論　文

組合せ最適化問題に対するメタ戦略について

柳浦　睦憲[†]　　茨木　俊秀[†]

On Metaheuristic Algorithms for Combinatorial Optimization Problems

Mutsunori YAGIURA[†] and Toshihide IBARAKI[†]

あらまし　組合せ最適化問題に対する効率的な近似解法の一般的枠組みとして，近年，遺伝アルゴリズム，アニーリング法，タブー探索法やそれらの変形版など，様々なアルゴリズムが提案されてきた．これらを総称してメタ戦略あるいはメタヒューリスティクスと呼んでいる．本稿では，これらメタ戦略にあらわれる様々なアイディアを，近似解法の基本戦略である局所探索法の一般化ととらえることで，体系的にまとめる．メタ戦略の一つの魅力は，その手軽さとロバスト性にある．この観点から，次に，メタ戦略の基本的なアイディアのみで構成したシンプルなアルゴリズムを，計算実験により比較した結果を述べる．これをもとに，手軽なツールとしてのメタ戦略の設計指針を与える．そのあと，より多くの手間をかけても，さらに性能の高いアルゴリズムを構成したい場合に有効となる，やや複雑なアイディアについても簡単に紹介する．最後に，メタ戦略の理論的解析の話題にも言及する．

キーワード　組合せ最適化，近似解法，メタ戦略，局所探索法，遺伝アルゴリズム，アニーリング法，タブー探索法

1. 序　　章

最適化問題は一般的に以下のように表される:

$$\begin{array}{ll} \text{最小化} & f(x) \\ \text{制約条件} & x \in F. \end{array} \qquad (1)$$

f を目的関数，F を実行可能領域，制約条件を満たす解 $x \in F$ を実行可能解と呼ぶ．F が組合せ的な構造を持つ場合，(1) は組合せ最適化問題 (combinatorial optimization problem) と呼ばれる．

多くの組合せ最適化問題について，厳密な最適解を求めることが極めて困難であることが，計算の複雑さの理論により明らかにされてきた．NP 困難性はその代表例であり，多くの組合せ最適化問題が NP 困難であることが知られている [50]．しかし，現実には，最適性の保証はなくとも，十分精度の高い解が求まれば満足のいく場合が多い．近似解法 (approximate algorithms) や発見的手法 (heuristics) はこのような目的に用いられる．近似解法の基本戦略として，欲張り法 (greedy method)，局所探索法 (local search，以下 LS 法と記す; 山登り法 (hill climbing method)，近傍探索法 (neighborhood search) などとも呼ばれる) などがあるが，最近では，これらを組合せるなどの方法により，多少時間はかかっても，より精度の高い解を求める枠組を提供しようとする，メタ戦略 (メタヒューリスティクス，metaheuristics) の研究が盛んである．代表的なものとして，ランダム多スタート局所探索法 (random multi-start local search，以下 MLS 法と記す)[87], [98], [99], [121], [131], [132]，遺伝アルゴリズム (genetic algorithm，以下 GA 法と記す; 進化的計算 (evolutionary computation) とも呼ぶ)[33], [65], [73], [91], [111], [145]，アニーリング法 (simulated annealing，以下 SA 法と記す) [1], [3], [24], [83], [89]，タブー探索法 (tabu search，以下 TS 法と記す)[52], [55], [59], [61], [72] などがある．これらの変形として，GA 法に LS 法を組込んだ遺伝的局所探索法 (genetic local search，以下 GLS 法と記す) [17], [18], [48], [79], [92], [110], [142], [147]，MLS 法の初期解生成ルーチンとして欲張り法を組合せた GRASP 法 (greedy randomized adaptive search procedure) [42], [44] ~ [46], [94], [96], [133]，MLS 法の初期解生成に過去の探索で得られた解を利用する反復局所探索法 (iterated local search，以下 ILS 法と記す)[82], [106]，

[†] 京都大学情報学研究科数理工学専攻，京都市
Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan

ILS 法の変形である variable neighborhood search 法 [19], [20], [70], [108]，SA 法を多少単純化した方法 (threshold accepting，以下 TA 法と記す)[40], [97]，TA 法の変形 (great deluge algorithm，以下 GDA 法と記す)[39], [140]，TS 法を単純化した guided local search 法 [88], [146], [149], [150] など，様々な戦略が提案されている．

これらメタ戦略は，1) 過去の探索の履歴を利用して新たな解を生成する，2) 生成した解を評価し，その情報を解の探索にフィードバックする，という操作の反復より成る．よって，生成された解のどのような情報を探索履歴として記憶するか，探索の履歴をどのように利用して新たな解を生成するか，に対する様々なアイディアの集合がメタ戦略であるといえる．本稿では，これらメタ戦略全体を MLS 法の一般化であると捉えることにより，様々なアイディアを体系的に整理する．近年では，様々なメタ戦略のアイディアを組合せて用いることにより，より強力なアルゴリズムを構成することが一般的であるが，その際，メタ戦略を系統的に整理しておくことは重要である．また，メタ戦略のこのような捉え方が，柔軟な発想に基づく新たなメタ戦略のアイディアの誕生のきっかけになることも期待できる．

メタ戦略の一つの魅力として，その簡単さとロバスト性が挙げられる．これらは，問題の数学的構造などに対するとりわけ深い洞察がなくても，アルゴリズムを簡単に作ることができ，しかも，ある程度良い結果が期待できる．また，対象とする問題や，アルゴリズム内部のパラメータやオペレータが多少変化しても，性能が大幅に劣化しない場合が多い．

4.では，この点に重点を置き，メタ戦略を「手軽なツール」として用いることを前提に，問題固有の性質にできるだけ依存しないような一般的傾向を観察することを目的に計算実験を行った結果について報告する ([152], [153], [155] にも結果の一部を報告した)．具体的な対象としては，1 機械スケジューリング問題 (single machine scheduling problem，以下 SMP と記す) と，充足最大化問題 (maximum satisfiability problem，以下 MAXSAT と記す) の 2 つを用いた．SMP は，与えられた n 個の仕事のコスト最小の順列を求める問題，MAXSAT は，n 個の変数とその否定の論理和より成る m 個の節と各節の重みが与えられたとき，真となる節の重みの和が最大となる変数の 0-1 割当を求める問題で，いずれも NP 困難であることが知られている．これらに対する実験結果と，アルゴリズムを実際にプログラムする際の手間 (後者はある程度著者の主観的評価) をもとに，手軽なツールとしてのメタ戦略の設計指針を与える．

一方，もう一つの魅力として，メタ戦略は，非常に柔軟性に富む枠組みであるため，内部に様々な工夫を施すことが可能であり，そのような工夫により，より強力なアルゴリズムを構成できる可能性を秘めている．また，メタ戦略アルゴリズムには，通常，探索を制御する様々なパラメータが含まれており，高い性能を引き出すには，これらをうまく調整する必要があるが，これらをアルゴリズム内部で自動的に調整することで，より実用性の高いアルゴリズムを作るという試みもある．**5.**では，このようなメタ戦略の性能向上に有効と思われる様々なテクニックを紹介する．

最後に，**6.**では，メタ戦略に対する理論的解析について，SA 法などの漸近収束性に関する話題と，計算の複雑さの立場から LS 法を解析しようとする，クラス PLS の話題，および LS 法により得られる解の近似精度の話題にふれる．

4.の計算実験に用いた問題の定義を以下に与えておく (問題例の生成など，より詳くは **4. 1**と **4. 4**を参照)．1 機械スケジューリング問題 (SMP) は，代表的なスケジューリング問題の 1 つであり，これまで多くの研究がある [12], [78]．与えられた n 個の仕事 $V = \{1, \ldots, n\}$ を 1 台の機械で処理するが，機械は 1 度に 1 つの仕事しか処理できず，処理の中断は許されない，さらに，処理間の無駄時間はないと考える．従って，スケジュールは n 個の仕事の順列 σ を与えることで決定される．$\sigma(k) = i$ は，k 番目に処理される仕事が i であることを意味する．各仕事 i は処理時間 p_i を要し，完了時刻が c_i であるときに $g_i(c_i)$ のコストがかかる．c_i は処理順序によって $c_i = \sum_{k=1}^{\sigma^{-1}(i)} p_{\sigma(k)}$ と一意に定まる．このとき，問題は以下のように定式化される．

1 機械スケジューリング問題 (single machine scheduling problem，SMP)

入力: n 個の仕事 $V = \{1, \ldots, n\}$ と，各仕事 i に対する処理時間 p_i，および完了時刻に対するコスト関数 g_i．

出力: $cost(\sigma) = \sum_{i \in V} g_i(c_i)$ を最小化する仕事の順序 σ．

充足最大化問題 (MAXSAT) は，代表的な組合せ最適化問題の 1 つであり，これまで多くの研究がある [69], [138], [139] . n 個の変数 $y_1, \ldots, y_n$ とその否定 $\bar{y}_1, \ldots, \bar{y}_n$ の論理和よりなる m 個の節 $C_1, \ldots, C_m$ (たとえば $C_i = y_1 \vee \bar{y}_3 \vee y_4$) と，各節に対する重み $w_1, \ldots, w_m \in Z_+$ が与えられる . 変数 $y_1, \ldots, y_n$ とその否定 $\bar{y}_1, \ldots, \bar{y}_n$ を総称してリテラルと呼ぶ . $v = \langle v_1, \ldots, v_n \rangle \in \{0,1\}^n$ を変数 $y = \langle y_1, \ldots, y_n \rangle$ への 0-1 割当とする . すなわち，$v_j = 1$ ならば $y_j = 1$ かつ $\bar{y}_j = 0$, $v_j = 0$ ならば $y_j = 0$ かつ $\bar{y}_j = 1$ とする . また , 0-1 割当 v に対応する節 C_i の値を $C_i(v)$ と表す . このとき，問題は以下のように定式化される .

充足最大化問題 (maximum satisfiability problem , MAXSAT)
入力: n 変数より成る節 $C_1, \ldots, C_m$ と各節の重み $w_1, \ldots, w_m$.
出力: $gain(v) = \sum_{i=1}^{m} w_i C_i(v)$ を最大にする 0-1 割当 $v \in \{0,1\}^n$.

なお , 充足可能性問題 (satisfiability problem , SAT) は $\sum_{i=1}^{m} C_i(v) = m$ となる v の存在を問う決定問題である . SAT は最初の NP 完全問題としてよく知られている [50] .

なお，説明の都合上，上述の 2 つの問題の他に，以下の 2 つをしばしば例として用いるので，ここでまとめて定義しておく .

グラフ彩色問題 (graph coloring problem , GCP)
入力: n 節点より成る無向グラフ .
出力: 枝の両端点には同じ色を塗らないという条件下で全ての節点を彩色するとき，色数最小の彩色 .

行商人問題 (traveling salesman problem , TSP)
入力: n 個の街の集合と街 i と j の間の距離 d_{ij} ($i, j = 1, \ldots, n$) .
出力: 与えられた街の全てをちょうど 1 度ずつ訪れて元に戻る距離最小の巡回路 (tour) .

これらはいずれも代表的な組合せ最適化問題で，NP 困難である [50] . 本稿では，TSP の距離行列 (d_{ij}) は対称，すなわち全ての i と j に対して $d_{ij} = d_{ji}$ であると仮定する . なお，TSP の解は街の訪問順序，すなわち順列で与えられるが，説明の都合上，巡回路において隣り合う街の対を枝と呼び，解が枝の集合として与えられると解釈する場合もある .

2. 代表的なメタ戦略

本節では , まず , メタ戦略を設計する際に重要となる探索空間について説明する . 次に , 近似解法の基本戦略である欲張り法と LS 法 , および代表的なメタ戦略として MLS 法 , GA 法 , SA 法 , および TS 法を簡単に説明する . これらの変形など，より詳しくは次節で述べる . なお，メタ戦略については [2], [75], [76], [93], [116] ~ [118], [123] ~ [125], [127], [128], [148] などに解説がある .

2.1 探索空間

探索の対象となる解集合を探索空間と呼ぶ . 実行可能解を生成することが簡単である場合には，実行可能領域をそのまま探索空間としてもよいが，そうでない場合には，問題に応じた工夫が必要となる . また，もとの実行可能領域と異なる探索空間を採用した場合には，実行可能性を考慮しつつ探索解を評価する基準を新たに定義する必要がある . 新たな基準は，その基準の下で最適な探索解を求めることで，元の問題の実行可能な最適解が求まるように定義することが望ましい . 例えば , TSP や SMP では，実行可能領域は任意の順列，MAXSAT では任意の 0-1 ベクトルとなるので，これらをそのまま探索領域とすればよい . 一方 , GCP では，定めた色数で実行可能な彩色を求めることは困難である (そのような彩色の存在の判定は NP 完全) . このような問題に対しては，例えば，以下のような方法が考えられる . なお，便宜上，色は正の整数で表されるものとする .

(i) 利用する色数を固定した上で，任意の彩色を探索解とし，両端上に同じ色が塗られている枝の本数を最小化する . この場合，固定する色の数はいろいろ試す .

(ii) 色数を特に定めず，任意の彩色を探索解とする . 両端が同色で塗られている枝数の合計，および用いた色数の二つの重みつき和を評価基準とする .

(iii) 節点の順列を探索解とし，順列の前から順に，利用可能な色のうち番号最小のものを割当てていくことで実行可能な彩色を求め，用いられた色の数をその順列の評価値とする .

