電気学会論文誌 C（電子・情報・システム部門誌）
IEEJ Transactions on Electronics, Information and Systems
Vol.135 No.2 pp.254–264　DOI: 10.1541/ieejeiss.135.254

論　文

解空間の階層構造に基づく組合せ最適化手法

正　員　落合　広樹[*]　正　員　金澤　貴彦[**]
正　員　田村　健一[*]　フェロー　安田恵一郎[*]

Combinatorial Optimization Method Based on Hierarchical Structure in Solution Space

Hiroki Ochiai[*], Member, Takahiko Kanazawa[**], Member, Kenichi Tamura[*], Member, Keiichiro Yasuda[*], Fellow

（2013 年 12 月 27 日受付，2014 年 9 月 12 日再受付）

In this paper, we introduce a new concept of hierarchical structure into solution space of a combinatorial optimization problem, and develop a novel combinatorial optimization method based on the concept. The introduction of the above new hierarchical structure concept “basin of attraction”, which is a set binding solutions by utilizing properties of local optimal solutions, enables us to construe solution space as not only set of solutions but also set of basins of attraction hierarchically. It is well known that the appropriate balance of two policies, intensification and diversification, is essential in the search of meta-heuristics. The proposed method clarifies the search policy by relating the hierarchical structure in solution space with intensification and diversification. In regard to diversification, we incorporate a movement strategy that has longer-term or more macroscopic viewpoint than before in the algorithm by utilizing the concept basin of attraction. The performance of the proposed combinatorial optimization method is inspected by numerical experiments using some typical benchmark problems of a traveling salesman problem, a knapsack problem, a flow-shop scheduling problem and a quadratic assignment problem.

キーワード：組合せ最適化，メタヒューリスティクス，近接最適性原理，局所探索法，集中化，多様化
Keywords: Combinatorial Optimization, Meta-heuristics, Proximate Optimality Principle, Local Search, Intensification, Diversification

1. はじめに

近年のシステムの大規模化・複雑化，システムの設計・運用・制御に対する要求の高度化に伴い，高い探索性能と汎用性を兼ね備えた最適化手法の開発は重要な課題となっている。良く知られているように，施設配置問題，配送計画問題，およびシステム運用のスケジューリング問題など，離散的構造を有する多くの実問題は，組合せ最適化問題として定式化できる。しかしながら，多くの組合せ最適化問題は，将来的に計算機の性能が飛躍的発展を遂げることを考慮したとしても，厳密な最適解を得るためには天文学的時間を要することが知られ，いわゆる $\mathcal{NP}$ 困難問題に位置付けられる[(1)(2)]。

一方，現実の問題においては，厳密な最適解を求めること以上に，実用的な時間内に十分な最適性を有する解（準最適解）を求めることが必要とされている。加えて，近年のコンピュータパワーの著しい向上に伴い，比較的低コストで長時間計算機を稼働させることが容易な状況が生まれ，最適化アルゴリズムおよび計算機を利用したシミュレーションなど様々な数値計算において，その恩恵を享受することが可能になった。

以上のような最適化分野を取り巻く環境：①対象の大規模化・複雑化，②計算時間の制約，③周辺技術の発展，などの変化に応じてメタヒューリスティクスという新たなパラダイムが誕生した[(3)(4)]。メタヒューリスティクスとは，経験的に有効性が知られているオペレーション，すなわちヒューリスティクスを有機的かつ柔軟に組み合わせた最適化手法であり，複雑なシステムへの応用を中心に注目されている。

メタヒューリスティクスの特徴の一つとして，自然現象や生物現象などにアナロジーを持った手法が多いことが挙げられる。例えば，遺伝的アルゴリズム[(5)(6)]は生物の進化，シミュレーティッド・アニーリング[(7)]は金属の焼きなまし

[*] 首都大学東京大学院
〒 192-0397　東京都八王子市南大沢 1-1
Tokyo Metropolitan University
1-1, Minami-Osawa, Hachioji, Tokyo 192-0397, Japan
[**] 九州電力（株）
〒 810-8720　福岡県福岡市中央区渡辺通 2-1-82
Kyushu Electric Power Corporation
2-1-82, Watanabe-Dori, Chuo-ku, Fukuoka, Fukuoka 810-8720, Japan

からアナロジーを得て提案された。多くの手法が，任意の時間で適用できる近似解法であり，対象の数式モデルなどを必要とせず，シミュレーション技術などと直接的に接続することが容易である。このような優れた柔軟性や汎用性は，メタヒューリスティクスの工学的な価値として挙げられる。著者らは，これらの特徴を踏まえた上で，自然現象などのアナロジーに立脚するのではなく，対象とする問題の解空間の特性や最適化手法として本質的に必要な戦略に着目することで組合せ最適化手法の構築および改良を行ってきた[19]~[21]。著者らは，このようなメタヒューリスティクスの研究におけるアナロジーに立脚したアプローチと最適化手法の基本構造に基づくアプローチの2つは独立した別の概念とは考えていない。アナロジーに立脚して開発されたメタヒューリスティクスの解析により，メタヒューリスティクスの最適化手法としての構造解析が実現され，この解析の知見を活用することで，新たなメタヒューリスティクスの開発はもちろんのこと，アナロジーに立脚して開発されたメタヒューリスティクスの改良も実現できるなど，双方のアプローチが互いに有機的な結合を持つことでメタヒューリスティクスの改良・開発が達成できると考えている。

ところで，メタヒューリスティクスにおける最適化の過程は解空間内の探索である。そして，優れたメタヒューリスティクスを構築するにあたり，探索の方針である集中化と多様化が重要なポイントとなる。集中化とは，解空間の限られた範囲を重点的に探索する，短期的な解の改善を目的とする探索方針である。多様化とは，限られた範囲に留まることを防ぎ，広い範囲を探索する，長期的な解の改善を目的とする探索方針である。例えば，メタヒューリスティクスの代表的手法である Tabu Search[8][10][11]は，Local Search[10][11][22]に基づく強力な集中化のメカニズムに加え，探索履歴を保持することで後戻りを禁止する多様化のメカニズムを備えている。そして，それぞれのメカニズムが互いに影響を及ぼし合いながら，バランスを取っている。しかし，既存の手法における多様化のメカニズムは過度な集中化の抑制といった意味合いが強く，長期的な観点から探索の方向付けを行っていない場合がほとんどである。

このような背景を踏まえ，著者らは従来の解空間における近傍レベルの移動戦略に加えて，Local Search に基づく新たな概念により構成される大域的構造において長期的な解の改善を可能とする移動戦略が構築可能なことを示した[20]。そして，解空間の新たな構造における移動戦略に基づく組合せ最適化手法構築の基礎的検討を行ない，巡回セールスマン問題を用いた数値実験により，Tabu Search を上回る探索性能を有することを報告した[20]。

本論文では，著者らのこれまでの研究成果[20]を踏まえ，導入した新たな概念 “引き込み領域” を，最良移動戦略の Local Search を適用した際に，到達する局所的最適解が一致する解同士の集合として明確に定義付け，解空間を引き込み領域の集合として再解釈できることを明らかにする。さらに，このような新たな解釈に基づくことで，個々の解と引き込み領域の双方から解空間を捉えることが可能となり，解空間の階層構造を探索に利用可能となることを示す。さらに，本論文では，Local Search を基礎とする Iterated Local Search や Tabu Search には存在しない「解同士の距離」を，距離の公理を満たすように定義・導入する。既に探索した局所的最適解から「解同士の距離」が増加するように近傍を選択し，移動を行うことで,「局所的最適解からの距離が増加するような移動」が決定論的かつ直接的に実現できることとなる。このことは，移動戦略において提案手法が Iterated Local Search[3][10]や Tabu Search と本質的に異なる点である。

これらの考察を踏まえ，引き込み領域により構成される上位構造の導入と，解同士の距離の定義・導入によって新たに定義された解空間の階層構造に基づき，下位構造における移動戦略を集中化，上位構造における移動戦略を多様化と対応付けることで，それぞれの探索方針を明確化した，新たな組合せ最適化手法を提案する。そして，多様な組合せ最適化問題を用いた数値実験により，提案手法の探索性能と汎用性を確認する。

