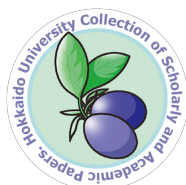




Title	遺伝的アルゴリズムによる最適系列分割問題の解法
Author(s)	荒木, 肇; 加地, 太一; 山本, 雅人 他
Citation	電気学会論文誌. C, 電子・情報・システム部門誌, 120(2), 215-221
Issue Date	2000-02-01
Doc URL	https://hdl.handle.net/2115/64535
Type	journal article
File Information	120_215.pdf



遺伝的アルゴリズムによる最適系列分割問題の解法

非会員 荒 木 肇 (北海道大学)

正 員 加 地 太 一 (小樽商科大学)

非会員 山 本 雅 人 (北海道大学)

非会員 鈴 木 恵 二 (北海道大学)

正 員 大 内 東 (北海道大学)

A genetic algorithm for the optimal sequential partitioning problem

Hajime Araki, Non-member (Hokkaido University), Taichi Kaji, Member (Otaru University of Commerce), Masahito Yamamoto, Non-member, Keiji Suzuki, Non-member, Azuma Ohuchi, Member (Hokkaido University)

The optimal sequential partitioning problem is defined as the problem to find the minimum cost partition of the nodes of a directed acyclic graph into subsets of a given size, subject to the constraint that the precedence relationships among the elements are satisfied. The heuristic algorithm based on a tabu search for this problem has been proposed⁽²⁾. However, there is a tendency for the solutions obtained using the tabu search approach to be trapped in bad local optima in the parallel graphs with random costs of edges

In this paper we present the genetic algorithm for the optimal sequential partitioning problem. We develop effective two point partial order crossover satisfying sequential conditions, which preserve better block that has the larger sum of edge costs of block. In this crossover we introduce roulette selection method to escape local optima. We also assess the effectiveness of the developed algorithm. The results show that this proposed algorithm outperforms, in terms of solution quality, any other algorithm using tabu search.

キーワード：グラフ分割問題、遺伝的アルゴリズム、最適系列分割問題、無閉路有向グラフ、近似解法

1. 序 論

グラフ分割問題とはあるグラフが与えられたとき、部分集合間のエッジのコストの総和が最小となるような各ノードの分割を求める問題である。ただし、制約条件として、各部分集合に属するノードのウェイトの総和がある上限を超えないという条件を持つ。グラフ分割問題は、重要な研究テーマとして長らく研究されている。その工学的応用として、VLSI システムのレイアウト設計やコンピュータのオブジェクト割り当てなどが挙げられる。

分割問題の研究のひとつとして、1970 年代に全順序関係のグラフに対する系列分割問題が Kernighan により提案されている⁽¹⁾。この問題は、全順序関係のグラフに対して、分割される各頂点部分集合がその番号の連続性を保持し、分割された頂点の各部分集合の重みの総和がある上限数以下に限定される条件の下で、カットエッジのコストの総和を最小化する問題である。

しかし、多くのシステムは単純な全順序関係よりも複雑

な半順序関係で構成される。そのため、工学的応用を考慮した場合、無閉路有向グラフを対象とする必要がある。それに対し、この問題を半順序関係の無閉路有向グラフに拡張した問題が加地により提案されており⁽³⁾、厳密解法、近似解法が検討されている。この無閉路有向グラフの頂点を先行順位の関係を保持して分割する問題を無閉路有向グラフの系列分割問題と呼び、本論文では無閉路有向グラフを系列分割するときに生ずるカット辺のコストの総和を最小化する問題を取り扱い、この問題を最適系列分割問題と呼ぶことにする。

多くの工学的問題への応用が期待される無閉路有向グラフの最適系列分割問題に対して、これまでに提案されている近似解法として、Tabu Search 法による解法がある⁽²⁾。その手法では、システム構造に多くみられる並列グラフ⁽²⁾に対して実験が行われており、近似度 5% 以内の有効な解が得られている。しかし、エッジコストの値が幅広く分散するグラフに対しては性能の低下が報告されている。このような、局所解に陥り易い問題に対して、大域的な探索が必要とさ

れる。

本論文では、最適系列分割問題の遺伝的アルゴリズム (Genetic Algorithm, GA) ⁽⁴⁾ による解法を提案する。GA の特徴として、集団として多数の解を保持しており、形質を保存した上で大域的な近傍探索が可能であるということが挙げられる。しかし、最適系列分割問題に GA を適用するとき、一般のバイナリビット列による個体表現では、分割の表現や解が制約条件を満たすかどうかの判定が困難である。このことが標準的な GA による解法の構成を困難としている。