戦略 (ii) において，両端が同色で塗られた枝 (以下同色枝と記す) の数にかかる重みを十分大きく設定すれば，探索空間における評価値最良の解が，元の問題の実行可能な最適解となる . この性質は，同色枝数

と色数をこの順に並べたベクトルに対する辞書式順序（小さいほどよい）を評価基準とした場合でも成り立つが，これは，同色枝数の重みを無限大にした場合の重みつき和と等価である．なお，同色枝数と色数をこれとは逆の順に並べた辞書式順序は，不適切な評価基準となってしまうので注意を要する．また，実行不可能解を含む探索空間を採用した場合には，探索をスムーズに行うために，通常，評価関数や探索のルールにさらに工夫が加えられる．この点については，**3.2** と **3.3** で触れる．

このように，問題の定式化が定まっても，探索空間と解の評価法には様々な工夫が可能であり，問題の性質に応じて，これらをうまく定めるか否かで，アルゴリズムの性能は大きく異なる．この点は極めて重要であるが，問題の構造に大きく依存する．よって，以下では探索空間と解の評価（小さいほどよいとする）は，とくに断らない限りあらかじめ与えられるものとし，探索空間を $\tilde{F}$，探索解の評価基準を $\tilde{f}$ と記す．なお，$\tilde{f}$ がベクトルである場合には，$\tilde{f}$ の大小関係は辞書式順序で定めるものとする．また，解の評価 $\tilde{f}$ に f 以外の関数を利用した場合，探索の途中で訪れた実行可能解の中で f を最小にする暫定解を，探索解とは別に保持しておく必要がある．

なお，GCP では，戦略（i）や（ii）を用いた場合，1 つの節点の色を変更して得られる解を評価する操作が基本操作としてよく用いられるが，このような解の小さな変形に対する評価値は，色の変化した節点に接続する枝のみを走査すれば計算できるので，比較的高速に出来る．一方，戦略（iii）では，順列に対する変形が小さい場合でも，得られる彩色は大きく異なる可能性があるため，1 回の評価に時間がかかる．よって，GCP に対しては，戦略（iii）は効率の観点から他の 2 つの戦略にやや劣ると思われるが，制約条件が複雑なスケジューリング問題などに対しては，このような考え方も有効である（例えば [113]）．

2.2 基本戦略

組合せ最適化問題に対する近似解法の基本戦略には，

- 欲張り法，
- 局所探索法（LS 法），

などがある．欲張り法は，目的関数への貢献度を示す局所的な評価値に基づいて，実行可能解を直接構成する方法である．LS 法は，適当な探索解 $x \in \tilde{F}$ に対し，x に少しの変形を加えることによって得られる解集合 $N(x) \subset \tilde{F}$（近傍と呼ばれる）内に，x よりも良い解 x'（すなわち $\tilde{f}(x') < \tilde{f}(x)$）があれば，$x := x'$ と移動する操作を，近傍内に改善解が存在しなくなるまで反復する方法である．$N(x)$ 内に改善解が存在しない x を局所最適解（locally optimal solution）と呼ぶ．欲張り法，LS 法ともに古くから用いられてきた手法であるが，早期の研究としては，たとえば [31] がある．この中では，TSP に対し，欲張り法で生成した解を LS 法で改良し，それをさらに複雑なアルゴリズムで修正するという手法を提案し，街数 42 の問題例に対する解を手計算により約 70 時間かけて求めたことが報告されている．

2.3 多スタート局所探索法

MLS 法は，複数個の初期解に対して LS 法を適用し，得られた最良の解を出力する方法で，メタ戦略の中では最も古くから広く用いられてきた手法の一つである [87], [98], [99], [131], [132]．初期解は，ランダムに生成する，欲張り法を利用する，などの方法が考えられる．また，過去の探索で得られた局所最適解の情報を，その後の探索の，(1) 初期解生成に利用する [131]，(2) 計算の効率化に利用する [98], [99]，などの重要なアイディアが，早期の研究ですでに提案されている．

2.4 遺伝アルゴリズム

GA 法は，生物の進化にアイディアを得た方法で，複数の解を同時に保持し，それらを集団として改善していくところに特徴を持つ．現在保持している解集合 P に対し，交叉（crossover）および突然変異（mutation）の操作を加え，新しい解集合 $Q \subset N(P)$（$N(P)$ は交叉と突然変異によって生成され得る解全体の集合とする）を生成したのち，P と Q から淘汰（selection）の規則に従って解集合 $P' \subset P \cup Q$ を選択し，$P := P'$ とする操作を反復するものである．交叉は，2 つまたはそれ以上の解を組合せることにより新たな解を生成する操作，突然変異は，1 つの解に少しの変形を加えることで新たな解を生成する操作である．GA 法は，Holland [73] により提案された．解説としては，[11], [33], [65], [91], [111], [129], [145] などがある．

2.5 アニーリング法

SA 法は，LS 法の移動ルールに工夫を加えた方法の一つで，$N(x)$ 内の各解に，解の良さに応じた遷移確率（良い解ほど移行し易い）を設定し，それに従って次の解を選ぶ．具体的には，

1. $N(x)$ 内の解をランダムに 1 つ選び x' とする．
2. x と x' の評価値の差を計算し，Δ とする（x' が改悪解ならば $\Delta > 0$）．

3. $\Delta \leqq 0$ ならば 1，そうでなければ $e^{-\Delta/t}$ の確率で $x := x'$ とする．

このように，改悪解であっても遷移する確率を与えることにより，局所最適解からの脱出を図るものである．遷移確率は，物理現象の焼きなまし (annealing) にアイディアを借りて，温度と呼ばれるパラメータ t により調整される．温度 t は，通常，探索の初期の段階では高めに設定しておき (すなわち，ランダムな移動が生じやすい)，探索が進むにつれて徐々に小さくしていく．$t \to 0$ では，SA 法の動作は通常の LS 法と同様になる．SA 法は Kirkpatrick ら [89]，および Černý [24] により提案された．精密な計算実験の報告が [83] に，また，解説としては [1], [3] などがある．

2.6 タブー探索法

TS 法は，SA 法と同様，LS 法の移動ルールに工夫を加えたものであり，$N(x) \setminus (\{x\} \cup T)$ 中の最良の解を次の解として選ぶ．T は，解の循環を避けるために設けられた，タブーリスト (tabu list，または短期メモリ) と呼ばれる解集合であり，通常は最近探索した解が含まれるように動的に制御される．このように，TS 法では，常に T 以外の解に移行するため，新しい解への移動が強制され，短い周期のサイクリングを防ぐことが出来る．TS 法では，さらに，特定の変数を変更した頻度や，特定の変数がある値をとり続けた期間の長さなど，探索してきた解の特徴を長期間に渡り記憶しておくこと (長期メモリと呼ばれる) により，未探索の領域へ探索を方向づけようとする手法が組合せて用いられることが多い．TS 法は Glover [52] により提案された．解説としては [55], [59] ~ [61], [72] などがある．

3. メタ戦略の一般的枠組

本節では，メタ戦略を LS 法の一般化と捉え，様々なアイディアを体系的にまとめる．以下では，メタ戦略を

I. 初期解 x を生成する，

II. x を (一般化された) LS 法により改善する，

という 2 つの操作を反復する方法とみなす．II の LS 法の基本的な枠組は **2.2**ですでに述べたが，ここではこれをもう少し一般化し，以下のように定義する．

LS 法: 与えられた初期解 x から始め，近傍 $N(x)$ 内の解に一定のルールで移動する操作を，終了条件が満たされるまで反復する．

なお，**2.2**で述べた基本的な LS 法をこのような一般的な枠組ととくに区別する場合には，単純 LS 法と呼ぶ．LS 法の動作を定めるには，

A. 近傍 $N(x)$ の定義，

B. 解の評価法，

C. 移動ルール，

D. 終了の基準，

を決める必要がある．A は，現在の解から新たな解をどのように生成するかを定める．B は，生成した解 x' の良さを判定する基準を定める．C は，近傍 $N(x)$ 内の解をどのような順序で調べ，どの解に移動するかを定める．最後に，D は，LS 法の探索をいつ終了するかを定める．たとえば，単純 LS 法は，B の解の評価に評価関数 $\tilde{f}$ を，C の移動ルールとして $\tilde{f}(x') < \tilde{f}(x)$ ならば $x := x'$ とする方法を，D の終了基準として x が局所最適解 (すなわち $\tilde{f}(x') \geq \tilde{f}(x),\ \forall x' \in N(x)$) ならば終了するというルールを採用したものである．

このように考えると，メタ戦略の主要なアイディアは，上述の I，II-A，II-B，II-C，II-D のいずれかに対する工夫であると捉えることが可能である．以下では，これらのそれぞれについて説明する．なお，このうち，II-A (近傍の定義) については，やや複雑なアイディアが主となるので，**5.**で説明を行う．

3.1 I: 初期解の生成

初期解の生成法として，最も簡単な方法はランダムな解を利用することである．これが **2.3**ですでに述べた MLS 法である．しかし，ランダムな解よりももう少し良い解から始めれば，性能の向上が期待できる．以下の様々なアイディアは，全てこの考え方に基づいている．

GRASP 法は，ランダム性を加えた欲張り法により生成された解を初期解とする方法で，Feo らのグループ (例えば [46]) により提案された．欲張り法の各ステップでは，通常，目的関数への貢献度を示す局所的な評価値の最も高い要素を選択しつつ解を構成していくが，GRASP 法の初期解生成では，評価値の高いものから順にいくつかを候補として持ち，それらの中からランダムに 1 つ選択を行うことにより，解の生成にランダム性を加える．いくつかの組合せ最適化問題に対する適用結果 [42], [45], [94], [96], [133] にあるように，一定の成果を収めている．解説としては，[44] などがある．なお，LS 法との組合せは行われていないが，GRASP 法の初期解生成のメカニズムのアイディアは，[43], [71] などですでに紹介されている．

ILS 法では，過去の探索で得られた良い解にランダムな変形を加えたものを初期解とする．ランダムな変形には，通常，近傍操作を用いるが，その際，LS 法に用いる近傍とはタイプの異なる近傍を利用する場合もある．ILS 法は，Johnson [82] により提案された．初期解生成に利用する解として，[82] の方法では常にこれまでの最良解を用いるが，この選択に，SA 法に類似のアイディアを導入した方法もある．ILS 法のこのような変形は chained local optimization 法，large-step Markov chain 法などと呼ばれる [101], [105], [106] （歴史的にはこちらが ILS 法提案のきっかけとなった [84]）．解のランダムな変形に用いる近傍を，初めは小さく設定しておき，初期解の生成に用いる解と LS 法により新たに得られた解を比べて改善が見られない場合には徐々に大きくしていき，改善解が得られた場合は初めの近傍に戻すという方法もある（LS 法の近傍は常に同じであることに注意）．このような方法は variable neighborhood search 法と呼ばれる [19], [20], [70], [108]．

GLS 法は，GA 法の内部に LS 法を組合せ，交叉や突然変異によって生成された解を LS 法の初期解とする方法である．ILS 法では，保持している解が 1 つであるのに対し，複数個の解を保持しておき，それらを自由に組合せることにより初期解を生成するところに GLS 法の特徴がある（なお，GA 法という言葉は GLS 法を含めた意味で用いられることもあるので，LS 法を組合せない場合をとくに区別するときには，単純 GA 法（simple GA）と呼ぶ．) GLS 法の基本的なアイディアは [18] ですでに提案されているが，比較的初期の研究としては [33], [65], [79], [80], [107], [109], [110], [142], [147] などがある．なお，genetic local search という用語を用いたのは，[147] が最初と思われる．また，GA 法よりもさらに自由な発想で複数個の解から新たな解を生成するメカニズムが scatter search 法として研究されている [53], [54]．

GLS 法では，これまでに得られた良い解のいくつかを保持しておき，それらを変形することによって次の初期解を生成するが，保持しておく情報として，解の部分的構造や統計量を用いる方法もある [17], [143]．類似の方法に，ant system 法（ant colony system ともいう）がある．これは，蟻の行列の巣から餌までのルートがフェロモンの情報によって定まるメカニズムにアイディアを得た方法である．欲張り法によって解を生成し，良い解に含まれている解の構成要素（たとえば TSP の枝など）に得点（フェロモンと呼ぶ）を加える．そして，欲張り法の局所利得を，得点の大きい構成要素がより選ばれやすくなるように変更しつつ欲張り法を反復するのである．この方法は [29], [35], [37] で提案された．初期の ant system 法には LS 法は組合されていないが，それでは通常十分な性能が得られないため，最近では，LS 法を組合せることで，一定の成果を納めている [36], [49], [102], [141]．

3. 2 II-B: 解の評価

解の評価法で最も基本的なものは，目的関数 f をそのまま $\tilde{f}$ として用いる方法である．探索空間に実行不可能解が含まれる場合には，この方法は使えず，実行不可能解にはペナルティを与えるなどの工夫が必要となる．また，$\tilde{f}$ に変形を加える方法としては，noising 法，guided local search 法，TS 法の長期メモリなどがある．

noising 法は，$\tilde{f}$ にランダムな変形を加えた評価関数を用いた LS 法を反復する方法である．探索の初期には加えるランダム性を大きく設定し，反復とともに徐々に小さくしていき，最後は元の $\tilde{f}$ を評価とする LS 法となる．また，各反復の初期解には前回の局所最適解を利用する．noising 法は [26] で提案された．この変形として，$\tilde{f}$ を用いた LS 法と，$\tilde{f}$ にランダム性を加えた評価値を用いた LS 法を交互に反復する，perturbation 法という方法もある [28]．また，ランダム性を含まない変形を用いる以下のようなアイディアもあり，探索空間平滑化法（search space smoothing method）と呼ばれる [66], [67]．パラメータ α（$\geqq 1$）を用意し，関数 $\tilde{f}^{(\alpha)}$ を，$\tilde{f}^{(1)} = \tilde{f}$ と $\tilde{f}^{(\infty)}(x) = \tilde{f}^{(\infty)}(x')$ $(\forall x, \forall x', x \neq x' \in \tilde{F})$ を満たすように定義する．たとえば，TSP では，街 i と j の間の距離 d_{ij}（ここでは $0 \leqq d_{ij} \leqq 1$ を仮定する）に対して $\bar{d} = \sum_{i \neq j} d_{ij}/(n(n-1))$ とし，

$$d_{ij}^{(\alpha)} = \begin{cases} \bar{d} + |d_{ij} - \bar{d}|^{\alpha}, & d_{ij} \geq \bar{d} \\ \bar{d} - |d_{ij} - \bar{d}|^{\alpha}, & d_{ij} < \bar{d} \end{cases}$$