なお，本論文の提案手法は，著者らの研究成果[20]における移動戦略に新たな機構を付加することで探索性能の維持・向上を実現すると同時に，巡回セールスマン問題のみならず，他の代表的な組合せ最適化問題であるナップサック問題，フローショップスケジューリング問題，二次割当問題への適用を可能としている。

以下本論文では，2 章において本論文で対象とする組合せ最適化問題の構造と諸定義を明確にする。続いて，3 章において，著者らがメタヒューリスティクスの基本戦略と考える近接最適性原理と集中化・多様化に関する考察を行い，4 章では，解空間の階層構造に基づく新たな組合せ最適化手法を提案する。最後に 5 章では，典型的なベンチマーク問題を用いた数値実験を通じて，提案する組合せ最適化手法の有用性を検証する。

2. 組合せ最適化問題と関連する諸定義

組合せ最適化手法の開発を行う上で必要となる，組合せ最適化問題の定式化，関連する諸定義をまとめた上で，本論文で導入する組合せ最適化問題の解空間における新たな距離を定義する。

〈2·1〉 組合せ最適化問題の定式化　最適化問題は一般に以下のように定式化できる。

$$\begin{array}{lll} \text{min (or max)} & f(\boldsymbol{x}) & (f : \text{Objective Function}) \\ \text{subj. to} & \boldsymbol{x} \in \boldsymbol{F} & (\boldsymbol{F} : \text{Feasible Region}) \end{array}$$

ここで，$f(\boldsymbol{x})$ は目的関数，あるいは評価関数と呼ばれ，$f : \boldsymbol{F} \to \mathbb{Z}$ (or $\mathbb{R}$) なる写像である。$\boldsymbol{x} \in \boldsymbol{F}$ は実行可能解，あるいは単に解と呼ばれる。そして，解がグラフ理論や順列に代表される組合せ的構造を有する問題が組合せ最適化問題に分類される。以下に，組合せ最適化における諸定義を示す。

定義 1　大域的最適解集合

実行可能領域内の全体において最適である解（大域的最適解）の集合で，次式で表現できる。

$$F^\star = \{x \in F \mid f(x) \le f(y),\ \forall y \in F\}$$

定義 2　局所的最適解集合

実行可能領域内の特定部分において最適である解（局所的最適解）の集合で，次式で表現できる。

$$F^* = \{x \in F \mid f(x) \le f(y),\ \forall y \in N(x)\}$$

ここで，$N(x)$ は x における近傍である。

定義 3　近傍

近傍 N は次の写像として定義される。

$$N : F \to 2^F$$

この写像は，特定の解 x に対して，任意の部分集合を充てることができる。しかし，メタヒューリスティクスの探索においては一般に，特定の解 x に対して，何らかの意味で近い，似通った構造の解集合と位置付けられる。実用的には，x に摂動を加えた結果生成される解集合として，問題ごとに定義されることが多い。

〈2·2〉 組合せ最適化問題の解空間における距離の導入

組合せ最適化問題の解空間は，実数値ベクトルを解とする連続型最適化問題に比べ，直観的な把握が困難であり，これが組合せ最適化問題に対する理解を妨げる一因となっている。解空間の把握が困難である主な理由として，一般に解同士の類似性に関する尺度がなく，位置関係を定量的に測ることができないことが挙げられる。

そこで，組合せ最適化問題の解空間に距離の概念[12][13]を導入する。集合・位相空間論においては，距離の公理† を満たす任意の集合に距離を定めることができ，以下のように定式化できる。

空間 X に対して，距離の公理を満足する写像

$$d\ :\ X \times X \to \mathbb{R}$$

が定義できるとき，d を空間 X の距離関数あるいは単に距離と呼び，対 (X, d) を距離空間と呼ぶ。

距離の概念は，公理さえ満たせば導入できる柔軟な枠組みであり，多くの組合せ最適化問題の解空間においても距離関数が定義できる。従って，距離関数を探索に利用することは，適用できる問題の範囲，すなわち汎用性に対する影響を最小限に留めながら，探索状況の把握や探索オペレーションの設計を容易にする。以下に本論文で用いる距離空間に基づく諸定義を示す。

定義 4　球体

球体 S（あるいは単に球）は，空間上のある一点から一定の距離内にある全ての点の集合であり，次式で定義される。

$$S(x, \varepsilon) = \{y \in F \mid d(x, y) < \varepsilon\}$$

定義 5　直径

距離空間の部分集合 X における直径 $\mathrm{diam}(X)$（あるいは単に径）は，その集合内に含まれる二点の距離の上界であり，次式で定義される。

$$\mathrm{diam}(X) = \sup\{d(x, y) \mid \{x, y\} \in X\}$$

直径の定義から，明らかに以下の定理が成り立つ。

$$X \subset Y \Rightarrow \mathrm{diam}(X) \le \mathrm{diam}(Y)$$

ところで，組合せ最適化問題における距離関数は，各問題の解の構造や近傍定義などの特性を踏まえ，適切に定義することが望ましい。著者らはこれまでに，さまざまな組合せ最適化問題の解の構造や近傍定義などの特性を踏まえた距離関数の定義に関する検討を行ってきた[19]~[21]。本論文では，これらの検討結果を踏まえ，ベンチマーク問題として扱う巡回セールスマン問題（TSP）[14]，0-1 ナップサック問題（0-1 KP）[15]，フローショップスケジューリング問題（FSP）[16][23]，二次割当問題（QAP）[3]における距離関数を以下のように定義する。

TSP の距離定義

TSP において，解は n 頂点の Hamilton 閉路[18]として表現される。距離関数は Hamilton グラフの共通しない辺数として次式で定義する。

$$d_{\mathrm{TSP}}(x, y) = \mid G(x) \setminus G(y) \mid$$

ここで，$G(x)$ は Hamilton グラフ x の辺集合である。

0-1 KP の距離定義

0-1 KP において，解は n bit の二値ベクトルとして表現される。距離関数は Hamming 距離[13]を用いて，次式で定義する。

$$d_{\mathrm{KP}}(x, y) = d_{\mathrm{H}}(x, y) = \mid \{i \mid x_i \ne y_i (i = 1, 2, \cdots, n)\} \mid$$

ここで，x_i は二値ベクトル x の i 番目の要素である。

FSP の距離定義

FSP において，解は n 個の製品における順位として表現できる。距離関数は Spearman's Footrule[13][24]を用いて，次式で定義する。

$$d_{\mathrm{FSP}}(x, y) = d_{\mathrm{SF}}(x, y) = \sum_{i=1}^{n} \mid x(i) - y(i) \mid$$

ここで，$x(i)$ は順位 x の対象 i の位置番号である。

QAP の距離定義

QAP において，解は n-順列として表現できる。距離関数は Cayley 距離[13]を用いる。順列 y に対する順列 x の Cayley 距離は，任意の要素対を入れ替える操作により x から y へ変換するために，必要となる最小操作回数で定義される。置換 $(x^{\mathrm{T}}, y^{\mathrm{T}})^{\mathrm{T}}$ の巡回置換 (cycle) の数 $\mid T((x^{\mathrm{T}}, y^{\mathrm{T}})^{\mathrm{T}}) \mid$ を用いて以下で算出できる。

$$d_{\mathrm{QAP}}(x, y) = d_{\mathrm{C}}(x, y) = n - \mid T((x^{\mathrm{T}}, y^{\mathrm{T}})^{\mathrm{T}}) \mid$$

† 任意の $x, y, z \in X$ において
(A)　正値性：$d(x, y) \ge 0$
(B)　非退化性：$d(x, y) = 0 \Leftrightarrow x = y$
(C)　対称性：$d(x, y) = d(y, x)$
(D)　三角不等式：$d(x, z) \le d(x, y) + d(y, z)$
が成立するとき，d を X 上の距離という。

3. メタヒューリスティクスの戦略

最適化アルゴリズムにおける探索では，対象の最適化問題の特性を抽出し，利用することが重要となる。従って，適用範囲に対して優れた汎用性を有する最適化アルゴリズムを構築するためには，多くの最適化問題に共通する構造を抽出することが必要である。以下では，本論文が立脚している近接最適性原理と集中化・多様化の概念，および Local Search について取りまとめる。

