () では遺伝子対策毎の新規規模、流用規模、サイクロマチック数を順に示している。規模はソースコードの行数で測定している。

致死遺伝子対策を以下の2つに分類する。

(1) 順列の置換やその表現方法に着目する方法

染色体を1から n 番目の都市の並べ方 (順列) と考え、遺伝子交叉を別の順列への置換と捉える。置換を互換の積として表現する。

- Grefenstette 法
- PMX 法
- OX 法
- CX 法
- 平野 OX 法 ・ ・ OX 法に改良を加えた方式

(2) 枝の性質に着目する方法

隣接する都市間の距離に関する情報や、既に生成された都市間のつながりに着目する方法である。

- 改良 EX 法
- SXX 法
- EXX 法
- 平野 EX 法 ・ ・ 改良 EX 法を簡易化した方法

3.3.3 致死遺伝子対策実現上の留意点

以下に各致死遺伝子対策の特記事項をまとめる。

(1) SXX 法

SXX 法では部分巡回路を選択する効率を向上させるため、どの部分巡回路を基準の部分巡回路として選択するかは、その長さや起点に関して一様乱数を発生させて確率的に決定する方式とした。当方式では、最適な部分巡回路を決定するのに、都市数 n として、網羅的検索では n^3 オーダかかっていた検索時間を、都市数 n^1 のオーダに削減している。

(2) 改良 EX 法

本研究では、最適解への収束効率を向上させるため、親の巡回路中で隣接する都市リストの中で、現在訪れている都市から最も距離の短い都市を、次に訪れる都市として選択する方式としている。

表 2. 巡回セールスマン問題に対する致死遺伝子対策

■: 提案方式

| 項番 | 致死遺伝子対策 | 説明 |
|----|---|--|
| 1 | 平野 EX 法 [1] (0.94, 17)* | 出発都市を任意に定める。2つの親の染色体を前向き方向に探索して、出発都市の次に現われる都市を探す。次に現われる都市の中で距離の短い方を、二番目に訪れる都市として選択する。同様に、これまで訪れていない都市のうち、現在訪問中の都市の次に現われる都市の中で距離の短い方を、次に訪れる都市として選択する。 |
| 2 | Grefenstette 法 [4] [18] (374, 0, 40) | 全都市の基準順位を決めておく。ある巡回路において次に訪れる都市がその基準順位の中で何番目に位置するかを親遺伝子の値として設定する。既に訪れた都市は基準都市順位から次々に削除していき、その新たな基準都市順位の中で次に訪れる都市が何番目を親遺伝子の値として設定する。本検討では、他の遺伝子交叉方式に合わせ、上記のように表現された両親の染色体に対して二点交叉を行う方式とする。 |
| 3 | 平野 OX 法 [1] (153, 0, 22) | 二点交叉をベースとして遺伝子交叉を行う。第1番目の親の交叉2点間の遺伝子を第1番目の子の同一位置にある遺伝子として複製する。第2番目の親の交叉2点間以外の遺伝子を、第2番目の子の同一位置にある遺伝子として複製する。第1番目の残りの遺伝子は、第2番目の親の遺伝子の出現順序で並べ直す。同様に、第2番目の残りの遺伝子は、第1番目の親の遺伝子の出現順序で並べ直す。 |
| 4 | PMX 法 [7] [8] (Partially Mapped Crossover) (179, 0, 33) | 二点交叉をベースとして遺伝子交叉を行う。第1番目の親の交叉2点間の遺伝子を同じ遺伝子座にある第2番目の親の遺伝子を互いに交換して第1番目の子と第2番目の子の遺伝子とする。交叉することによって生じた重複遺伝子については、交叉時に訪問する都市の遺伝子座が同じだった親の遺伝子と交換する。重複がなくなるまで交換処理を続ける。 |
| 5 | OX 法 [9] [10] (Order Crossover) (378, 0, 69) | 二点交叉をベースとして遺伝子交叉を行う。第1番目の親の交叉2点間の遺伝子を同じ遺伝子座にある第2番目の親の遺伝子と交換して第1番目の子の遺伝子とする。同じく、第2番目の親の交叉2点間の遺伝子を同じ遺伝子座にある第1番目の親の遺伝子と交換して第2番目の子の遺伝子とする。交叉することによって生じた重複遺伝子以外の遺伝子は、第2の交叉点から始めて、親の遺伝子の並び順で並べる。 |
| 6 | CX 法 [7] (Cycle Crossover) (164, 0, 22) | 第2番目の親の遺伝子は第1番目の親の遺伝子の順列に対する置換後の遺伝子と考える。任意の置換は巡回置換 (ある巡回数列に対する置換) の積となることが知られており、第1番目の親の遺伝子座にある遺伝子と同じ遺伝子座にある第2番目の親の遺伝子の組み合わせを巡回置換の文字列の一部と見なす。第1番目の親の第1遺伝子座から始まる巡回置換について、その巡回置換に含まれる文字を実現値として持つ第1番目の親の遺伝子を第1番目の子供の遺伝子とみなす。第1番目の子の残りの遺伝子については、同じ遺伝子座にある第2番目の親の遺伝子と交換する。第2番目の親の第1遺伝子座から始まる巡回置換を探すことから始めて、第2番目の子についても同様な考え方で遺伝子座を生成する。 |

