

巡回セールスマン問題の局所解構造解析

谷 昌史・片山 謙吾*・成久 洋之*

岡山理科大学大学院工学研究科修士課程情報工学専攻

*岡山理科大学工学部情報工学

(1999年11月4日 受理)

1. まえがき

組合せ最適化問題の代表的なものとして巡回セールスマン問題 (Traveling Salesman Problem, TSP) がある. この TSP の近似解を比較的短時間に求める解法が数多く提案されている. その中に Genetic Algorithm, Simulated Annealing, Ant System, Tabu Search などがあり, これらの解法の多くは, 局所探索法 (Local Search, LS) に対して, 自然界のふるまいを巧みに導入することで実現される. この LS を用いて得られた局所解を複数個利用する解法として, 遺伝的局所探索法 (Genetic LS, GLS) などがある. GLS は, 有限個の局所解の集合に対して遺伝的操作 (選択, 交叉, 突然変異) を適用し, より優れた局所解へ探索を進めることができる. 特に, Boese ら¹⁾の観測結果を利用した GLS は, 極めて良好な近似解を算出可能であることが知られている.

Boese らは TSP において近傍構造, 及び近傍探索オペレーターを用いることにより「Big Valley Structure」が発現することを明らかにした. この構造を巧妙に利用したものとして Merz ら⁵⁾, Katayama ら⁴⁾の解法がある. また, Wolpert ら²⁾の「一定条件のもとではどんなアルゴリズムであっても, あらゆる目的関数について平均をとればアルゴリズムの性能はどれもみな等しい」とする主張を認めると, 一般にどんな問題に対しても優れた解法というのは存在しえない. すなわち優れた解法とは, 対象となる問題の性質を的確に活用したものといえる. こうした観点から本研究では, TSP に対する優れた解法を構築するには, 問題の性質, または探索空間の構造上の特性を調査する必要があると考え, 次の三つの解法で得られた解と最適解の間に存在する関係について解析する.

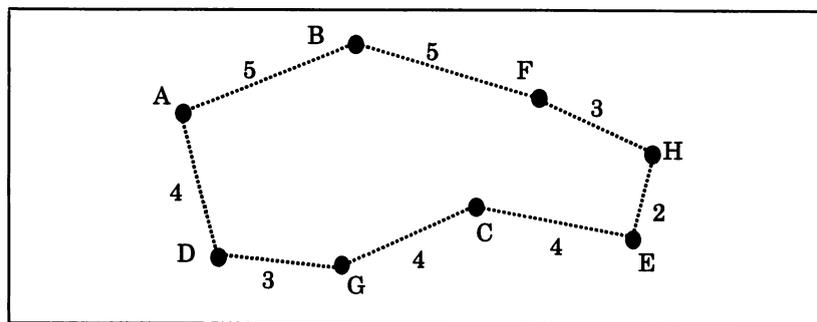
- a) ランダムに生成された解に 2-Opt 法を実施して算出した局所解.
- b) Nearest Neighbor 法で算出した解.
- c) Nearest Neighbor 法で算出された解に対して, さらに 2-Opt 法を実施して算出した局所解.

2. TSP の概要

TSP とは, n 個の点 (都市) $\{c_1, c_2, \dots, c_n\}$, および各二都市間の長さ $d(c_i, c_j)$ が与えられた時, すべての点をただ一度経由する巡回路 π (ハミルトン閉路) において, 次式を最小化する問題である.

$$f(\pi) = \sum_{i=1}^{n-1} d(c_{\pi(i)}, c_{\pi(i+1)}) + d(c_{\pi(n)}, c_{\pi(1)})$$

本研究で扱う TSP は, $1 \leq i, j \leq n$ において, 二点 c_i, c_j 間の双方向の長さが等しい ($d(c_i, c_j) = d(c_j, c_i)$) 対称巡回セールスマン問題であり, 実行可能解の数は $(n-1)!/2$ になる. 図 1 に TSP の簡単な問題例を示す. 図 1 の節点 A~H および閉路は, それぞれ都市およびある巡回路を表す. また, 枝の重みは都市間の距離を表す. なお, 図に示した巡回路を形成するもの以外の枝は省略している.

