

## 大規模 TSP を対話的応答時間内で最適化するための多段知能型 GA

櫻井 義尚<sup>†</sup> 小野山 隆<sup>††</sup> 久保田 仙<sup>††</sup> 中村 嘉宏<sup>†</sup> 鶴田 節夫<sup>†</sup>

提案する多段知能型 GA は、大規模巡回セールスマン問題 (TSP) を対話リアルタイム応答時間内に専門家レベルの最適化で解くことを目的とする。GA において最適化問題を効率的に解くためには問題固有の特性に従った探索をする必要があるが、これにより偏った探索になる可能性がある。提案する GA は、TSP に有効なヒューリスティクスあるいは知識を導入した新しい初期解生成、交叉、突然変異を用いることにより、効率的な探索を実現し、探索特性の異なる相補的性質を持つ GA を複数設計し、これらを組み合わせて用いることにより、探索の偏りを防ぎ、安定的に解を求める事ができる。

### A Multi-step Intelligent Genetic Algorithm to Interactively Optimize Large-scale TSP

Yoshitaka Sakurai<sup>†</sup> Takashi Onoyama<sup>††</sup> Sen Kubota<sup>††</sup>  
Yoshihiro Nakamura<sup>†</sup> Setsuo Tsuruta<sup>†</sup>

To optimize large-scale distribution networks, solving about 1000 middle scale (around 40 cities) TSPs (Traveling Salesman Problems) within an interactive length of time (max. scores of seconds) is required. Yet, expert-level (less than 3% of errors) accuracy is necessary. To realize the above requirements, a Multi-step intelligent GA method was developed. This method combines a high-speed GA with an intelligent GA holding problem-oriented knowledge that is effective for some special location patterns. If conventional methods were applied, solutions for more than 20 out of 20,000 cases were below expert-level accuracy. However, the developed method could solve all of 20,000 cases at expert-level.

## 1. はじめに

物流の効率化は経済面からだけではなく、環境・社会面から見ても CO<sub>2</sub> や排気ガスの節減などにもつながるため重要な問題であり、多数の配送先と多数のトラックにより構成されるより大規模な配送ネットワークの最適化の要求が高まっている。

このような配送ネットワークの構築では、まず製品の製造を行う工場や、輸送の中継に用いるデポの配置、部品輸送用のトラック配置等の条件設定を人間が行う。次に輸送コストと CO<sub>2</sub> 等の排出量を算出して、経済面及び環境面から人間が総合的に評価する。これらの算出のためには、千近くの商品メーカーとデポ・工場を結ぶ数百の集配ルートを作成して走行距離を求める必要がある。このルートの最適性の評価には単純な走行コストだけでなく、社会面からの人間の判断が不可欠になる。このため、システムとしては、対話的な応答性能が要求される。

つまり、大規模な配送ネットワークにおける数百から千に及ぶ配送ルートを数十秒で作成しなければならない。1ルートあたりではルート生成を数十ミリ秒で行う必要がある。それぞれのルートの生成は数十都市の TSP(巡回セールスマン問題, Traveling Salesman Problem) を解くのと等価である。ゆえに TSP に対して対話的な応答性と高い最適性の両方を可能とする近似解法が必要である。

人間の専門家は、数学的な最適解に対して 2~3% の誤差を出すことも多いが、それ以上の誤差を出し実用上の問題を起こすことはない。一方、従来の近似解法 [1] を用いると、数学的な最適解を得られる確率は高いが、誤差を常に 2~3% 程度以下に出来る保証がない。このため、従来の近似解法は実用上、特に上記の問題への適用には適していない。TSP の厳密解法としては、Branch-and-cut 法 [2] や Dynamic Programming (DP) を用いた解法 [3] が提案されているが、精度は高いが計算時間が長い。近似解法としては、Simulated Annealing (SA) 法やタブーサーチを用いる方式 [4], [5] が提案されているが、計算時間が長く、対話的な応答性能が要求される問題には適していない。TSP の解法として

<sup>†</sup> 東京電機大学

Tokyo Denki University

<sup>††</sup> (株)日立ソフトウェアエンジニアリング

Hitachi Software Engineering Co., Ltd.

Lin-Kernighan(LK)法やその改良手法[6]が提案されているが、専門家レベルの最適度を常に保証することができない[7].