表2. 続き

■: 提案方式

| 項番 | 致死遺伝子対策 | 説明 |
|----|---|---|
| 7 | 改良 EX 法 [11] (Edge Recombination Crossover) (509, 0, 64) | 隣接都市の中で距離の最も短い都市を次々に辿る最近近傍アルゴリズム [17] の応用である。各々の都市について、親 CX と親 CY の閉路上で隣接する都市の和集合を考え、それを各都市の隣接リストと呼ぶ。第 1 番目の子の最初の訪問都市は親 CX の最初の訪問都市とし、その訪問先の隣接リストの中からまだ訪問していない都市の中で最も距離の短い都市を二番目に訪れる都市とする。こうして隣接リストから次に訪れる都市を次々に選択する。隣接リストに訪れる都市がなく、まだ訪問先が残っている場合は未訪問先の中で最も距離の短い都市を次の訪問先として選ぶ。次に訪問する都市がなくなるまでこの処理を続けて、子供 1 の巡回路を決定する。第 2 番目の子の最初の訪問都市を親 CY の最初の訪問都市とすることから始めて、同様な手順で第 2 番目の子の巡回路を決定する。 |
| 8 | SXX 法 [3] [12] [13] (Subtour Exchange Crossover) (459, 0, 31) | 含まれる都市が同じ都市から成る巡回路の部分集合 S を親 CX と親 CY から探しそれぞれの部分巡回路を SX と SY とする。選ばれた部分巡回路から、以下の手続きで、4 人の子供を作る。本実験では、他の遺伝子交叉法との整合をとるため、4 人の中から適応度の高い 2 人の子供を選択する。(a) 親 CX の部分巡回路 SX を SY または SY の逆巡回路 (SY) に交換、(b) 親 CY の部分巡回路 SY を SX または SX の逆巡回路 (SX) に交換。本実験では、他の遺伝子交叉法との整合をとるため、4 人の子供の中から適応度の高い 2 人の子供を選択して残す。 |
| 9 | EXX 法 [19] (Edge Exchange Crossover) (453, 0, 51) | 親の巡回路における枝と枝のつながりに関する情報、すなわち部分巡回路とその巡回方向に関する情報を残すように、子供の枝を生成していく。親の巡回路 CX 及び巡回路 CY を、隣接する都市間の枝列 ϵX , ϵY として表現する。例えば巡回路が abcd なら (ab) (bc) (cd) (da) として表現する。枝 (ab) の始点を a, その終点を b と呼ぶ。 <ステップ 1> ϵX からひとつの枝 $i1$ を選ぶ。そして $i1$ と同じ始点を持つ ϵY の枝を $i2$ とする。 <ステップ 2> $i2$ の終点と同じ始点を持つ ϵX の枝を $j1$ とし、 $i1$ の終点と同じ始点を持つ ϵY の枝を $j2$ とする。 <ステップ 3> 巡回路 ϵX 上の枝 $i1$ と、巡回路 ϵY 上の枝 $i2$ を交換してできる 2 つの枝の列を、子候補 1, 子候補 2 とする。交換後に、 $i1$ と $i2$ の終点と同じであれば処理を終了する。 <ステップ 4> 子候補 1 では、 $i1$ の次に訪れる都市に矛盾がなく親の枝間の連結情報を引き継ぐため、巡回路 ϵX の $i1 \sim j1$ 間の部分巡回路を逆順にする。 <ステップ 5> 同様に、子候補 2 では、 $i2$ の次に訪れる都市に矛盾がなく親の枝間の連結情報を引き継ぐため、巡回路 ϵY の $i2 \sim j2$ 間の部分巡回路を逆順にする。 <ステップ 6> 生成された子候補 1, 2 において子の枝と枝が繋がらない箇所、かつそれを入れ替えればその点ではつながるような枝は $j1, j2$ であるから、 $i1=j1, i2=j2$ として、ステップ 2 へ。以後、ステップ 3 の終了条件を満たすまで、ステップ 2 からステップ 6 の処理を繰り返す。 |
| 10 | 遺伝子交叉法交代法 (24, 0, 11) (main 関数の一部) | 世代 (解の安定度) によって上記 1 から 9 に示される遺伝子交叉法を交代させる方式。交叉法や交叉時期は任意に選択可能とする。 |
| | 合計 (2693, 94, 360) | |