〈3·1〉 近接最適性原理（POP） 多くの組合せ最適化問題において，“良い解同士は似通った構造をもっている”という特性が存在することが経験的に知られている。この特性は，近接最適性原理（Proximate Optimality Principle：POP）[(4)(10)] と呼ばれる。メタヒューリスティクスに分類される手法には，探索過程で得られた良い解の情報を活用して探索を進めるという共通した基本戦略が存在する。メタヒューリスティクスが様々な最適化問題に対して良好な結果を残しているのは，この基本戦略が POP を根幹としているからだと考えられる。

POP は漠然とした原理であり，“良い解”，“似通った構造”といった表現に曖昧さを有している。探索において，POP を明確に利用するためには，“解の良さ”，“解同士の類似性”を定量的に評価できる尺度を設けることが要求される。本論文では，評価値（目的関数値），距離を尺度として，POP を以下のように再解釈する。

POP：“評価値の優れた解同士は距離が近い”

著者らは，この特性を念頭に置いて，新たな最適化アルゴリズムの構築を志向する。

〈3·2〉 集中化と多様化 メタヒューリスティクスの探索は POP を根幹としていると考えることができる。そして，重要な探索の方針として，集中化と多様化が挙げられる。

(1) **集中化** 現時点で得られている良い解の情報を基に，さらに良い解があると予想される範囲の探索を強める方針である。主として狭い範囲に探索を絞り込み，短期的な解の改善を目的とする。探索アルゴリズムにおける集中化のメカニズムは，最終的に得られる解の精確さに寄与する。しかし，集中化のみでは，限られた（短期的，あるいは局所的）情報のみから生成される範囲内の下限を得られるに過ぎない。

(2) **多様化** 探索範囲や利用情報が偏ることを防ぎ，広い範囲に探索を向ける方針である。長期的な解の改善を目的とし，解を改善していくための基本的な土台を築く。これは多様化のメカニズムにより，大規模な移動や多様な情報の収得ができれば，集中化のメカニズムによって得られる解の下限が変化するためである。しかし，多様化のみでは，良好な領域への移動などに成功したとしても，その領域の下限を求める能力が欠けるため，十分な最適性を有する解を得ることができない。

つまり，集中化と多様化は相補う関係にあると考えられる。優れた探索を実現するためには，集中化あるいは多様化のいずれかが突出するのではなく，それぞれの長所を最大限発揮できるような適切なバランスを持つことが望ましい。既存のメタヒューリスティクスの多くにおいて，集中化と多様化のメカニズムは連動的に働き，どちらかに偏った探索となることを防いでいる。これは同時に，それぞれのメカニズムは相互に抑制し合ってしまうことも意味している。

また，多様化のメカニズムは，未探索範囲への移動や多様な情報の収集という観点よりも，単に過度な集中化の抑制と捉えられていることが多い。しかし，著者らは，より長期的な視点に立った解の改善を実現することが，望ましい多様化のあり方であると考えている。著者らが考える長期的視点に立った移動戦略とは，

- 集中化のみでは到達し得ない未探索範囲への移動戦略
- 解改善の基本的な土台として有益な解の存在が期待できる範囲への移動戦略

の 2 つである。

以上を踏まえ，本論文では，集中化と多様化，それぞれの方針に特化したオペレーションを設け，探索状況に応じて交互に切り替えることで優れた探索の実現を目指す。

〈3·3〉 Local Search ここで，メタヒューリスティクスにおける基本的な移動戦略である Local Search（LS）† について紹介する。LS は，強力な集中化のメカニズムを有する戦略であり，定められた近傍での移動を繰り返し，局所的最適解を発見する。最良移動戦略の LS のアルゴリズムは以下のとおりである。

【Local Search のアルゴリズム】

Step 0 :[準備] 探索開始点 $\boldsymbol{x}^0$ を入力する。$t = 0$ とする。

Step 1 :[終了判定] $\boldsymbol{x}^t$ が局所的最適解ならば，探索を終了し，$\boldsymbol{x}^t$ を出力する。さもなければ Step 2 へ進む。

Step 2 :[更新] $\boldsymbol{x}^{t+1} = \arg\min\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in N(\boldsymbol{x}^t)\}$ とする。$t := t + 1$ として Step 1 へ戻る。

本論文では，後述の提案手法において，LS を活用する。また，入力 $\boldsymbol{x}$ に対する LS の出力を $\boldsymbol{y}$ としたとき，LS の演算を以下のように表記する。

$$\boldsymbol{y} = \mathcal{LS}(\boldsymbol{x})$$

4. 解空間の階層構造に基づく組合せ最適化手法

最良移動戦略の Local Search を適用した際に，到達する

† LS の代表的な移動戦略として，以下が挙げられる。
(A) 最良移動戦略：近傍 $N(\boldsymbol{x}^t)$ 内すべての解を評価し，最良解に移動する
(B) 即時移動戦略：近傍 $N(\boldsymbol{x}^t)$ をランダムな順序で評価し，最初に見つかった改善解へ移動する

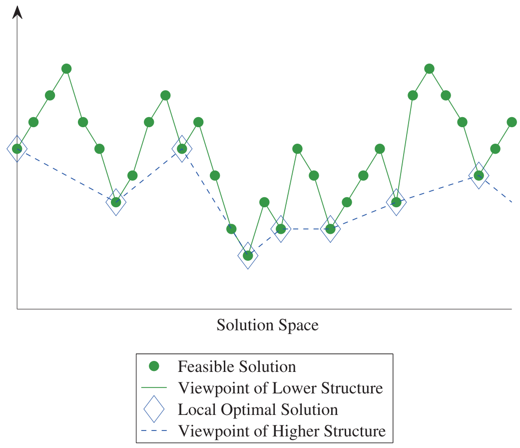


Fig. 1.　Hierarchical Structure in Solution Space

局所的最適解が一致する解同士の集合として定義できる「解空間における上位概念」を導入する。続いて，この上位概念の導入によって新たに定義された解空間の階層構造に基づき，下位構造における移動戦略を集中化，上位構造における移動戦略を多様化と対応付けることで，新たな組合せ最適化手法を提案する。

〈4·1〉　解空間における上位概念の導入　　新たな手法を構築する上で，集中化の方針に特化したオペレーションは，最良移動戦略の LS により実現できる。一方で，多様化の方針に特化したオペレーションの実現は，容易でない。一般に，解空間のある一定範囲を定める定義がなく，広範囲の探索の方針を立てることが困難である。また，未探索範囲への移動を実現するためには長期的な探索履歴の活用を必要とする一方，膨大な量の情報を参照することは計算時間の観点から好ましくない。

そこで，著者らは組合せ最適化問題の解空間において，新たに個々の解同士を括る上位概念として，“引き込み領域 (Basin of Attraction)” を導入する。引き込み領域は，最良移動戦略の LS により到達する局所的最適解に着目した同値関係[12][17]から定義づける。上位概念の導入により，解空間をマクロに捉えることが可能となり，引き込み領域の集合として再解釈できる。そして，従来の個々の解の集合としての捉え方と併せた，解空間の階層構造を探索に利用していく。著者らが考える解空間の階層構造を Fig. 1 に示す。

上位概念，引き込み領域について，より具体的に以下に記述する。

$\boldsymbol{x}$ および $\boldsymbol{y}$ に，最良移動戦略の LS を適用した場合，同じ局所的最適解に到達するという，以下の関係

$$\boldsymbol{x} \sim \boldsymbol{y} : \mathcal{LS}(\boldsymbol{x}) = \mathcal{LS}(\boldsymbol{y})$$

が同値律† を満足していることは容易に確認できる。従って，上記の関係 ~ によって解空間 $\boldsymbol{F}$ を直和分割（類別）することができる。同値類の代表元は一般に任意に決定できるが，ここでは到達する局所的最適解 $\boldsymbol{x}^*$ とする。これは，同値類の中でも局所的最適解が以下の性質を有するためである。

(1)　同値類における最良解であり，下限値を与える
(2)　同値類の中の任意の点から容易に求めることができる
(3)　同値類内の構造は単峰性であり，同値類の他の解より相対的に中心に位置することが期待できる