本研究では、TSP に有効なヒューリスティクスあるいは知識を導入した新しい初期解生成、交叉、突然変異を導入した複数の遺伝的アルゴリズム (Genetic Algorithm, GA) [8]を組み合わせることにより、多様な巡回拠点配置パターンにおいて、対話的応答時間内に専門家レベルの最適で解くことを保証する近似解法を提案し、実験によりその有効性を検証する。

## 2. 多段知能型 GA (MIGA)

### 2.1. 多段知能型 GA

相補的な性質を持つ2つのGAを組み合わせることにより様々な巡回拠点配置パターンに対して、専門家レベルの最適性と対話的応答性を実現する多段知能型 GA (Multi-step Intelligent GA; MIGA)を提案する。

GA で効率的な最適化を行うためには、問題の特徴に合ったGAのオペレータを利用する必要がある。そこで、TSP に有効なヒューリスティクスをGAのオペレータに組み込む。この手法は、主にシンプルで汎用性の高いヒューリスティクスを用いているため高速に最適化が行える2opt型GAと計算コストは高いが最適化能力と局所最適解からの脱出効果の高いブロック型GAの2つの異なる探索特性を持つGAを組み合わせる事により、多様な巡回拠点配置パターンで局所最適解に陥るのを回避しつつ、高い応答性を保証する。具体的には、2つのGAを独立に試行し、良い結果を採用する。

**2opt型GA:**ランダムNI生成により初期集団を生成し、NI型交叉、2opt型突然変異により解の改善を行う。シンプルで汎用性の高いヒューリスティクスである2opt法を用いることにより、素早くツアーを改善する。しかし、局所最適解に陥るリスクも持っている。

**ブロック型GA:**ランダム生成、挿入順制御NI生成を使うことにより高い最適度の初期集団を生成し、NI型交叉により解の改善を行う。また、ブロック型突然変異によりツアーの局所的に最適化された部分を再構築し、局所最適解に陥ることを防ぐ。計算コストは高いが高い最適化が望めるヒューリスティクスであるNI(Nearest Insertion)法を主に用い

ている。これにより2opt型GAとは違う探索特性を持つ。

### 2.2. 多段知能型 GA の構成要素

本解法では、解の候補となる個体はTSPにおけるツアー(配送ルート)を表現する染色体を持ち、その遺伝子型は巡回する順番にノード番号(配送先のID番号)を並べた構造(各遺伝子はノード番号を表現する)になっている。各個体の評価(適応度)は、その世代の個体集団の中での最小のツアー長を $L_{Min}$ とするとき、ツアー長が $L$ の個体の適応度を $F=L_{Min}/L$ で計算する。この適応度に基づき、全個体(親個体集団と子個体集団)の中から適応度の高い順に次の世代への個体の選択を行う。また、たとえ適応度が低くても改善の機会を与えるため個体数の10%はランダムに決定される。

#### (1) 初期集団生成:

局所最適解に収束するのを防ぎ、最適度の高い解を得るためには、初期集団の多様性が重要になる。しかし迅速に高い最適度を得るためには、初期集団の生成段階からある程度の最適化が必要になる。このため以下のように最適化の強さの違う3種類の方式を用いる。

##### a) ランダム生成

ノードの順序をランダムに並べることによりツアーを生成する。

##### b) ランダムNI生成

ランダムな順序でノードをNI法を用いてツアーに挿入していくことによりツアーを生成する。

##### c) 挿入順制御NI生成

専門家がツアーを生成するとき、彼らはたいてい全体のルートの概観を考えながら近いものから順に巡回する順番を決めていく。このような専門家の全体指向的な決め方をモデル化することにより、挿入順制御NI生成法は提案された。具体的には、以下のような手順でツアーを生成する。

- ① NI法でノードをツアーに挿入する場合の、ツアーの距離の増分値を求める。
- ② ①で求めた値と、ノードを挿入するリンクの長さに一定倍率を掛けた値を比較する。
- ③ ①で求めた値が小さい場合にだけNI法でノードをツアーに挿入する。
- ④ ①から③の操作を、③の条件を満足するノードがなくなるまで繰り返す。
- ⑤ ツアーに挿入されていないノードがあれば、先の