このような問題点に対して、系列化グラフの概念を使用する。この概念を使用することにより、分割の表現および制約条件の判定が容易になるが、解空間の増大を招く。それに対し、Kernighan の算法 ⁽¹⁾ と組み合わせることにより解空間の縮約を行う。そして、常に先行順序関係を満たす解を生成できる交叉法として 2 点部分順序交叉を提案する。さらに、交叉点の決定において、形質として頂点の部分集合に着目し、より良好な頂点の部分集合がより高い確率で保存されるように、交叉点の決定にルーレット選択手法を導入する。その上で、各種のオペレータの比較を試み、本論文で提案する戦略の有効性を示す。そして、Tabu Search 法において性能の低下が指摘されているエッジコストの値が幅広く分散する並列グラフに対して数値実験を行い、その有効性について検討する。

2. 無閉路有向グラフの最適系列分割

本章では提案されている無閉路有向グラフの最適系列分割問題について簡潔に述べる。

V は頂点集合、 E は有向辺の集合とするととき先行、後続の関係をもつ要素集合の構造は、その要素を頂点とし、要素とその後続要素からなる要素の有向対を辺とする無閉路有向グラフ $D(V, E)$ によって表すことができる。系列分割の問題では単一の入口と出口を持つ無閉路有向グラフ $D(V, E)$ を考察の対象とする。

任意の頂点 $u, v \in V$ に対して v から u への有向路が存在するならば、関係 $v \preceq u$ が成立するといひ、 $v \preceq u$ かつ $v \neq u$ は $v \prec u$ で表される。要素間の先行順位関係を保持して頂点集合 V を互いに素で網羅的な部分集合の族 $\{V_1, V_2, \dots, V_k\}$ に分割し、各部分集合を先行順に並べることができるとき、このような分割を系列分割と呼ぶ。

この定義は以下のようにまとめられている。

〔定義 1〕 空でない部分集合 $A \subset V$ から誘導された D の部分グラフを $D(A)$ で表す。 $D(A)$ の任意の 2 点を結ぶ D 内の有向路がすべて $D(A)$ の有向路となるととき、 $D(A)$ は系列を保持する D の部分グラフであるといひ。

〔定義 2〕 V の部分集合 A が、 $A^c \times A$ から選んだ 2 元対 (x, y) に関して、 x と y が $\preceq$ において比較可能ならば常に $x \prec y$ が成立するとき、 (A^c, A) を A によって定まる V の切断といひ。そして、 A をこの切断の上組、 A^c を下組といひ。

〔定義 3〕 V の互いに素な部分集合 X と Y がそれぞれ V の

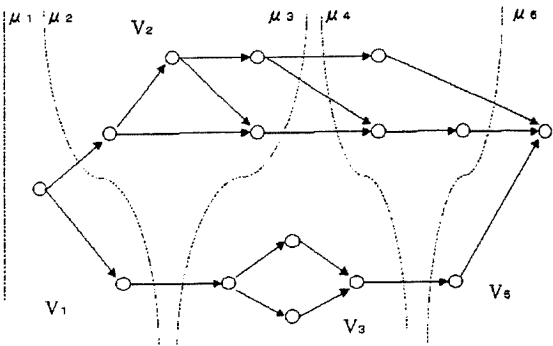


図 1 無閉路有向グラフの系列分割
Fig. 1. Sequential partitions of a directed acyclic graph.

ある切断の下組と上組に含まれるならば、 X と Y は切断により分離されるといひ、 $X|Y$ で表す。

$X|Y$ は定性的には “ X が Y より前にある” ことを、また X と Y を結ぶ辺が存在するときには “それらの辺はすべて同じ向きをもつ” ことを表している。ここで、無閉路有向グラフの系列分割が次の様に定義されている。

〔定義 4〕 $D(V, E)$ の頂点集合 V の分割 $\{V_1, V_2, \dots, V_k\}$ がその任意の成分集合 V_i, V_j に対し、 $i < j$ のとき $V_i|V_j$ が成立するとき、この分割を $D(V, E)$ の系列分割といひ、この分割による各 $D(V_i)$ は系列保持の性質をもつ。

図 1 は無閉路有向グラフの系列分割の一例であり、切断を破線で表し、その右側が上組、左側が下組となることを示す。各成分集合 V_i に上組、下組 からなる切断 $\mu_i = (A_i^c, A_i)$ を対応づける。切断の順序列 $\mu_1, \mu_2, \dots, \mu_5$ を作ると隣り合う切断の上組 A_i, A_{i+1} から、 $V_i = A_i - A_{i+1}$ となり、系列分割 $V_1|V_2|\dots|V_5$ が得られる。