このとき，局所的最適解 $\boldsymbol{x}^*$ を代表解とする同値類（引き込み領域）は，$\boldsymbol{B} \rightarrow 2^{\boldsymbol{F}}$ なる写像を用いて，以下のように表現できる。

$$\boldsymbol{B}(\boldsymbol{x}^*) = \{\boldsymbol{x} \in \boldsymbol{F} \mid \boldsymbol{x} \sim \boldsymbol{x}^*\}$$

このように，引き込み領域の概念を導入し，解空間を階層化して整理することで，

- 集中化：引き込み領域内の移動戦略
- 多様化：引き込み領域間の移動戦略

という形で集中化および多様化の方針を移動戦略と対応付けることができ，両者の役割がより明瞭になる。

〈4·2〉　階層構造を利用する移動戦略　　本論文では，解空間に対して，引き込み領域の集合としての解釈を上位構造，個々の解の集合としての解釈を下位構造と表現する。

(1) **集中化：引き込み領域内の移動戦略**　引き込み領域内の移動は，下位構造における移動に相当する。探索の方針は，集中化とする。引き込み領域の構造が単峰性であるため，下限値をもつ局所的最適解を得た時点で，現在の引き込み領域が探索済みであるとみなすことができる。従って，具体的な移動戦略としては，最良移動戦略の LS により速やかに局所的最適解を求めることとする。

(2) **多様化：引き込み領域間の移動戦略**　引き込み領域間の移動は，上位構造における移動に相当する。探索の方針は，多様化とし，これまでに探索していない引き込み領域への移動を狙う。そのためには，まず現在の引き込み領域から脱出する必要がある。具体的には，現引き込み領域の代表元から離れるように近傍を制限し，移動を方向付ける。

さらに，これだけでは既に探索した引き込み領域へ戻る可能性があるため，それらからも徐々に離れるようなオペレーションを加える。ここで，直径の定義に基づく直径対集合を以下のように定義する。

定義　直径対集合

距離空間の部分集合 X における直径 $\mathrm{diam}(X)$ をなす点の集合族を直径対集合 $D(X)$ とし，次式で定義できる。

$$D(X) = \{\{\boldsymbol{x}, \boldsymbol{y}\} \in X \mid d(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{diam}(X)\}$$

ただし，X の元が $\boldsymbol{x}$ ただ一つの場合，$D(X) = \{\{\boldsymbol{x}, \boldsymbol{x}\}\}$ とする。

† 集合 X における関係 ~ について，以下の同値律
(A)　反射律：X のすべての元 $\boldsymbol{x}$ に対して，$\boldsymbol{x} \sim \boldsymbol{x}$
(B)　対称律：X の元 $\boldsymbol{x}, \boldsymbol{y}$ に対して，$\boldsymbol{x} \sim \boldsymbol{y} \Rightarrow \boldsymbol{y} \sim \boldsymbol{x}$
(C)　推移律：X の元 $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ に対して，$\boldsymbol{x} \sim \boldsymbol{y} \wedge \boldsymbol{y} \sim \boldsymbol{z} \Rightarrow \boldsymbol{x} \sim \boldsymbol{z}$
が成立するとき，~ は X における同値関係であるという。

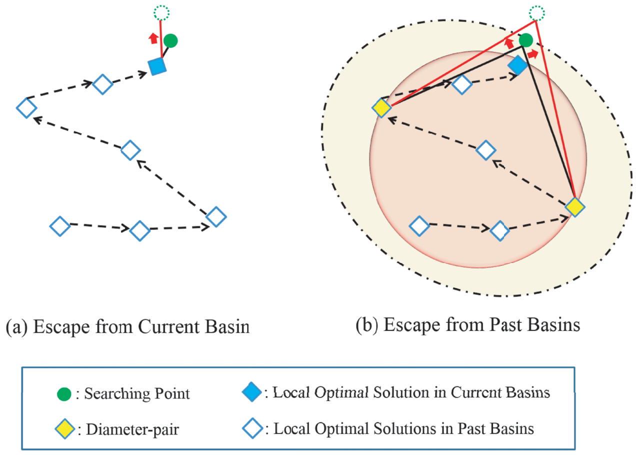


Fig. 2. Movement Strategy between Basins of Attraction

探索済みの局所的最適解の集合の直径対集合に着目する。直径対集合の要素は，探索済みの局所的最適解の集合のうち互いに最も遠い対である。直径対集合を球面の要素としたとき，他の探索済みの局所的最適解はその球に内包される。従って，直径対集合の要素から離れるように探索を方向付ければ，これまでに探索した引き込み領域への回帰を抑止できると考える。これは，Euclid 空間からの類推で考えれば，直径を構成する二つの局所的最適解を焦点とする楕円の外側へ向かうよう探索を方向付けている。

Fig. 2 (a) は，上述の引き込み領域の移動戦略の一つ目の移動制限である現引き込み領域から脱出するための制限のイメージを表している。一方，Fig. 2 (b) は，これまでに探索してきた引き込み領域から脱出するための制限のイメージを表している。Fig. 2 (a) の黒い線は，現時点での探索点（解）と，その探索点が直近で発見した局所的最適解との距離を表しており，Fig. 2 (b) の黒い線は，現時点での探索点（解）と，探索済みの局所的最適解集合の中で互いに最も遠い距離にある対との距離を表している。Fig. 2 (a) の赤の矢印は，現時点での探索点（解）から離れる移動方向を表しており，赤の線は赤の矢印の方向に移動した新たな探索点との距離を表している。Fig. 2 (b) の 2 つの赤い矢印は，それぞれ探索済みの局所的最適解集合の中で互いに最も遠い距離にある対から離れる方向を表しており，赤い線は最も遠い距離にある対から同時に離れる方向に移動した新たな探索点と，最も遠い距離にある対からの距離を表している。

また，現引き込み領域からの脱出判定であるが，評価値が改善された時点で脱出できたと判断する。これは，現引き込み領域の代表元から離れる移動の中で評価値が改善された場合，異なる引き込み領域に到達した可能性が高いと考えられるためである。

〈4·3〉 解空間の階層構造に基づく組合せ最適化手法

これまでに述べたオペレーションを組み合わせた「解空間の階層構造に基づく組合せ最適化手法」のアルゴリズムは，以下の通りである。

【解空間の階層構造に基づく組合せ最適化手法】

Step 0 :[準備]
最大反復回数 $T_{\max}$，脱出領域サイズ s を与える。

Step 1 :[初期化]
初期点 $\boldsymbol{x}^{(0,0)}$ を生成する。$t = 0$，$k = 0$ とする。ここで，t は反復回数のカウンタであり，k は領域間の移動における移動回数のカウンタである。

Step 2 :[領域内の移動]
$\boldsymbol{x}^{(t,*)} = \mathcal{LS}(\boldsymbol{x}^{(t,k)})$ とする。ここで，$\boldsymbol{x}^{(t,*)}$ は局所的最適解である。

Step 3 :[終了判定]
$t = T_{\max}$ ならば，探索終了。さもなければ，$\boldsymbol{x}^{(t+1,0)} = \boldsymbol{x}^{(t,*)}$，$t := t+1$，$k = 0$ として Step 4 へ進む。

Step 4 :[領域間の移動]

Step 4-1 :[直径対集合の算出]
直径対集合の元 $\{\boldsymbol{x}_\alpha, \boldsymbol{x}_\beta\}$ を得る。
$t = 1$ ならば，$\boldsymbol{x}_\alpha = \boldsymbol{x}_\beta = \boldsymbol{x}^{(0,*)}$,
$2 \le t \le s+1$ ならば,
$\{\boldsymbol{x}_\alpha, \boldsymbol{x}_\beta\} = \boldsymbol{D}(\{\boldsymbol{x}^{(0,*)}, \boldsymbol{x}^{(1,*)}, \ldots, \boldsymbol{x}^{(t-2,*)}\})$,
$s+1 < t$ ならば,
$\{\boldsymbol{x}_\alpha, \boldsymbol{x}_\beta\} = \boldsymbol{D}(\{\boldsymbol{x}^{(t-(r+1),*)}, \boldsymbol{x}^{(t-r),*)}, \ldots, \boldsymbol{x}^{(t-2,*)}\})$
とする。ただし，直径対集合の元が複数存在する場合，より新しい局所的最適解を含む元を選択する。

