

# 系列分割問題に対する確率的複合移動による Simulated Annealing 法の適用

加地 太一<sup>†</sup> 大内 東<sup>††</sup>

本論文では無閉路有向グラフの系列分割問題に対する Simulated Annealing 法の適用とその有効性について検討している。本問題はその特徴として、解の成分集合の個数、各成分集合の要素数がともに不定であり、これが解の近傍処理などの操作を複雑にし探索法の構成を難しくしている。そのため、通常の近傍移動のみでは近似度のより良い結果は望めない。これに対して、本論文では一列化グラフとブレイク・ポイントの集合を用いたデータ構造を利用し、グラフ分割問題における従来の近傍移動である left-to-right 移動, right-to-left 移動の組織的で多重的な適用と、部分的な最適化を取り入れ、複合的な処理方法を用い効果的な近傍移動を実現した。さらに、通常の Simulated Annealing 法の採択基準と異なる局所的なコスト差による新たな基準を設け効果をはかっている。以上の複合的な近傍移動により解に大きな変化をほどこし Simulated Annealing 法の性能をより強く引き出し、その特徴も維持する結果となった。さらに、今回考案した確率的複合移動の効果を数値実験により示し、Tabu Search 法による解法と比較し本算法の性能を明らかにした。それによると、提案する算法は Tabu Search 法により求まる解より良質な解を導き、また、その計算時間もほぼ頂点数に線形に増加する傾向が示された。

## A Simulated Annealing Approach Using Random Compound Move for Sequential Partitions Problem

TAICHI KAJI<sup>†</sup> and AZUMA OHUCHI<sup>††</sup>

This paper presents an approach based on simulated annealing algorithm for finding a minimum cost partition of the nodes of a directed acyclic graph into subsets of a given size, subject to the constraint that the precedence relationship are satisfied. The simulated annealing algorithm imposes different randomized search and acceptance criteria on local search method in order to escape poor quality local minima. The method generally can obtain solutions arbitrarily close to an optimum. But a standard simulated annealing approach can't find good solutions for this problem, because the problem is complex multiple partitioning problems in which the number of subsets and the number of nodes in each subset are unsettled. For this problem, we use appropriate data structure for this method, and develop effective neighbourhood structure and new acceptance criterion. And, we assess the effectiveness of the developed algorithm. The results show that this algorithm outperform the algorithm using tabu search in solution quality and computational time. It is effective in obtaining near-optimal solution to this problem. The running time of the procedure is proportional to the number of nodes in the graph.

### 1. はじめに

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系列分割問題と呼ぶ。本論文では無閉路有向グラフを系列分割するときを生ずるカット・

エッジ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“総カット値を最小化する系列分割問題”と呼ぶことにする。その応用例として、コスト関数や制約条件を問題の特性に適合させることによって、先行順位を持ついくつかの生産プロセスを各ステーション間の輸送コストの総和が最小になるように、ステーションごとに利用可能な資源量の範囲内で配置する問題、およびライン・バランス<sup>3),10)</sup>の問題など多様な問題に適応させることが可能である。

この総カット値を最小化する系列分割問題は一般に指数的オーダーの計算時間を必要とし、実用的な

<sup>†</sup> 小樽商科大学商学部社会情報学科

Department of Information and Management Science,  
Faculty of Commerce, Otaru University of Commerce

<sup>††</sup> 北海道大学工学部システム情報工学専攻

Division of Systems and Information Engineering, Faculty of Engineering, Hokkaido University

時間内での厳密解の導出は困難である．これに対し、Tabu Search 法<sup>7),8)</sup>による近似解法が提案されている<sup>13)</sup>．この解法では並列構造を持つ無閉路有向グラフに対して、相対誤差5%以内の精度で解を得ることができた．しかし、ランダムなグラフに対しては局所解に落ち込みやすい傾向なども示された．

本論文では、局所解からの脱出性をより強めるため、Simulated Annealing 法<sup>2)</sup>の考えに基づく解法を提案する．本問題は、その特徴として、解の成分集合の個数、各成分集合の要素数がともに不定であり、これが解の近傍移動などの処理を複雑にし、標準的な Simulated Annealing 法による解法の構成を難しくしている．本論文では、一列化グラフと分割を表すブレイク・ポイント<sup>6)</sup>の列を用いたデータ構造を利用し、一列化グラフ上での系列分割を保持する頂点の多重移動、およびブレイク・ポイント列の最適配置を組み合わせた効果的かつ計算しやすい独自の近傍移動を提案する．また、この頂点移動の過程において独自の基準を用いて Simulated Annealing 法の考えを取り入れている．さらに本算法について使用した複合移動の効果を数値実験で示し、その効果を明らかにする．また、Tabu Search 法による解法との比較を行い、この提案算法では Tabu Search 法より良質な解が導出され、頂点数に線形な計算時間で求められることを示す．