本論文では、最適系列分割問題と呼ばれる各成分に課せられた制約条件のもとで、総カット値を最小化する分割を求める問題について取り扱う。対象とする無閉路有向グラフ $D(V, E)$ のすべての頂点 $v \in V$ には重み $w(v)$ が、各有向辺 $(u, v) \in E$ にはコスト $c(u, v)$ が付与されている。これらの値は、すべての $u, v \in V$ について、条件 $0 < w(v) \leq B$ 、および $c(u, v) \geq 0$ を満たす実数であり、 B はブロックサイズと呼ばれる問題に固有な正の実数である。最適系列分割問題では、容量制約のもとで、分割成分である各部分グラフ間を連結するコストの総和を最小にする分割を求める。 k を固定されない定数とするととき、最適系列分割問題は以下のように定式化される。

Minimize
$$\sum_{u \in V_i, v \in V_j, i < j, (u, v) \in E} c(u, v) \dots\dots\dots (1)$$

subject to

$$|V_i| = \sum_{v \in V_i} w(v) \leq B, i = 1, \dots, k, \dots\dots\dots (2)$$

$$V_1|V_2|\dots|V_k, \dots\dots\dots (3)$$

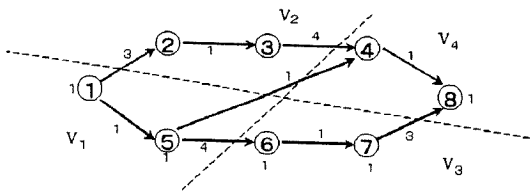


図2 無閉路有向グラフの系列分割

Fig. 2. Sequential partitions of a directed acyclic graph.

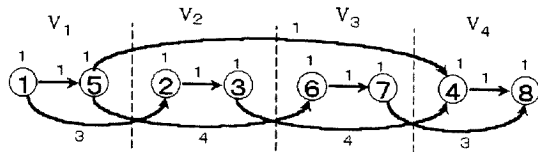


図3 図2に対する一列化グラフの系列分割

Fig. 3. Sequential partitions of sequential graph in Fig. 2.

3. 遺伝的アルゴリズムによる解法

(3・1) 個体表現

最適系列分割問題を表現する手法として、無閉路有向グラフを一列化グラフ⁽²⁾で表現し、系列分割を一列化(トポロジカルソート)された頂点列と分割点を表すブレイク・ポイント⁽¹⁾の列で表現する手法がある。一列化グラフとは、無閉路有向グラフ $D(V, E)$ の頂点集合 V を一列化し、その順に頂点を左から右に並べ換えて得られる同型なグラフのことである。図2のグラフを一列化グラフで表現した例が図3で表されている。一列化された頂点列を $\{v_1, v_2, \dots, v_n\}$ とすると、連続した頂点の並びからなる部分列 $\{v_p, v_{p+1}, \dots, v_q\}$ は系列を保持する V の部分集合となる。

それゆえ、無閉路有向グラフの系列分割は一列化グラフとその頂点列上のブレイク・ポイントの並びを与えることによって生成できる。ここで部分列 $\{v_p, v_{p+1}, \dots, v_q\}$ をブロックと呼び、 $[v_p, v_q], [v_p, v_{q+1})$ などの記法を用いて表す。ブレイク・ポイントの挿入位置を示すインデックスを $1 \leq p_1 < p_2 < \dots < p_k \leq n$ とする。ブレイク・ポイントの並びを $b_i = v_{p_i}; i = 1, 2, \dots, k$ とすると、ブロック $[b_i, b_{i+1}); i = 1, 2, \dots, k$ の系列は V の系列分割を生成し、 $[b_i, b_{i+1})$ はその成分集合 V_i となる。ただし、頂点 $n+1$ をダミーとして与え、 b_{k+1} を $n+1$ とする。

この表現を用いることにより、無閉路有向グラフの一列化された頂点列とブレイク・ポイントの列のデータを管理することで全ての解を表現することができる。その利点として、制約条件の一つである先行関係の保持を満たす全ての分割を容易に生成することができることが挙げられる。よって容量制約のみに焦点を置くことにより実行可能解を導くことができる。しかし、一列化グラフの表現による解候補と問題の解候補は多対1の対応となるため冗長性が増す。そこで、一列化グラフに対して容量制約を満たす最適な分割配置を決定することができる Kernighan の算法⁽¹⁾を用いた。これにより一列化グラフから得ることが可能な解集合の中から最適な解を決定することにより解空間の縮約を図