図1 8都市 ($n=8$) の TSP の例

3. 大谷構造 (Big Valley Structure)

大谷構造とは、ある目的関数を最小にする問題を考えるとき、対象となる問題に応じた近傍構造、および近傍探索オペレーターを用いることにより、地形が大局的に見ると真の最適解である極小値に向かって1つの大きな谷を持つような構造していることである。図2に「Big Valley Structure」を示す。

地形とは、探索空間の構造の複雑さを表す尺度である。直感的には $f(x)$ の最適化における $y = f(x)$ のグラフの様子と考えることが出来る。たとえばグラフがすり鉢状で唯一の極値を持つとき f の地形は単純で、簡単な降下法によって容易に最適解が求まる。このような観点から地形 Λ は探索空間 S と目的関数 f の間の関係と定義できる。しかし実際には特に組合せ問題の場合、問題の複雑さ、難解さが、解法に用いられる探索オペレータにも依存するため、 S と f だけでは不十分である。そこで Reeves ら6)は地形 Λ を探索空間 S 、目的関数 f 、そして探索空間上に定義された距離 d の組として $\Lambda = (S, f, d)$ と表現した。本研究では、TSP を題材とした探索空間とする。

もし探索空間の地形 Λ が大谷構造を持っていれば、局所解は集団となって存在する。したがって、1つ優秀な解が発見されれば、さらに優秀な解が、その解の比較的近くに存在すると考えられる。つまり、対象となる問題が大谷構造の性質を持つならば、現在までに得られている最良の解をもとに、新しい解を構築することが有効であると考えられる。



図2 Big Valley Structure (大谷構造)

4. 局所探索法

実行可能領域に含まれる解の集合 F を与えたとき、その近傍 N は以下の写像と定義される3)。

$$N: F \rightarrow 2^F$$

$x \in F$ で $f(x) \leq f(y), \forall y \in N(x)$ を満足する x を局所解と呼ぶ。局所探索法の戦略は、現在の解 $x \in F$ に対してその近傍 $N(x)$ から選ばれる解 $x' \in N(x)$ を生成し、その近傍が現在の解よりよい解 $f(x') < f(x)$ であれば、その解に改善 ($x = x'$) する処理を繰り返すものである。最終的に得られる解は局所解になる。

TSP における有名な局所探索法として 2-Opt 法がある。図3に 2-Opt 近傍を示す。これは二本の枝を他の二本の枝と置き換えるものである。

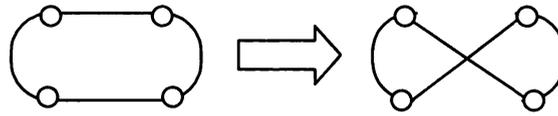


図3 2-Opt 近傍の一例

5. TSP における Nearest Neighbor 法

TSP における Nearest Neighbor 法は、 n 個の点 $\{c_1, c_2, \dots, c_n\}$ からランダムに開始点 c_i を選ぶ。ここで、一度選ばれた点は未使用の点群から使用済の点群に移すものとする。次に未使用の点群の中から点 c_i に最も近い点 c_j を選び、点 c_i と c_j をつなぎ、 $i=j$ とする。前述の操作を、未使用の点群が無くなるまで繰り返し、未使用の点群が無くなった場合、現在の点 c_i と開始点をむすび、ハミルトン閉路を形成して終了する。

前章の局所探索法とは異なり、開始点となることができる点は n 個なので、Nearest Neighbor 法で得られる解の個数は最大 n 個と限られることがわかる。

本研究では、与えられた問題例に対し、各点（都市）を開始点として、Nearest Neighbor 法を実施し、与えられた問題例の点の数（都市数）と同数の解を用いるものとする。

6. TSP の性質を利用した解法

TSP の性質を利用している解法として Katayama らの遺伝的反復局所探索法と、比較の対象として従来の反復局所探索法を簡単に説明する。これらの解法の詳細に関しては文献4)を参照。