## 2. 無閉路有向グラフの最適系列分割

先行、後続の関係を持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．ここで  $V$  は頂点集合、 $E$  は有向辺の集合である．これから扱う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  $v \preceq u$  が成立するという．また、 $v \preceq u$  かつ  $v \neq u$  を  $v \prec u$  で表す．ここで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時、これを系列分割と呼ぶ．この定義をより正確に以下にまとめる．

**定義 1.** 空でない部分集合  $A \subset V$  から誘導<sup>15)</sup>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  $D(A)$  の有向路となる時、 $D(A)$  は系列を保持する  $D$  の部分グラフであるという．

**定義 2.**  $V$  の部分集合  $A$  が、 $A^c \times A$  から選んだ

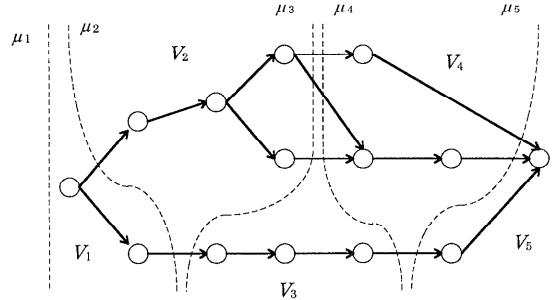


図1 無閉路有向グラフの系列分割  
Fig. 1 Sequential partitions of a directed acyclic graph.

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、つねに  $x \prec y$  が成立するとき、 $(A^c, A)$  を  $A$  によって定まる  $V$  の切断という．そして、 $A$  をこの切断の上組、 $A^c$  を下組という．

**定義 3.**  $V$  の互いに素な部分集合  $X$  と  $Y$  がそれぞれ  $V$  の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い、 $X|Y$  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持つ”ことを表している．ここで、無閉路有向グラフの系列分割を次のように定義する．

**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  $V_i|V_j$  が成立し、かつ  $V_1|V_2|\dots|V_k$  を満たすように並べることができる時、この分割を  $D(V, E)$  の系列分割といい、この分割による各  $D(V_i)$  は系列保持の性質を持つ．

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  $A_i^+ = V_i \cup V_{i+1} \cup \dots \cup V_k$ 、下組  $A_i^- = V_1 \cup V_2 \cup \dots \cup V_{i-1}$  からなる切断  $\mu_i = (A_i^+, A_i^-)$  を対応づけ、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．

系列分割上で、各成分に課せられた制約条件のもとで、最良の評価値を持つ分割を求める問題を系列分割問題と呼ぶ．本論文では“総カット値を最小化する系列分割問題”を扱う．以下にその定式化について述べる．

総カット値を最小化する系列分割問題で考えるネットワークは、多重辺を持たない無閉路有向グラフであり、単一の入口と出口を持つ  $n$  個の頂点からなる．これを  $D(V, E)$  として与える． $D(V, E)$  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。総カット値を最小化する系列分割問題では、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。これを定式化すると

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

subject to

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

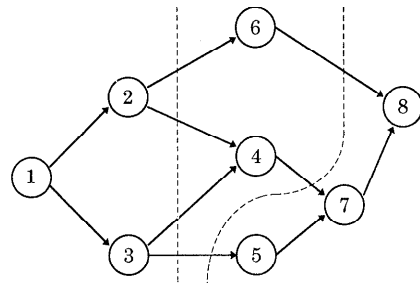
$$V_1 | V_2 | \dots | V_k$$

となる。このとき、 $k$  は固定されていない値である。以後、この問題を単に最適系列分割問題と呼ぶ。

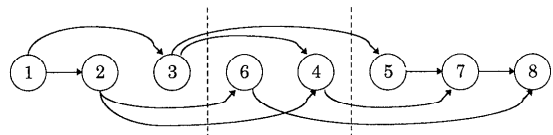
### 3. 無閉路有向グラフの系列化グラフによる表現

最適系列分割問題の探索法による近似解法を構成するにあたって、無閉路有向グラフを系列化グラフで表現し、系列分割を系列化された頂点列とブレイク・ポイントの列で表現する。この表現をとることによって、問題自体の理解と、系列分割の計算処理が容易になるばかりでなく、次章で提案する算法中に使用する最良のブロック分割を求める部分問題の最適化算法の構成を容易にする。