る。Kernighan の算法は、動的計画法に基づいた手法であり、一列化グラフに対する容量制約を満たす最適分割配置をグラフのエッジ数に比例した計算量で決定することができる効率的な手法である。

以上のように、個体として一列化された頂点列を持つことにより有効な探索が行われる。個体 p_i^k の適応度 $f(p_i^k)$ は、カット・エッジのコストの総和を s とするとき、コスト関数と同等の以下の式を使用した。

$$f(p_i^k) = \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n} c(v_i, v_j) - s \dots \dots \dots (4)$$

ここで p_i^k は、 k 世代における i 番目の個体であり、集団内の個体数を M とするとき、一列化グラフの頂点列を用いて、以下のように表現する。

$$p_i^k = \{v_1 v_2 \dots v_n\}, (1 \leq i \leq M) \dots \dots \dots (5)$$

(3・2) 交 叉

本アルゴリズムでは個体表現として、一列化グラフによる表現を使用している。この個体表現においては、頂点の順序が非常に重要になってくる。そこで交叉手法としては、2点順序交叉⁽⁵⁾⁽⁸⁾が有効であると考えられる。しかし、通常の2点順序交叉では、先行関係が保持されない致死遺伝子を生成する可能性がある。この問題点に対して、2点順序交叉における遺伝子組み替えを部分的に行い、先行順位関係を保持する有効な交叉法として2点部分順序交叉を提案する。

そして、本分割問題における個体が保存する形質としてブロックに注目した。ここで提案する2点部分順序交叉では、ブロック内のエッジのコストの総和がより大きいブロックを有効な形質として保存し、さらに解の偏りをなくするためにルーレット選択法の考え方を導入した。

以下に、2点部分順序交叉の手順、交叉点の決定法の詳細について記述する。

(3・2・1) 2点部分順序交叉

前述の通り、通常の2点順序交叉では、先行関係が保持されない致死遺伝子を生成する可能性がある。そこで、先行順位関係を保持する親が与えられたとき、先行順位関係を保持する子を必ず生成する2点順序交叉を構成し、2点部分順序交叉と呼ぶ。その手順は以下のように表される。

親 A の個体が頂点列 $\{v_1 \dots v_n\}$ からなる時、頂点列 $\{v_i v_{i+1} \dots v_j\}, (1 \leq i < j \leq n)$ を優れた形質として残すとする。 $\{v_1 \dots v_{i-1}\}$ を親 B の順序に従って要素を割り当ててできた頂点列を $\{v'_1 \dots v'_{i-1}\}$ とする。また、 $\{v_{j+1} \dots v_n\}$ を親 B の順序に従って要素を割り当ててできた頂点列を $\{v'_{j+1} \dots v'_n\}$ とする。この時、頂点列 $\{v'_1 \dots v'_{i-1} v_i \dots v_j v'_{j+1} \dots v'_n\}$ を子 A とする。同様に、子 B も親 B のブロックをもとに親 A の順序に従って生成する。

例として、親 A の $\{3\ 8\}$ 、親 B の $\{3\ 4\}$ を保存する場合について考える。親 A の $\{1\ 6\ 2\ 7\}$ が、親 B の順序に従っ

て {1 2 6 7}, {4 9 5 10} が {4 5 9 10} に組み替えられる。同様に親 B の {1 2 6} が親 A の順序に従って {1 6 2}, {7 8 5 9 10} が {7 8 9 5 10} に組み替えられる。この時、以下のような 2 個の子 A, B が生成される。

親 A : {1 2 6 7} 3 8 4 9 5 10
 親 B : {1 2 6} 3 4 7 8 5 9 10
 ↓ 2 点順序交叉
 子 A : {1 2 6 7} 3 8 4 5 9 10
 子 B : {1 6 2} 3 4 7 8 9 5 10