6.1. 反復局所探索法

局所探索法を用いて、優れた近似解を得るためのもっとも簡単な方法は、ランダムな解を発生させ、そのランダム解に対して局所探索法を許容される時間内に複数回実行する方法がある。これは“ランダム多点局所探索法”と呼ばれ、実用上良好な近似解を算出可能である。しかし、全く新しいランダム解から局所探索法を繰り返したのでは、効率の面での無駄が多い。そこで局所探索法から得られた局所解に適切な変化を与えて、局所解ではない解にし、その解から再び局所探索法を実施し局所解を得る解法を“反復局所探索法” (Iterated Local Search, ILS) とよぶ。

6.2. 遺伝的反復局所探索法

遺伝的反復局所探索法 (Genetic ILS, GILS) 4) は、従来の ILS とは異なり、二つの近似解に含まれる情報を利用し、最良の局所解からの脱出を行う方法として、遺伝的変換 (Genetic Transformation, GT) を用いる。GT は、二つの近似解に対して遺伝的な操作を行うことで一つの新しい近似解を生成する。GT の過程で生成される解は、後に局所探索法によって改善されるため、局所解でない解に変換されると共に異なる探索空間へ移動する処理が施される。

7. 実験方法

TSPLIB から選んだ 100, 532, 783 都市の問題例 (kroA100, kroB100, kroC100, kroD100, kroE100, rd100, lin318, att532, rat783) に対し、以下の三つの方法を用いて TSP の解の構造を解析する。

- a) ランダムに生成された解に 2-Opt 法を実施し、これを 2000 回行う。このとき生成された 2000 個の解 (局所解) を最適解と比較する。同時に 2000 個の局所解のうち、解質が良好な順に 10, 50, 100 個の解を選択する。それらの解の巡路がもつすべての枝を列挙し、すべてに共通する枝がある場合、それが最適巡路の枝を何%含んでいるか等を調べる。

- b) すべての各点（都市）を出発点として Nearest Neighbor 法を実施し，得られた解を最適解と比較する．同時に得られた解の内，解質の良好な順に 10, 50, 100 個の解を選択する．それらの解の巡路がもつすべての枝を列挙し，すべてに共通する枝がある場合，それが最適巡路の枝を何%含んでいるか等を調べる．
- c) 既に b) において，得られているすべての解に 2-Opt 法を実施し，得られた解（局所解）と最適解を比較する．b) と同様に選択を行い，それらの解の巡路がもつすべての枝を列挙し，すべてに共通する枝がある場合，それが最適巡路の枝を何%含んでいるか等を調べる．

ここで，解質は次式で与えられる．

$$\text{解質} = \frac{\text{得られた解長} - \text{最適解長}}{\text{最適解長}} \times 100 (\%)$$

また，b), c) において，Nearest Neighbor 法を実施して得られる解の個数は問題例の都市数に依存するため，a) とは異なり，得られた解の個数だけで比較を行うものとする．

8. 実験結果

TSPLIB から選んだ 100, 532, 783 都市の問題例 (kroA100, att532, rat783) に対して，それぞれランダムに生成された解に 2-Opt 法を実施して得られた 2000 個の局所解と最適解を比較した結果を図 4 (a) ~ (c) に示す．同様に，同じ問題例に対して Nearest Neighbor 法を実施し，その得られたすべての解と最適解を比較した結果を図 5 (a) ~ (c) に，更に，それらの解に 2-Opt 法を実施して得られた解を最適解と比較した結果を図 5 (a) ~ (c) に示す．各図は縦軸が局所解と最適解との長さの差 (cost)，横軸は最適巡路と共通していない枝の本数 (edge) とする．

表 1 の (a) ~ (c) の内容は，TSPLIB から選んだ 100, 532, 783 都市の各問題例 (上段から kroA100, kroB100, kroC100, kroD100, kroE100, rd100, lin318, att532, rat783) において，前章で示した三つの方法で得られた解に対し，良好な解質の順に選択された 10, 50, 100 個の局所解を平均した解質をそれぞれ示したものである．

表 2 の (a) ~ (c) は，表 1 と同様に各問題例に対して，選択された解の巡路において，すべてに共通する枝の本数を算出し，その共通する枝と最適巡路の枝との一致率 (%)，それらの解において，共通する枝を含む含有率 (%) を示したものである．

