

## 車両配送問題のタブー探索による解法の実験的解析

中出元樹, 朴成浩, 今井浩(東大), 西村茂樹, 下浦弘, 天目健二(住友電工)

東京大学大学院理学系研究科情報科学専攻  
〒113 東京都文京区本郷 7-3-1  
住友電気工業株式会社

車両配送問題は複数のトラックを使って品物を倉庫から複数の店に配送して戻って来る総巡回路長を最小化する問題であり、コンビニエンスストアチェーンのトラック配送計画等、数多くの重要な実用的応用を持つ。この問題はNP困難に属するため多くの近似解法の研究がなされてきたが、複数の手法の性能比較研究が多く、1つの手法で実際に良い性能を得るための知見はまだ充分とはいえない。

本研究では有力なメタ戦略の1つであるタブー探索に基づいた車両配送問題の近似解法において、主要な設計要素が解にどう影響するかを実験的に明らかにした。これらの実験の知見により、目的に応じた近似解法設計の指針を得ることができる。

## Experimental Analysis of Tabu Search for Vehicle Routing Problem

Motoki Nakade\*, Narihiro Park\*, Hiroshi Imai\*  
Shigeki Nishimura†, Hiroshi Shimoura†, Kenji Tenmoku†

\*Department of Information Science, Faculty of Science, University of Tokyo  
7-3-1 Hongo, Bunkyo-ku, Tokyo 113, Japan

†Sumitomo Electric Industries, Ltd., 1-1-3 Shimaya, Konohana-ku, Osaka 554, Japan

Vehicle Routing Problem is a problem to find the minimum total cost routes of trucks delivering goods from the depot to the customers. It has various practical applications such as truck delivery scheduling for convenience store chain. Since it belongs to NP-hard, many researches of approximate algorithms, mainly focus on comparison among several methods, have been made. However, it is still not obvious how to design one effective method.

We have made large amount of experiments of Tabu Search and have found several factors which affect the behavior of the search. Our result provides a good indication in designing effective approximate algorithms to meet their own purposes.

## 1 はじめに

車両配送問題 (Vehicle Routing Problem: VRP) は複数のトラックを使って品物を倉庫から複数の店に配送して戻って来る総巡回路長を最小化する問題であり、コンビニエンスストアのチェーンにおけるトラック配送計画等の数多くの現実的な応用を持つ。VRP の解は店集合をクラスターに分割し分割された各集合に対して巡回セールスマン問題を解くことによって得ることができ、2つの難しい組合せ最適化問題が階層構造になっていると捉えることもできるので、理論的にも非常に興味深い問題である。ところがNP 困難に属するために、ごく小規模の問題を除いては現実的な時間で厳密解を出力するアルゴリズムを構成するのは難しい。従って厳密解法の研究 [7] [8] と共に効率の良い近似解法の研究も非常に重要となっている。

VRP の近似解法は従来セービング法 [5] が主流であったが、近年メタヒューリスティックスによる解法が目覚ましい成果をあげており [2][4][3]、従来よりもはるかに良い近似解を高速に得られるようになってきている。メタヒューリスティックスは最適化問題の近似解法の強力なパラダイムの一つであり、しばしば難しい組合せ最適化問題を現実的時間内に解く唯一の手段となる。シミュレーテッドアニーリング、遺伝的アルゴリズム、タブー探索、GRASP 等がこのパラダイムに属する。これらのアルゴリズムは概して良い結果を出すのが、残念ながらその性能はあまり安定していない。これらの性能は探索に用いられる各種パラメータに左右されるだけでなく、問題の構造を探索に如何にうまく取り込むかにも影響を受ける。これらのアルゴリズムの性能を理論的に解析することは困難なため、実験的な解析が非常に重要となる。

タブー探索 [1] は最も成功しているメタヒューリスティックスの一つであり色々な研究がなされているが、それらの多くは特定の問題に対する複数のメタヒューリスティックス同士、例えばタブー探索とシミュレーテッドアニーリング法、の性能の比較 [2][4] が主である。

他方、問題に特化した手法を追求した結果、基本的なタブー探索の枠組からかなり離れてしまっている傾向も見られる [3]。

以上より、そもそものメタヒューリスティックスの魅力である枠組の単純さという長所を生かし、かつ特定の手法で実際にアルゴリズムを設計するための指針がまだ明確でないといえる。そこで我々はタブー探索の枠組で実際の問題に適用するアルゴリズムを設計するに当たって主要な設計要素の選択が探索の挙動にどう影響するのかを系統的な実験によって明らかにした。