このように生成された子はいずれも、先行関係を保持している。それを定理として以下に証明する。

〔定理 1〕 先行順位関係を保持する親 A, B が与えられたとき、2 点部分順序交叉により生成される子は先行順位関係を保持する。

証明 親 A, B をそれぞれ $V_A = \{u_1 \cdots u_n\}$, $V_B = \{v_1 \cdots v_n\}$ とする。有向辺の集合を $E = \{(u, v) | (u, v) \in V \times V\}$ とする。 V_A, V_B が先行順位関係を保持するとき、 $(u_i, u_j) \notin E, (n \geq i > j \geq 1)$, $(v_i, v_j) \notin E, (n \geq i > j \geq 1)$ の関係を満たす。 V_A の中の頂点列 $V_{A2} = \{u_l u_{l+1} \cdots u_m\}, (1 \leq l < m \leq n)$ を保存し、 $V_{A1} = \{u_1 \cdots u_{l-1}\}$, $V_{A3} = \{u_{m+1} \cdots u_n\}$ を組み替えるとき、 $(i, j) \notin E, (\forall j \in V_{A1}, \forall i \in V_{A2}, V_{A3})$, $(i, j) \notin E, (\forall j \in V_{A1}, V_{A2}, \forall i \in V_{A3})$ は自明である。従って、 V_{A1}, V_{A3} を V_B の順に並び変えて生成される $V'_{A1} = \{u'_1 \cdots u'_{l-1}\}$, $V'_{A3} = \{u'_{m+1} \cdots u'_n\}$ が先行順位関係を満たせばよい。仮定より、 $(v_i, v_j) \notin E, (i > j)$ であり、 V'_{A1} と V'_{A3} は B の順序によって並び変えられることから、 V'_{A1} と V'_{A3} の両方とも、 $(u'_i, u'_j) \notin E, (l > i > j \geq 1)$, $(u'_i, u'_j) \notin E, (n \geq i > j > m)$ が成り立つ。よって、 $V'_A = \{u'_1 \cdots u'_{l-1} u_l \cdots u_m u'_{m+1} \cdots u'_n\}$ では、 $(u'_i, u'_j) \notin E, (n \geq i > j \geq 1)$ となる。 □

〈3・2・2〉 交叉点の決定法

子孫に遺伝すべき形質としてブロックに着目した。最適系列分割問題はカット・エッジのコストの総和の最小化問題である。これは、カット・エッジ以外のエッジ、つまりブロック内エッジのコストの総和の最大化問題に置き換えることができるので、ブロック内エッジのコストの総和が大きいブロックが優れた形質として考察される。よって、ブロック内エッジのコストが、より大きいブロックを保存するように交叉点を決定することが有効であると考えられる。

しかし、ある一定のブロックのみが保存されるような交叉点の決定法では、局所解に陥りやすいと考えられる。そこで、交叉点の決定にルーレット選択を導入し、ブロック内のエッジのコストの総和に比例した確率でブロックを保存するように交叉点を決定する。系列化の表現では、ブロックはブレイクポイントによって特定されるので、ブロックの両端のブレイクポイントが交叉点となる。

ブロック V_i 内のエッジのコストの総和を S_i とするとき、

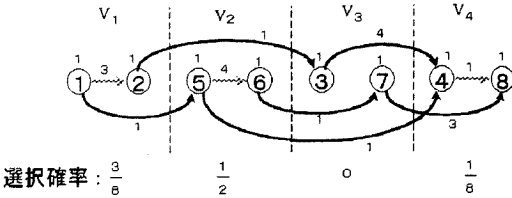


図 4 交叉点の決定法
 Fig. 4. A strategy to decide crossover point.

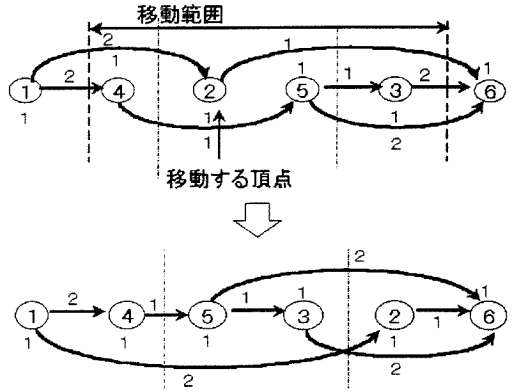


図 5 突然変異の操作
 Fig. 5. An operation of mutation.

ブロック V_i の両端が交叉点となる確率 $CPselect_i$ は、式 6 で表される。

$$CPselect_i = \frac{S_i}{\sum_{j=1}^n S_j} \cdots \cdots (6)$$

図 4 では、ブロック V_1, V_2, V_3, V_4 がそれぞれ $\frac{3}{8}, \frac{1}{2}, 0, \frac{1}{8}$ の確率で優良なブロックとして保存される。

〈3・3〉 突然変異

突然変異は、個体の各頂点が突然変異率の確率で移動する操作である。突然変異による頂点の移動は、個体の先行関係を満たす範囲内で行われる。頂点 v の移動範囲は v を終点とする最短な流入辺の始点 u の直後から、 v を始点とする最短な流出辺の終点 w の直前までで、その範囲内でランダムな場所を選択しその場所へ移動する。図 5 の場合、頂点 2 を移動するとき、頂点 2 の移動範囲が頂点 1 の直後から頂点 6 の直前であるので、その範囲内の頂点 3 と頂点 6 の間に移動している。