*: 括弧 (A, B, C) は各遺伝子対策毎に A=新規規模, B=流用規模, C=サイクロマチック数を示す。規模は C ソースコードの行数で測定している。

4. 巡回セールスマン問題を解く遺伝的アルゴリズムのCプログラム開発

4.1 開発目的

巡回セールスマン問題を解く遺伝的アルゴリズムの各致死遺伝子対策を中心に性能改善施策について検討する。その改善施策を、漸進的C言語プログラミング開発 [15] と Excel 等による分散分析等の統計的データ分析実験により考察する。

4.2 開発方法

まず、平野広美著「遺伝的アルゴリズムプログラミング」[1] で提供されている遺伝的アルゴリズムを実現するCプログラムの基本部 (tspapp.c, pp. 271~278, 595行 (コメント行含む)) を Windows の VisualC++ 環境に移植した。そこでは、(1) 改良 EX 法において次に訪問する都市の探索方向を巡回路の方向のみに限定した平野 EX 法、(2) 2点交叉における交換対象遺伝子を2つの交叉点間にある遺伝子と排他的な位置にある遺伝子とするように OX 法を拡張した平野 OX 法が提案されている。このCソースコードを基にして、Grefenstette 法、PMX 法、OX 法、CX 法、改良 EX 法、SXX 法、EXX 法の各致死遺伝子対策を実現する遺伝的アルゴリズムCプログラムを順に設計・製造・試験し、遺伝的アルゴリズムのオンライン性能評価実験を行った。最後に、これらのプログラムモジュールを統合し、「世代による致死遺伝子対策の交代機能」を加えて、総規模3,431行(コメント行含む) から成るC言語 [16] で開発されたソフトウェアを実現した。

4.3 開発条件

巡回セールスマン問題を解く遺伝的アルゴリズムの性能規定要因分析を可能とするため、以下のプログラム起動パラメータについてその実現値を幾つかの候補の中から選択可能とした。

●遺伝子交叉方式・・・3節で述べた10個の致

死遺伝子対策の選択

- 逐次型世代交代方式/一括型世代交代方式
- 2オプト法等の有無。
- 子供の生成方法 (間引きあり, なし)
- 遺伝子交叉確率
- 乱数種
- 人口数
- 都市数 (<=15)