無閉路有向グラフ  $D(V, E)$  の頂点集合  $V$  はいつでも系列化 (トポロジカルソート) が可能である。頂点をこの系列化の順に左から右に並べ換えて得られる同型なグラフを系列化グラフと呼ぶことにする。系列化された頂点列を  $\{v_1, v_2, \dots, v_n\}$  とすると、系列化グラフの各頂点はこの順に左から右に並んでおり、すべての辺の向きはつねに左から右に向かっている。また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  $V$  の部分集合となることを容易に示すことができる。それゆえ、無閉路有向グラフの系列分割は系列化グラフとその頂点列上の分割点 (ブレイク・ポイント<sup>6)</sup>) の並びを与えることによって生成できる (図2参照)。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わす。ブレイク・ポイントの挿入位置を示すインデックスを  $1 \leq p_1 < p_2 < \dots < p_k \leq n$  とし、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



(a) 無閉路有向グラフの系列分割



(b) (a)に対する系列化グラフの系列分割

図2 無閉路有向グラフと系列化グラフの関係

Fig. 2 Relation between a directed acyclic graph and a sequential graph.

$V_i$  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。したがって、無閉路有向グラフの系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によって解とその移動を表現し、近傍移動の処理を効果的に構成することが可能になる。

### 4. 近似解法の実現

本問題はその特徴として、解の成分集合の個数、各成分集合の要素数がともに不定であり、これが解の近傍処理などの操作を複雑にし探索法の構成を難しくしている。本論文ではグラフ分割問題における従来の近傍操作である left-to-right 移動, right-to-left 移動<sup>14)</sup> をブロック順を考慮して多重的に適用することにより、容量制約を考慮せずに、系列化された頂点の並びをより望ましい系列化の並びに積極的に変化させる多重頂点移動を構成する。次に、この新しい系列化に対して、容量制約を満たす最良のブロック分割を求める部分問題を解くことによって、実行可能性の回復と近似度のさらなる向上を計る。以上によって構成された解の複合移動の形成過程に Simulated Annealing 法<sup>2),4),5)</sup> の考え方を取り入れた近似解法を提案する。標準的な Simulated Annealing 法においては解の移動の前後におけるコスト差により採択基準を決定するのに対し、本近似解法では多重頂点移動における局所的な頂点移動ごとに採択基準を定め、この基準のもとで確率的に採択された頂点移動のみを合成し、それに基づき解の

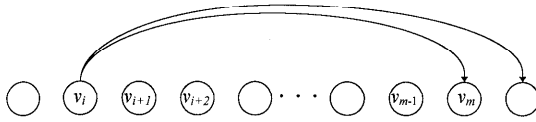


図3 一列化グラフの頂点の並び

Fig. 3 The sequence of the nodes of sequential graph.

確率的複合移動を形成する。この複合移動により、解に大きな変化をほどこし Simulated Annealing 法の性能を引き出す。ここで提案する方策の有効性とその特徴については 5 章の数値実験の結果と合わせて述べる。

#### 4.1 近傍構造の構成

近傍構造の構成は、基本的には無閉路有向グラフを変換した一列化グラフの性質を壊すことなく、頂点を左あるいは右へ移動することによって構築する。 $d(v, u)$  を  $v \in V$  を始点、 $u \in V$  を終点とする有向辺が存在するなら 1、そうでなければ 0 の値をとる関数とする。このとき、 $d(v_i, v_{i+1}) = d(v_i, v_{i+2}) = \dots = d(v_i, v_{m-1}) = 0$  かつ  $d(v_i, v_m) > 0$  が成立するならば、 $v_i$  は  $v_{i+1}$  の直後から  $v_m$  の直前の位置までの範囲内で、一列化の性質を破壊することなく右移動が可能である。図 3 の一列化の並びでは  $v_{i+1}$  の直後への移動とは  $v_{i+1}$  と  $v_{i+2}$  の間に  $v_i$  を移動することであり、 $v_m$  の直前に移動するとは  $v_{m-1}$  と  $v_m$  の間に  $v_i$  を移動することである。この  $v_m$  を  $v_i$  に対する一列化グラフ上での最近接後続点と呼ぶ。同様に、 $d(v_{i-1}, v_i) = d(v_{i-2}, v_i) = \dots = d(v_{q+1}, v_i) = 0$  かつ  $d(v_q, v_i) > 0$  が成立すれば、 $v_i$  は  $v_q$  の直後から  $v_{i-1}$  の直前の範囲内で左移動が可能である。この  $v_q$  を  $v_i$  の最近接先行点と呼ぶこととする。以上のように、一列化グラフ上の任意の頂点はその最近接後続(先行)点によって定まる範囲内で、任意の場所に移動可能である。しかし、本論文で提案する解法では頂点  $v_i \in V_i$  に対して、右移動の場合、最近接後続点  $v_m$  が  $v_m \notin V_i$  ならば  $v_m$  の直前へ、また左移動の場合、最近接先行点  $v_q$  が  $v_q \notin V_i$  ならば  $v_q$  の直後への移動に限定する。なお、前提条件が満たされないときは頂点の移動は行わず、頂点の並びはそのままとする。すなわち、一列化グラフ上でできうる限り遠くへ頂点を移動する方策をとる。以後、頂点  $v$  の最近接後続点の直後への頂点移動、および  $v$  の最近接先行点の直前への頂点移動を、**longmove-right**( $v$ )、**longmove-left**( $v$ ) で表す。