と定義する．そして $\tilde{f}^{(\alpha)}$ を巡回路に含まれる枝 $\{i, j\}$ の $d_{ij}^{(\alpha)}$ の総和と定義するなどである [67]．探索の初めには α を十分大きく設定して LS 法を行い，LS 法が局所最適解で止まるたびに α を徐々に小さくしていくことで，探索の初めにコストに大きな影響を与える解の構造を把握し，そのあと徐々に細かい構造を修正していくというアイディアである．

guided local search 法は，まず，前回の探索の局所

最適解 x の構成要素 (e.g., TSP の枝など) の中でコストの大きいものにペナルティを導入する．次の探索では，x を初期解とし，前回の評価関数にこのようなペナルティを付加した評価値を用いて LS 法を行う．このとき，十分なペナルティを与えることで，新しい評価の下では x が局所最適解でなくなり，x からの LS 法の探索が続行するので，x の近傍解の中で，元の評価関数 $\tilde{f}$ の値に大きなコストを与えていた要素を含まないような解を探索することができる．この方法は [149] で提案された .[88], [146], [150] などにいくつかの適用事例がある．guided local search 法よりもやや単純だが，同様のアイディアが充足可能性問題 (SAT) に対する weighting 法という名前で提案されている [138]．

同様のアイディアは，TS 法の長期メモリとしても利用されている．長期メモリには，この他に，特定の変数の値を変更した回数など，探索の履歴の様々な情報を利用するアイディアが提案されている [60]．これまで頻繁に値の変更が行われている変数の値を再び変えようとする移動には大きなペナルティを与え，長い周期での探索のサイクリングを防ぐ効果をねらうなどである．なお，長期メモリは，このようにサイクリングを防ぐ目的だけでなく，未探索の領域に探索を誘導する目的にも利用される．また，これまでに得られた良い解の情報を記憶しておき，それらの近くを調べるなど，**3. 1**で述べたようなアイディアも含む．このように，長期メモリは，**5. 1**に述べる探索の集中化と多様化を実現するための広い概念である．

探索空間に元の問題の実行不可能解が含まれる場合には，**2. 1**で述べたように，評価関数 $\tilde{f}$ は，通常，制約違反に対するペナルティと元の目的関数 f の 2 要素から成るベクトル，または，これら 2 つの重みつき和，というような形で与えられる．このような場合には，制約違反のペナルティの重みを軽くした重みつき和を用いて，実行不可能領域も頻繁に訪れるように探索を制御することにより，性能が向上する場合がある．また，制約が複数存在する場合は，制約ごとに異なるペナルティ重みを与えることもしばしば有効である．このような方法は，ペナルティ関数法と呼ばれ，組合せ最適化に限らず様々な最適化問題に対して古くから利用されてきた手法である．適用例として [51], [83], [157], [159], [160] を挙げておく．なお，ペナルティ重みは，軽すぎると実行可能解が得られなくなる可能性があるが，これを避けるため，長期にわたり満たされていない制約の重みを徐々に大きくするなど，適応的に調整する方法もある (例えば [51], [157])．

元の目的関数 f と全く異なる評価関数を $\tilde{f}$ に用いることも可能である．一例として，グラフ彩色問題 (GCP) に対して Johnson ら ([83] の Part II) が用いた評価関数を紹介する．ある彩色 x において，利用されている色数を k とし，各 i ($i = 1, \ldots, k$) について，色 i で塗られている節点の集合を V_i，両端の節点が V_i に含まれる枝集合を E_i と記す．このとき，

$$\tilde{f}(x) = -\sum_{i=1}^{k} |V_i|^2 + \sum_{i=1}^{k} 2|V_i| \cdot |E_i| \qquad (2)$$

とする方法である．**2. 1**の戦略 (ii) のように，評価法に色数を用いた場合は，色数が変化する場合にしか目的関数への寄与を評価できない．一方 (2) では，第 1 項が，ある節点の色を $|V_i|$ のより大きな色で塗り直す操作に高い評価を与えるので，全体の色数が変化しない場合でも評価が可能となる上，結果的に色数が最小化されるような評価基準となっている．(2) の第 2 項は，制約を満たさない枝，すなわち同色枝に対するペナルティである．色数を定めず，実行不可能解も含めた任意の彩色を探索空間とし，さらに，1 つの節点の色を変えることにより得られる解集合を近傍とした場合，評価 (2) の下での任意の局所最適解は，実行可能な彩色になることが以下のように示せる．現在の彩色が実行不可能であるとし，同色枝が E_i に含まれているとする．その枝の一方の端点の色を $k+1$ (すなわち新しい色) に変更すると (2) の第 2 項は少なくとも $2|V_i|$ 減るが，第 1 項の増加は高々

$$|V_i|^2 - ((|V_i| - 1)^2 + 1^2) = 2|V_i| - 2$$

である．よって，実行不可能な彩色に対しては，新たな色を導入して同色枝を減らすことにより (2) の評価値は減少する．しかし，一方で，この性質は，色数を増やすことで同色枝を減らす解が常に改善解となってしまうことを示唆しており，色数を増やさずに (2) の値を減らせる場合には，そのような解を優先するように移動ルールを考慮する必要がある．

3. 3 II-C: 移動ルール

移動ルールの最も基本的なものは，常に改善解 (解の良さの判定には，**3. 2**で述べたような様々な関数が利用され得ることに注意) に移動するというものである．この場合，代表的なものとして，近傍 $N(x)$ 内をランダムな順序で調べて最初に見つかった改善解に移動する，即時移動戦略 (first admissible move strategy)

と，$N(x)$ 内の解を全て調べて最良解に移動する，最良移動戦略 (best admissible move strategy) の 2 つがある．

改善解以外への移動も許す方法としては，**2. 5**と **2. 6**ですでに述べた SA 法と TS 法が代表的である．また，SA 法の変形として，TA 法や GDA 法もある．

TA 法では，SA 法の反復のステップ 3 を，

3′. $\Delta < \tau$ ならば $x := x'$ とする，

で置き換える．τ は探索を制御するパラメータで，閾値 (threshold) と呼ばれる．閾値 τ は，SA 法における温度 t と同様に，探索の初期の段階では大きめに設定しておき，探索が進むにつれて徐々に小さくしていく．SA 法では，ステップ 3 において，乱数を 1 回生成し，確率を表す指数関数を 1 回計算する必要があるが，TA 法ではそれらが不要となる．それゆえ，各反復での計算が多少効率化される．TA 法は，Dueck ら [40] により提案された．[4], [97], [147] などに，他のメタ戦略との比較結果がある．

GDA 法も SA 法の変形の 1 つである．説明の都合上，本節の GDA 法の説明に限って最大化問題を仮定する．具体的には，SA 法の反復のステップ 2 と 3 を，

2′. x' の評価値が ω よりも大きければ，$x := x'$ とする，

で置き換える．ω は探索を制御するパラメータで，水位 (water level) と呼ばれる．ω は，通常，探索の始めには十分小さい値にしておき，反復毎に，新たに採用された x の目的関数値と現在の ω の値の間の適当な値に更新することによって，徐々に大きくしていく．大洪水 (great deluge) で水位が増してくる時，出来るだけ高い山に逃れたいが，水位より下には移動出来ない人間の行動を想定してアルゴリズムが開発されたため，この名前がつけられている．GDA 法は [39] で提案された．他のメタ戦略との比較実験が [140] に報告されている．

2. 6の TS 法の説明では，$N(x) \setminus (\{x\} \cup T)$ (T はタブーリスト) 内の最良解に常に移動するものとしたが，$N(x) \setminus (\{x\} \cup T)$ 内に改善解が存在する場合は即時移動戦略を用いるとか，$N(x) \setminus (\{x\} \cup T)$ 内の解の中で，見込みのありそうなもののみに候補を絞り，その中の最良解に移動する，候補リスト戦略 (candidate list strategy) などの変形も可能である．また，$N(x)$ 内の解 x' が $x' \in T$ であっても，たとえば $\tilde{f}(x')$ がこれまで探索したどの解よりも小さいなど，その解を採用することが十分意味があると判断される場合は，T に含まれる解への移動を許す場合もある．このようなルールを特別選択基準 (aspiration criteria) と呼ぶ．

TS 法で利用されるアイディアに戦略的振動 (strategic oscillation) と呼ばれる方法がある．これは，探索空間に実行不可能解が含まれるときに主に利用されるが，実行可能領域と実行不可能領域を周期的に交互に訪れるように，タイプの異なる近傍操作を探索の状況に応じて使い分けて制御するという方法である．グラフ彩色問題やナップサック問題など，単純 LS 法があまり効果的でないと思われる問題などに有効である [57], [58], [68]．このように，TS 法には様々なアイディアが含まれている．詳細は [60] などを参照のこと．

ランダムな移動ルールに候補リスト戦略に類似のアイディアを組合せた方法として，SAT に対する WALKSAT 法がある [139]．これは SAT 専用のアルゴリズムとして開発されたが，アイディアは汎用性がある．WALKSAT 法の探索空間は変数への任意の 0-1 割当，近傍は 1 つの変数の 0-1 割当の反転により得られる解集合 (**4. 5**に後述の N_1)，解の評価値は満たされていない節の数である．移動ルールを以下に示す．まず，満たされていない節の中から 1 つをランダムに選ぶ．そして，選んだ節に含まれるどれかの変数の 0-1 割当を反転することにより得られる解の中から次の解を選ぶが，変数の選択ルールには以下の 2 つがあり，どちらを用いるかを毎回ランダムに定める．

ルール 1: 0-1 割当の変更により，満たされる節の数が最も大きくなる変数を選ぶ．

ルール 2: ランダムに 1 つ選ぶ．

近傍内に改善解が存在すれば，そのような解では現在満たされていない節のどれかが満たされるようになるので，このルールによる改善解への移動確率は正となる．また，1 つの節に含まれる変数の数は通常少ないので，1 回の移動にかかる時間が非常に小さくなり，結果的に多くの解を探索できるようになる．ルールは単純だが，このように問題の性質がうまく取り入れられており，[139] には良好な計算結果も報告されている．なお，各節に含まれるリテラル数が 2 以下の場合には，上記のルール 2 のみを用いて $O(n^2)$ (n は変数の数) 回の移動を行うと，全ての節を満たす 0-1 割当が存在すれば，そのうちの 1 つが求まる確率をいくらでも 1 に近づけることができることが示されており [119]，それがこの方法の起源になっている．

3.4 II-D: 終了基準

1 回の LS 法の終了基準は，II-C（移動ルール）に応じて定める必要がある．II-C において，常に改善解へ移動するルールを用いた場合は，近傍内に改善解がなくなった時点で探索を終了するというのが基本的なルールである．一方，SA 法や TS 法のように，探索をいくらでも長く続けられるようなルールの場合は，(1) あらかじめ定められた移動回数や計算時間内で終了する，(2) あらかじめ定められた移動回数の間に $\tilde{f}$ の最良値が改善されなければ終了する，というようなルールが用いられる．また，たとえば，SA 法の場合は，温度 t を初めは大きく設定しておき，探索の進行とともに徐々に小さくしていくので，t が十分小さくなった時点で終了するというルールも考えられる．このように，探索を制御する特定のパラメータに基づいて終了条件を設定することも可能である．

4. 手軽なツールとしてのメタ戦略

本節では，メタ戦略を「手軽なツール」として用いることを前提に，基本的アイディアのみから成る代表的なメタ戦略を SMP と MAXSAT の 2 つの問題に対する計算実験を通して比較する．

計算実験は，SMP については Sun SPARC station IPX 上で，MAXSAT については Sun Ultra 2 Model 2300 上で行い，言語は C 言語を用いた．解の精度は，SMP では全実験中に求まった最良解からの誤差，MAXSAT では最適値からの誤差で計る．また，アルゴリズムの効率は，計算時間ではなく，コストの評価回数（以下，サンプル数と記す）を基準とする．これは，計算時間が，コンピューターやプログラム技術など，アルゴリズムの本質とは関係のない要因に左右されやすいことを考慮したためである．

4.1 1 機械スケジューリング問題

各仕事 i の完了時刻に対するコスト関数 $g_i(c_i)$ の一例として，以下の実験では，

$$g_i(c_i) = h_i \max\{d_i - c_i, 0\} + w_i \max\{c_i - d_i, 0\}$$

を用いた．ここに $d_i \in Z_+$（正整数の集合）は仕事 i の納期，$h_i, w_i \in Z_+$ はそれぞれ仕事 i の進み，遅れに対するペナルティの重みである．この定義に限らず，多くの $g_i(c_i)$ について SMP の NP 困難性が知られている．問題例は，$n = 100$ に対して 10 問を [78] に従ってランダムに生成した．

4.2 SMP に対するメタ戦略の概要

SMP に対し，MLS 法，GRASP 法，ILS 法，GA 法，GLS 法，SA 法，TA 法，GDA 法，および TS 法を構成した．これらの概要を以下に述べる．これらの詳細と，各アルゴリズムのより詳しい振舞いは [155] を参照のこと．

MLS 法では，I の初期解はランダムに生成し，II の解の改善には単純 LS 法という，最も単純なルールを用いる．これ以外のメタ戦略でも，とくに説明を行わない部分は，この単純ルールを採用するものとする．なお，II-A の近傍には $N_{ins}(\sigma) = \{\sigma$ の 1 つの仕事を他の位置に挿入することにより得られる解 $\}$（挿入近傍と呼ぶ）と，$N_{swap}(\sigma) = \{\sigma$ の 2 つの仕事の位置を交換することにより得られる解 $\}$（交換近傍と呼ぶ）の 2 通りを調べ，II-C の移動ルールは，即時移動戦略と最良移動戦略の 2 通りを調べた．この内，移動ルールについては，即時移動戦略を用いる方が最良移動戦略を用いた場合に比べてより良い解が速く求まることが観測された．なお，単純 LS 法により得られる局所最適解の平均精度には両者の間であまり差がないことも観測されている [152]．最良移動戦略では，1 つの局所最適解を求めるために要するサンプル数が即時移動戦略に比べて非常に大きい．よって，サンプル数を一定にして比較した場合には，即時移動戦略による方が反復回数が多くなり，その結果より良い性能が得られたと考えられる．以上より，MLS 法以外の枠組においても同様の傾向となることが予想されるため，他のアルゴリズムでは移動ルールを即時移動戦略に限定した．