9. 考察

本実験では，Reeves ら6)の地形 Λ を表すものとして，探索空間 S を TSP の実行可能解の集合とし，目的関数 f は最適解と近似解の解長の差，距離 d は最適解と近似解の差（近似解の枝のうち，最適巡路に含まれない枝の数）にそれぞれ対応している．つまり図 4 (a) ~ (c)，図 5 (a) ~ (c)，図 6 (a) ~ (c) の縦軸は目的関数 f ，横軸は距離 d に対応している．

図 4 (a) ~ (c) より，ランダムに生成された解に 2-Opt 法を実施された局所解は，最適巡路で構成される枝を約 50%以上含んでいることがわかった．また，解質が良くなるにつれて最適巡路の枝を多く共有していくことがわかった．よって局所解が最適解に構造的に近づいていくにつれ，解質が良くなる傾向から，大谷構造を持つことが確認できる．

表 1 (a)，表 2 (a) より，解質が良くなるにつれ共通な枝は高い確率で最適巡路に含まれるが，表 2 の att532, rat783 の場合のように，共通する枝がすべて最適巡路と一致しない場合が生じた．この理由として，小規模な問題例に比べ，平均の解質が悪くなったことと，局所解の選択数が少ないことなどが考えられる．これらの理由から，局所解の選択数を 2 個とした場合，共通する枝の含有率は上がるかもしれない

が、最適巡路との一致率は下がると考えられる。つまり、GLS ではいくつかの局所解の集合から、二つの解を親として選ぶ。このとき、共通する枝を継承する交叉法等を利用する場合、その二つの局所解の巡路にはいくつかの共通する枝が存在するが、最適巡路ではない枝を子に遺伝させてしまう恐れがある。このような枝が原因で、解が改善されにくくなる可能性があることも考慮できる。

図5 (a) ~ (c) より、Nearest Neighbor 法で算出された解の巡路は、最適巡路で構成される枝を大体約 70%程度含んでいることが確認できた。また、Nearest Neighbor 法で算出された解は、その時点で解の枝の約 70%は最適巡路に含まれる枝である。これらの解に 2-Opt 法を実施すると、図6 (a) ~ (c) から、前時点より最適解の枝を含む比率が引き上げられる傾向がある。これらから、最適解の枝を多く取りいれて生成した解に、2-Opt 法を実施すると大体比率を保つか、もしくは引き上げる効果が期待できる。

ここで、TSP の性質を的確に利用した解法との違いとして、GILS と ILS の性能の差について考える。

どちらの解法も現在得られている最良の解を用いて、解を再構成していることから、大谷構造の特性を利用していることがわかる。よって、GILS と ILS の性能の差は、GILS が局所解の特性を利用して ILS より積極的に最適巡路に近づける点だと考えられる。GILS は、局所解どうしに共通する枝は最適巡路に含まれる枝である確率が高いことから、二つの解にある枝の中から最適巡路の枝を遺伝的操作により選定し、それをもとに GT 技法によって新しい解を構成している。GT 技法と Nearest Neighbor 法は解を構成する操作が類似していることから、Nearest Neighbor 法と同じ様な特性を持つと予想される。そうすると、ILS と比べた場合、GILS のほうが多くの最適巡路に含まれる枝を含んでいる局所解を算出すると思われる。これが ILS より GILS のほうが高品質な解を算出可能な理由の一つであると思われる。

10. むすび

本論文では、2-Opt 法により算出された局所解、Nearest Neighbor 法により算出された解、Nearest Neighbor 法により算出された解に 2-Opt 法を実施した局所解、これら三つの方法で得られた解と最適解の間に存在する関係の解析を行った。解析の結果から、Nearest Neighbor 法により算出された解の性質、また 2-Opt 法により算出された局所解の性質を知ることができた。そしてこれらの性質から、TSP の探索空間における大谷構造の存在を確認し、GLS における遺伝的操作として、共通する枝を継承する交叉法等を使用したときの問題点を提示することができた。また、TSP の性質を多く取り入れた解法が有効であることを、ILS と GILS の比較によって考察した。