以上の頂点移動をベースとし、さらに順次連続するブロックに適用し非実行可能解の範囲まで探索を広げる多重的な近傍移動を構築する。これにより、

解に大きな変化をほどこしより強く最良な解の方向へ移動させ近似度の改善を行う。まず、任意の探索過程での解  $x = (V_1, V_2, \dots, V_k)$  はある一列化グラフ  $D$  の頂点の並びと、ブレイク・ポイントの集合  $\{b_1, b_2, \dots, b_k\}$  によって表現される。このとき、longmove-right 移動の系列 longmove-right( $v_1$ )、longmove-right( $v_2$ )、 $\dots$ 、longmove-right( $v_{k-1}$ ) を、出発解  $x = (V_1, V_2, \dots, V_k)$  から始めて連続適用し、解  $x' = (V'_1, V'_2, \dots, V'_k)$  を得る。ただし、 $v_i$  はブロック  $V_i = [b_i, b_{i+1}]$  中から順次、ランダムに選択した頂点とする。この際、ブレイク・ポイントの移動は考慮しないので、ブロックの容量制約は無視して頂点移動を行うことになる。この頂点の連続した移動を**多重 longmove-right 移動**と呼び、 $x' = \overrightarrow{\text{multi}N}(x)$  で表す。この結果に対して、再び longmove-left 移動の系列 longmove-left( $v_k$ )、longmove-left( $v_{k-1}$ )、 $\dots$ 、longmove-left( $v_2$ ) を連続適用することにより解の改善を行う。このとき各頂点  $v_k, v_{k-1}, \dots, v_2$  は  $V'_k, V'_{k-1}, \dots, V'_2$  から順次ランダムに選び出すこととする。これを**多重 longmove-left 移動**と呼ぶこととし、同様に、 $x'' = \overleftarrow{\text{multi}N}(x')$  で表す。

また、多重 longmove-right 移動、多重 longmove-left 移動だけでは、解の実行可能性が失われるばかりでなく、成分集合の個数、大きさに大きな変化をもたらすことは望めない。そこで、実行可能性を回復し、さらに近似度を高めるためにブレイク・ポイントの移動、付加、削除を試みる。すなわち、多重頂点移動によって得られた一列化グラフに対して容量制約を満たす最良なブロック分割を求める部分的な最適化を組み入れる。これによって、自動的に最良な分割数が決定される。ここで、多重移動によって得られた解  $x'' = (V''_1, V''_2, \dots, V''_k)$  がある一列化グラフとブレイク・ポイントの単調増加列  $\{b_1, b_2, \dots, b_k\}$  により表されているとする。このとき、ブレイク・ポイントの移動、新たな付加、および削除などによって実行可能性を回復して得られる実行可能解の集合を  $\lambda$  と定義し、 $\lambda$  の中で最も低いコストを示す解  $x'''$  への移行を示す関数を  $dp(x'')$  とする。ここで  $dp(x'')$  は Kernighan の一列化グラフの最適系列分割問題を解く算法<sup>(6), (12)</sup>をそのまま利用して実現できる。この算法の計算量は  $O(n)$  であり、与えられた一列化グラフのもとで最適なブレイク・ポイント列  $\{b'_1, b'_2, \dots, b'_k\}$  を見出す。その結果、頂点のブロック間での移動によるコストの改善と、実行可能性の回復が行われる。これらの処理を一反復過程において、 $x''' = dp(\overleftarrow{\text{multi}N}(\overrightarrow{\text{multi}N}(x)))$  の順序で行い、良好な近似効果を持つ複合移動を達成

する。

### 4.2 確率的複合移動による Simulated Annealing 法の構成

通常の Simulated Annealing 法では、反復過程において現在の解  $x$  から上記の複合移動により導かれた解  $x'$  に対して、そのコスト差  $\text{cost}(x) - \text{cost}(x')$  を基準として確率的採択の計算を行い、移動の採否を決定する。これに対して本研究では、各頂点移動  $\text{longmove-right}(v)$ ,  $\text{longmove-left}(v)$  ごとに局所的な採択基準量  $\delta^\pm(v)$  を計算し、改善されたときと、また非改善のときは確率  $\exp(-\delta^\pm(v)/tmp)$  で頂点移動の処理を行い、これらを合成して複合移動を構成する。したがって、実際の算法では前記の多重頂点移動過程中に確率的要素を導入することとなる。