GRASP 法では，初期解生成に用いる欲張り法に様々な工夫が可能である．詳細は略すが，ここでは [152] で用いた 12 通りの欲張り法を調べた．

ILS 法では，(1) 初期解生成に利用する解の選択方法，(2) 初期解生成の際に加えるランダムな変形に利用する操作（以下では突然変異と記す）について，様々な工夫が可能である．(1) については，通常，過去の探索における最良解（σ_{best} と記す）と，前回の反復における局所最適解（σ_{last} と記す）のいずれかを利用する．ここでは，[106] の流儀に従い，以下の方法をとる．

(1) $\Delta := cost(\sigma_{last}) - cost(\sigma_{best})$ とし，$\Delta \leqq 0$ ならば確率 1 で，$\Delta > 0$ ならば確率 $e^{-\Delta/t}$ で σ_{last} を選択する．σ_{last} が選ばれなかったときには σ_{best} を選択する．ただし，t は固定されたパラメータとし，SA 法のような制御は行わない．

(2) 突然変異は，解 σ に対し，近傍 $N(\sigma)$ 中から解

σ' をランダムに 1 つ選び，$\sigma := \sigma'$ とする．近傍 N には，N_{ins} と N_{swap} の 2 通りを考え，それぞれ INS，SWAP と記す．

GA 法と GLS 法では，交叉，突然変異，および淘汰の規則に様々なものが考えられる．ここでは，交叉には order crossover [32], [110]，partially mapped crossover [64]，cycle crossover [115] などを調べた．これら交叉法の比較については [152] を参照のこと．また，突然変異には，ILS 法と同様に INS と SWAP の 2 通りを調べた．淘汰には以下を用いた．1 世代に生成される子集合 Q のサイズ $|Q|$ を 1 とする．2 つの親 σ_A と σ_B をランダムに選び，そのいずれかに突然変異を加え，その後 σ_A と σ_B を交叉させることにより，子 σ_C を生成する．そして，$P \cup Q$ 内で最大のコストを持つ σ_{worst} と $P := P \cup \{\sigma_C\} \setminus \{\sigma_{worst}\}$ のように入れ換える．なお，解集合 P には重複解を許さないものとし，子を生成した時点で $\sigma_C \in P$ であるときは，解の入れ換えを行わない．

SA 法では，温度 t の制御に様々な工夫が可能である．ここでは [83] と同様の枠組みを用いた．パラメータとして，$init_prob$，$term_ratio$，$size_factor$，$temp_factor$ の 4 つを用意する．まず，初期温度 t は，初期解 σ に対し，改悪解への移動確率が $init_prob$ 程度となるように定める．次に，一定の温度 t を用いて **2.5** で述べたステップ $1 \sim 3$ のループを $size_factor \cdot |N|$ 回反復した後，$t := temp_factor \cdot t$ とする操作を繰り返す．この反復は，改善解への移動が生じないループが連続して $term_ratio \cdot |N|$ 回以上生起した時点で終了する．このルールで反復を終了した時点で，サンプル数が指定した数値に達していない場合には，ランダムな初期解より再び反復を行う．TA 法の閾値 τ も同様の制御が可能なため，同じ規則を用いた．

GDA 法については，パラメータ ω の制御ルールを与える必要がある．パラメータ $rain_speed$ （$0 < rain_speed \leqq 1$）を用意する．初期解 σ に対し，初期水位を $\omega := cost(\sigma)$ とし，各反復では $N(\sigma)$ 中よりランダムに 1 つ選んだ解 σ' に対し，$cost(\sigma') < \omega$ ならば $\omega := \omega - rain_speed(\omega - cost(\sigma'))$ とする．終了条件は SA 法と同様とした．

TS 法では，移動ルールやタブーリスト T の定義などに様々な工夫が可能である．ここでは，$N(\sigma)$ の解を順番に調べて行き，$cost(\sigma') < cost(\sigma)$ かつ $\sigma' \notin \{\sigma\} \cup T$，または，$cost(\sigma') < cost(\sigma_{best})$（特別選択基準）をみたす最初の解 σ' に移動するというルールを用いた．σ_{best} は過去の探索中に見つかった最良解である．以上のルールが $N(\sigma)$ に適用できない場合（即ち，σ は（T によって制限された近傍における）局所最適解）には，$N(\sigma) \setminus (\{\sigma\} \cup T)$ 内の最良解に移動する．タブーリストは T_{job} と T_{pos} の 2 つを試みた．これらは，$T_{job}(\sigma) = \{\sigma' \in N(\sigma) \mid \sigma$から$\sigma'$への移動は，過去 $tabu_tenure$ 回の移動において位置が変わったある仕事の位置を再び変更する $\}$，$T_{pos}(\sigma) = \{\sigma' \in N(\sigma) \mid \sigma$から$\sigma'$への移動は，過去 $tabu_tenure$ 回の移動の間に位置が変わったある仕事をもとの位置にもどす $\}$ と定義される．ただし，$tabu_tenure$ はタブー期間（tabu tenure）と呼ばれるパラメータである．また，過去の探索において，仕事 i が位置 k から他へ移った移動の回数や，仕事 i が位置 k にスケジュールされていた期間の長さ（移動回数で計る）を長期メモリとして保持しておき，同じ移動が繰り返される場合に解の評価関数にペナルティを加える方法も併用した．

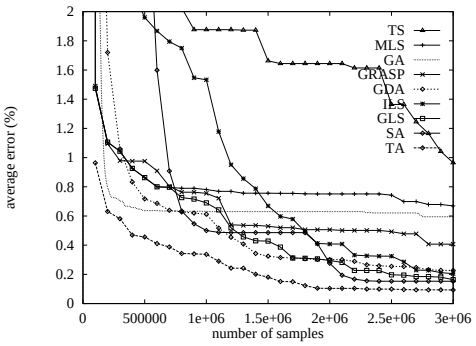

図 1　サンプル数に対する解の精度 (%)（近傍 N_{ins}，突然変異 INS の場合）．
Fig. 1　Average error in % against the number of samples (neighborhood N_{ins} and mutation INS are used).

4.3　SMP に対するメタ戦略の比較

本節では，**4.2**で述べた様々なメタ戦略の比較を行う．図 1と 2に，$n = 100$ の問題例に対して，各メタ戦略で得られた最良解の平均誤差 (%) の，サンプル数に対する変化の様子を示す．挿入近傍を用いた場合の結果が図 1，交換近傍を用いた場合の結果が図 2である．GA 法については，N_{ins} と N_{swap} に対応して，突然変異にそれぞれ，INS と SWAP を用いた場合の結果を示す．各アルゴリズムのパラメータ設定等は，各アルゴリズムに対していろいろ試した後，最良と思われ

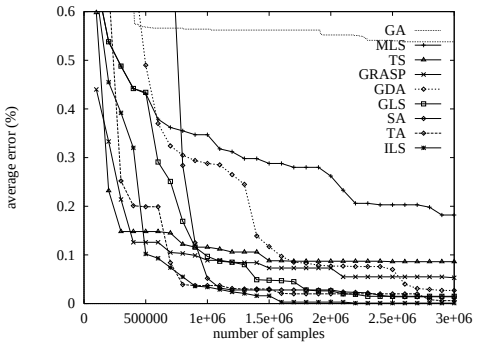

図 2　サンプル数に対する解の精度 (%)（近傍 N_{swap}，突然変異 SWAP の場合）.

Fig. 2　Average error in % against the number of samples (neighborhood N_{swap} and mutation SWAP are used).

るものを用いた．

図より，以下のことが観測できる．(1) GA 法により得られる解の精度はあまり高くない．(2) GRASP 法，ILS 法，GLS 法，SA 法，TA 法，および，GDA 法は，いずれも MLS 法よりも高い性能を示している．その中でも，同じ近傍を用いた場合には，ILS 法，GLS 法，SA 法，TA 法，および，GDA 法が特に良好な結果を得ている．(3) GA 法以外の全ての方法の性能は，近傍に大きく依存し，N_{swap} が N_{ins} より優れた結果を与えている．

4.4　充足最大化問題

MAXSAT の問題例は [134] で用いられた $n = 100$，$m = 850$ の問題例 (URL: `http://www.research.att.com/~mgcr/data/maxsat.tar.gz` より入手) を用いた .[134] で用いられた問題例は ,DIMACS (URL: `ftp://dimacs.rutgers.edu/pub/challenge/sat/benchmarks/cnf/`) のベンチマーク問題 `jnh` ([74] に従って生成された問題例) に，各節の重みを [1, 1000] の正数からランダムに選んだ値とすることで生成されたものである．

4.5　MAXSAT に対するメタ戦略の概要

MAXSAT に対し，MLS 法，GRASP 法，ILS 法，GA 法，GLS 法，SA 法，TA 法，および TS 法を構成した．これらの概要を以下に述べる．各メタ戦略の枠組は SMP と同様とし，問題の性質に応じて変更した点のみ説明する．

MLS 法では，II-A （近傍）には，$N_k(v) = \{v' \in \{0,1\}^n \mid v'$ と v のハミング距離は $k\}$（k フリップ近傍と呼ぶ) を用い，$k = 1$ と 2 を調べた．この内，N_2 については，MLS 法による解の精度は改善されるものの，定めたサンプル数内で可能な多スタートの回数が N_1 に比べて極めて少なくなり，メタ戦略の効果が確認できるほどの反復が不可能であったため，他のアルゴリズムでは N_1 のみを利用した．また，移動戦略については，即時移動戦略と最良移動戦略の両方を調べたが，SMP の場合と同様に，即時移動戦略を用いる方がより良い解が速く求まる傾向が観測されたので，他のアルゴリズムでは即時移動戦略のみを利用した．

GRASP 法の初期解生成には ,[81] と [134] の欲張り法とそれらの変形をあわせて 4 通りを調べた．なお ,[81] の欲張り法では，確率 1/2 でランダムに定めた 0-1 割当による $gain$ の期待値以上の解が得られることが保証されている．

ILS 法の突然変異は，$nperturb$ をパラメータとし，解 v に対し，近傍 $N_{nperturb}(v)$ 中から解 v' をランダムに 1 つ選び，$v := v'$ とする操作とする．

GA 法と GLS 法では，交叉法に 0-1 ベクトルに対する 1 点交叉，2 点交叉，および一様交叉 [33], [65] を用いた．なお，MAXSAT の場合は，変数の順序にはとくに意味がないにもかかわらず，1 点交叉と 2 点交叉では変数の順序により生成され得る解集合が異なってしまうため，交叉を行うたびに変数の順序をランダムに並べ直す方法も試した．突然変異は，解 v に対し，近傍 $N_1(v)$ 内の解 v' をランダムに選び，$v := v'$ とする操作とした．

TS 法では，タブーリストに，$T_{var}(v) = \{v' \in N_1(v) \mid v$ から v'への移動は過去 $tabu_tenure$ 回の移動の間に 0-1 割当が変わったある変数の割当を再び変更する $\}$ を用いた．また，短期メモリのみによる TS 法によって一定の性能が得られたので，長期メモリは用いない．

4.6　MAXSAT に対するメタ戦略の比較

図 3に，**4.5**で述べた様々なメタ戦略による最良解の平均誤差 (%) のサンプル数に対する変化の様子を示す．各アルゴリズムのパラメータ設定などは，各アルゴリズムに対していろいろ試した後，最良と思われるものを用いた．

図より，以下が観測できる．(1) GA 法の性能は MLS 法と同程度である．(2) それ以外の方法は MLS 法よりも高い性能を示す．

4.7　手軽なツールとしてのメタ戦略の利用法

4.3と **4.6**の実験の結果を総合して，以下が観測さ

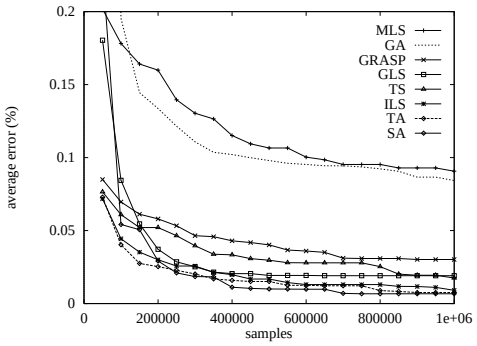

図 3 サンプル数に対する解の精度 (%) ($n = 100$, $m = 850$ の場合).
Fig. 3 Average error in % against the number of samples ($n = 100, m = 850$).