他の組合せ最適化問題に対する大谷構造の存在の有無に関する研究は、最近行われ始めたばかりであり、効率的に問題を解くための有益なカギになり得る。今後、更に解析を行い、TSP や他の最適化問題に対する新しいまたは既存する解法の改良を模索する予定である。

参考文献

- 1). 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).
- 2). D.H. Wolpert, and W.G. Macready, "No free lunch theorems for search," *Santa Fe Institute Report*, SFI-TR-95-02-110, Santa Fe Institute (1995).
- 3). 久保幹雄, "メタヒューリスティックス," 離散構造とアルゴリズムⅣ, 近代科学社, pp.171 - 230, (1995).
- 4). K. Katayama and H. Narihisa, "Iterated local search approach using genetic transformation to the traveling salesman problem," *Proc. 1999 Genetic and Evolutionary Computation Conference*, Orlando, Florida, USA, Jul.13 - 17, vol.1, pp.321 - 328 (1999).
- 5). P. Merz and B. Freisleben, "Genetic local search for the TSP : New results," *Proc. 1997 IEEE Int. Conf. on Evolutionary Computation*, pp.159 - 164 (1997).
- 6). C. R. Reeves, and C. Hohn, "Are Long Path Problems Hard for Genetic Algorithms ?," *4th International Conference on parallel problem solving from Nature*, pp.134-153 (1996).

表 1 (a) ランダムに発生させた解に 2-Opt 法を実施して算出した
解の集合から、解質の良好な順に各選択数で選出したときの平均解質(%)

問題例 \ 選択数	10	50	100	2000
kroA100	1.371	2.189	2.762	8.922
kroB100	1.651	2.725	3.340	8.451
kroC100	1.174	2.362	3.033	9.730
kroD100	2.150	3.243	3.730	9.041
kroE100	2.279	3.214	3.696	8.793
rd100	1.862	3.191	3.972	10.071
lin318	5.844	6.449	6.849	10.540
att532	7.136	7.741	8.080	10.991
rat783	9.781	10.276	10.544	12.762

表 1 (b) Nearest Neighbor 法により算出した
解の集合から、解質の良好な順に各選択数で選出したときの平均解質(%)

問題例 \ 選択数	10	50	100	全体
kroA100	20.72	23.96	27.09	
kroB100	17.80	21.86	26.05	
kroC100	15.26	20.43	26.18	
kroD100	17.93	24.85	29.00	
kroE100	14.04	19.42	24.09	
rd100	21.05	24.47	27.82	
lin318	18.67	21.05	22.04	25.30
att532	20.89	22.29	23.28	27.21
rat783	19.83	21.83	23.01	27.31

表 1 (c) Nearest Neighbor 法で算出した解に 2-Opt 法を実施して算出した
解の集合から、解質の良好な順に各選択数で選出したときの平均解質(%)

問題例 \ 選択数	10	50	100	全体
kroA100	3.12	4.24	6.02	
kroB100	3.03	4.65	6.39	
kroC100	3.66	5.87	7.30	
kroD100	3.99	6.94	9.16	
kroE100	1.87	4.02	6.05	
rd100	4.07	5.65	7.76	
lin318	5.53	6.47	6.93	8.55
att532	5.86	6.35	6.69	8.18
rat783	6.66	6.97	7.19	8.43

表2 (a) ランダムに発生させた解に 2-Opt 法を実施して算出した解の集合から、解質の良好な順に各選択数で選出したとき、それらすべてに共通した枝と最適巡路の枝との関係