以下、まずタブー探索と車両配送問題について述べ、その後アルゴリズムの主な設計要素を列挙する。その後それらの設計要素に関して行なった系統的实验について説明し、最後に結論を述べる。

## 2 タブー探索の概略

タブー探索は局所探索をベースとした近似解法であり、局所解に陥らないための機構として過去何回かの探索のステップで訪れた解を近傍から除外する(「タブー」にする)という方法をとる。疑似コードで書くと以下のように表現できる。

```
procedure tabu_search
  t := 0
  x0 := 適当な初期解
  xbest := x0
  Ξ := φ
  while 停止条件 ≠ yes do
    xt+1 := move(xt)
    Ξ := Ξ ∪ {xt} \ {xt-L}
    t := t + 1
    if f(xt) < f(xbest) then
      xbest := xt
  return xbest
```

ここで  $move(x)$  は各探索ステップで近傍  $N(x)$  の中から選択される解を表し、以下のように定義される。

$$move(x) = \begin{cases} x' & f(x') \leq f(y) \quad \forall y \in N(x) \setminus \Xi \\ \phi & N(x) \setminus \Xi = \phi \end{cases}$$

③はタブーリストと呼ばれ、過去  $L$  回のステップで選択した解を記憶している。 $L$  をタブーリスト長と呼ぶ。タブーリストは解そのものを記憶しなくても良く、解の変化を記憶しておいても構わない。例えば VRP において 2 つの店のたどる順番を入れ換えた解を近傍とした場合、入れ換える前の巡回路そのものではなく入れ換えた 2 つの店をタブーリストに入れることによって過去の探索の履歴を保持することもできる。

タブー探索ではある解の近傍にはその解自身は含まないように定義する (すなわち  $x \notin N(x)$ )。すると例えば局所解  $x$  においては  $f(\text{move}(x)) > f(x)$  となるので探索の過程は必ずしも単調減少するわけではない。

以上のようにタブー探索はメタな枠組なので本研究の車両配送問題以外にも様々な組合せ最適化問題に広範に適用することができる。

### 3 車両配送問題の定式化

車両配送問題 (Vehicle Routing Problem: VRP) は容量制限のあるトラックが倉庫から  $n$  個の店を各々一度ずつ訪れて品物を届けるための最短巡回路を求める問題である。各店  $i$  が要求する品物の量  $d_i$  の合計が 1 台のトラックの容量制限  $M$  を越えたらトラックが複数台必要になるので、VRP の解は一般に複数の巡回路が倉庫を共有した形になる。VRP は以下のような組合せ最小化問題に定式化できる。ただし  $c_{ij}$  を店  $i, j$  間のコスト (所要時間等) とし、トラック  $t$  がまわる巡回路に含まれる店の集合を  $V_t$ 、その巡回路に含まれる枝 (まわり順が連続する店の組)  $\langle i, j \rangle$  の集合を  $E_t$  と表す。

$$\begin{aligned} & \text{minimize} \sum_t \sum_{\langle i, j \rangle \in E_t} c_{ij} \\ & \text{s.t.} \sum_{i \in V_t} d_i \leq M \quad \forall t \end{aligned}$$

図 1 は 75 店の車両配送問題の解の例である。ここでは 10 台のトラックを用いている。すべての巡回路が共有している中央の点が倉庫であり、各トラックはここから出発して各々の巡回路に沿って品物を店に配送してくる。

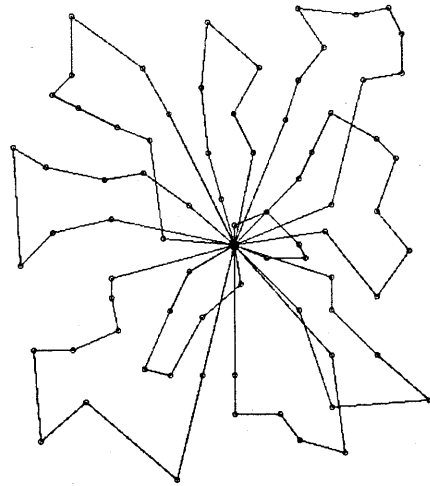


図 1: 75 店の車両配送問題の解の例

## 4 アルゴリズムの設計戦略

タブー探索はメタな枠組なので、適用する問題に応じてアルゴリズムの細部を決定しなければならない。具体的には 1) 初期解をどう構成するか、2) 近傍をどう定義するか、3) タブーリスト長をどうするか、4) 探索をいつ終了するか、等が問題となる。これらを適切に設計しないと効率の良いアルゴリズムは得られない。以下に車両配送問題に対するタブー探索の設計方法を、従来の方法と我々の方法を比較して述べる。