Step 4-2 :[移動方向の限定]
$\boldsymbol{N}_1$，$\boldsymbol{N}_2$ を以下で構成し，$\boldsymbol{N}_{1,2} = \boldsymbol{N}_1 \cap \boldsymbol{N}_2$ を得る。

$$\boldsymbol{N}_1 = \{\boldsymbol{y} \in \boldsymbol{N}(\boldsymbol{x}^{(t,k)}) \mid d(\boldsymbol{x}^{(t,k)}, \boldsymbol{x}^{(t,0)}) < d(\boldsymbol{y}, \boldsymbol{x}^{(t,0)})\}$$

$$\boldsymbol{N}_2 = \{\boldsymbol{y} \in \boldsymbol{N}(\boldsymbol{x}^{(t,k)}) \mid d(\boldsymbol{x}^{(t,k)}, \boldsymbol{x}_\alpha) + d(\boldsymbol{x}^{(t,k)}, \boldsymbol{x}_\beta) < d(\boldsymbol{y}, \boldsymbol{x}_\alpha) + d(\boldsymbol{y}, \boldsymbol{x}_\beta)\}$$

ただし，$\boldsymbol{N}_{1,2} = \emptyset$ となる場合，$\boldsymbol{N}_{1,2} = \boldsymbol{N}_1$，または，$\boldsymbol{N}_{1,2} = \boldsymbol{N}_2$ とする。なお，$\boldsymbol{x}^{(t,0)} = \boldsymbol{x}^{(t-1,*)}$ である。

Step 4-3 :[領域間での移動]
$\boldsymbol{x}^{(t,k+1)} = \arg\min\{f(\boldsymbol{x}) \mid \boldsymbol{x} \in \boldsymbol{N}_{1,2}\}$ とする。$k := k+1$ とする。
$f(\boldsymbol{x}^{(t,k)}) < f(\boldsymbol{x}^{(t,k-1)})$，または，$\boldsymbol{N}_1 = \emptyset$ ならば，Step 2 へ戻る。さもなければ，Step 4-2 へ戻る。

上記のアルゴリズムにおける脱出領域サイズ s は，探索済みとして扱う局所的最適解の最大数を与えるパラメータである。提案手法における脱出領域サイズ s は，Tabu Search におけるタブーリストのサイズ，すなわちタブーリストレングスに相当する役割を担っている。著者らの数値実験を通じて，タブーリストレングスと同様に脱出領域サイズ $s = \infty$ と設定すると探索性能が低下すること，および適切な脱出領域サイズ s が存在することを確認している。

一方，アルゴリズム中，Step 2 が引き込み領域内の移動

戦略であり，Step 4 が引き込み領域間の移動戦略である。Step 4-2 および Step 4-3 では，未探索の領域への移動を進めながらも，$\boldsymbol{N}_1 \cap \boldsymbol{N}_2$ の中で最も評価値が優れている解へ移動することで，探索過程で構築してきた優れた解情報を徒に組み替えることを避け，より優れた解が存在する領域への移動を狙っている。Step 4-2 において，$\boldsymbol{N}_1$ が空集合となり，引き込み領域の代表解から遠ざかる近傍解が存在しなくなった場合は，脱出に十分な距離の移動を行ったと判断して，評価値の変動に関係なく領域間の移動を終了する。

ところで，局所探索のみでは初期値によって定まる局所的最適解に収束してしまうため，局所探索を基礎とする大域的最適化手法には，局所探索法の持つ初期値依存性を解消もしくは軽減する何らかの機構がアルゴリズムに埋め込まれている。Tabu Search においては，過去の探索履歴情報を記憶し，未探索領域への移動を可能とするためのタブーリストの存在と，タブーリストの活用による未探索領域への移動がこの機構に相当する。局所探索を基礎とする代表的な手法である Reactive Tabu Search[9] は，通常の Tabu Search では固定されているタブーリストレングスを探索状況に応じて調整することで，解くべき問題の構造や初期値などの影響に対する適応性を向上させた手法と位置付けられる。これに対して提案手法は，Reactive Tabu Search の持つような適応的な機能はなく，Tabu Search と同一のカテゴリに属する手法として位置付けられるが，提案手法における脱出領域サイズ s を適応的に調整するなどすることで，提案手法に適応的な機能を付加することも可能であると考えている。

5. 数値実験による検証

提案手法の性能を評価するために典型的なベンチマーク問題を用いた数値実験を行う。比較手法の選定や計算量に関する基準など，本論文における数値実験および検証の基本方針を述べた後，検証の具体的な方法をまとめ，最後に数値実験結果を示し，考察を行う。

〈5·1〉 数値実験および検証の基本方針　メタヒューリスティクスには，多様な手法とその改良版が存在する上に，それぞれのメタヒューリスティクスにはいくつもの調整すべきパラメータが存在するため，メタヒューリスティクスの研究においては，性能評価のための数値実験のあり方，とりわけ比較手法の選定は大変重要かつ難しい課題である。既存の多くの方法との公平な比較は極めて困難であると同時に，慎重な選定と比較が行われない場合には比較そのものが意味を持たなくなる場合もあり得るため，適切な類似手法との比較が重要となる。

本論文の提案手法と Tabu Search は，一般的な最適化手法の分類の観点からは，「単一目的の離散型最適化問題のための最適化手法で，アルゴリズムが単点探索かつ決定論的最適化手法」であり，同時に，「アルゴリズムのパラメータが固定型の最適化手法」として特徴づけられる。さらに，Tabu Search は，上記によって特徴づけられるメタヒューリスティクスの中でも，様々な問題に対する有効性が多数の研究者によって検証されている手法[3][4][8][10]である。これらのことを踏まえ，本論文では提案手法の比較手法として Tabu Search を選定することとする。

一方，アルゴリズムの実行時間は，使用するコンピュータの性能やプログラミングの技量などによって大きく変わることが知られている。さらに，多くのメタヒューリスティクスが決定変数情報と評価値情報のみを用いて探索を行う，いわゆる解直接探索型の手法であることを踏まえ，一般的には「同数の決定変数情報と評価値情報の下での性能評価」がメタヒューリスティクスのアルゴリズムの評価をする上で適切であるとされている[4][10]。本論文でも，このことを踏まえて，同数の決定変数情報と評価値情報の下での性能評価を行うこととする。

〈5·2〉 検証の具体的な方法　提案手法（Proposed Method）の性能を検証するために，提案手法と同様に LS に立脚した単点型最適化手法である Tabu Search と提案手法を TSP，0-1 KP，FSP，QAP に適用し，結果を比較した。

対象問題は，48, 107, 150 都市の TSP（TSPLIB の att48, pr107, ch150），100, 500, 1000 bit の 0-1 KP，20 製品 5 機械，50 製品 5 機械，50 製品 10 機械の FSP（Taillard's Instances の ta001, ta031, ta041），問題サイズが 30, 64, 81 の QAP（QAPLIB の Nug30, Sko64, Sko81）を用いた。近傍定義はそれぞれの距離定義に基づく近傍，TSP では 2-opt 近傍，0-1 KP ではハミング距離が 1 の近傍，FSP では Sperman's Footrule が 4 以内の近傍，QAP では交換近傍とした。比較対象の Tabu Search におけるタブーリストに記憶する属性を以下に示す。TSP において，記憶する属性は，取り除かれた枝とし，記憶された枝が一つでも加えられる操作を禁止する。0-1 KP において，記憶する属性は，変更された bit 番号とし，記憶された bit 番号が再度変更される操作を禁止する。FSP および QAP において，記憶する属性は，変更されたジョブ（QAP では施設）と処理番号（QAP では地点）の組とし，記憶されたジョブが組となる処理番号に再度変更される操作を禁止する。