れる．

(i) GA 法の性能は MLS 法と同程度かそれ以下である．
(ii) TS 法の性能は問題や近傍に大きく依存する．
(iii) ILS 法，GLS 法，SA 法 (およびその変形) では，MLS 法よりも高い性能が得られ，手法による性能差は MLS 法からの改善の度合いに比べれば小さい．

このうち (ii) の原因については本節の後半で考察する．また (i) と (iii) の原因については，**5. 1**の POP に基づく説明が可能である (詳細は**5. 1**で述べる)．2 つの問題に対する限られた実験結果のみから一般的な結論を導くことは困難であるが，上述の傾向は 2 つの問題に共通して観測されたものであり，他の多くの問題に対しても似通った傾向が得られることが期待できる．また，紙面の都合上省略したが，前節までに述べた比較結果の他に，各アルゴリズムの内部オペレータやパラメータに対する振る舞いについても詳細な実験を行った．その中で観測された傾向は，本節の後半でアルゴリズムごとに簡単に紹介する．

以上の結果に，アルゴリズムの実現の手間などの後述の他の要因を加味して，総合的に判断すると，メタ戦略を「手軽なツール」として利用する際の設計指針としては，

(1) まず MLS 法を試み，その際近傍は十分工夫する，
(2) より高い精度が要求される場合には，ILS 法を試みる，
(3) ILS 法であまり効果が得られなかった場合には，GLS 法か SA 法を試みる，

のが良いと思われる．なお，指針 (3) には，定評のある枠組みの一つとして SA 法を挙げたが，本研究の実験結果からは，その変形である TA 法や GDA 法でも十分と思われる．

各アルゴリズムのオペレータやパラメータに対して観測された振る舞いと，アルゴリズム実現の手間に関する考察を，上述の指針の理由として以下に挙げておく．なお，アルゴリズム実現の手間に関する評価は，著者の主観的なものであることを断っておく．

MLS 法: MLS 法は LS 法を反復するだけなので，実現は容易である．また，MLS 法以外のメタ戦略において，アルゴリズム内部のオペレータが適切でなかったり，パラメータが十分に調整されていないと，MLS 法よりも性能が悪くなることが観測された．SMP では，2 通りの近傍を試したが，交換近傍を用いた場合の MLS 法の性能は，挿入近傍を用いた場合のメタ戦略の最良のものとほぼ同程度であった．よって，近傍の性能を調べたり，様々なアルゴリズムとの比較の基準を与えるアルゴリズムとして，まず MLS 法を作成することが推奨される．

ILS 法，GLS 法，SA 法: これらはいずれも，パラメータをうまく選べば MLS 法よりもかなり高い性能が得られ，しかも，様々なパラメータや内部オペレータに対して比較的ロバストな挙動を示した．MLS 法からの改善の度合いに比べれば，これらの枠組の違いによる性能差は小さいといえる．以下にアルゴリズムの実現の手間を議論しておく．ILS 法と GLS 法は，初期解生成に工夫が加わるだけなので，MLS 法がすでに出来ていれば，単純 LS 法のルーチンをそのままモジュールとして利用でき，移動ルールに変更が加わる SA 法などよりも実現がやや容易と思われる．ILS 法と GLS 法の間では，複数個の候補解を扱ったり，交叉を作る手間が不要な ILS 法の方が容易である．

GA 法: $|P|$ の値や交叉法などをいろいろ試した最良の結果が MLS 法と同程度であった．既に多くの研究者により認識されている (たとえば [129]) ように，単純 GA 法ではあまり高い性能は期待できず，性能を上げるには，LS 法や他の近似解法とのハイブリッドが必要である．

GRASP 法: 性能は初期解生成に用いる欲張り法やパラメータの値に大きく依存し，MLS 法よりも悪くなる場合もあった．また，欲張り法による初期解の

精度を上げても GRASP 法の性能が向上するとは限らないことが観測された．うまく設計された GRASP 法の解の精度は，MLS 法よりはよいが，他の様々な戦略と比べてとりわけ良いというわけではなかった．一方，GRASP 法に必要な欲張り法を構築する手間は，通常，単純 LS 法を構築する手間と同程度か，それ以上に大変であった．以上が GRASP 法を手軽なツールとしては推奨できない理由である．しかし，GRASP 法の 1 回の LS 法により得られる解の精度は，通常，MLS 法に比べて平均的に向上するため，制限時間が短く，LS 法の反復を少数しか行えないような状況では有利であると考えられる．また，初期解生成の欲張り法には，問題固有の性質を取り入れる様々な工夫を加えやすく，そのような工夫による良い結果も多く報告されていることを断っておく．

TS 法: MAXSAT では，短期メモリのみを用いた簡単な実現でも十分な性能が得られたが，性能はパラメータ $tabu_tenure$ に敏感であった．また，SMP では，短い周期のサイクリングが起こりやすく，アルゴリズムがあまりうまく動作しなかった．もちろん，SMP に対しても，タブーリストを工夫するとか，TS 法の他の様々なアイディアを組合せるなどにより，より性能の高いアルゴリズムを構成できる可能性はあるが，これでは手軽なツールとはいえない．さらに，TS 法の基本動作は，近傍内のタブーでない最良解に移るというものであるが，これを単純に実現すると，1 回の移動に時間がかかりすぎる．これを避けるため，通常，TS 法を構築する際には，1 回の移動にかかる計算時間を小さくするために，データ構造などの工夫を加える．本研究の実験における TS 法の性能がそれほど良くなかったのは，手軽なツールとしての性能を比較するため，このような工夫を一切加えなかったことも原因の一つと思われる．

一方，TS 法の枠組は，**5.**で述べるような高度なテクニックを組合せやすいものとなっており，そのような工夫により非常に性能の高いアルゴリズムを構成したという報告が数多くある．すなわち，TS 法は上級者向けのアルゴリズムであるといえよう．

なお，本研究と動機は異なるが，メタ戦略のいくつかを比較した文献としては，[4], [30], [38], [82], [84], [102], [140], [147] などがある．ただし，比較実験による結果は，取り扱う問題により異なるので，一般的な結論は導き難い．

5. 高性能アルゴリズムの設計

前節で調べたような，メタ戦略の簡単な実現よりも，さらに高い性能が望まれる場合は，「どの戦略を用いるか」よりもむしろ「どれだけ問題の構造をうまく利用するアイディアを加えたか」による影響の方が大きいと思われる．また，実用的なツールとして用いる場合は，問題が変わるたびにパラメータ調整が必要では困るので，重要なパラメータを自動的に調整するメカニズムを加えるなどの工夫も要る．これらのアイディアを説明する前に，メタ戦略アルゴリズムの基本概念である，proximate optimality principle (以下 POP と記す)，および集中化 (intensification) と多様化 (diversification) についてふれておく．

5.1 メタ戦略の基本概念

POP は，「よい解同士は似通った構造を持っている」という漠然とした概念である．メタ戦略のアルゴリズムは，この考え方に基づく部分が大きい．すなわち，過去の探索で見つかった良い解の構造を多く含むような解を，次々と新たに生成していくアルゴリズムになっている．これを実現するためのメカニズムが，例えば近傍操作，突然変異，交叉などのオペレータである．メタ戦略の設計には，良い解が共有している構造が何かをうまく見つけ出すことが成功の鍵といえる．たとえば，TSP では，解に含まれる枝集合 (巡回路において隣合う街の対の集合) が重要であることが知られている．TSP に対する近似解法としては，Lin と Kernighan による LS 法 [99] が有名であり，この方法によって得られた局所最適解は非常に精度が高いことが知られているが，2 つの局所最適解に共通する枝が平均的に 85%もあることが [99] において観測されている．

また，MAXSAT では，各変数 y_j への 0-1 割当 v_j，すなわち v の各要素に対し POP が成立していることが，以下のデータにより分かる．図 4は，$n = 100$，$m = 850$ の問題例の 1 つに対し，横軸に最適解からのハミング距離，縦軸に最適解からの誤差をとり，解が分布する様子を示したものである．ただし，全ての解をプロットするのは無理なので，各ハミング距離に対し，1000 個の解をランダムに生成した結果である．図より，最適解からのハミング距離と解の精度に強い正の相関があることが確認できる．このような，基本オペレータと目的関数の相関に対する考察は，[17], [86], [104], [130], [161] などにある．ま

た，MAXSAT に対する POP のより詳しい実験結果は [158] にある．

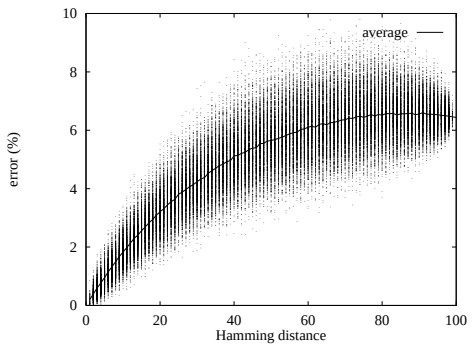

図 4　最適解からのハミング距離と解の誤差 (%) の分布．
Fig. 4　Error in % against Hamming distance from the optimal solution.

探索法の設計においてしばしば言及される集中化は，良い解が見つかったときは，その解の近くを丹念に探索することを (POP の具現化)，多様化は，それとは逆に時々は未探索の領域を探索することを示すものである．これらは TS 法の基本概念として用いられるが，この考え方は全てのメタ戦略に共通して重要である．メタ戦略の様々なアイディアがどちらを目的としているかをよく理解し，これらの互いに相反する動作をバランスよく組込むことが求められる．

4. 7で述べた傾向 (i) と (iii) は POP によってある程度説明できる．まず (i) については，GA 法の交叉によって得られる解は，通常，LS 法の近傍操作によって得られる解に比べて，元の解の構造を十分に保存していない場合が多く，従って，GA 法では集中化の能力が十分でない (または多様化を頻繁に行いすぎる) ことが原因と考えられる．また (iii) については，ILS 法，GLS 法，SA 法 (およびその変形) の動作は，過去の探索によって得られた良い解の付近を LS 法でさらに詳しく調べるという原理に基づいており，集中化の能力が他の手法に比べて高いことが原因と考えられる．

5. 2　近　　傍

3.の枠組の II-A の近傍の定義は，LS 法の設計において最も重要なものの 1 つである．近傍を設計する際には，**5. 1**で述べた良い解が共有する構成要素をうまく取り入れることが重要である．たとえば，TSP は街の訪問順序を求める問題なので，**4. 2**の SMP に対する実験で用いた交換近傍や，挿入近傍を用いることも出来る．また，他の近傍としては，挿入近傍を少し拡張して，現在の巡回路において隣合う高々3 つまでの街を現在の位置から抜き出し，他の位置へ挿入することにより得られる解集合を近傍とする Or-opt 近傍や，現在の巡回路と異なる枝数が高々λ 本 (λ は定数; $\lambda = 2, 3$ がよく用いられる) である解集合を近傍とする λ-opt 近傍などがよく知られている．これらの中では，枝を基本に考えた λ-opt 近傍が強力であることが知られている [84], [95]．このように，問題によっては様々なタイプの近傍の定義が可能であり，それによって性能も大きく変わりうる．また，複数の近傍が可能な場合には，それらのうちいくつかを組合せて用いることも可能である [157], [159], [160]．

より高度な近傍の構成法として，可変深度近傍探索法 (variable depth search) と呼ばれるものがある．これは，Lin と Kernighan により提案され，グラフ分割問題 [87] と TSP [99] に対して成功を収めたアイディアで，単純な近傍操作を，連鎖的に複数回反復することで生成され得る解集合を，改めて近傍と定義する方法である．しかし，このルールを単純に実行したのでは，生成され得る解の数が指数的に増大してしまうので，問題構造を利用して，次に行う近傍操作の選択や反復の打ち切りを行う必要がある．同様のアイディアは TS 法でも時々用いられており，その場合は ejection chain とも呼ばれる [21], [56], [122], [157]．また，[13] では，同様のアイディアを guided local search と呼んでいるが，**3. 2**で説明したものとは異なる．このようにアルゴリズムの呼び方は統一されていない場合もあるので注意を要する．より詳しくは，Lin と Kernighan の原著 [87], [99]，解説 [121]，または最近の適用事例 [21], [122], [157], [159], [160] を参照のこと．

近傍探索の作業を効率化することも重要な課題の 1 つである．たとえば，$N(x)$ 内に改善解が存在すれば，それを逃さず発見できるという条件を満足しつつ，探索する解を $N(x)$ の一部に限定してしまうという手法は，様々な問題に対して可能である．TSP に対する 2-opt 近傍の例を説明しておく．現在の巡回路の枝 $\{i, j\}$ と $\{k, l\}$ を取去り，新たに $\{i, k\}$ と $\{j, l\}$ を加えることにより改善解が得られるとする．すなわち $d_{ij} + d_{kl} > d_{ik} + d_{jl}$ である．このとき，$d_{ij} > d_{jl}$ と $d_{kl} > d_{ik}$ の内少なくとも一方が成り立つことがいえる．これを用いると，街 i に関する枝交換の操作を (1) 枝 $\{i, j\}$ を取去り $\{j, l\}$ を加える，(2) 枝 $\{k, l\}$ を取去り $\{i, k\}$ を加える，と 2 段階に分け，第 1 段階で加

える枝の候補を $d_{jl} < d_{ij}$ を満たすもののみに限定できる．ただし，各街につながっている 2 つの枝に対して方向を変えて同様の探索を行う必要がある．第 2 段階の操作は，第 1 段階に対して枝の候補が一意に定まる．この方法により近傍探索の手間を大幅に縮小できる．なお，TSP の各街が座標により平面上の点として与えられる場合 (Euclidean TSP と呼ばれる) には，枝 $\{i, j\}$ に対して $d_{jl} < d_{ij}$ を満たすような街 l の集合を効率よく列挙する K-d 木という特別なデータ構造も研究されている [16]．同様の近傍縮小は 3-opt 近傍についても可能である．Lin と Kernighan の方法は，このような近傍縮小に用いられたルールを一般の λ に対する λ-opt 近傍まで拡張したものと捉えることもできる (ただし，λ-opt 近傍内に改善解があればそれを逃さないという保証はない)．また，LS 法の様々な基本操作を効率的に行うために解を記憶するためのデータ構造を工夫する研究もある [47]．これらの詳細については [84] を参照のこと．

同様の考え方に基づく近傍縮小の例として ,[156] では，MAXSAT に対して，2 フリップ近傍 N_2 と 3 フリップ近傍 N_3 の縮小を行い，さらに，縮小された近傍のサイズに対する解析を行っている．たとえば，2 フリップ近傍については，何も工夫をしない場合には $|N_2| = O(n^2)$ であるが，縮小近傍では，ランダムに与えられた解 v に対して，$N_2(v)$ 内で探索される解の数の期待値が $O(n + m)$ で抑えられることを証明し，これが現実によく合うことを計算実験により確認している．N_3 に対しても同様の結果が示されている．

5.3 その他の工夫

メタ戦略アルゴリズムには，通常，探索を制御するいくつかのパラメータが含まれる．その中には，解くべき問題例の性質に敏感に反応し，十分な性能を得るためには，問題例のサイズやタイプが変わるたびに調整し直す必要があるものもある．これを避けるため，探索の状況に応じてパラメータを自動調整しようという試みがある．たとえば TS 法のタブー期間 (**4.2** と **4.5** の *tabu_tenure*)[14], [112] や，ペナルティ関数法 (**3.2**) のペナルティ重み [51], [157] がその対象として挙げられる．

解こうとする問題が整数計画問題として定式化できる場合，そのラグランジュ緩和や LP (linear programming) 緩和を解くことにより得られる解や相対コスト (relative cost，reduced cost) の情報を探索に利用することもできる．比較的古くから知られているが [15]，最近でも，集合被覆問題 (set covering problem) に対する優れた結果が報告されるなど [22], [23]，さらなる発展が期待できる．これまで提案されている手法の多くは，(1) 相対コストの情報を利用して一部の変数の値を固定し，探索の対象となる変数を減らす，(2) 緩和問題に対する最適解 (または十分良い解) を修正して元問題の実行可能解を生成する，(3) 相対コストの情報を欲張り法の局所利得に利用する，という考え方に基づいている．これとは別の考え方として，相対コストの情報を利用して LS 法における近傍を縮小するというアイディアも可能である [157]．類似の試みとして，代替双対問題 (surrogate dual) の情報を TS 法の制御に用いた例もある [68]．このように，緩和問題の情報の利用法はいろいろ考えられるので，メタ戦略に組合せて用いる新たなアイディアが今後も出てくることが期待できる．

解こうとする問題の部分問題などに専用の厳密解法や近似解法が利用できる場合は，これらを基本オペレータとして利用することも可能である．たとえば ,[154] では，順序づけ問題に対し，保持している候補解の 2 つに共通する半順序関係に矛盾しない解の中で最適なものを動的計画法 (dynamic programming，DP) を用いて求めるという操作を GA 法の交叉オペレータとして利用することを提案し，一定の成果を納めている．同様のアイディアが，独立節点集合問題 (independent set problem) に対して提案されており，optimized crossover と呼ばれている [5]．このような混成アプローチについては [77], [129] などに解説がある．

実用性の観点からは，解きたい問題に対してその都度アルゴリズムを作る代わりに，汎用性の高い問題 (問題 A とする) に対してメタ戦略アルゴリズムを用意しておき，解きたい問題を問題 A に帰着して，用意しておいたアルゴリズムを利用できれば，アルゴリズム開発の手間が省ける．この観点から，汎用ツールとしてのメタ戦略アルゴリズムを開発しようという試みもすでに始まっており ,[112] では，様々な問題を制約充足問題 (constraint satisfaction problem, CSP) に帰着することにより同一のアルゴリズムで解き，良好な成果を納めている．また ,[100] では 0-1 混合整数計画問題に対して TS 法に基づく汎用解法を提案している．充足可能性問題に対する WALKSAT 法のアイディアを整数計画問題に拡張し，汎用解法としての性能を詳しく調べた報告もある [151]．スケジューリング

問題に対する汎用解法の開発の試みもある [113] .