頂点移動  $\text{longmove-right}(v)$ ,  $\text{longmove-left}(v)$  を実行すると、それとともなって目的関数値が変化する。その効果は通常、増分コストを用いて評価する。しかし、本論文では、以下で新たに定義する量  $\delta^+(v)$ ,  $\delta^-(v)$  を使用して頂点移動の効果を測定する。まず、これらを定義するために次の記号を導入する。一列化グラフ上の任意の頂点  $v_p$  を左端点または右端点とし、大きさ  $b$  の容量制約を満足し最大の重みを持つブロックをそれぞれ  $V^+(v_p, b)$  または  $V^-(v_p, b)$  で表す。すなわち、 $w(v)$  を頂点  $v$  の重みとして、制約  $\sum_{i=0}^q w(v_{p+i}) \leq b$  を満たす  $q$  の最大値を  $r$ ,  $\sum_{i=0}^q w(v_{p-i}) \leq b$  を満たす  $q$  の最大値を  $s$  とすれば、それらはそれぞれ  $V^+(v_p, b) = [v_p, v_{p+r}]$  または  $V^-(v_p, b) = [v_{p-s}, v_p]$  と定義される。頂点移動  $v$  の最近接後続点を  $u$ , 最近接先行点を  $u'$  として  $\delta^+(v)$ ,  $\delta^-(v)$  を

$$\delta^+(v) = \sum_{s \in V^-(v, B)} c(s, v) - \sum_{t \in V^+(u, B-w(v))} c(v, t) \tag{1}$$

$$\delta^-(v) = \sum_{t \in V^+(v, B)} c(v, t) - \sum_{s \in V^-(u', B-w(v))} c(s, v) \tag{2}$$

と定義する。これらはいずれも頂点  $v$  を実際に移動する前に計算可能な量である。また、これは頂点移動によって一列化グラフ上で大きなコストを持つ辺の長さ（頂点の添字の差）をより短くし、できるだけ容量  $B$  のブロック内にまとめると良い結果が得られるという経験則を反映させ、従来の増分コストの定義を修正して導びいたものである。さらに、この量は現在のブレイク・ポイントの位置を意識せずに計算でき、そのため計算処理を簡素化できる利点がある。

確率的採択基準に  $\delta^+(v)$ ,  $\delta^-(v)$  を用い Simulated Annealing 法の考えに準拠して多重頂点移動を構成

Procedure 確率的多重  $\text{longmove-right}$  移動

```

for  $i=1$  to  $k$  do begin
 $v = V_i$  の中からランダムに移動可能な頂点の一つを選択する。;
if  $\delta^+(v) \leq 0$  then
     $\text{longmove-right}(v)$ ;
else
    if  $\exp(-\delta^+(v)/tmp) \geq \text{random}[0,1)$  then
         $\text{longmove-right}(v)$ ;
end;
```

図 4 確率的多重  $\text{longmove-right}$  移動の手続き

Fig. 4 A procedure of random multi  $\text{longmove-right}$  move.

する算法について具体的に説明する。多重  $\text{longmove-right}$  移動の場合、頂点  $v \in V_i$  に対して  $\delta^+(v) < 0$  は解の総コストが改善される傾向を示している。この条件  $\delta^+(v) < 0$  が成立した場合には無条件で、また成立しない場合には確率  $\exp(-\delta^+(v)/tmp)$  で、最近接後続点の直前への頂点移動を行う。この処理をブロック  $V_1, V_2, \dots, V_k$  の順に適用した結果、新たな解  $x'$  への移動が行われる。これを新たに確率的多重  $\text{longmove-right}$  移動と呼び、その手続きを図 4 に示す。

同様に、確率的採択基準  $\delta^-(v)$  を用いて確率的多重  $\text{longmove-left}$  移動を定義する。

解  $x$  に対して、この確率的多重  $\text{longmove-right}$  移動を作用させた結果を  $x' = \overleftarrow{\text{RandMultiN}}(x)$  とし、この結果に確率的多重  $\text{longmove-left}$  移動を作用させた結果を  $x'' = \overrightarrow{\text{RandMultiN}}(x')$  で表す。これに基づく解の複合移動を確率的複合移動という。

この確率的複合移動をベースとし、Simulated Annealing 法の考えに従って図 5 の近似解法を構成する。ここで、 $tmp$  は温度パラメータであり、順次冷却しながら各温度での準平衡解を得て近似解への収束をはかる。また、各温度での準平衡解を回復するには少なくともある一定の推移を起こす受理回数が必要であり<sup>2)</sup>、これは各温度での  $r$  回のループによって実現される。このループの最終値は準平衡解であり、これを次の温度での準平衡解を求めるための出発解とする。また、温度の低下の減少にともなって受理確率が減少するので、 $r$  の数を増大していくことも必要となる。これらを制御するパラメータ ( $tmp, \text{stoptmp}, r, \text{phi}, \text{tau}$ ) に依存して、計算時間と解の質が変化することは明らかである。これらの値は数値実験による経験則から導かれる。