その他の検証条件として，各問題の大域的最適解の評価値 $f(\boldsymbol{x}^\star)$，目的関数の呼出回数（Number of Function Calls），タブーリストレングス L_T，脱出領域サイズ s を Table 1 に示す。評価回数は探索中に評価値を呼び出した回数である。評価値の呼び出しは主な計算負荷となることから評価回数を探索時間の尺度と考える。評価回数は各問題で長短二つの場合を設定し，設定した評価回数以内で得られた結果を，二つの手法で比較する。タブーリストレングス L_T，および脱出領域サイズ s は，事前に数値実験を行い，問題，評価回数ごとに優れた結果が得られた値を選択している。

以上の条件で，対象ごとにランダムに生成した 50 の初期点から各手法を適用し，得られた評価値の平均値（Mean），最良値（Best），最悪値（Worst），標準偏差（SD）を求めた。

〈5·3〉 検証結果と考察　数値実験による検証の結果は，Table 2 の通りである。表中の括弧内の値は厳密な最適

Table 1.　Numerical Experiment Condition

		$f(\boldsymbol{x}^\star)$	Number of Function Calls		L_T		s	
			Case 1	Case 2	Case 1	Case 2	Case 1	Case 2
Traveling	att48	10628	1×10^6	3×10^6	20	18	4	3
Salesman	pr107	44303	1×10^7	3×10^7	36	36	12	4
Problem	ch150	6528	5×10^7	1.5×10^8	40	40	4	12
0-1	100 bit	2053	1×10^5	3×10^5	14	14	16	16
Knapsack	500 bit	10490	2×10^6	6×10^6	60	65	80	80
Problem	1000 bit	20900	2×10^7	6×10^7	75	70	190	190
Flow-shop	ta001	1278	1×10^5	3×10^5	22	22	6	2
Scheduling	ta031	2724	1×10^6	3×10^6	120	110	4	8
Problem	ta041	2991	2×10^6	6×10^6	45	35	12	10
Quadratic	Nug30	6124	5×10^5	1.5×10^6	12	12	3	3
Assignment	Sko64	48498	5×10^6	1.5×10^7	30	20	6	4
Problem	Sko81	90998	1×10^7	3×10^7	36	36	6	10

値に対する相対誤差率を示している。Tabu Search と提案手法（Proposed Method）のうち優れた結果にアスタリスク（*）を付している。

Table 2 の結果から，TSP，0-1 KP の全ての問題において，Tabu Search に比べ，提案手法が優れていることが確認できる。QAP では，問題や探索時間によっては Tabu Search に比べ，提案手法が劣る場合はあるものの，全体を見れば同等以上の結果が得られているといえる。とりわけ探索時間を長く取った場合には，全ての対象の平均値で Tabu Search を上回る結果が得られた。一方，FSP においては，ほとんどの場合で Tabu Search が提案手法を上回る結果となった。

以上の代表的な組合せ最適化問題に対する数値実験において，多くの場合で既存の代表的な手法に優る結果が得られていることから，提案手法の有用性が確認できる。比較手法の Tabu Search とは，近傍に基づく移動を行うこと，および探索履歴を活用することは共通していることから，解空間における上位概念を導入し，階層構造を利用することが有効な戦略であったと考えられる。

FSP の全般において，優れた結果を得ることができなかった原因は，解 - 評価値空間の構造上の特性，いわゆるランドスケープ(4)(10)にあると推測する。FSP は解空間全域における評価値の上下限の差が比較的小さい問題であり，探索過程の近傍レベルでは全ての評価値が一致することさえある。一方，提案手法の引き込み領域間の移動では，評価値の改悪および改善の遷移を，領域の境界線と対応付けて探索方針を切り替えている。したがって，領域脱出中に評価値が平坦な推移をとることが多く，適切なタイミングで探索方針の切り替えを行えなかったことが，結果に影響したと考えられる。

ここで，パラメータについて考察する。脱出領域サイズ s は探索済みとして扱う局所的最適解の最大数を与えるパラメータであるため，s が大きければ探索済みとして扱われる空間の直径も大きくなり，多様化が促進されると考えられる。Table 1 からわかるように，0-1 KP では，良い結果を得られた脱出領域サイズ s が他の問題に比べ，大きくなっている。これは近傍定義および局所的最適解の数が影響していると考えられる。今回の実験で用いた近傍の要素数は，

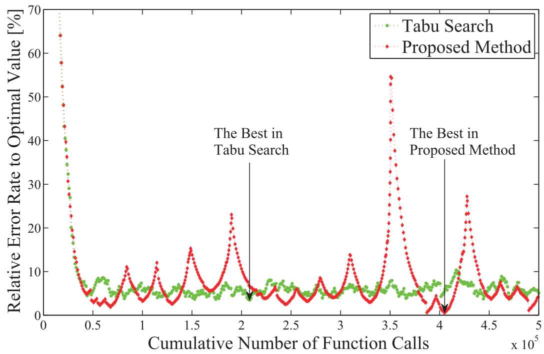


Fig. 3.　Transition of Objective Function Value

問題サイズ n としたとき，TSP，FSP，QAP において n の二次関数となるが，0-1 KP においては n となる。従って，0-1 KP では解空間全体の規模に対する局所的最適解の数が他の問題より多くなる傾向にあると考えられる。そのため，有効な多様化を実現する上で，広い空間を探索済みとして扱うには，より大きな s を設定する必要があったと考えられる。

Fig. 3 は，48 都市の巡回セールスマン問題（att48）に対して，同一の初期点から提案手法と Tabu Search を適用した時の探索過程の一例を示したものである。縦軸は，最適値に対する探索点の評価値の相対誤差であり，横軸は目的関数の呼出回数（Number of Function Calls）の累計，すなわち探索の進展を表している。Fig. 3 から，提案手法の探索過程は Tabu Search の探索過程に比べて探索点における評価値が大きく変動しており，提案手法で期待されている直径対集合の外側の引き込み領域への移動が実現できていることが分かる。

6.　おわりに

本論文では，組合せ最適化問題の解空間に，最良移動戦略の LS に基づく新たな概念“引き込み領域”を導入し，解空間を階層的に再解釈した。解空間の下位構造における移動戦略を集中化，上位構造における移動戦略を多様化と対応付け，それぞれの戦略を明確化した最適化手法を提案した。特に，多様化については，引き込み領域の概念を活用