〈3・4〉 選択淘汰

選択淘汰は、交叉により生み出された子の個体に突然変異の操作を行って評価したものと親の個体とを合わせた集団の中から選択淘汰を行う。代表的な選択手法としてはルーレット選択法⁽⁶⁾、トーナメント選択法⁽⁷⁾ などがある。ここでは、交叉法として順序交叉を使用しており、両方の親から強く影響を受けるので、適応度が低い親を順序を保存する親として選択してしまうと、解が悪化してしまう。そこで、より適応度が高い個体同士を組み合わせたエリート組合せ手法を採用し、優位性を実験で示す。

〈3・5〉 実 装

```
procedure
begin
  k := 0;
  Initialize( $P^k$ );
  Fitness( $P^k$ );
  while stop-criterion <> yes do begin
     $C^k$  := Crossover( $P^k$ );
    Mutation( $C^k$ );
    Fitness( $C^k$ );
     $P^{k+1}$  := Selection( $P^k \cup C^k$ );
    k := k + 1;
  end;
end;
```

図 6 アルゴリズム
Fig. 6. Algorithm.

これまでに説明した交叉 (Crossover), 突然変異 (Mutation), 選択淘汰 (Selection) を用いてアルゴリズムを構成する。アルゴリズムは図 6 のとおりである。 P^k は k 世代における親の集団, C^k は k 世代における子の集団を表す。

Initialize 関数は, 集団サイズ M に対して M 個の個体をランダムに生成する関数である。Fitness 関数は, Kernighan の算法により, 集団内の全個体の適応度とブレイクポイントの決定を行う関数である。終了条件は打ち切り世代数に達した時点とする。

4. 実験

本章ではこれまでに述べたことについて実験を行う。最初に本論文で使用する GA について各オペレータの比較実験を行う。ここでは, 交叉点の決定手法, 個体の選択オペレータについて比較実験を行い最良のオペレータを決定する。さらに, システム構造で多く見られる並列グラフに対する Tabu Search 法による研究があるが, その Tabu Search 法との比較実験により GA の有効性を示す。

〈4・1〉 GA のオペレータ決定に関する実験

最初に, エッジコストが 1~50 までランダムに分散した値を持つ頂点数 102 の各種の並列グラフに対して, 交叉オペレータにおける交叉点の決定法の比較実験を行った。交叉点の決定法として, ランダムに決定した手法, ブロック内エッジのコストの総和が最大であるブロックを保存するように決定した手法, 提案するルーレット選択で決定した手法, 以上の 3 つの手法を用いて比較実験を行った。GA のパラメータは集団数 100, 打ち切り世代数 150, 突然変異率を 0.01 とし, 個体の選択オペレータはエリート組合せ法を採用した。10 回の実験を行い, その最良値, 平均値, 最悪値を表 1 に示す。各手法は, 並列数が小さいときには, あまり効果に違いが生じていないが, 並列数が大きくなるに従って, ルーレット選択で決定したものが最良となることが示されている。交叉点の決定にルーレット選択を導入することは, グラフの複雑度の増加に伴い有効な効果をもたらしている。

次に, 個体の選択オペレータとして, トーナメント選択法, ルーレット選択法, エリート組合せ法の 3 つの選択オペレータについて比較実験を行った。この実験においても, 同様の頂点数 102 の各種の並列グラフを対象にして行った。集団

並列数		2 並列	3 並列	4 並列	5 並列
ランダム	最良値	122	123	131	189
	平均値	123	126.5	140	199
	最悪値	124	130	149	209
ブロック内 エッジコスト 最大	最良値	122	121	129	184
	平均値	122	124.6	141.3	193
	最悪値	122	130	151	207
ルーレット	最良値	122	121	128	163
	平均値	122	121	130.8	172
	最悪値	122	130	139	184

表 1 各並列グラフに対する交叉点の決定法の実験結果 (頂点数 102)

Table 1. Computational results for the strategies to decide crossover points in various parallel graphs.

並列数		2 並列	3 並列	4 並列	5 並列
ルーレット選択	最良値	122	121	128	164
	平均値	124	121	134.25	175.3
	最悪値	128	121	145	201
トーナメント選択	最良値	122	121	128	164
	平均値	122	121	132.5	178
	最悪値	122	121	144	191
エリート組合せ	最良値	122	121	128	163
	平均値	122	121	130.8	172
	最悪値	122	121	139	184

表 2 各並列グラフにおける個体の選択オペレータの実験結果 (頂点数 102)

Table 2. Computational results for selection strategies in various parallel graphs.