### 4.1 初期解の構成法

初期解の構成法は、従来はセービング法 [5] というヒューリスティクスが用いられていた [2][4]。

ランダムな初期解構成法も試されていたが、近年の手法ではほとんど採用されていない。我々はこのランダムな初期解からの探索を系統的に試行し、セービング法による初期解からの探索と比較実験した。Taillard ら [3] はより複雑な初期解構成法を提案しているが、彼らはタブー探索そのものの枠組も拡張しているので、ここでは言及しないことにする。

一般に複数のランダム初期解から探索を実

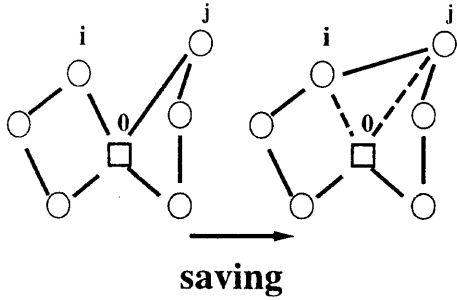


図 2: セービング法による巡回路の合併

行するとその中の最良解は良い精度になる可能性があるが、これは問題及び探索の性質に強く依存する。この実験の主旨は車両配送問題でタブー探索を用いた場合に複数のランダム初期解から探索を実行することが最終的にその労力に見合うかどうかを見極めることにある。

以下、セービング法による解の構成法を説明する。まず初期解として店を1つだけ回る巡回路を店の数だけ生成する。任意の異なる2つの巡回路について倉庫とつながっている店をそれぞれから1つずつ選び  $i, j$  とした時、コスト改善  $s_{ij} = c_{ij} - c_{i0} - c_{0j}$  が最大となるような組合せを見つけ、その  $i, j$  を結んで2つの巡回路から新しい1つの巡回路を生成する(図2参照)。ただし容量制約を満たす組合せを選ぶようにする。

この操作を、それ以上実行できなくなるまで繰り返して得られた解がセービング法の解となる。

#### 4.2 近傍の定義

近傍の定義には、従来は Osman ら [2] が提唱していた shift 近傍と interchange 近傍がよく用いられてきた。しかし局所探索は近傍の定義によって探索の性能が大きく左右されるので、我々は上記の近傍の他に可変近傍と枝近傍という2種類の新しい近傍を定義し、探索の性能を比較した。以上3種類の近傍の定義を順番に述べる。

最初の shift 近傍はある解の1つの巡回路の中の1つの店を別の巡回路に移してできる解の集合として定義される。interchange 近傍

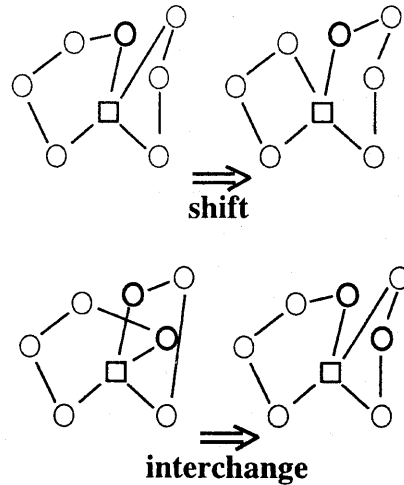


図 3: shift 近傍と interchange 近傍

は異なる2つの巡回路から店をそれぞれ1つずつ選びとり、それらを移し合っできる解の集合として定義される(図3参照)。これらの近傍定義ではあくまでも巡回路を集合として捉えていて、近傍探索が集合分割を部分的に変えていくことに対応することに注意する。従って近傍の評価関数を計算するためには厳密には各部分集合について巡回セールスマン問題(TSP)を厳密に解かなければならないが、実際には計算時間との兼ね合いで通常TSPの高速な近似解法が適用される。