Table 2.　Numerical Experiment Result

			Objective Function Value Case 1		Objective Function Value Case 2	
			Tabu Search	Proposed Method	Tabu Search	Proposed Method
Traveling Salesman Problem (Minimization Problem)	48 cities att48	Mean	10817.7 (1.79 %)	10674.1 * (0.43 %)	10743.2 (1.08 %)	10647 * (0.18 %)
		Best	10661 (0.31 %)	10628 * (0 %)	10638 (0.09 %)	10628 * (0 %)
		Worst	10951 (3.04 %)	10798 * (1.6 %)	11163 (5.03 %)	10726 * (0.92 %)
		SD	58.2 (0.55 %)	40.7 (0.38 %)	81.3 (0.76 %)	21.3 (0.2 %)
	107 cities pr107	Mean	45010.9 (1.6 %)	44640.3 * (0.76 %)	44794.5 (1.11 %)	44421.1 * (0.27 %)
		Best	44433 (0.29 %)	44326 * (0.05 %)	44433 (0.29 %)	44326 * (0.05 %)
		Worst	46005 (3.84 %)	45421 * (2.52 %)	45643 (3.02 %)	45158 * (1.93 %)
		SD	295.1 (0.67 %)	248.5 (0.56 %)	177 (0.4 %)	134.4 (0.3 %)
	150 cities ch150	Mean	6702 (2.67 %)	6641.9 * (1.74 %)	6665.3 (2.1 %)	6604.3 * (1.17 %)
		Best	6612 (1.29 %)	6556 * (0.43 %)	6609 (1.24 %)	6555 * (0.41 %)
		Worst	6844 * (4.84 %)	7043 (7.89 %)	6831 (4.64 %)	6807 * (4.27 %)
		SD	54.6 (0.84 %)	95.1 (1.46 %)	40.4 (0.62 %)	40.2 (0.62 %)
0-1 Knapsack Problem (Maximization Problem)	100 bit	Mean	2010.7 (2.06 %)	2023.3 * (1.45 %)	2010.7 (2.06 %)	2032.1 * (1.02 %)
		Best	2036 (0.83 %)	2046 * (0.34 %)	2036 (0.83 %)	2046 * (0.34 %)
		Worst	1982 (3.46 %)	1996 * (2.78 %)	1982 (3.46 %)	2018 * (1.7 %)
		SD	11.6 (0.56 %)	11.7 (0.57 %)	11.5 (0.56 %)	6.4 (0.31 %)
	500 bit	Mean	10139.9 (3.34 %)	10240 * (2.38 %)	10148.6 (3.25 %)	10248.8 * (2.3 %)
		Best	10208 (2.69 %)	10308 * (1.73 %)	10240 (2.38 %)	10308 * (1.73 %)
		Worst	10059 (4.11 %)	10190 * (2.86 %)	10074 (3.97 %)	10214 * (2.63 %)
		SD	31.8 (0.3 %)	25 (0.24 %)	31 (0.3 %)	23.2 (0.22 %)
	1000 bit	Mean	20251.6 (3.1 %)	20398.8 * (2.4 %)	20260.3 (3.06 %)	20408.1 * (2.35 %)
		Best	20350 (2.63 %)	20529 * (1.78 %)	20349 (2.64 %)	20529 * (1.78 %)
		Worst	20124 (3.71 %)	20301 * (2.87 %)	20108 (3.79 %)	20319 * (2.78 %)
		SD	57.5 (0.28 %)	48.9 (0.23 %)	55.8 (0.27 %)	43 (0.21 %)
Flow-shop Scheduling Problem (Minimization Problem)	20 jobs 5 machines ta001	Mean	1294.9 * (1.32 %)	1295.2 (1.35 %)	1291.5 * (1.06 %)	1292.4 (1.13 %)
		Best	1278 (0 %)	1278 (0 %)	1278 (0 %)	1278 (0 %)
		Worst	1297 (1.49 %)	1297 (1.49 %)	1297 (1.49 %)	1297 (1.49 %)
		SD	5 (0.39 %)	5.1 (0.4 %)	7 (0.54 %)	7.4 (0.58 %)
	50 jobs 5 machines ta031	Mean	2735.3 * (0.42 %)	2740.4 (0.6 %)	2730.4 (0.23 %)	2730.1 * (0.23 %)
		Best	2724 (0 %)	2724 (0 %)	2724 (0 %)	2724 (0 %)
		Worst	2752 * (1.03 %)	2791 (2.46 %)	2740 * (0.59 %)	2741 (0.62 %)
		SD	6.9 (0.25 %)	13.6 (0.5 %)	3.7 (0.14 %)	5.3 (0.19 %)
	50 jobs 10 machines ta041	Mean	3092.1 * (3.38 %)	3130.3 (4.66 %)	3059.9 * (2.3 %)	3081.8 (3.04 %)
		Best	3060 (2.31 %)	3057 * (2.21 %)	3036 * (1.5 %)	3042 (1.71 %)
		Worst	3126 * (4.51 %)	3236 (8.19 %)	3098 * (3.58 %)	3126 (4.51 %)
		SD	18.1 (0.61 %)	34.9 (1.17 %)	12.3 (0.41 %)	21.7 (0.73 %)
Quadratic Assignment Problem (Minimization Problem)	Nug30	Mean	6147.2 * (0.38 %)	6148.3 (0.4 %)	6137.6 (0.22 %)	6136.9 * (0.21 %)
		Best	6124 (0 %)	6124 (0 %)	6124 (0 %)	6124 (0 %)
		Worst	6180 * (0.91 %)	6202 (1.27 %)	6160 * (0.59 %)	6174 (0.82 %)
		SD	15.2 (0.25 %)	18.7 (0.31 %)	12.2 (0.2 %)	12.7 (0.21 %)
	Sko64	Mean	48742.6 (0.5 %)	48729.2 * (0.48 %)	48643.8 (0.3 %)	48623.8 * (0.26 %)
		Best	48524 * (0.05 %)	48536 (0.08 %)	48524 (0.05 %)	48502 * (0.01 %)
		Worst	49234 (1.52 %)	49192 * (1.43 %)	48922 (0.87 %)	48870 * (0.77 %)
		SD	139.5 (0.29 %)	134.3 (0.28 %)	97.7 (0.2 %)	102.7 (0.21 %)
	Sko81	Mean	91486.8 * (0.54 %)	91505.5 (0.56 %)	91309.7 (0.34 %)	91291.6 * (0.32 %)
		Best	91134 (0.15 %)	91094 * (0.11 %)	91062 * (0.07 %)	91092 (0.1 %)
		Worst	92190 (1.31 %)	91950 * (1.05 %)	91976 (1.07 %)	91752 * (0.83 %)
		SD	256.7 (0.28 %)	239 (0.26 %)	208 (0.23 %)	158.5 (0.17 %)

することで，特定の範囲を探索済みとして扱い，未探索範囲へ移動を方向付ける新たな移動戦略を組み込んだ。そして，ベンチマーク問題を用いた数値実験により，提案手法が，代表的な組合せ最適化手法である Tabu Search と比べ，多くの場合で優れた結果を得られることを確認した。

数値実験に基づく提案手法の課題としては，①特定の問題において，探索方針を適切なタイミングで切り替えることができなかったと推測されるため，多様化のオペレーション，すなわち未探索領域への移動と脱出判定のさらなる改良，②問題の種類，規模，近傍定義などの違いと設定パラメータや探索ダイナミクスの関係性を分析し，手法の改良に活かすこと，などが挙げられる。

その他の検討課題としては，〈3·2〉項で長期的視点に立った移動戦略として述べた “有益な解の存在が期待できる範囲への移動” に着目した手法の改良が挙げられる。POP の成立を考慮すれば，探索過程で得られた良質な解情報を基に移動を適切に方向付けられることが移動戦略として重要となる。しかし，これは同時に探索済みの範囲への回帰を意味しており，単点探索の限界を示唆している。従って，提案手法の改良の方向性としては，探索点の多点化により，情報を共有し移動戦略に相互作用を与え，探索性能を向上させることが考えられる。

文　献

（1） M. Sipser: “Introduction to the Theory of Computation”, PWS Publishing (1997)
マイケル・シプサ（著），渡辺　治，太田和夫（監訳），阿部正幸，植田広樹，田中圭介，藤岡　淳（訳）:「計算理論の基礎」，共立出版（2000）

（2） J. Hromkovic: “Algorithmics for Hard Problems”, Springer-Verlag, (2003)
J・ホロムコヴィッチ（著），和田幸一，増澤利光，元木光雄（訳）:「計算困難問題に対するアルゴリズム理論」，丸善出版（2012）

（3） 久保幹雄，J・P・ペドロソ:「メタヒューリスティクスの数理」，共立出版（2009）

（4） 相吉英太郎・安田恵一郎（編著）:「メタヒューリスティクスと応用」，オーム社（2007）

（5） D.E. Goldberg: “Genetic Algorithms in Search, Optimization, and Machine Learning”, Addison-Wesley (1989)

（6） 伊庭斉志:「遺伝的アルゴリズムの基礎—GA の謎を解く—」，オーム社（1994）

（7） E. Aarts and J. Korst: “Simulated Annealing and Boltzmann Machines”, John Wiley & Sons (1989)

（8） F. Glover and M. Laguna: “Tabu Search”, Kluwer Academic Publishers (1997)

（9） R. Battiti and G. Tecchiolli: “The Reactive Tabu Search”, ORSA Journal of Computing, 6, pp.126-140 (1994)

（10） 柳浦睦憲・茨木俊秀:「組合せ最適化—メタ戦略を中心として—」，朝倉書店（2001）

（11） E. Aarts and J.K. Lenstra (Editor): “Local Search in Combinatorial Optimization”, A Wiley-Interscience Publication (1997)

（12） 矢野公一:「距離空間と位相構造」，共立出版（1999）

（13） J.I. Marden: “Analyzing and Modeling Rank Data”, Chapman & Hall (1995)