数 100, 打ち切り世代数 150, 突然変異率を 0.01, 交叉点の決定法にルーレット法を使用して, 10 回の実験を行い, その最良値, 平均値, 最悪値を表 2 に示す。結果から大きな違いは見られないが並列数が増加するに従い, 最悪値においてエリート組合せ法の若干の優位性が示されるので, 本アルゴリズムではエリート組合せ法を採用する。

〈4・2〉 Tabu Search 法との比較実験

並列グラフに対して, 有効な結果が得られている Tabu Search 法との比較実験を行い, 本アルゴリズムの有効性を示す。

実験は, Sun Ultra SPARC1 上で C++ 言語によって作成したプログラムを実行することで行った。GA は, 交叉部分のプログラムなどが計算時間に大きく影響する。また, 本プログラムにおいては交叉による新たな解生成に $O(n^3)$ の計算時間がかかるものとなっており, それにより, Tabu Search より計算時間を多く必要とする結果となった。そこで, 今回の実験では実験環境を統一するために, 解を生成する回数に着目し, すなわち近傍解に移動した探索数を統一することにより, 実験環境の統一性をもたせた。また, 計算時間の改善については, 交叉のプログラムにおいて一列化グラフの頂点列の順番を継承する処理においてハッシュ法などをもちいて $O(1)$ で順位の認識を行うことによりスピード化がはかれる。また, 交叉処理は独立性が高いため, その機能分割型の並列化アルゴリズム, および, 集団のデータ分割型並列アルゴリズムに移行しやすく容易に計算時間の短縮がはかれる。

GA のパラメータとして, Tabu Search 法のステップ数

並列数		2 並列	3 並列	4 並列	5 並列
GA	最良値	72	121	128	107
	平均値	72	124.6	130.8	111.8
	最悪値	72	130	139	121
Tabu Search		90	133	158	173

表 3 各並列グラフにおける Tabu Search 法との実験結果 (頂点数 102)

Table 3. A comparison of the genetic algorithm and the tabu search in various parallel graphs.

頂点数		50	100	150	200	250
GA	最良値	79	72	162	170	254
	平均値	79	72	162	171.5	254.5
	最悪値	79	72	162	173	256
Tabu Search		87	90	165	173	279

表 4 頂点数に対しての Tabu Search 法との実験結果 (2 並列グラフ)

Table 4. A comparison of the genetic algorithm and the tabu search for various number of nodes.

が 14510 であることから、評価の回数がほぼ同等の条件となるように設定した。GA は集団数 100、打ち切り世代数 150 とした。突然変異率 0.01 で実験を行った。また、GA のオペレータとして、交叉点の決定法はルーレット選択法、個体の選択オペレータはエリート組合せ手法を使用した。GA は 10 回実験を行い、その中の最良値、平均値、最悪値を記述している。

最初に、エッジコストが 1~50 までにランダムに分散した値を持つ頂点数 102 の各種の多並列グラフに対し実験を行い、表 3 に結果を示す。表 3 から、多並列になるに従い、GA の有効性が示されている。並列グラフでは並列数が増加するに従って、系列化グラフの組合せ数が増大して解空間が広がるため、複雑度が増し広域的な近傍移動が有効となる。

次に、エッジコストが 1~50 までにランダムに分散した値を持つ 2 並列グラフに対して頂点数を増加させ比較実験を行い、表 4 にその結果を示す。各頂点数の 2 並列グラフにおいて GA は Tabu Search 法よりも良好な値を得ている。

以上より、ランダムなエッジコストを持つ並列グラフにおいて大域的な探索を行う本アルゴリズムを用いることによって、著しい有効性がもたらされる結果となった。

5. 結 論

本論文では、遺伝的アルゴリズムによる最適系列分割問題の解法を提案した。特に、系列化グラフによる個体表現と Kernighan の算法との組み合わせによる解空間縮約を試みた。さらに、優良な形質を保存し、常に先行順位関係を満たす 2 点部分順序交叉を構成し、本問題に有効な遺伝的アルゴリズムを提案した。

交叉法として通常の 2 点順序交叉を使用すると個体の先行順位関係が破壊される。しかし、2 点部分順序交叉では、組み換える遺伝子を限定し、他方の親の順序列を継承させることによって先行順位関係を保持した。そして、交叉点の決定法として、より良好なブロックが確率的に高く採択されるルーレット選択法を導入することによって効果を図った。