まず、 $tmp, \text{stoptmp}$  の値を受理比<sup>2)</sup> (= 受理された移動数/ 要求された移動数) より決定する。理論的にはこの値を 1.0 から 0.0 へと変化させれば、良質な解が得られることはすでに知られている。しかし、こ

```

Procedure simulated annealing
begin
  各パラメータ( $tmp, stoptmp, r, phi, tau$ )
  の値を設定する。;
   $x :=$  initial solution;  $x^* := x$ ;
  while  $tmp > stoptmp$  do begin
    for  $i = 1$  to  $r$  do begin
       $x' = RandMultiN(x)$ ;
       $x'' = RandMultiN(x')$ ;
       $x''' := dp(x'')$ ;
      if  $f(x''') < f(x^*)$  then  $x^* := x'''$ ;
       $x := x'''$ ;
    end;
     $tmp := tmp \times phi$ ;
     $r := r \times tau$ ;
  end;
  best solution :=  $x^*$ ;
end;

```

図5 simulated annealing に基づくアルゴリズム

Fig. 5 An algorithm based on simulated annealing.

のような変化に基づき実行すれば、計算時間を多大に必要とせざるをえない。本問題では傾向として0.85から0.15まで受理比を変化させることにより実用的計算時間内で良質な解が得られ、その範囲を超える変化により解を求めても、著しい解の改善は見られなかった。したがって、受理率が0.85から0.15まで変化するように  $tmp$  と  $stoptmp$  を設定する。これは単純な予備実験で容易に求まる。このとき、初期温度は無閉路有向グラフの辺のコストの大きさに依存し、その最大値の大きさに対して1.4倍前後の値をとればよいことが判明した。

$phi$  については0.8, 0.85, 0.9, 0.95とパラメータの値を変化させ効果を確認してみた。値が高いほど良質な解が求まるが、それに反して計算時間が必要となる。ここでは、0.9を採用することにより解の質および計算時間の観点から良好な結果が得られている。

同様に、 $r$  の値は10から300までの範囲を調査し、その結果、 $r$  の初期値は50から100が適正であることが判明した。また、 $tau$  は1.0から1.2の範囲を調べた結果1.15以上になると比較的計算時間が大きくなることから、1.1を採用する。

## 5. 数値実験

最初に、提案した確率的複合近傍移動が、単純に構成した近傍移動より効果的であることを実験により示す。次に、この確率的複合移動を用いたSimulated Annealing法に基づく算法が他の戦略より優れた結果を導くことを実験の結果より示す。

### 5.1 確率的複合移動の効果

確率的複合移動を用いて構築した提案算法と以下で述べる3つの近傍移動を用いた戦略の効果を数値実験

表1 各種戦略に対する数値結果

Table 1 Computational result for various strategies.

	100	200	300	400	500
algo1	2623	5741	9145	12051	16468
algo2	2645	5993	9763	13282	17795
algo3	2669	5856	9323	12373	16879
algo4	2916	6601	10709	14481	19367

的に検討する。代表的な実験例として、頂点数100から500のランダムグラフに対する結果を表1に示す。このグラフの辺数は頂点数の3倍とする。そこに示される値は10回<sup>2)</sup>の実験で求まる解のコストの平均値をそれぞれ示したものである。また、その他のグラフに対する実験でも同様な結果が得られている。

まず、今回提案する近傍移動による算法を algo1 とする。これに対して、頂点をできうる限り一列化グラフ上で遠くへ移動する提案の効果を実証するために、頂点をその頂点が現在存在するブロックに隣接するブロックへ移動する短距離的な移動を採用した近傍移動による戦略 algo2 と比較する。次に、複数回連続的に頂点を移動する多重頂点移動の効果を測るために、ランダムに選択した頂点を左右交互に一度のみ移動する近傍を用いた algo3 との比較を試みる。最後に局所的な段階で確率的要素を採用し Simulated Annealing 法を構成した本算法の性能を示すために、解そのもののコストを基準として確率的採択を行う基本的な Simulated Annealing 法を構成し比較する。これを algo4 とする。以上の戦略により求まる複数のコストを平均し、解の質を判定する。このときの各戦略の計算時間はほぼ等しい結果である。

表1の実験結果より、algo1の優越性は明らかであり、他の近傍移動、および基本的な Simulated Annealing 法の採用に対して、今回提案する確率的複合移動の本問題に対する効果が示された。