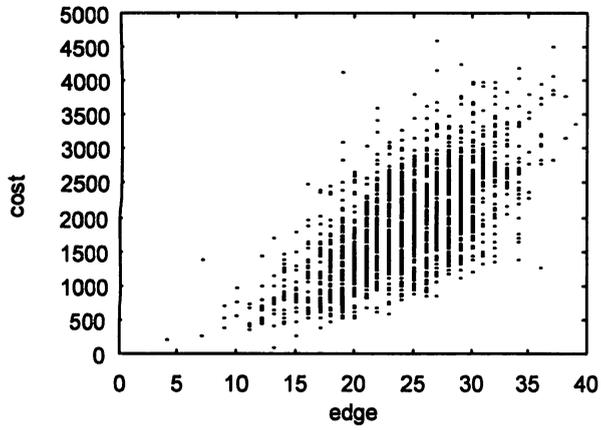
問題例 \ 選択数	10			50			100		
	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)
kroA100	62	100.0	62.0	42	100.0	42.0	30	100.0	30.0
kroB100	48	100.0	48.0	27	100.0	27.0	21	100.0	21.0
kroC100	69	100.0	69.0	36	100.0	36.0	32	100.0	32.0
kroD100	45	100.0	45.0	31	100.0	31.0	25	100.0	25.0
kroE100	47	100.0	47.0	24	100.0	24.0	20	100.0	20.0
rd100	57	100.0	57.0	28	100.0	28.0	18	100.0	18.0
lin318	136	100.0	42.8	73	100.0	23.0	53	100.0	16.7
att532	124	98.4	23.3	58	100.0	10.9	37	100.0	7.0
rat783	141	99.3	18.0	53	100.0	6.8	39	100.0	5.0

表2 (b) Nearest Neighbor 法により算出した解の集合から、解質の良好な順に各選択数で選出したとき、それらすべてに共通した枝と最適巡路の枝との関係

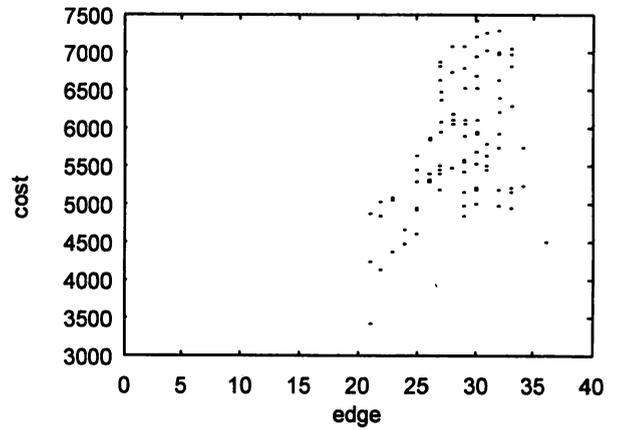
問題例 \ 選択数	10			50			100		
	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)
kroA100	58	87.93	58.0	42	92.86	42.0	30	93.33	30.0
kroB100	74	85.14	74	51	92.16	51.0	31	96.78	31.00
kroC100	85	78.82	85.0	45	93.33	45.0	34	100.00	34.0
kroD100	85	72.94	85	48	81.25	48.0	33	90.91	33.0
kroE100	69	78.26	69.0	44	95.45	44.0	33	96.97	33.0
rd100	64	70.31	64.0	38	86.84	38.00	30	86.67	30.0
lin318	212	88.68	66.67	188	90.96	59.12	157	96.82	49.37
att532	398	77.39	74.81	307	84.69	57.71	289	85.12	54.32
rat783	743	72.41	94.89	439	88.38	56.07	408	89.22	52.11

表2 (c) Nearest Neighbor 法で算出した解に 2-Opt 法を実施して算出した解の集合から、解質の良好な順に各選択数で選出したとき、それらすべてに共通した枝と最適巡路の枝との関係

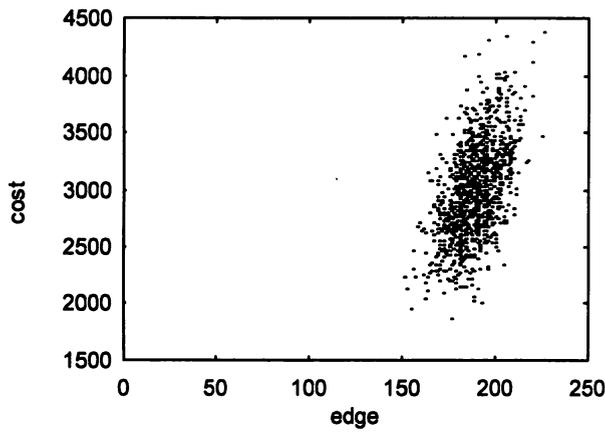
問題例 \ 選択数	10			50			100		
	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)	本数	一致率 (%)	含有率 (%)
kroA100	80	92.5	80.0	51	100.0	51.0	37	100.0	37.0
kroB100	65	98.5	65.0	38	100.0	38.0	23	100.0	23.0
kroC100	64	96.9	64.0	36	100.0	36.0	25	100.0	25.0
kroD100	61	93.4	61.0	37	100.0	37.0	26	100.0	26.0
kroE100	80	77.5	80.0	35	100.0	35.0	30	100.0	30.0
rd100	55	92.7	55.0	33	100.0	33.0	18	100.0	18.0
lin318	183	99.5	57.5	134	100.0	42.1	114	100.0	35.8
att532	284	53.4	93.0	133	99.3	25.0	95	100.0	17.9
rat783	323	95.0	41.3	168	98.8	21.5	136	99.3	17.4