倍率(ステップ②の倍率)を大きくして、再度①から④の操作を繰り返す。

**(2) 交叉オペレータ(NI型交叉):**

交叉で二つの親の良い形質を子に継承してGAで迅速な解の収束を実現するために、NI法を用いた交叉オペレータNI型交叉を提案する。これは図1のような手順で、親1の交叉点より前の部分ツアーにこれを除いた親2のノードをNIにより挿入することにより新しい個体を生成する。

この交叉オペレータは、二つの親個体に含まれているノードの順序を子個体に継承でき、GAの収束性を高めることができると考えられる。

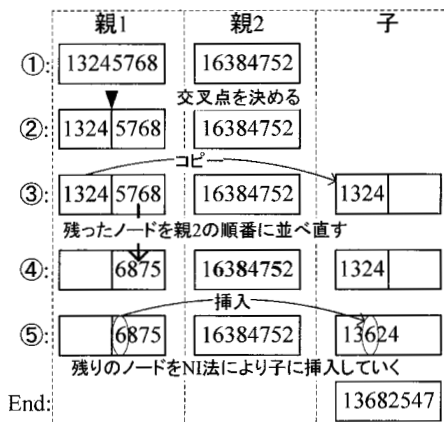


図 1. シミュレーションの過程

**(3) 突然変異オペレータ:**

**a) 2opt型突然変異**

これはランダムに選択した遺伝子を局所探索法である 2opt 法で改良する方式である。まず、遺伝子が表すツアー中の二つのリンクを取り出して、その二つのリンクを交換してツアー長を評価する。もしリンクの交換でツアー長が短くなれば、二つのリンクを交換する。このリンクの交換をツアー長の改善がなくなるまで繰り返す。

**b) ブロック型突然変異**

2opt 型突然変異は効率的にツアーを改良でき短時間に良好な解を得ることができる。しかし、このような効率的な探索は逆に局所最適解に陥る危険性を高めてしまう。より最適性の高い解を得るためには局所最適解から脱出するためにツアーの一部を破壊する操作が必要である。このため、次のブロック突然変異を提案する。これは、まずツアー中の1つのノードをランダムに選択して、そのノードの近

傍に含まれるノードをツアーから削除する。この近傍のサイズもランダムに選択する。次に、これらのツアーから削除したノードを再び NI 法で再挿入してツアーを再構築する。

**3. 評価実験**

**3.1. 評価実験の設定**

本実験では、Intel Pentium II(450MHz)プロセッサと256MBのメモリを装備しているPCを実験に使用し、C言語によりプログラムを作成した。

ターゲットとしている配送ネットワークでは、一台のトラックでは一日に40拠点以上巡回することは不可能なので、40都市のTSPで各解法の評価を行う。現実には様々な拠点の配置が発生するために、40都市をランダムに配置した20,000個の異なる巡回拠点配置パターンを用意した。そして、各解法の効果の評価するために、それぞれの解法で100回、テストパターンを解き、数10ミリ(30ミリ)秒以内での最適解との誤差が3%以内の解の出現頻度を調べた。

GAのパラメータは、集団個体数を100とし、交叉率と突然変異率はそれぞれ10%に設定した。ランダム生成と挿入順制御NI生成の混合では、両生成法による初期集団の50%ずつを混ぜ合わせて用いた。また、挿入順制御NI生成の閾値wには4段階の倍率(0.5,1.0,1.5,制限なし)を用いた。これらのパラメータは幾つかのパラメータセットによる予備実験の結果に基づいて決定した。

表 1. 30ミリ秒以内に実行可能な世代数

#	手法	初期化	突然変異	世代数
1	2opt型	ランダムNI	2opt型	24
2	ブロック型	ランダム + 挿入順制御NI	ブロック型	20

表 2. 誤差率3%以内の解の割合

#	手法	最適解	誤差率3%以内
1	2opt型GA	84.45%	99.885%
2	ブロック型GA	83.75%	99.785%
3	多段知能型GA	92.05%	100.0%

**3.2. 実験結果**

表1に30ミリ秒以内で計算可能な各GA方式の世代数を、表2に各GA方式における最適解の獲得率、専門家レベルの誤差率3%以内の解の獲得率を

示す。これを見ると、40 都市 TSP において 30 ミリ秒以内に誤差率 3% 以内の解を 100% 獲得できたのは多段知能型 GA だけである。これにより提案手法多段知能型 GA の有効性が示された。