尚、突然変異はなく、親の選択はルーレット方式固定である。

4.4 品質条件, 試験条件

致死遺伝子対策毎に遺伝子交叉の途中過程を標準端末にダンプ出力する遺伝子交叉部を実現するソフトウェアを実現し、机上の遺伝子交叉結果と計算機上でのソフトウェア実行結果が一致していることを目視確認した後、各遺伝子交叉部を、遺伝的プログラムを実現する本体部と結合させた。

4.5 機能, 規模

TSP を遺伝的アルゴリズムで実現するCプログラム構造を以下に整理する。

4.5.1 C関数とその機能, 規模

C関数とその機能, 規模(新規・改造規模, 流用規模), 複雑度を付録の付表1にまとめる。規模(ソースコードの行数)は、コメント行・空白行を含めて、トータル3,431行(新規3,049行, 流用404行), 46関数, 494サイクロマチック数 (= (条件文数+1)/関数) である。

(1) 遺伝的アルゴリズム共通部

付表1に示す関数のうち18関数を流用(再利用)した。再利用率は11%(=404/3431)であった。

(2) 巡回セールスマン問題共通部

適用率計算 (dist_fitness) 23行, 集団評価 (statistic) 45行, 親の都市と都市の間のつなが

りに関する有効な情報を残す2オプト法40行である。2オプト法はある種の優性遺伝に関わる問題を解決している。

(3) 巡回セールスマン問題固有部

表2に示される致死遺伝子対策を実現する10個のcrossover関数と、その詳細関数から成る。平野EX法(約94ステップ)を含み、2,763行で実現した。

4.5.2 プログラム制御構造のグラフ表現

各遺伝子交叉方式の一括型世代交代方式に関するプログラム制御構造を、付録の付図1のグラフで表現する。このグラフから、モジュール共通化の度合いを閉領域数=エッジ数(関数の制御数)-ノード数(関数数)+2で計測すると、6となることがわかる。

5. 実験

5.1 実験データ

図5の空間に位置する15都市について実験を行った。当空間はユークリッド空間である。各都市間の所要距離はピタゴラスの定理で求めている。

5.2 起動条件

遺伝子交叉方式(致死遺伝子対策)について、その方式の違いがどの程度性能に影響を与えているのかを、同一の条件で比較するため、遺伝子交叉方式以外のプログラムの起動パラメータを以下の通り固定して比較評価することとした。

- 世代交替方式 ・ ・ 一括型世代交代方式
- 2オプト法 ・ ・ 有り
- 子供の生成方法 ・ ・ 間引きあり
- 親の選択 ・ ・ ルーレット方式固定
- 遺伝子交叉確率 ・ ・ 0.8
- 突然変異 ・ ・ なし
- 乱数種* (1,10,20,30,40,50,60,70,80,90,99)
- 人口数 ・ ・ 100
- 都市数 ・ ・ 15
- 最大世代数 ・ ・ 100

性能は、最大適応度や平均適応度が安定した状態になるまでに、何回遺伝子交叉を行ったかで比較する。これはDe Jongが定義したオンラインパフォーマンス(各世代における最大適応度の世代平均)と同等の定義である。両親の遺伝子交叉で2人の子供を産むが、2人の子供の