そして、数値実験では、Tabu Search 法で性能の低下が示

されていた。エッジコストの値がランダムに分散する並列グラフに対して、Tabu Search 法よりも良好な解が得られた。

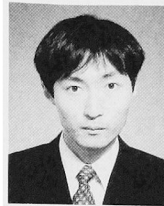
また並列数が増加するに従って、系列化グラフの組合せ数が増大するので解空間が広がる。このため、並列数が増加するに従って広域的な近傍移動が有効となり、特に GA の有効性が示された。

(平成 11 年 2 月 22 日受付, 同 11 年 8 月 17 日再受付)

文 献

- (1) Kernighan, B. W. . Optimal Sequential Partitions fo Graphs, J. ACM, Vol. 18, No. 1, pp. 34-40(1971).
- (2) 加地 太一, 大内 東: Tabu Search を用いた無閉路有向グラフの最適系列分割問題の解法, 電気学会論文誌 C 分冊, Vol. 116-C, No. 10, pp. 1149-1157(1996).
- (3) 加地 太一, 大内 東: 要素間に先行順位をもつシステムの配置問題, 電気学会論文誌 C 分冊, Vol. 117-C, No. 2, pp. 136-142(1997).
- (4) 北野 宏明: 遺伝的アルゴリズム, 産業図書, 1995
- (5) 柳浦 睦憲, 茨木 俊秀: 順序問題における遺伝的交叉法に対する一考察, 電気学会論文誌 C 分冊, Vol. 114-C, No. 6, pp. 713-720(1994).
- (6) Goldberg, D. E. Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley, Reading, Mass. , 1989.
- (7) Goldberg, D. E. and Deb, K. . A Comparative Analysis of Selection Schemes used in Genetic Algorithms, Foundations of Genetic Algorithms, Rawlins, G. J. E. (ed.), Morgan Kaufmann, 1991.
- (8) I. M. Oliver, D. J. Smith and J. R. C. Holland, "A Study of Permutation Crossover Operators on the Traveling Salesman Problem", Proceedings of the Second International Conference on Genetic Algorithms, pp. 224-230(1987).

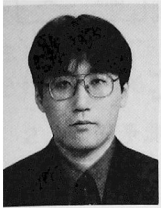
荒 木 肇 (非会員) 昭和 49 年 3 月 29 日生。平 8 年 3 月北海道大学工学部情報工学科卒。平 10 年 3 月同大学院工学研究科システム情報工学専攻修士課程修了。同年 4 月より、建設省東北地区建設局に勤務。遺伝的アルゴリズムに興味を持つ。



加 地 太 一 (正員) 1988 年北海道大学大学院工学研究科情報工学専攻修士課程修了。同年 (株) 東芝入社。1989 年北海道情報大学経営情報学部助手。1994 年小樽商科大学商学部社会情報学科助教授。博士 (工学) システム工学、情報科学の研究に従事。日本 OR 学会、情報処理学会各会員。

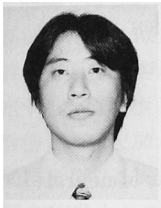


山 本 雅 人（非会員） 1991 年北海道大学工学部情報工学科卒



業。1993 年同大学院工学研究科修士課程情報工
学専攻修了 1996 年同博士後期課程修了。博士
(工学)。1996 年日本学術振興会特別研究院 (PD)。
1997 年北海道大学大学院工学研究科複雑系工学
講座助手。マルチエージェントシステム，DNA
コンピューティングなどに関する研究に従事。情
報処理学会，人工知能学会，電子情報通信学会，
計測自動制御学会，日本 OR 学会各会員。

鈴 木 恵 二（非会員） 1988 年北海道大学工学部精密工学科卒



業。1990 年同大学院工学研究科修士課程精密工
学専攻修了。1993 年同博士後期課程修了 博士
(工学)。同年，北海道大学工学部助手。1996 年同
大学院工学研究科複雑系工学講座助教授。ニュー
ラルネットワーク，遺伝的アルゴリズム，マルチ
エージェントなどに関する研究に従事。情報処理
学会，日本機械学会各会員。

大 内 東（正員） 1974 年北海道大学大学院工学研究科博



士課程修了 博士 (工学)。北海道大学大学院工学
研究科複雑系工学講座教授。システム情報工学，
応用人工知能システム，医療システムの研究に
従事。情報処理学会，電気学会，計測自動制御学
会，日本 OR 学会，医療情報学会，病院管理学会，
IEEE-SMC 各会員。