さらに、頂点数400のランダムグラフを対象とし、提案算法の受理率とコストの温度 ( $tmp$ ) に対する変化の様子を図6、図7に表す。この曲線は基本的な Simulated Annealing 法の変化<sup>1),2)</sup>と同様な傾向を示しており、提案算法が Simulated Annealing 法の特性に従って挙動していることを示している。

### 5.2 Tabu Search 法との比較

Simulated Annealing 法は近傍構造をベースとする近似解法である。これと同様に近傍構造をベースとする近似解法として Tabu Search 法<sup>7)~9),11)</sup>があり、昨今、種々の問題で優れた効果を示している<sup>14),16)</sup>。本問題に対しても今回採用した複合移動を用いて効果的な近似解が得られている<sup>13)</sup>。ただし、この場合、頂点

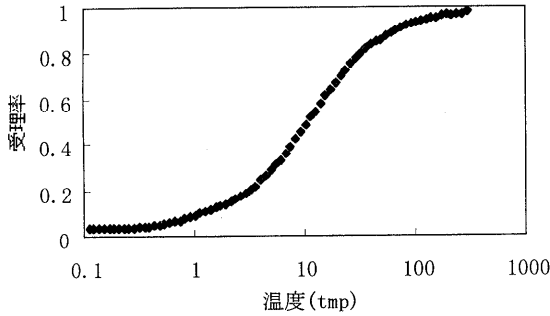


図6 受率と温度

Fig. 6 Acceptance ratio vs. temperature.

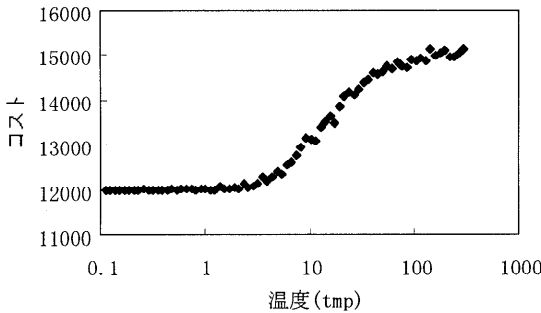


図7 コストと温度

Fig. 7 Cost vs. temperature.

の移動は短距離的な移動を採用することが望ましい。また、確率的要素は当然採用していない。この Tabu Search 法と今回提案した Simulated Annealing 法に基づく解法による解のコストと計算時間を表 2 (表 1 と同一なランダムグラフを使用) で比較することにより、提案算法の優越性を示す。また、計算時間の増加傾向に関しては特に図 8 で示す。両者のコストの質を比較するため、Tabu Search 法においても同一の反復回数となるよう設定しておく (Tabu Search 法で求まるコストは決定的な値である)。Simulated Annealing 法による値は 10 回導出されたコストの平均である。また、その中での最良値と最悪値を付記しておく。ただし、ここで用いられるパラメータは上記の推奨値に基づき、 $initial\ temperature = 40$ ,  $stoptmp = 2.0$ ,  $r = 100$ ,  $\phi = 0.9$ ,  $\tau = 1.1$  とし、両者の反復数を 14510 回と一致させる。計算環境は CPU が Intel 486DX2 および OS として Linux を採用している。

表 2 より、本問題に対して Simulated Annealing 法に基づく今回提案した戦略が Tabu Search 法と比較し、解の質が改善され、最悪値においても Tabu Search 法のコストを改善する結果となった。さらに、計算時間に関する実験では基本的に両者とも大差はないが若干

表 2 頂点数による影響と計算時間

Table 2 Effect and running time by increasing number of nodes.

	100	200	300	400	500
Annealing	2023	5741	9145	12051	16468
最良値	2602	5669	9068	11960	16329
最悪値	2652	5828	9209	12163	16578
time (sec)	53	117	176	243	317
Tabu	2675	5910	9260	12414	16979
time (sec)	56	120	186	251	322

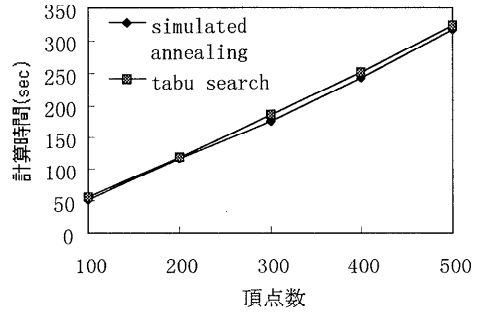


図8 頂点数と計算時間

Fig. 8 Number of nodes vs. computing times.