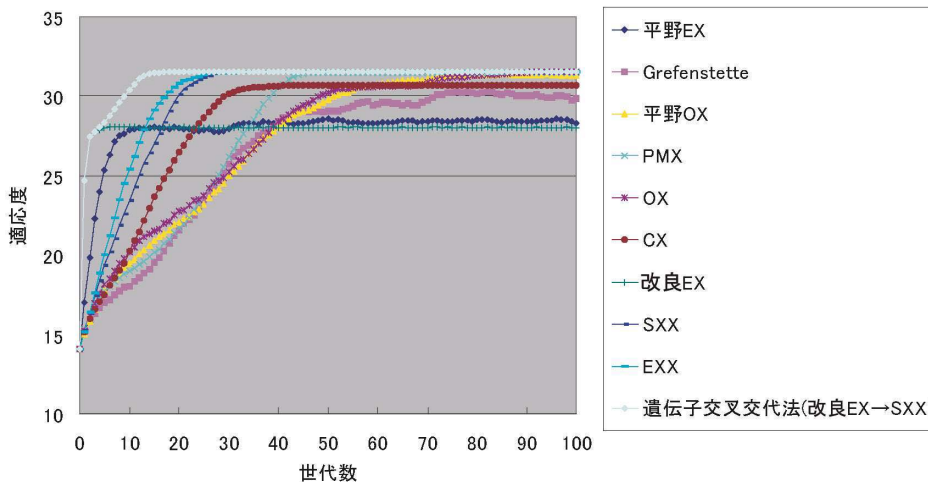


図4. 平均適応度の推移

うち適応度の高い子供のみを残す間引き方式では、2人の子供を無条件に残す方式に比較して2倍の遺伝子交叉回数が必要となる。

5.3 実験結果

15都市の巡回セールスマン問題に対する各遺伝子交叉方式(致死遺伝子対策)のオンライン性能を比較した結果を図4,表3にまとめる。図4では致死遺伝子対策毎に平均適応度の世代に対する推移状況を示している。表3では、各致死遺伝子対策毎に平均適応度と最大適応度に関し、それが安定状態に入った時期とその値を整理している。表3では、その性能が良い順番に致死遺伝子対策を並べている。図4,表3から以下のことが読み取れる。

① Grefenstette法やPMX法,CX法など「順列の互換やその表現法に着目する方法」では、枝に関する情報(距離の短さ)が欠落するため、必ずしも適用度の高い子供が生成される可能性がない。このため、安定するまでに時間がかかっている。

② 親の隣接都市間の所要距離に関する情報のみで次に訪れる都市を決定する改良EX法は局所解に到達するが、必ずしも最適解に到達し

ない。このため、比較的早い時期に、平均適応度が低い状態で安定したままとなっている。

③ 親の遺伝子の適用度が高い場合は、ある都市からある都市までに到る部分集合が一致している枝を交換するように遺伝子交叉を行うSXX法や、親の遺伝子の並び順を考慮して2点交叉を行うOX法については、比較的適用度の高い子供が生成される可能性が高い。我々の実験によれば、枝のつながりに着目したSXX法の方が都市の並び順に着目したOX法よりやや性能が良い。

5.4 遺伝子交叉オペレータ交代法

以上のことから、任意の世代で遺伝子交叉法を交代させて遺伝子交叉を実現する遺伝子交叉オペレータ交代法(A method of Changing Crossover Operators(CXO))を考案し、その方式が有効であることを、Cプログラミング実験で確認することとした。実験では3世代目で改良EX法をSXXに交代させた。表3に示すように、当交代方式では17世代目に平均適応度が最高値31.5の安定状態に入り、SXX法単独で平均適応度が最高値31.5の安定時期に入る34世代目より早く安定時期に入っていること、安定時期が9世代目と早いものの28.02と低い適応度で安定している改良EX法に比べて適応度の高いモデル(この場合は最適解)を選択できていることがわかる。

CXO法が発見した最適巡回路を図5に示す。

表3. 平均適応度と最大適応度の安定時期とその解

| 致死遺伝子対策 | 平均適応度 (安定時期) | 最大適応度 (安定時期) |
|-------------------------|-----------------|-----------------|
| 遺伝子交叉オペレータ交替法(改良EX→SXX) | 31.5 (17世代) | 31.5 (10世代) |
| EXX法 | 31.5 (32世代) | 31.5 (27世代) |
| SXX法 | 31.5 (34世代) | 31.5 (27世代) |
| PMX法 | 31.46(46世代) | 31.47(38世代) |
| OX法 | 31.42(85世代) | 31.5 (83世代) |
| 平野OX法 | 31.32(79世代) | 31.47(74世代) |
| CX法 | 30.69(43世代) | 30.69(34世代) |
| Grefensttete法 | 30.23(74世代) | 30.49(69世代) |
| 平野EX法 | 28.33(55世代) | 30.93(52世代) |
| 改良EX法 | 28.02 (9世代) | 28.95(13世代) |