6. メタ戦略の理論

本節では，メタ戦略に対する理論的成果として，SA 法などに対する漸近収束性と，LS 法の計算の複雑さ，および近似精度に関する話題にふれておく．

SA 法については，ある仮定の下で厳密な最適解への漸近収束性が成立することが知られており (例えば [90], [103])，それが SA 法の魅力の一つととらえられている．TA 法や GA 法，さらには，TS 法に確率的動作を導入した確率的 TS 法などについても漸近収束に関する同様の研究がある [7], [41], [137] . しかし，漸近収束性はあくまでも「十分に計算時間をかけた」時の理論的性質であり，実際に用いられるアルゴリズムの性能に直接関係するとは考えない方がよい．実際，SA 法において漸近収束までに必要な計算時間については，多項式時間アルゴリズムが存在するマッチング問題に対してすら指数時間必要となるなど [136]，否定的な結果が知られている．実用的手法の観点からは，物理現象のアナロジーや理論的成果にあまりとらわれず，柔軟な姿勢でアルゴリズムを構築していくことが重要であると思われる．

単純 LS 法では，近傍 $N(x)$ を 1 回の移動に必要な操作が多項式時間で収まるように定めるのが普通である．しかし，初期解から局所最適解に到達するまでに要する移動回数は多項式オーダーで収まるとは限らない．実際，たとえば，TSP に対する 2-opt 近傍については，$\Theta(2^{n/2})$ 回以上かかるような問題例が存在することが知られている．最悪ケースに対するこのような否定的な結果は他にもいろいろある [84] . もちろん，単純 LS 法では，1 回の移動で評価関数 $\tilde{f}$ の値が真に減少するので，移動回数を $\tilde{f}$ が取り得る値の種類数 (多くの場合は擬多項式オーダー) で抑えることはできる．これらを計算の複雑さの観点から議論するため，単純 LS 法における局所最適解の 1 つを求める問題の全体を数学的にきちんと定義し，クラス PLS (polynomial-time local search) と呼んでいる [85] . この概念に関し，様々な解析が行われており，例えば，TSP に対する λ-opt 近傍 (λ は十分大) や，MAXSAT に対する 1 フリップ近傍の PLS 完全性が示されている [163] . これは，クラス PLS に含まれる全ての問題が多項式時間で解けない限りは，局所探索法に限らず，いかなる方法を用いても，PLS 完全問題の局所最適解を 1 つ求めることが最悪の場合多項式時間では出来ないことを意味する．

一方，現実の単純 LS 法が局所最適解を発見するまでに要する移動回数は，通常，十分実用的である．そこで，問題例の生成に対する確率的な仮定をおいて期待計算時間を解析することが試みられている．たとえば，TSP において，平面上にランダムに街の座標が与えられ，距離をユークリッド距離とした問題例では，2-opt 近傍による移動回数の期待値が $O(n^{10}\log n)$ で抑えられることなどが示されている [25] . 最近では，これを λ-opt 近傍に拡張した結果もある [114] . しかし ,[84] の実験結果では，移動回数の平均値は $O(n\log n)$ 程度と，理論値よりもかなり小さくなっており，理論と実際とのギャップが大きいことが伺える．

局所最適解の精度についても，理論的な解析が行われている．アルゴリズムによって得られる解 x の目的関数値 $f(x)$ と最適値 $f(x^*)$ の比 $f(x)/f(x^*)$ を近似度 (approximation ratio) と呼ぶ．近似度は，最小化問題ならば 1 以上，最大化問題ならば 1 以下となり，1 に近いほどよい．例えば，TSP については，$P \neq NP$ の仮定の下では，局所最適性の判定が多項式時間で可能な場合，近似度を定数ではおさえられないことが知られている [120] (クラス P と NP については [50] 参照 .) なお，TSP については，$P \neq NP$ ならば，多項式時間アルゴリズムでは近似度 $2^{poly(n)}$ ($poly(n)$ は n の任意の多項式) を実現できないことも知られているが [135]，単純 LS 法は多項式時間で終了する保証がないので，この結果をそのまま LS 法には適用できない．問題例に制限を加えた場合は，以下のもう少し肯定的な結果が [25] に示されている ([84] にも結果の紹介がある)．距離行列 (d_{ij}) が任意の i,j,k に対して $d_{ij} \leqq d_{ik} + d_{kj}$ (三角不等式と呼ばれる) を満たす場合は，近似度の下界は，2-opt 近傍で $(1/4)\sqrt{n}$，λ-opt 近傍で $(1/4)n^{1/(2\lambda)}$ (λ($\geqq 3$)は任意) となり，上界は 2-opt 近傍で $4\sqrt{n}$ となる．また，街の座標が k 次元空間 (k は定数) に与えられ，距離行列がそれら間のユークリッド距離で与えられる場合は，2-opt 近傍による近似度の上界は $O(\log n)$ となり，下界は $k=2$ で $\Omega(\log n/\log\log n)$ となる．さらに，街の座標が k 次元空間にランダムに与えられる場合は，2-opt 近傍による近似度の期待値は定数でおさえられる．しかし，LS 法に基づかない方法では，三角不等式が成り立つ場合には，近似度 1.5 [27]，k 次元空間上のユークリッド距離の場合には，任意の $\varepsilon > 0$ に対して近似度 $1+\varepsilon$ の多項式時間アルゴリズムが知られており [8] ([144]

に解説がある），これらに比べると，局所最適解の近似度は理論的には劣っているといえる．

一方，[8] の近似度 $1+\varepsilon$ のアルゴリズムは，計算時間が $O(n((\log n)/\varepsilon)^{100/\varepsilon})$ となり [144]，理論的にはほぼ線形時間であるが実用的ではない．また，[84] の計算実験では，例えば，2 次元平面上にランダムに街を置き，距離行列をそれらの間のユークリッド距離で与えた $n = 1000$ の問題例に対して実際に観測された近似度の平均値は，[27] のアルゴリズムで 1.097，ランダムな初期解から始めた単純 LS 法の局所最適解では，2-opt 近傍で 1.079，3-opt 近傍で 1.038 と報告されており，いずれも理論的な保証値よりも遥かに良い値であることが確認できる．

MAXSAT に対しても同様の解析がある．節に含まれるリテラル数の最小値が q である問題例においては，近傍に N_1 を用いた場合の局所最適解の近似度は $q/(q+1)$ であり，また，これ以上の改善は不可能であることが知られている [69]（文献 [69] は各節の重みを 1 に限定して議論しているが，これを正の重みつきの場合に拡張することは容易である．また，近似度の下界については，特定の問題例を与えるにとどまっているが，これを一般の q とサイズに拡張することも容易である．）

一方，MAXSAT に対しては，各変数に 1 を割当てる確率を与えてランダムな 0-1 割当を定めた場合の目的関数の期待値に基づく近似度の解析が盛んである．代表的な近似度として，Johnson による 0.5 [81]，Yannakakis による 0.75 [162]，Goemans と Williamson による 0.75 [62] および 0.7584 [63] などがある．このうち，Goemans と Williamson の線形計画法（LP）や半正定値計画法（SDP）による緩和問題を利用した解法は，近似度に対する理論的保証を与える汎用的な手法として，近年着目されている．これらの解説は [9] にある．また，最近では，近似度 0.7846 のアルゴリズムも提案されている [10]．以上は q にとくに制限を与えない場合の結果であるが，q を用いた場合は，例えば Johnson のアルゴリズムによって近似度 $1-1/2^q$ が実現できる．よって，これらに比べると，近傍 N_1 による局所最適解の近似度は劣るといえる．

しかし，実際の計算実験では，例えば **4. 4**の問題例に対して以下の傾向が観測されている（なお，**4. 6**は $m = 850$ の 17 問に対する結果であるが，以下の結果は $m = 800$ と 900 の場合を含めた 44 問に対する近似度の平均値である．)[9] では，Johnson のアルゴリズムと Goemans と Williamson の LP 緩和に基づくアルゴリズム，およびその変形をいくつか調べている．まず，各変数に 1 を与える確率に従ってランダムに 0-1 割当を定めるときの近似度の期待値の平均値は，Johnson のアルゴリズムで 0.935 程度，LP 緩和に基づく手法の最良のもので 0.95 程度である．また，これらの確率を利用した欲張り法によって得られた解の近似度は期待値よりも良いことが保証されるが，実際には，Johnson のアルゴリズムで 0.997 程度，LP 緩和に基づく手法の最良のもので 0.998 程度と，期待値よりもかなり良くなる．これに対し，単純 LS 法による局所最適値の近似度を求めるための実験を行ったところ，その平均値は，N_1 では 0.9927，N_2 では 0.9970，N_3 では 0.9982 程度となることが観測された．また，スタート回数を 1000 回とした MLS 法による近似度は，N_1 で 0.9988，N_2 で 0.99988 程度となる．なお，これら MLS 法の計算時間は，Sun Ultra 2 Model 2300 を用いて，1 問当り N_1 で 3.9 秒，N_2 で 16.8 秒程度と，十分高速である．結局，MAXSAT に対しても，理論的な近似度の保証と実際に得られる解の精度には大きなギャップがある．

以上の他にも理論的解析の試みはいろいろある．たとえば，GA 法については，スキーマ理論など，GA 法に独特な様々な解析がなされている [11], [65]．また，GA 法に類似の多点探索法で go with the winners 法と呼ばれるものを提案し，確率的解析によって，1 点による LS 法のような探索法よりも有利な状況が存在することを示した研究もある [6], [34]．しかし，メタ戦略がなぜうまく行くのかを説明するような理論的成果は今のところ見つかっておらず，今後の発展が望まれている．

7. ま と め

メタ戦略を局所探索法の一般化と捉えることで体系的に整理し，様々なメタ戦略アルゴリズムのアイディアを紹介した．次に，メタ戦略を手軽なツールとして利用することを念頭に置き，1 機械スケジューリング問題と充足最大化問題に対する計算実験を通していろいろなメタ戦略アルゴリズムを比較した．その結果をもとに，メタ戦略アルゴリズムの簡単な設計指針を提示した．最後に，メタ戦略をさらに高性能にするためのアイディアや，これまでに知られている理論的解析の試みについて紹介した．

現在では，メタ戦略は，大規模な組合せ最適化問題

に対処するために必要不可欠なツールとして広く認知されている．しかし，その理論的性質などはほとんど解明されておらず，まだまだ斬新なアイディアが必要とされている．本稿がメタ戦略の今後の発展の一助となれば幸いである．

謝辞　注意深く読んでいただき，貴重なご助言をいただいた査読者の方々に感謝いたします．なお，本研究は一部文部省科学研究費の援助を得たものである．

文　献

[1] E.H.L. Aarts and J.H.M. Korst, *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, 1989.

[2] E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, 1997.

[3] E.H.L. Aarts, J.H.M. Korst and P.J.M van Laarhoven, "Simulated annealing," in: E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp.91–120, 1997.

[4] E.H.L. Aarts, P.J.M. van Laarhoven, J.K. Lenstra and N.L.J. Ulder, "A computational study of local search algorithms for job shop scheduling," *ORSA J. Computing*, vol.6, pp.118–125, 1994.

[5] C.C. Aggarwal, J.B. Orlin and R.P. Tai, "Optimized crossover for the independent set problem," *Operations Research*, vol.45, pp.226–234, 1997.

[6] D. Aldous and U. Vazirani, ""Go with the winners" algorithms," *Proc. 35th Annual Symposium on Foundations of Computer Science*, pp.492–501, 1994.

[7] I. Althöfer and K.-U. Koschnick, "On the convergence of "threshold accepting"," *Applied Mathematics and Optimization*, vol.24, pp.183–195, 1991.

[8] S. Arora, "Nearly linear time approximation schemes for Euclidean TSP and other geometric problems," *Proc. 38th Annual Symposium on Foundations of Computer Science*, pp.554–563, 1997.

[9] 浅野孝夫，"論理システム解析のための高性能近似アルゴリズム — 四角い問題を丸くして解く，" 情報処理，vol.39, pp.683–688, 1998.

[10] T. Asano and D.P. Williamson, "Simpler approximation algorithms for MAX SAT," Technical Report of IEICE, COMP99-14, pp.33–40, 1999.