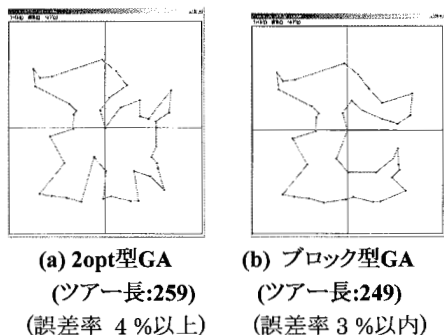


図 2. 特定の拠点配置パターンにおけるツアー形状の違い

### 3.3. 評価

#### (1) ブロック型 GA の効果:

2opt 型 GA による実行結果から誤差率 3% 以上の解を解析したところ、図 2(a) のようなギアホイール型のツアー形状をした解が多数見られた。人間の専門家は、大抵、図 2(b) のような、より直線状のツアー形状を生成する。もし専門家が図 2(a) のような非効率的なツアーを見つけたら、彼らはそのシステムを信頼でないものとして使うことをやめるだろう。

ブロック型 GA を使った場合は、そのような巡回拠点配置パターンにおいても図 2(b) のような解を得ることができるであろう。なぜなら、このような局所的な不効率なルートはブロック型突然変異により改善される可能性が高いからである。事実このような形状のツアーはブロック型 GA による結果では見られなかった。

#### (2) 多段知能型 GA (MIGA) の効果:

実験の結果によると、2opt 型 GA の場合、20000 テストケースの内 43 ケースで解の誤差率が 3% 以上であった。ブロック型 GA の場合は、23 ケースで解の誤差率が 3% 以上であった。しかしながら、多段知能型 GA では 20000 テストケースの全てで誤差率が 3% 以内であった。これは、多段知能型 GA が 2opt 型 GA とブロック型 GA というタイプの違い、相補的特性を持った GA の良い方の解を採用することにより、幅広い巡回拠点配置パターン

に対応できるためである。これにより、技術目標であった対話的応答性(各ルート数 10 ミリ秒以内、千ルートで数 10 秒)と専門家レベルの最適性(誤差率 3% 以内)を実現できた。

### 4. おわりに

本研究では、様々な巡回拠点配置パターンで専門家レベルの最適性と対話的応答性を実現するため、特性の違う 2 つの GA、2opt 型 GA とブロック型 GA を組み合わせて用いる多段知能型 GA (MIGA) を提案した。

2opt 型 GA は迅速に収束することにより、短時間で高い最適性の解を得ることができたが、ある特定の巡回拠点配置パターンでは、専門家レベルの最適性を保証できなかった。しかし、ブロック型 GA により計算コストは多少増えたが 2opt 型 GA の弱点パターンを克服することができた。この 2 つを組みあせて用いる多段知能型 GA により、大規模な配送ネットワークの対話的シミュレーションに必要な最適性と応答性で TSP を解くことができることが示された。

#### <参考文献>

- [1] 山本, 久保: 巡回セールスマン問題への招待, 朝倉書店(1997).
- [2] Grotschel, M.; and Holland, O., "Solution of large-scale symmetric traveling salesman problems", *Mathematical Programming*, Vol.51, pp.141-202, 1991.
- [3] Arora, S., "Polynomial Time Approximation Schemes for Euclidean TSP and Other Geometric Problems", *Journal of the ACM*, Vol. 45, No.5, pp.753-782, 1998.
- [4] 茨城俊秀. 離散最適化とアルゴリズム, 岩波書店 (1993)
- [5] Hooker, J.H.; and Nataraj, N.R. 1995. Solving a General Routing and Scheduling Problem by Chain Decomposition and Tabu Search, *Transportation Science*, Vol.29, No.1, pp.30-44.
- [6] S.Lin, and B.W.Kernighan, An effective heuristic algorithm for the traveling salesman problem, *Operations Research*, Vol. 21, No.2, Feb.1972, pp.498-516.
- [7] Kubota S.; Onoyama T.; Onayagi K. and Tsuruta S., "Traveling Salesman Problem Solving Method fit for Interactive Repetitive Simulation of Large-scale Distribution Network", *Proc. IEEE SMC'99*, pp. 533-538, 1999.
- [8] L.Davis. editor. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, NY, 1991.