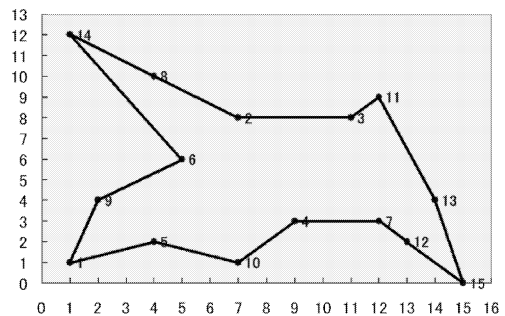


図5. 遺伝子交叉方式交代法(改良EX→SXX)が発見した最適巡回路

本実験から、「比較的初期の世代では改良 EX 法を適用し、その解が安定した比較的後期の世代では SXX 法を適用する」ような、世代が経るにつれ遺伝子交叉方法を替える遺伝子交叉法 (CXO 法) が有効であることがわかった。

6. 結論と今後の課題

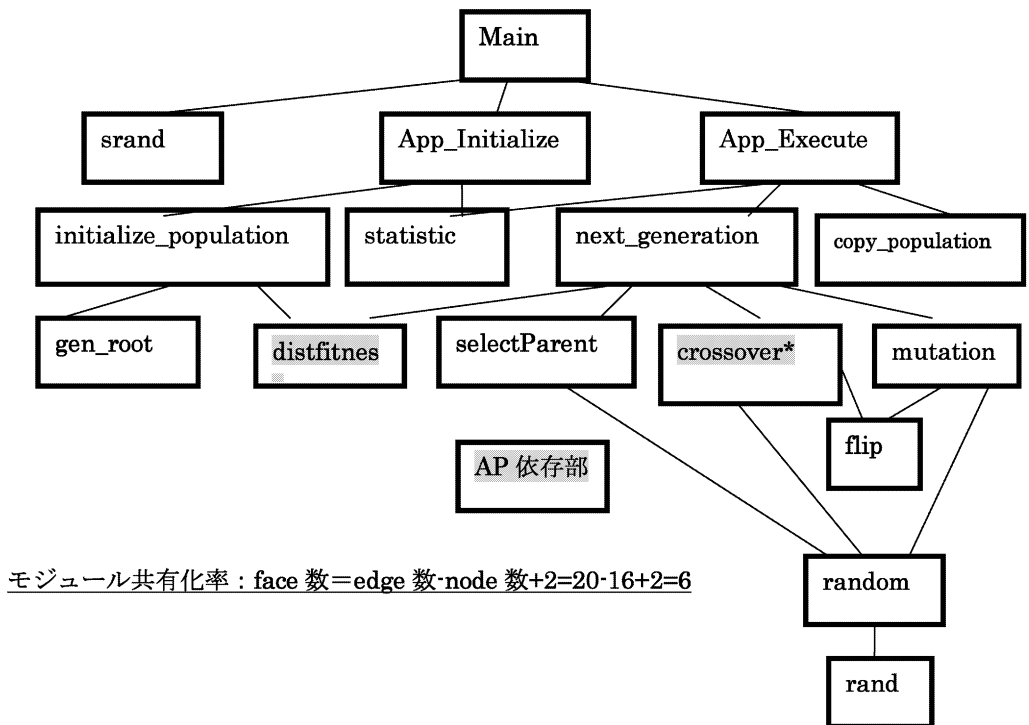
本研究では、時間の経過と共に遺伝子交叉の方法を変化させるような遺伝的アルゴリズムの実現方法の有効性を、15 都市を対象とした小規模 C プログラミング実験により検証した。この結果、単独で改良 EX 法や SXX を適用するより、初期の段階では改良 EX 法を適用し後期の段階では SXX を適用する方式が有効であることを検証した。この実験結果は、都市数や、各都市間の位置関係や距離の偏り等、巡回セールスマン問題で対象としているデータの特徴を考慮して、遺伝子交叉方式を変化させる方式が最適巡回路の選択に有効であることを示唆している。都市数の拡大等、実験空間の拡張による本方式の有効性の検証については、今後の課題とする。