（14） 山本芳嗣・久保幹雄:「巡回セールスマン問題への招待」，朝倉書店（1997）

（15） 久保幹雄・松井知己:「組合せ最適化 [短編集]」，朝倉書店（1999）

（16） 黒田　充・村松健児（編）:「生産スケジューリング」，朝倉書店（2002）

（17） 斎藤　毅:「集合と位相」，東京大学出版会（2009）

（18） 藤　重悟:「グラフ・ネットワーク・組合せ論」，共立出版（2002）

（19） K. Yasuda, H. Jinnai, and A. Ishigame: “Multi-Point Combinatorial Optimization Method with Distance Based Interaction”, *IEEJ Trans. EIS*, Vol.130, No1, pp.6-13 (2010) (in Japanese)
安田恵一郎・神内宏幸・石亀篤司:「距離に基づく相互作用を用いた多点探索型組合せ最適化手法」，電学論 C，Vol.130, No.1, pp.6-13 (2010)

（20） T. Kanazawa and K. Yasuda: “A Basic Study of Metaheuristics Based on Higher Level Structure in Solution Space of Combinatorial Optimization Problem”, *IEEJ Trans. EIS*, Vol.131, No.4, pp.934-935 (2011) (in Japanese)
金澤貴彦・安田恵一郎:「組合せ最適化問題の解空間における上位構造に基づくメタヒューリスティクスの基礎検討」，電学論 C，Vol.131, No.4, pp.934-935 (2011)

（21） K. Yaguchi, K. Tamura, K. Yasuda, and A. Ishigame: “Combinatorial Optimization Method Based on Quantitative Evaluation of Proximate Optimality Principle”, *IEEJ Trans. EIS*, Vol.133, No.6, pp.1-11 (2013) (in Japanese)
矢口航太・田村健一・安田恵一郎・石亀篤司:「近接最適性原理の定量的評価に基づく組合せ最適化手法」，電学論 C，Vol.133, No.6, pp.1-11 (2013)

（22） R.J.M. Vaessens, E.H.L. Aarts, and J.K. Lenstra: “A Local Search Template”, Computers and Operations Research, Vol.11, pp.969-979 (1998)

（23） E. Taillard: “Benchmarks for Basic Scheduling Problems”, ORWP89/21 (1989)

（24） P. Diaconis and R.L. Graham: “Spearman’s Footrule as a Measure of Disarray”, Journal of the Royal Statistical Society. Series B, Vol.39, No.2, pp.262-268 (1977)

付　録

1. 数値実験に用いた組合せ最適化問題

付録では，本論文の数値実験に用いた組合せ最適化問題である，0-1 ナップサック問題，巡回セールスマン問題，フローショップスケジューリング問題，二次割当問題について説明する(3)(14)～(16)(23)。

〈1·1〉 巡回セールスマン問題 (Traveling Salesman Problem: TSP)　巡回セールスマン問題とは，n 個の都市の集合 $V = \{1, \cdots, n\}$ と都市 i と都市 j の間の距離 $d(i, j)$, $i, j \in V$ が与えられたとき，すべての街をちょうど一度ずつ訪問したあと元に戻る巡回路のうち，距離が最小になるものを求める問題である。この問題の解は，$e_{(i,j)}$ を都市 i と都市 j を頂点としたときの都市 i, j 間の辺とすれば，n 個の点のハミルトングラフ $\boldsymbol{x} = (V, \{e_{(x_1,x_2)}, e_{(x_2,x_3)}, \cdots, e_{(x_{n-1},x_n)}\})$, $x_i \in V$ で与えられる。巡回セールスマン問題は次のように定式化される。

$$\min \qquad f(\boldsymbol{x}) = \sum_{k=1}^{n-1} d(x_k, x_{k+1}) + d(x_n, x_1)$$

〈1·2〉 0-1 ナップサック問題 (0-1 Knapsack Problem: 0-1 KP)　ナップサック問題とは，各荷物 i に対する，重量 a_i，価値 c_i，および重量制限 b が与えられたとき，与えられた n 個の荷物集合 $V = \{1, 2, \cdots, n\}$ からいくつかを選び，選ばれた荷物の重量の合計が重量制限を超えないという条件の下で，価値の合計を最大にする問題である。この問題の解は，x_i をナップサックに入れる荷物 $i \in V$ の個数としたときに $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$ で与えられる (x_i は非負整数)。特に，$x_i \in \{0, 1\}$, $i \in V$ の場合を 0-1 ナップサック問題と呼ぶ。0-1 ナップサック問題は次のように定式化される。

$$\max \qquad f(\boldsymbol{x}) = \sum_{i=1}^{n} c_i x_i$$

$$\text{subj. to} \quad \sum_{i=1}^{n} a_i x_i \leq b$$

〈1·3〉 フローショップスケジューリング問題 (Flow-shop Scheduling Problem: FSP) フローショップスケジューリング問題とは，与えられた n 個の製品集合 $V=\{1,\cdots,n\}$ を S 台の機械により順に加工する加工順序をなんらかの評価基準を最大化あるいは最小化するように決める問題である。n 個の製品はすべての機械で同じ順序で加工されるとし，各機械は一度に 1 つの製品しか加工できず，一度ある製品の加工にとりかかると，それが完了するまで他の製品は加工できない。

機械 i における製品 j の処理時間は $t_p(i,j)$ として与えられるものとする。この問題の解は x_k を k 番目に加工される製品としたときに，n 個の製品の加工順序 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)$ で与えられる。加工順序 $\boldsymbol{x}$ で加工した場合での各機械における各仕事の加工終了時間 $t_c(i,j)$ は次のように計算される。

$$t_c(i,x_k)=\max\{t_c(i-1,x_k),\ t_c(i,x_{k-1})\}+t_p(i,x_k)$$
$$i=1,2,\cdots,S,\ \ k=1,2,\cdots,n$$

ただし，$t_c(0,j)=t_c(i,x_0)=0$ とする。本論文では，評価基準をすべての製品の全機械における加工が完了する時間とし，その最小化問題を考える。この場合，フローショップスケジューリング問題は次のように定式化される。

$$\min \qquad f(\boldsymbol{x})=t_c(S,x_n)$$

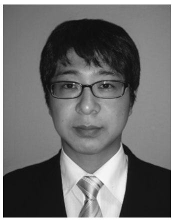
〈1·4〉 二次割当問題 (Quadratic Assignment Problem: QAP) 二次割当問題とは，配置すべき n 個の施設の集合 $V=\{1,\cdots,n\}$ と n 箇所の地点があり，施設 i から施設 j への物の移動量 $a(i,j),\ i,j\in V$，および，地点 k から地点 l までの距離 $b(k,l),\ k,l\in V$ が与えられたとき，物の総移動距離を最小とする施設の配置を求める問題である。この問題の解は，地点 x_j に施設 i を配置することを表す順列 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)$ で与えられる。二次割当問題は次のように定式化される。

$$\min \qquad f(\boldsymbol{x})=\sum_{i=1}^{n}\sum_{j=1}^{n}a(i,j)b(x_i,x_j)$$

落 合 広 樹

（正員） 2014 年 3 月首都大学東京大学院理工学研究科電気電子工学専攻博士前期課程修了，同年 4 月東日本旅客鉄道 (株) 入社，現在に至る。システム最適化，主として組合せ最適化に関する研究に従事。

金 澤 貴 彦

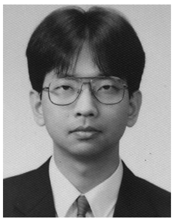


（正員） 2003 年 3 月東京都立大学大学院工学研究科電気工学専攻修士課程修了，同年 4 月九州電力株式会社入社，現在に至る。システム最適化手法，および電力システム工学に関する研究に従事。

田 村 健 一

（正員） 2008 年 9 月，慶應義塾大学大学院理工学研究科総合デザイン工学専攻後期博士課程修了。同年 10 月首都大学東京理工学研究科助教となり，現在に至る。PID 制御，適応制御，システム最適化の研究に従事。博士 (工学)。

安 田 恵一郎



（フェロー） 1989 年 3 月北海道大学大学院 工学研究科電気工学専攻博士課程修了。同年 4 月東京都立大学工学部助手，1991 年同大助教授，2006 年首都大学東京理工学研究科教授となり，現在に至る。システム最適化および電力系統工学の研究に従事。工学博士。