[11] T. Bäck, D.B. Fogel and Z. Michalewicz, eds., *Handbook of Evolutionary Computation*, IOP Publishing Ltd and Oxford University Press, 1997.

[12] K.R. Baker and G.D. Scudder, "Sequencing with earliness and tardiness penalties: a review," *Operations Research*, vol.38, pp.22–36, 1990.

[13] E. Balas and A. Vazacopoulos, "Guided local search with shifting bottleneck for job shop scheduling," *Management Science*, vol.44, pp.262–275, 1998.

[14] R. Battiti and G. Tecchiolli, "Local search with memory: benchmarking RTS," *OR Spektrum*, vol.17, pp.67–86, 1995.

[15] J.E. Beasley, "Lagrangian heuristics for location problems," *European J. Operational Research*, vol.65, pp.383–399, 1993.

[16] J.J. Bentley, "Fast algorithms for geometric traveling salesman problems," *ORSA J. Computing*, vol.4, pp.387–410, 1992.

[17] K.D. Boese, A.B. Kahng and S. Muddu, "A new adaptive multi-start technique for combinatorial global optimizations," *Operations Research Letters*, vol.16, pp.101–113, 1994.

[18] R.M. Brady, "Optimization strategies gleaned from biological evolution," *Nature*, vol.317, pp.804–806, 1985.

[19] J. Brimberg, P. Hansen, N. Mladenović and É.D. Taillard, "Improvements and comparison of heuristics for solving the multisource weber problem," Technical Report IDSIA-33-97, IDSIA, Corso Elvezia 36, CH-6900 Lugano, Switzerland, 1997 (available as `http://www.idsia.ch/~eric/articles.dir/IDSIA-33-97.ps.Z`).

[20] J. Brimberg and N. Mladenović, "A variable neighborhood algorithm for solving the continuous location-allocation problem," *Studies in Locational Analysis*, vol.10, pp.1–12, 1996.

[21] B. Cao and F. Glover, "Tabu search and ejection chains — application to a node weighted version of the cardinality-constrained TSP," *Management Science*, vol.43, pp.908–921, 1997.

[22] A. Caprara, M. Fischetti and P. Toth, "A heuristic method for the set covering problem," *Proc. 5th Conference on Integer Programming and Combinatorial Optimization*, pp.72–81, 1996, (full version is available at `http://promet4.deis.unibo.it/techrep.html`).

[23] S. Ceria, P. Nobili and A. Sassano, "A Lagrangian-based heuristic for large-scale set covering problems," *Mathematical Programming*, vol.81, pp.215–228, 1998.

[24] V. Černý, "Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm," *J. Optimization Theory and Applications*, vol. 45, pp.41–51, 1985.

[25] B. Chandra, H. Karloff and C.A. Tovey, "New results on the old k-opt algorithm for the TSP," *SIAM J. Computing*, to appear.

[26] I. Charon and O. Hudry, "The noising method: a new method for combinatorial optimization," *Operations Research Letters*, vol.14, pp.133–137, 1993.

[27] N. Christofides, *Worst-case analysis of a new heuristic for the traveling salesman problem*, Report 388, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh, PA, 1976.

[28] B. Codenotti, G. Manzini, L. Margara and G. Resta, "Perturbation: an efficient technique for the solution of very large instances of the Euclidean TSP," *INFORMS J. Computing*, vol.8, pp.125–133, 1996.

[29] A. Colorni, M. Dorigo and V. Maniezzo, "Distributed optimization by ant colonies," *Proc. European Conference on Artificial Life*, pp.134–142, 1991.

[30] H.A.J. Crauwels, C.N. Potts and L.N. van Wassenhove, "Local search heuristics for single-machine scheduling with batching to minimize the number of late jobs," *European J. Operational Research*, vol.90, pp.200–213, 1996.

[31] G.A. Croes, "A method for solving traveling-salesman problems," *Operations Research*, vol.6, pp.791–812, 1958.

[32] L. Davis, "Applying adaptive algorithms to epistatic domains," *Proc. 9th International Joint Conference on Artificial Intelligence*, pp.162–164, 1985.

[33] L. Davis, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, 1991; (嘉数侑昇，三上貞芳，皆川雅章，川上敬，高取則彦，鈴木恵二（訳），遺伝アルゴリズムハンドブック，森北出版，1994).

[34] T. Dimitriou and R. Impagliazzo, "Towards an analysis of local optimization algorithms," *Proc. 28th Annual ACM Symposium on the Theory of Computing*, pp.304–313, 1996.

[35] M. Dorigo, *Optimization, Learning, and Natural Algorithms*, PhD Thesis, Politecnico di Milano, 1992.

[36] M. Dorigo and L.M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol.1, pp.53–66, 1997.

[37] M. Dorigo, V. Maniezzo and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics — Part B*, vol.26, pp.29–41, 1996.

[38] J. Dorn, M. Girsch, G. Skele and W. Slany, "Comparison of iterative improvement techniques for schedule optimization," *European J. Operational Research*, vol.94, pp.349–361, 1996.

[39] G. Dueck, "New optimization heuristics: the great deluge algorithm and the record-to-record travel," Heidelberg Scientific Center Research Report TR89.06.011, IBM, Germany, 1989.

[40] G. Dueck and T. Scheuer, "Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing," *J. Computational Physics*, vol.90, pp.161–175, 1990.

[41] U. Faigle and W. Kern, "Some convergence results for probabilistic tabu search," *ORSA J. Computing*, vol.4, pp.32–37, 1992.

[42] T.A. Feo and J.L. González-Velarde, "The intermodal trailer assignment problem," *Transportation Science*, vol.29, pp.330–341, 1995.

[43] T.A. Feo and M.G.C. Resende, "A probabilistic heuristic for a computationally difficult set covering problem," *Operations Research Letters*, vol.8, pp.67–71, 1989.

[44] T.A. Feo and M.G.C. Resende, "Greedy randomized adaptive search procedures," *J. Global Optimization*, vol.6, pp.109–133, 1995.

[45] T.A. Feo, M.G.C. Resende and S.H. Smith, "A greedy randomized adaptive search procedure for maximum independent set," *Operations Research*, vol.42, pp.860–878, 1994.

[46] T.A. Feo, K. Venkatraman and J.F. Bard, "A GRASP for a difficult single machine scheduling problem," *Computers and Operations Research*, vol.18, pp.635–643, 1991.

[47] M.L. Fredman, D.S. Johnson, L.A. McGeoch and G. Ostheimer, "Data structures for traveling salesmen," *J. Algorithms*, vol.18, pp.432–479, 1995.

[48] B. Freisleben and P. Merz, "A genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems," *Proc. IEEE International Conference on Evolutionary Computation*, pp.616–621, 1996.

[49] L.M. Gambardella, É.D. Taillard and M. Dorigo, "Ant colonies for the quadratic assignment problem," *J. Operations Research Society*, vol.50, pp.167–176, 1999.

[50] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, 1979.

[51] M. Gendreau, A. Hertz and G. Laporte, "A tabu search heuristic for the vehicle routing problem," *Management Science*, vol.40, pp.1276–1290, 1994.

[52] F. Glover, "Tabu search — part I," *ORSA J. Computing*, vol.1, pp.190–206, 1989; part II, ditto, vol.2, pp.4–32, 1990.

[53] F. Glover, "Genetic algorithms and scatter search: unsuspected potentials," *Statistics and Computing*, vol.4, pp.131–140, 1994.

[54] F. Glover, "Scatter search and star-paths: beyond the genetic metaphor," *OR Spektrum*, vol.17, pp.125–137, 1995.

[55] F. Glover, "Tabu search and adaptive memory programming — advances, applications and challenges," in: Barr, Helgason and Kennington, eds., *Interfaces in Computer Science and Operations Research*, Kluwer Academic Publishers, 1996.

[56] F. Glover, "Ejection chains, reference structures and alternating path methods for traveling salesman problems," *Discrete Applied Mathematics*, vol.65, pp.223–253, 1996.

[57] F. Glover and G.A. Kochenberger, "Critical events tabu search for multidimensional knapsack problem," in: I.H. Osman and J.P. Kelly, eds., *Meta-Heuristics: The Theory and Applications*, Kluwer Academic Publishers, pp.407–427, 1996.

[58] F. Glover, G.A. Kochenberger and B. Alidaee, "Adaptive memory tabu search for binary quadratic problems," *Management Science*, vol.44, pp.336–345, 1998.

[59] F. Glover and M. Laguna, "Tabu search," in: *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publishing, pp.70–141, 1993.

[60] F. Glover and M. Laguna, *Tabu Search*, Kluwer Academic Publishers, 1997.

[61] F. Glover, É.D. Taillard and D. de Werra, "A user's guide to tabu search," *Annals of Operations Research*, vol.41, pp.3–28, 1993.

[62] M.X. Goemans and D.P. Williamson, "New 3/4-approximation algorithms for the maximum satisfiability problem," *SIAM J. Discrete Mathematics*, vol.7, pp.656–666, 1994.

[63] M.X. Goemans and D.P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. Association for Computing Machinery*, vol.42, pp.1115–1145, 1995.

[64] D.E. Goldberg and R. Lingle, "Alleles, loci, and the traveling salesman problem," *Proc. 1st International Conference on Genetic Algorithms*, pp.154–159, 1985.

[65] D.E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[66] J. Gu, "Optimization by multispace search," Technical Report UCECE-TR-90-001, Department of Electrical and Computer Engineering, University of Calgary, 1990.

[67] J. Gu and X. Huang, "Efficient local search with search space smoothing: a case study of the traveling salesman problem (TSP)," *IEEE Transactions on Systems, Man, and Cybernetics*, vol.24, pp.728–735, 1994.

[68] S. Hanafi and A. Freville, "An efficient tabu search approach for the 0-1 multidimensional knapsack problem," *European J. Operational Research*, vol.106, pp.659–675, 1998.

[69] P. Hansen and B. Jaumard, "Algorithms for the maximum satisfiability problem," *Computing*, vol.44, pp.279–303, 1990.

[70] P. Hansen and N. Mladenović, "An introduction to variable neighborhood search," in: S. Voß, S. Martello, I.H. Osman and C. Roucairol, eds., *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, pp.433–458, 1999.

[71] J. Hart and A. Shogan, "Semi-greedy heuristics: an empirical study," *Operations Research Letters*, vol.6, pp.107–114, 1987.

[72] A. Hertz, É.D. Taillard and D. de Werra, "Tabu search," in: E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp.121–136, 1997.

[73] J.H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, The University of Michigan Press, 1975, and MIT Press, 1992.

[74] J.N. Hooker and C. Fedjki, "Branch-and-cut solution of inference problems in propositional logic," *Annals of Mathematics and Artificial Intelligence*, vol.1, pp.123–139, 1990.

[75] 茨木俊秀，"組合せ最適化の手法 — 巡回セールスマン問題の例から —，" 電学論 C，vol.114, pp.411–419, 1994.

[76] 茨木俊秀，"組合せ最適化とスケジューリング問題: 新解法とその動向，" 計測と制御，vol.34, pp.340–346, 1995.

[77] T. Ibaraki, "Combination with other optimization methods," in: T. Bäck, D.B. Fogel and Z. Michalewicz, eds., *Handbook of Evolutionary Computation*, IOP Publishing Ltd and Oxford University Press, 1997.

[78] T. Ibaraki and Y. Nakamura, "A dynamic programming method for single machine scheduling," *European J. Operational Research*, vol.76, pp.72–82, 1994.

[79] P. Jog, J.Y. Suh and D. van Gucht, "The effects of population size, heuristic crossover and local improvement on a genetic algorithm for the traveling salesman problem," *Proc. 3rd International Conference on Genetic Algorithms*, pp.110–115, 1989.

[80] P. Jog, J.Y. Suh and D. van Gucht, "Parallel genetic algorithms applied to the traveling salesman problem," *SIAM J. Optimization*, vol.1, pp.515–529, 1991.

[81] D.S. Johnson, "Approximation algorithms for combinatorial problems," *J. Computer and System Sciences*, vol.9, pp.256–278, 1974.

[82] D.S. Johnson, "Local optimization and the traveling salesman problem," *Proc. 17th Colloquium on Automata, Languages and Programming*, pp.446–461, 1990.

[83] D.S. Johnson, C.R. Aragon, L.A. McGeoch and C. Schevon, "Optimization by simulated annealing: an experimental evaluation; part I, graph partitioning," *Operations Research*, vol.37, pp.865–892, 1989; "part II, graph coloring and number partitioning," ditto, vol.39, pp.378–406, 1991.

[84] D.S. Johnson and L.A. McGeoch, "The traveling salesman problem: a case study," in: E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp.215–310, 1997.

[85] D.S. Johnson, C.H. Papadimitriou and M. Yannakakis, "How easy is local search?," *J. Computer and System Sciences*, vol.37, pp.79–100, 1988.

[86] T. Jones and S. Forrest, "Fitness distance correla-

tion as a measure of problem difficulty for GAs," *Proc. 6th International Conference on Genetic Algorithms*, pp.110–115, 1995.

[87] B.W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical J.*, vol.49, pp.291–307, 1970.

[88] P. Kilby, P. Prosser and P. Shaw, "Guided local search for the vehicle routing problem with time windows," in: S. Voß, S. Martello, I.H. Osman and C. Roucairol, eds., *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, pp.473–486, 1999.

[89] S. Kirkpatrick, C.D. Gelatt, Jr. and M.P. Vecchi, "Optimization by simulated annealing," *Science*, vol.220, pp.671–680, 1983.

[90] 木瀬洋，"スケジューリング問題に対するシミュレーティドアニーリング法，" オペレーションズ・リサーチ，vol.40, pp.268–273, 1995.

[91] 北野宏明（編），遺伝的アルゴリズム，産業図書，1993; 第 2 巻，同上，1995; 第 3 巻，同上，1997．

[92] A. Kolen and E. Pesch, "Genetic local search in combinatorial optimization," *Discrete Applied Mathematics*, vol.48, pp.273–284, 1994.