巡回セールスマン問題を遺伝的アルゴリズムで解く時の性能 (遺伝子交叉回数) は、本論文で述べた致死遺伝子対策 (「致死遺伝子」を回避するための「遺伝子交叉方式」) 以外に、(1) 即時、一括世代交代方式、(2) 性能向上対策 (① 2 オプト法の有無、② 生成した子供の間引の有無)、(3) 世代を構成する集団の規模とその初期条件 (① 各世代の集団を構成する個体数 (解候補数)、② 乱数種 (初期個体群生成時等に使用される初期パラメータ) 等の起動パラメータに依存する。上記「致死遺伝子対策」や起動パラメータ等性能規定要因のうち何が最も遺伝的アルゴリズムの性能に効いているかの分散分析については今後の課題とする。

参考文献

- [1] 平野廣美, 「遺伝的アルゴリズムプログラミング」, パーソナルメディア社, pp. 67-87, pp. 271-278, 1995.
- [2] 三宮信夫, 喜多 一, 玉置 久・岩本貴司, 遺伝的アルゴリズムと最適化, pp. 37-51, 朝倉書店, 1998.
- [3] 北野 宏明編集, 小林重信, 山村雅幸, 遺伝的アルゴリズム (1), 産業図書, pp. 43-60, 1993.
- [4] 伊庭斉志, 遺伝的アルゴリズムの基礎, オーム社, pp. 29-37, 1994.
- [5] J.H. Holland, *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press (1975), MIT Press (1992)
- [6] E. Vonk, L.C. Jain, and R.P. Johnson, *Automatic Generation of Neural Network Architecture Using Evolutionary Computation*, World Science, 1997.
- [7] D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.
- [8] D.E. Goldberg and R. Linger, Jr., *Alles, Loci, and the Traveling Salesman Problem*, Proc. Of 1st Int. Conf. on Genetic Algorithms and Their Applications, pp. 154-159 (1985).
- [9] I.M. Oliver, D.J. Smith, and J.R. C. Holland, *A Study of Permutation Crossover Operations on the Traveling Salesman Problem*, Proc. Of 2nd Int. Conf. on Genetic Algorithms, pp. 224-230 (1987).
- [10] L. Davis, *Applying Adaptive Algorithms to Epistatic Domains*, Proc. Of 9th Int. Joint Conf. on Artificial Intelligence, pp. 162-164 (1985).
- [11] D. Whitely, T. Starkweather and D'Ann Fuquary, *Scheduling Problems and Traveling Salesman: The Genetic Edge Recombination Operation*, Proc. Of 3rd Int. Conf. on Genetic Algorithms, pp. 133-140 (1989).
- [12] 山村雅幸, 小野 功, 小林重信: 形質の遺伝を重視した遺伝的アルゴリズムに基づく巡回セールスマン問題の解法, 人工知能学会誌, Vol. 17, No. 6, pp. 1049-1059 (1992).
- [13] 柳浦睦憲, 永持 仁, 茨木俊秀, サブツアー交換交叉に関する 2 つのコメント, 人工知能学会誌, Vol. 10, No. 3, pp. 464-467 (1995).
- [14] 岩間一雄, アルゴリズム理論入門, 昭晃堂, pp. 119-152, 2001.
- [15] Mint, ソフトウェア開発のすべて, 日本実業出版社, p. 63, 2000.