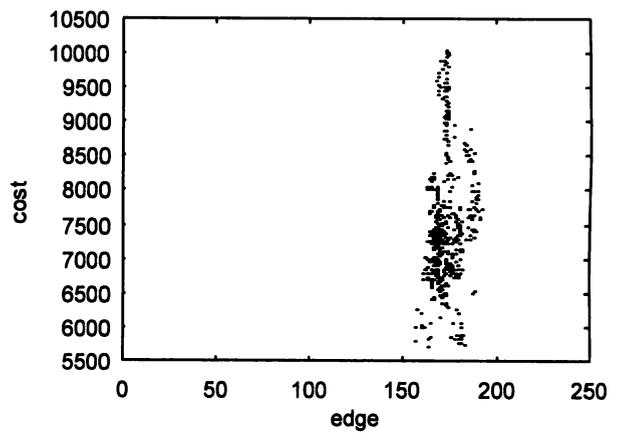
(a) kroA100



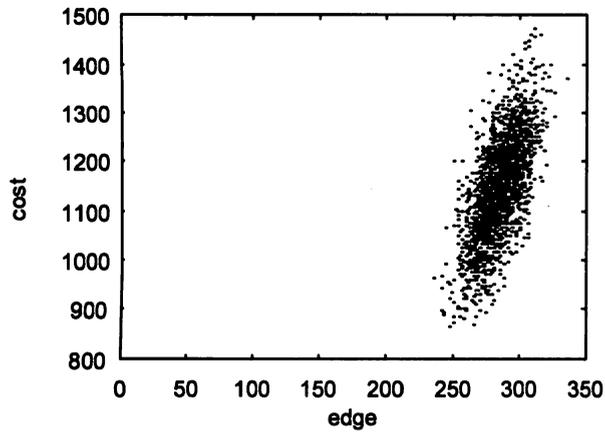
(a) kroA100



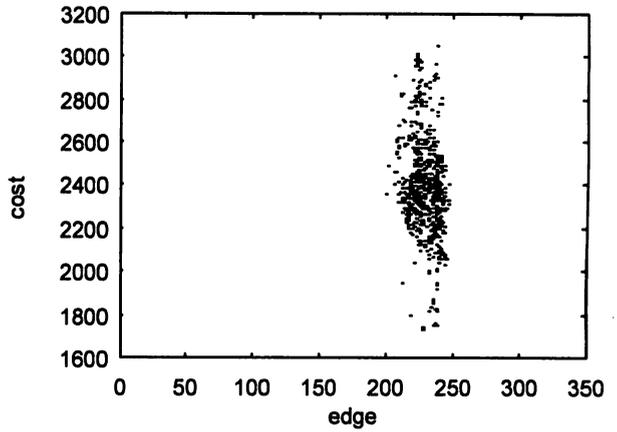
(b) att532



(b) att532



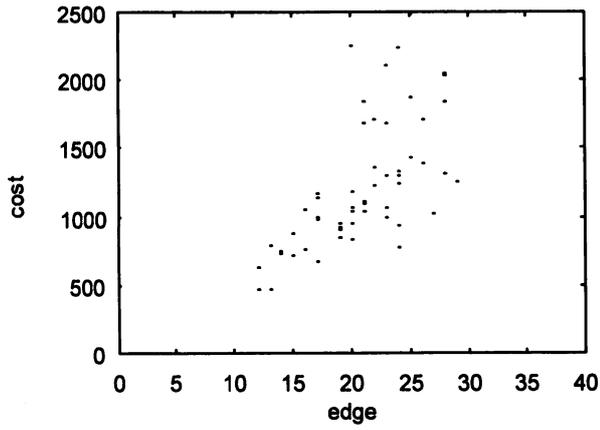
(c) rat783



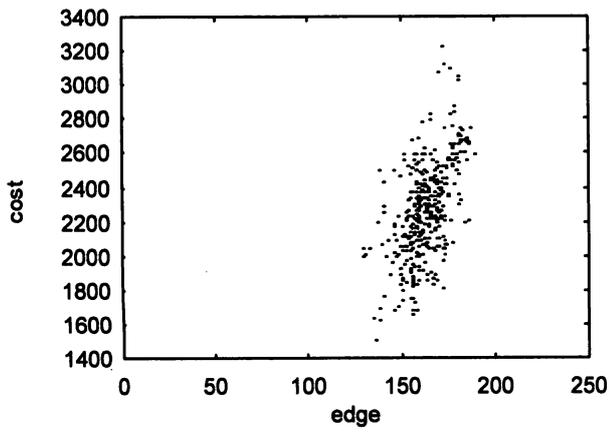
(c) rat783

図4 ランダムに発生させた解に2-Opt法を実施して算出した解の集合と最適解との関係

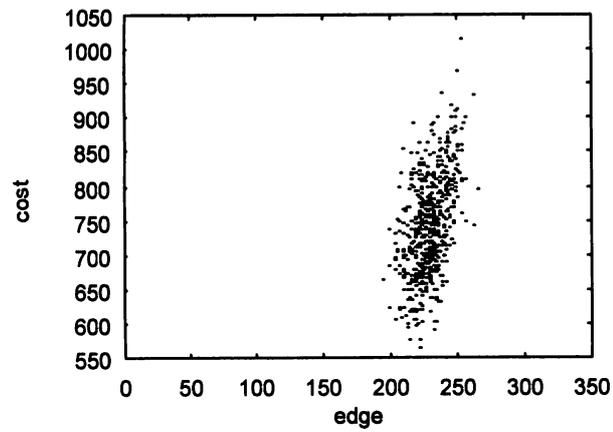
図5 すべての都市を開始点とした Nearest Neighbor法で算出した解の集合と最適解との関係



(a) kroA100



(b) att532



(c) rat783

図6 Nearest Neighbor 法により算出した解に 2-Opt 法
を実施して算出した解の集合と最適解との関係

Analysis of Local Optima Structure in the Traveling Salesman Problem

Masafumi TANI, Kengo KATAYAMA and Hiroyuki NARIHISA*

Graduate school of Engineering

**Department of Information & Computer Engineering*

Faculty of Engineering

Okayama University of Science,

Ridai-cho 1-1, Okayama 700-0005, Japan

(Received November 4, 1999)

The Traveling Salesman Problem (TSP) is one of the most well known combinatorial optimization problems. Many heuristic algorithms have been developed to find near-optimal solutions within reasonable times, for example the Genetic Algorithm, Simulated Annealing, Ant System, Tabu Search, etc. Generally these algorithms are based on a Local Search (LS). A Genetic LS (GLS) is an algorithm that incorporates the LS into a Genetic Algorithm. GLS algorithms are able to find good local optima by applying genetic operators (selection, crossover, mutation) to several local optima in a population. Particularly, GLS that use an observation result of Boese et al. are known to be able to find excellent solutions.

Boese et al. have reported that the solution space of the TSP has a character called "big valley structure". The algorithms proposed by Merz et al. and Katayama et al. utilize this character to obtain much improved solutions. However, according to Wolpert et al., there are no algorithms that can obtain very good solutions without utilizing suitable characteristics of a specific problem. From this point of view, it is important to know the characteristics of the TSP. In this paper, we analyze relations between the globally optimal solutions and each solution obtained by the following three algorithms:

- a) The local optimal solutions obtained by a 2-Opt local search starting from random solutions.
- b) The solutions obtained by a nearest neighbor heuristic.
- c) The local optimal solutions obtained by a 2-Opt local search starting from solutions obtained by a nearest neighbor heuristic.