次の可変近傍は shift 近傍の拡張であり、異なる2つの巡回路にある複数の店同士の入れ換えを考える。ただし複数個同士の入れ換える組合せをすべて試すのではなく、店の入れ換えを shift 近傍の移行操作の積み重ねで段階的に検討する。厳密に表現すると以下ようになる。なお、以下の表記で暫定解とは近傍探索の過程で得られた実行可能解(容量制約を満たす解)のうち最良のものを指し、検討解とは実行可能ではないが解の値が一番良いものを指す。

1. 異なる2つの巡回路  $i, j$  の組すべてについて以下の操作を行う。
2.  $i_0 = i, j_0 = j$  とし、 $i_0, j_0$  を0番目の検討解とする。
3.  $n$  番目の検討解  $i_n, j_n$  の間で1都市の shift

によって得られる解のうち、容量、タブー制約を満たし最もコストを改善するものを  $n + 1$  番目の「暫定解」とする。

- 上記の操作で容量制約を無視して最もコストを改善するものを  $n + 1$  番目の「検討解」とする。
- ステップ 3,4 を  $s_1 + s_2 + \dots + s_n > threshold$  の間繰り返す。ただし  $s_n$  ( $n$  段階目の近傍チェックで良くなるコスト)、 $threshold$  は次の通り。

$$s_n = (\text{cost}(i_{n-1}) + \text{cost}(j_{n-1})) - (\text{cost}(i_n) + \text{cost}(j_n))$$

$$threshold = \min(0, s_1) \quad (\leq 0)$$

- 得られた暫定解のうち最良のものを巡回路の組  $i, j$  の暫定解とする。すべての組  $i, j$  の中で最良の暫定解を採用する。

以上に加えて、より良い解を得るための工夫として次の点を実装に採り入れている。

- タブー制約を無視すると現在の最良解を更新できるような検討解が存在する場合はタブー制約を無視して採用
- 検討解の採用が終わった後、修正された各配送路に TSP 近似解法 (3-opt 法) をかけてより改善する。

最後の枝近傍が上記の 2 種類の近傍と大きく異なる点は、VRP の解を集合分割と各巡回路の最適化問題の階層構造では捉えずに、枝の集合として捉えている点である。厳密には枝を  $n$  本取り去った後異なる枝を「任意本」加えて実行可能解となる時、それらの解の集合を  $n$ -delete 近傍と定義する。ここで注意しなければならないのは、取り除いた枝の数と新たに加える枝の数が必ずしも一致しないということである。例えば図 4 は上下共に右の図が左の図の 2-delete 近傍であるが、上の近傍は新しい枝を 2 本付け加えているのに対して、下の枝は 3 本付け加えている。従って 2-delete は一般に対称でない。即ち  $y = N(x)$  の時に必ずしも  $x = N(y)$  とならない。

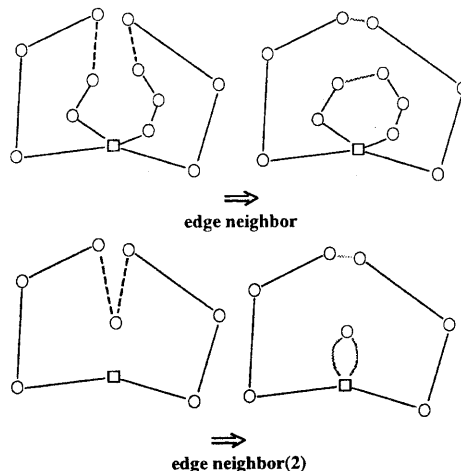


図 4: 枝近傍 (上) 2 本の枝をつなぎ換えた例 (下) 除いた枝と加えた枝の数が一致しない例

これらの近傍は一般に対象範囲が広いほど局所解に陥りにくいので強力であるといえるが、同時に一回の近傍探索に手間がかかってしまうため、実際には最終的に得られる解の質と探索時間のトレードオフで採用する近傍は決定される。

### 4.3 タブーリスト

タブーリストは過去に探索した解に関する情報を何らかの形で保持する機構である。解そのものを保持する方式が直接的であるがこれは非常に多くの記憶容量を必要とするので、その代わりに解の変化を記憶しておく方法が通常用いられる。この方式だとタブーリストが短か過ぎると以前に探索した解をすぐに忘れてしまうため局所解から抜け出しにくくなり、タブーリストが長過ぎると逆に禁止される近傍が多くなり過ぎて、有望な解への移行も禁止してしまいやはり性能が悪くなる。従ってタブーリストには中間的な適切な長さがあると予想できる。