- [16] Kernighan-Richie 著, 石田晴久訳, プログラミング言語 C (ANSI 規格準拠), 共立出版, 1989.
- [17] 廣田薫編著, 知能工学概論, pp. 4-6, 昭晃堂, 1996.
- [18] J. Grefenstette, R. Gopal, B. Rosmaita, D. Van Gucht, Genetic Algorithms for the Traveling Salesman Problem, Proc. of 1st Int. Conf. on Genetic Algorithms and Their Applications, pp. 160-168 (1985).
- [19] 前川景示, 玉置 久, 喜多 一, 西川禪一, 遺伝的アルゴリズムによる巡回セールスマン問題の1解法, 計測自動制御学会論文集, Vol. 31, No. 5, pp. 598-605, 1995.



付図1. 一括型世代交代方式の場合の遺伝的アルゴリズムのCプログラム構造

付表1. 巡回セールスマン遺伝的アルゴリズムを実現する主なC関数

| NO | 関数名 | 機能 | (新規規模, 流用規模, サイクロマチック数) |
|----------------------------------|-----------------------|--|-------------------------|
| 1 | main | 遺伝的プログラムの主制御関数。一様乱数の初期値を srand () で設定する。初期設定部 (App_Initialize) と中核部 (App_Execute()) の実行を制御する。 | (137, 0, 35) |
| 2 | App_Initialize | 以下の実行を制御する。 ・初期集団の生成 (initialize_population()) ・集団の評価 (statistic()) | (0, 18, 3) |
| 3 | App_Execute | 以下の機能の実行を制御する。(即時型を含む) ・次期世代個体生成 (next_generation()) ・世代一括交代 (copy_population()) ・集団の評価 (statistic()) | (18, 16, 8) |
| 4 | initialize_population | 以下の機能の実行を制御する。 ・個体の生成 (gen_chrom()) ・個体適応率計算 (dist_fitness()) | (0, 12, 2) |
| 5 | copy_population | 現世代の個体と次世代の個体を交代させる。(即時+一括) | (95, 11, 27) |
| 6 | next_generation | 以下の機能の実行を制御し, 次世代個体を生成する。 ・親の選択 (selectParent()) ・遺伝子交叉 (crossover()) ・突然変異 (mutation()) ・適応率計算 (dist_fitness()) | (85, 107, 30) |
| 7 | mutation | 以下の関数を利用して, 突然変異操作を行う ・変異操作の有無の確率的決定 (flip()) ・変異対象座の一様乱数 (random()) による決定 | 使用せず |
| 8 | crossover | 以下の関数を利用して, 遺伝子交叉操作を行う。(即時型含む) ・交叉操作の有無の確率的決定 (flip()) ・交叉位置の一様乱数 (random()) による決定 | (2669, 94, 349) |
| 9 | selectParent | 累積適応率が基準値を満たした個体を親として選択する。以下の関数を利用して, 基準値を決定する。 ・一様乱数 random() による基準値の確率的決定 | (0, 14, 2) |
| 10 | gen_root | 初期個体集団の生成 | (0, 22, 6) |
| 11 | flip | 一様乱数が基準値以下か否かを判断する。 | (0, 8, 3) |
| 12 | randomrand | 関数を利用して, 0 から 1 までの一様乱数を発生させる。 | (0, 8, 2) |
| 13 | statistic | 現世代の集団について (1) 平均適応率 (2) 最大適応率を出力する。個体の遺伝子型を出力する。 | (0, 45, 10) |
| 14 | dist_fitness | 個体の適応率を計算する。 | (23, 0, 3) |
| 15 | srand | 一様乱数の初期値を与える。 | (0, 4, 1) |
| 16 | rand | 一様乱数を発生させる。 | (0, 5, 1) |
| 17 | improve_crossover | 2 オプト法の実現。(間引き法含む) | (0, 40, 12) |
| 合計 (新規規模, 流用規模, 複雑度 (サイクロマチック数)) | | | (3027, 404, 494) |