提案算法の方が低い値を示す。また、両算法とも頂点数に線形な計算時間を要することが示される。この場合、精度を上げるため比較的多めに反復を行っているが、パラメータを  $r = 50$  (これは上記のパラメータの推奨値の範囲である) とすることにより、解の質をほとんど低下させることなく、反復数および計算時間をほぼ 1/2 程度に下げることが可能である。その他のグラフにおいても以上の傾向が示され、提案算法の有効性が示された。

## 6. おわりに

本論文では無閉路有向グラフの系列分割問題に対する Simulated Annealing 法の適用とその有効性について検討した。本問題は系列性を保持した多分割問題であり、成分集合の個数、各成分集合の要素数は不定である複雑なグラフ分割問題であるため、通常の近傍移動のみでは近似度のより良い結果は望めない。これに対して、本論文では一列化グラフとブレイク・ポイントの列を用いたデータ構造を利用し、近傍移動の設計では本問題の構成に合わせて変形した left-to-right 移動、および right-to-left 移動を多重的に適用し、さらに部分的な最適化を取り入れ、複合的な処理方法を用い効果的な近傍移動を実現した。さらに、より計算しやすい採択基準を用いた局所的なコスト差による確率的採択を採用し効果をはかっている。従来の近傍移動

に対して、この複合的な近傍移動の採用により解に大きな変化をほどこし Simulated Annealing 法の性能をより強く引き出し、その特徴も維持する結果となった。

また、今回考案した複合移動の本問題への効果と、Tabu Search 法による解法との比較によって本算法の性能を数値実験により明らかにした。それによると、確率的複合移動の有効性が示されると同時に、提案算法は Tabu Search 法により求まる解より良質な解が導かれ、その計算時間もグラフの構造などに影響されることなく、ほぼ頂点数に線形に増加する傾向が示された。

今後の課題として、目的関数、制約条件を変化させることによって構成できる種々の問題への Simulated Annealing 法の適用および効果について検討を試みたい。

### 参考文献

- 1) Aarts, E., Korst, J. and Laarhoven, P.: A Quantitative Analysis of the Simulated Annealing Algorithm: A Case Study for the Traveling Salesman Problem, *Journal of Statistical Physics*, Vol.50, pp.189-206 (1988).
- 2) Aarts, E. and Korst, J.: *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons (1989).
- 3) Betts, J. and Mahmoud, K.I.: A Method for Assembly Line Balancing, *Engineering Costs and Production Economics*, Vol.18, pp.55-64 (1989).
- 4) Drexel, A.: A Simulated Annealing Approach to the Multiconstraint Zero-One Knapsack Problem, *Computing*, Vol.40, pp.1-8 (1988).
- 5) Johnson, D., Aragon, C., McGeoch, L. and Schevon, C.: Optimization by Simulated Annealing: An Experimental Evaluation; Part I, Graph Partitioning, *Operations Research*, Vol.37, No.6, pp.865-892 (1989).
- 6) Kernighan, B.W.: Optimal Sequential Partitions of Graphs, *J. ACM*, Vol.18, No.1, pp.34-40 (1971).
- 7) Glover, F.: Tabu Search Part I, *ORSA J.C.*, Vol.1, No.3, pp.190-206 (1989).
- 8) Glover, F.: Tabu Search Part II, *ORSA J.C.*, Vol.2, No.1, pp.4-32 (1990).

- 9) Reeves, C.R.: *Modern Heuristic Techniques for Combinatorial Problems*, BlackWell (1993).
- 10) Salvesson, M.E.: The Assembly Line Balancing Problem, *Journal of Industrial Engineering*, May-June, pp.18-25 (1955).
- 11) 茨木：離散最適化法とアルゴリズム (岩波講座応用数学), 岩波書店 (1993).
- 12) 加地, 大内：最適系列分割問題に対する効率的分枝限定法の構築と諸特性解析, 情報処理学会論文誌, Vol.35, No.3, pp.364-372 (1994).
- 13) 加地, 大内：Tabu Search 法による無閉路有向グラフの最適系列分割問題の解法, 電学論 (C), Vol.116-C, No.10, pp.1149-1157 (1996).
- 14) 久保：メタヒューリスティックス, 離散構造とアルゴリズム IV, 近代科学社, pp.171-230 (1995).
- 15) 西関, 秋山：グラフとダイグラフの理論, 共立出版 (1981).
- 16) 藤沢, 久保, 森戸：Tabu Search のグラフ分割問題への適用と実験的解析, 電学論 (C), Vol.114, No.4, pp.430-437 (1994).

(平成 8 年 5 月 29 日受付)

(平成 9 年 9 月 10 日採録)



加地 太一 (正会員)

1960 年生。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, 電気学会, 日本経営工学会各会員。



大内 東 (正会員)

1945 年生。1974 年北海道大学大学院工学研究課博士課程修了。工学博士。北海道大学工学部システム情報工学専攻教授。システム情報工学, 複雑系工学, 調和系工学, 医療システムの研究に従事。電気学会, 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, 医療情報学会, 病院管理学会, IEEE-SMC 各会員。