### 4.4 その他

4) の探索終了判定は、ある一定回数以上最良暫定解の更新がなされなかった所で打ち切

るという方式に統一した。

## 5 実験と解析

### 5.1 実験のベンチマーク問題

実験の詳細を説明する前にまず実験した車両配送問題のベンチマークについて説明する。実験には Christofides[6] にある店数 50-199 の 7 つのベンチマークを用いた。これらのうち 5 つ (店数 50, 75, 100, 150, 199) は正方形内にランダム一様分布した店の問題であり、2 つ (店数 100, 120) は店の分布がいくつかのグループ毎にクラスターになっている問題である。実験はこれらすべてについて行い以下の知見はどの問題にも共通して得られたが、グラフには 3 つの問題 c100(100 店一様分布)、c120(120 店クラスター)、c150 (150 店一様分布) についてまとめた。

### 5.2 実際の計算の流れ

ここでは実装したアルゴリズムについて少し具体的に触れる。アルゴリズムは 1) 初期解を構成し、2) タブー探索を実行し、3) 終了条件を満たした時点で各巡回路に TSP の厳密解法をかけて更なる解の改善をはかり終了する。終了条件はどの実験も暫定最良値が 10000 反復の間改善されなかった時点で満たされるとした。

### 5.3 初期解の構成法に関する実験

まず初期解の構成法が最終的に得られる解の質にどのように影響するか調べた。具体的にはセービング法による初期解とランダム初期解 100 通りからそれぞれタブー探索を始めて、最終的に得られた解の質を比較した。実験結果を図 5 に示す。この図は横軸が得られた解の値 (総巡回路長) なので短いほど良い。縦軸は対応する解の値の区間に含まれた探索回数を示す。この時のタブーリスト長はセービング法で最も良い解が得られた値に統一している。

また、セービング法からスタートした探索で得られた解は各問題それぞれ以下のように

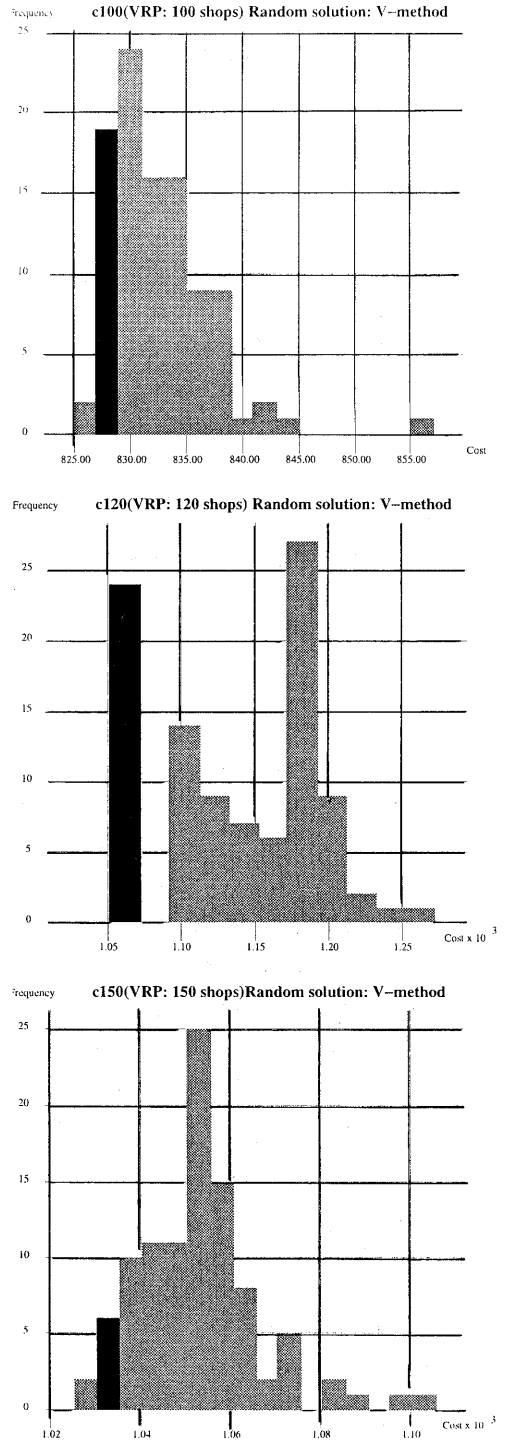


図 5: ランダム解 100 個から探索した場合の最終解の値 (c100, c120, c150)