[93] 久保幹雄，"メタヒューリスティックス，" in: 室田一雄編，離散構造とアルゴリズム IV，近代科学社，pp.171–230，1995．

[94] M. Laguna, T.A. Feo and H.C. Elrod, "A greedy randomized adaptive search procedure for the two-partition problem," *Operations Research*, vol.42, pp.677–687, 1994.

[95] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys, eds., *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley & Sons, 1985.

[96] Y. Li, P.M. Pardalos and M.G.C. Resende, "A greedy randomized adaptive search procedure for the quadratic assignment problem," *DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, vol.16, pp.237–261, 1994.

[97] C.K.Y. Lin, K.B. Haley and C. Sparks, "A comparative study of both standard and adaptive versions of threshold accepting and simulated annealing algorithms in three scheduling problems," *European J. Operational Research*, vol.83, pp.330–346, 1995.

[98] S. Lin, "Computer solutions of the traveling salesman problem," *Bell System Technical J.*, vol.44, pp.2245–2269, 1965.

[99] S. Lin and B.W. Kernighan, "An effective heuristic algorithm for the traveling salesman problem," *Operations Research*, vol.21, pp.498–516, 1973.

[100] A. Løkketangen and F. Glover, "Solving zero-one mixed integer programming problems using tabu search," *European J. Operational Research*, vol.106, pp.624–658, 1998.

[101] H.R. Lourenço, "Job-shop scheduling: computational study of local search and large-step optimization methods," *European J. Operational Research*, vol.83, pp.347–364, 1995.

[102] H.R. Lourenço and D. Serra, "Adaptive approach heuristics for the generalized assignment problem," Technical Report: Department of Economics and Management, Univerisitat Pompeu Fabra, R. Trias Fargas 25-27, 08005 Barcelona, Spain, 1998 (available at `http://www.econ.upf.es/ deehome/ what/ wpapers/ listwork.html`).

[103] M. Lundy and A. Mees, "Convergence of an annealing algorithm," *Mathematical Programming*, vol.34, pp.111–124, 1986.

[104] B. Manderick, M. de Weger and P. Spiessens, "The genetic algorithm and the structure of the fitness landscape," *Proc. 4th International Conference on Genetic Algorithms*, pp.143–150, 1991.

[105] O.C. Martin and S.W. Otto, "Combining simulated annealing with local search heuristic," *Annals of Operations Research*, vol.63, pp.57–75, 1996.

[106] O. Martin, S.W. Otto and E.W. Felten, "Large-step Markov chains for the TSP incorporating local search heuristic," *Operations Research Letters*, vol.11, pp.219–224, 1992.

[107] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 1992.

[108] N. Mladenović and P. Hansen, "Variable neighborhood search," *Computers and Operations Research*, vol.24, pp.1097–1100, 1997.

[109] H. Mühlenbein, "Parallel genetic algorithms, population genetics and combinatorial optimization," *Proc. 3rd International Conference on Genetic Algorithms*, pp.416–421, 1989.

[110] H. Mühlenbein, M. Gorges-Schleuter and O. Krämer, "Evolution algorithms in combinatorial optimization," *Parallel Computing*, vol.7, pp.65–85, 1988.

[111] H. Mühlenbein, "Genetic algorithms," in: E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp.137–171, 1997.

[112] K. Nonobe and T. Ibaraki, "A tabu search approach to the constraint satisfaction problem as a general problem solver," *European J. Operational Research*, vol.106, pp.599–623, 1998.

[113] K. Nonobe and T. Ibaraki, "Formulation and tabu search algorithm for the resource constrained project scheduling problem (RCPSP)," Technical Report #99010, Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, 1999.

[114] 岡田正浩，田地宏一，高橋豊，"巡回セールスマン問題に対する λ-opt の確率的多項式性，" NAIST-IS-TR98002，奈良先端科学技術大学院大学情報科学研究科，1998．

[115] I.M. Oliver, D.J. Smith and J.R.C. Holland, "A study of permutation crossover operators on the trav-

eling salesman problem," *Proc. 2nd International Conference on Genetic Algorithms*, pp.224–230, 1987.

[116] I.H. Osman and J.H. Kelly, "Meta-heuristics: an overview," in: I.H. Osman and J.P. Kelly, eds., *Meta-Heuristics: Theory and Applications*, Kluwer Academic Publishers, pp.1–21, 1996.

[117] I.H. Osman and J.P. Kelly, eds., *Meta-Heuristics: Theory and Applications*, Kluwer Academic Publishers, 1996.

[118] I.H. Osman and G. Laporte, "Metaheuristics: a bibliography," in: G. Laporte and I.H. Osman, eds., *Metaheuristics in Combinatorial Optimization*, Annals of Operations Research 63, Baltzer, pp.513–623, 1996.

[119] C.H. Papadimitriou, "On selecting a satisfying truth assignment (extended abstract)," *Proc. 32nd Annual Symposium on Foundations of Computer Science*, pp.163–169, 1991.

[120] C.H. Papadimitriou and K. Steiglitz, "On the complexity of local search for the traveling salesman problem," *SIAM J. Computing*, vol.6, pp.76-83, 1977.

[121] C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, 1982; re-issued by Dover Publications, 1998.

[122] E. Pesch and F. Glover, "TSP ejection chains," *Discrete Applied Mathematics*, vol.76, pp.165-181, 1997.

[123] E. Pesch and S. Voß, "Strategies with memories: local search in an application oriented environment," *OR Spektrum*, vol.17, pp.55–66, 1995.

[124] M. Pirlot, "General local search heuristics in combinatorial optimization: a tutorial," *Belgian J. Operations Research, Statistics and Computer Science*, vol.32, pp.7-67, 1992.

[125] M. Pirlot, "General local search methods," *European J. Operational Research*, vol.92, pp.493–511, 1996.

[126] C.N. Potts and L.N. van Wassenhove, "A decomposition algorithm for the single machine total tardiness problem," *Operations Research Letters*, vol.1, pp.177–181, 1982.

[127] V.J. Rayward-Smith, I.H. Osman, C.R. Reeves and G.D. Smith, eds., *Modern Heuristic Search Methods*, John Wiley & Sons, 1996.

[128] C.R. Reeves, ed., *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publications, 1993; re-issued by McGraw-Hill, 1995; (横山隆一，奈良宏一，佐藤晴夫，鈴木昭男，荻本和彦，陳洛南(訳)，モダンヒューリスティックス — 組合せ最適化の最先端手法，日刊工業新聞社，1997).

[129] C.R. Reeves, "Genetic algorithms for the operations researcher," *INFORMS J. Computing*, vol.9, pp.231–250, 1997.

[130] C.R. Reeves, "Landscapes, operators and heuristic search," *Annals of Operations Research*, to appear.

[131] S. Reiter and D.B. Rice, "Discrete optimizing solution procedures for linear and nonlinear integer programming problems," *Management Science*, vol.12, pp.829–850, 1966.

[132] S. Reiter and G. Sherman, "Discrete optimizing," *J. Society for Industrial and Applied Mathematics*, vol.13, pp.864–889, 1965.

[133] M.G.C. Resende and T.A. Feo, "A GRASP for satisfiability," *DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, vol.26, pp.499–520, 1996.

[134] M.G.C. Resende, L.S. Pitsoulis and P.M. Pardalos, "Approximate solution of weighted MAX-SAT problems using GRASP," *DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, vol.35, pp.393–405, 1997.

[135] S. Sahni and T. Gonzalez, "P-complete approximation problems," *J. Association for Computing Machinery*, vol.23, pp.555–565.

[136] G.H. Sasaki and B. Hajek, "The time complexity of maximum matching by simulated annealing," *J. Association for Computing Machinery*, vol.35, pp.387–403, 1988.

[137] L.M. Schmitt, C.L. Nehaniv and R.H. Fujii, "Linear analysis of genetic algorithms," *Theoretical Computer Science*, vol.200, pp.101–134, 1998.

[138] B. Selman and H.A. Kautz, "Domain-independent extensions to GSAT: solving large structured satisfiability problems," *Proc. 13th International Joint Conference on Artificial Intelligence*, pp.290–295, 1993.

[139] B. Selman, H.A. Kautz and B. Cohen, "Noise strategies for improving local search," *Proc. 12th National Conference on Artificial Intelligence*, pp.337–343, 1994.

[140] M. Sinclair, "Comparison of the performance of modern heuristics for combinatorial optimization on real data," *Computers and Operations Research*, vol.20, pp.687–695, 1993.

[141] T. Stützle and H. Hoos, "$\mathcal{MAX}$-$\mathcal{MIN}$ ant system and local search for the traveling salesman problem," *Proc. 1997 IEEE International Conference on Evolutionary Computation*, 1997.

[142] J.Y. Suh and D. van Gucht, "Incorporating heuristic information into genetic search," *Proc. 2nd International Conference on Genetic Algorithms*, pp.100–107, 1987.

[143] É.D. Taillard, P. Badeau, M. Gendreau, F. Guertin and J.-Y. Potvin, "A tabu search heuristic for the vehicle routing problem with soft time windows," *Transportation Science*, vol.31, pp.170–186, 1997.

[144] 玉木久夫，"巡回セールスマン問題の近似アルゴリズム: 天オアローラによる 20 年ぶりの急進展," 情報処理，vol.39,

pp.566-573, 1998.

[145] 玉置久，喜多一，岩本貴司，三宮信夫，“遺伝アルゴリズム — I — GA の基礎,” システム制御情報学会誌，vol.39, pp.295–302, 1995; 玉置久，喜多一，“II — GA による最適化計算 1,” 同上，vol.39, pp.400–407, 1995; 玉置久，喜多一，“III — GA による最適化計算 2,” 同上，vol.39, pp.431–438, 1995; 岩本貴司，“IV — GA の理論,” 同上，vol.39, pp.507–514, 1995; 玉置久，喜多一，岩本貴司，“V — GA の拡張,” 同上，vol.40, pp.69–76, 1996; 玉置久，喜多一，岩本貴司，“VI — 進化型計算の動向,” 同上，vol.40, pp.170–177, 1996.

[146] E. Tsang and C. Voudouris, “Fast local search and guided local search and their application to British Telecom’s workforce scheduling problem,” *Operations Research Letters*, vol.20, pp.119-127, 1997.

[147] N.L.J. Ulder, E.H.L. Aarts, H.-J. Bandelt, P.J.M. van Laarhoven and E. Pesch, “Genetic local search algorithms for the traveling salesman problem,” *Proc. 1st International Workshop on Parallel Problem Solving from Nature*, pp.109–116, 1990.

[148] S. Voß, S. Martello, I.H. Osman and C. Roucairol, eds., *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, 1999.

[149] C. Voudouris and E. Tsang, “Guided local search,” Technical Report CSM-247, Department of Computer Science, University of Essex, 1995.

[150] C. Voudouris and E. Tsang, “Guided local search and its application to the traveling salesman problem,” *European J. Operational Research*, vol.113, pp.469–499, 1999.

[151] J.P. Walser, *Domain-independent local search for linear integer optimization*, Doctoral dissertation, Technischen Fakultät, Universität des Saarlandes, 66041 Saarbrücken, Germany, 1998 (available at `http://www.ps.uni-sb.de/~walser/`).

[152] M. Yagiura and T. Ibaraki, “Genetic and local search algorithms as robust and simple optimization tools,” in: I.H. Osman and J.P. Kelly, eds., *Meta-Heuristics: Theory and Applications*, Kluwer Academic Publishers, pp.63–82, 1996.

[153] M. Yagiura and T. Ibaraki, “Metaheuristics as robust and simple optimization tools,” *Proc. IEEE International Conference on Evolutionary Computation*, pp.541–546, 1996.

[154] M. Yagiura and T. Ibaraki, “The use of dynamic programming in genetic algorithms for permutation problems,” *European J. Operational Research*, vol.92, pp.387–401, 1996.

[155] 柳浦睦憲，茨木俊秀，“メタ戦略のロバスト性について,” 第 8 回 RAMP シンポジウム論文集，pp.109–124, 1996.

[156] M. Yagiura and T. Ibaraki, “Analyses on the 2 and 3-flip neighborhoods for the MAX SAT,” *J. Combinatorial Optimization*, to appear.

[157] M. Yagiura, T. Ibaraki and F. Glover, “An ejection chain approach for the generalized assignment problem,” Technical Report #99013, Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, 1999.

[158] M. Yagiura and T. Ibaraki, “Local search,” Prepared for a chapter in: P.M. Pardalos and M.G.C. Resende, eds., *Handbook of Applied Optimization*, Oxford University Press.

[159] M. Yagiura, T. Yamaguchi and T. Ibaraki, “A variable depth search algorithm for the generalized assignment problem,” in: S. Voß, S. Martello, I.H. Osman and C. Roucairol, eds., *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, pp.459–471, 1999.

[160] M. Yagiura, T. Yamaguchi and T. Ibaraki, “A variable depth search algorithm with branching search for the generalized assignment problem,” *Optimization Methods and Software*, vol.10, pp.419–441, 1998.

[161] 山田武士，C.R. Reeves, “フローショップスケジューリング問題の地形解析と遺伝的局所探索法による解法,” 情報処理学会論文誌，vol.39, pp.2112–2123, 1998.

[162] M. Yannakakis, “On the approximation of maximum satisfiability,” *J. Algorithms*, vol.17, pp.475–502, 1994.

[163] M. Yannakakis, “Computational complexity,” in: E.H.L. Aarts and J.K. Lenstra, eds., *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp.19–55, 1997.

（平成 11 年 4 月 1 日受付，8 月 31 日再受付）

柳浦　睦憲　（正員）

平 3 京大・工・数理卒．平 5 同大学院修士課程了．平 6 同大学院博士課程中退．現在同助手．組合せ最適化問題に対する近似解法の研究に従事．博士（工学）．

茨木　俊秀　（正員）

昭 38 京大・工・電気卒．昭 40 同大学院修士課程了．昭 44 京大助手，昭 48 同助教授．昭 58 豊橋技術科学大学教授，昭 60 京大・工・数理工学科教授．Illinois 大学，Waterloo 大学，Simon Fraser 大学，Rutgers 大学，Erasmus 大学等客員．組合せ最適化，アルゴリズム，計算の複雑さ等の研究に従事．著書「アルゴリズムとデータ構造」，「最適化の手法」，「Resource Allocation Problems」など．工学博士．