なった。

問題	解の値
c100	829.45
c120	1042.11
c150	1031.42

これらの解は図4の黒い帯の部分に入る。これによるとセービング法からスタートして得られた解はほとんどのランダム初期解から得られた解よりも良かったが、いくつかのランダム初期解からスタートした探索ではセービング法からスタートしたケースを上回っていることが分かる。従って、計算力に余力がある時は、多くのランダム初期解から探索をスタートさせる価値があるといえる。

#### 5.4 近傍とタブーリストに関する実験

次に近傍とタブーリストに関する実験をまとめる。各近傍にそれぞれ対応したタブーリストを定義し、探索で得られる解の質がタブーリストの長さによってどのように変わるか実験で調べた。各近傍とそれぞれに対応するタブーリストの定義を以下に述べる。

近傍は3種類実験した。1つ目の shift、interchange 近傍に対してはタブーリストに過去  $L$  回の間に入れ換えが起こった店と巡回路の組を保存する。2つ目の可変近傍ではすべての入れ換えた店と元の巡回路の組をタブーにする。従って4つの店が各々異なる巡回路に移ったならばタブーリストには4つの組が入る。3つ目の枝近傍では解の変化によって取り除かれた枝を個別に保存する。以上のようにアルゴリズムを設計し、各探索におけるタブーリスト長と探索で得られた解をプロットしたのが図6である。各グラフで Sh&Int, V-method, Edge-Nbr はそれぞれ shift, interchange 近傍、可変近傍、枝近傍を表し、Best-known は現時点で知られている最良解を表す。

グラフより、最良解が得られるタブーリスト長は問題のサイズや近傍の定義に依存するが、タブーリスト長と解の質との相関関係は近傍の種類に関わらず原則的に凸形をしていることが分かる。これは有用なタブーリスト長を

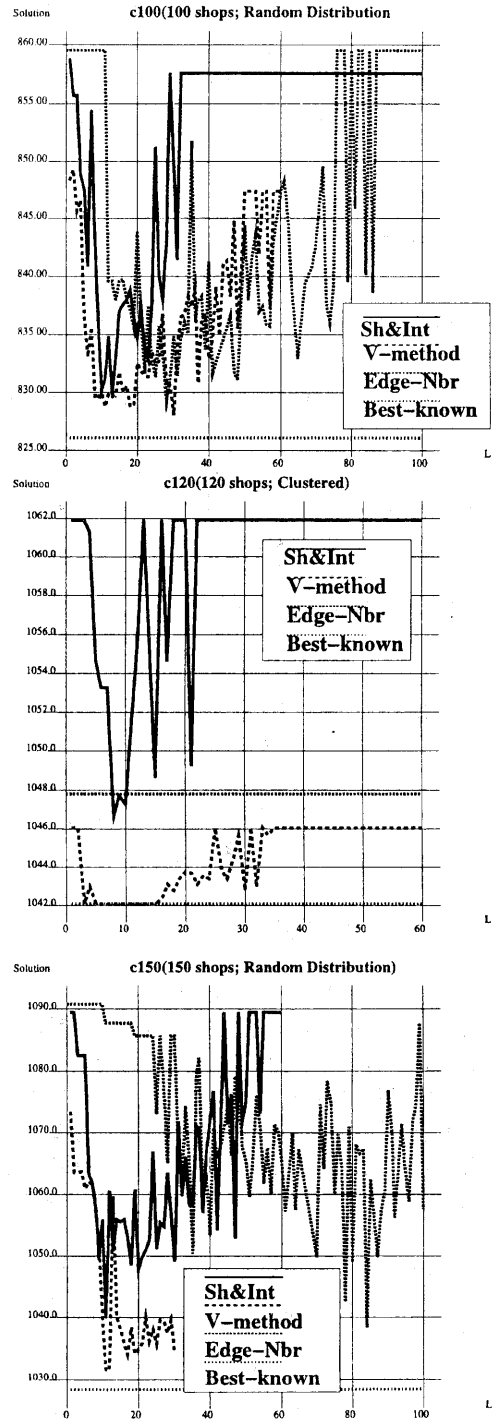


図6: タブーリスト長と得られる解の質の関係 (c100,c120,c150)

効率良く決定する重要な指標となる。なぜなら近似的に凸形を仮定して良いならばタブーリスト長の最適値は疑似的に2分探索のような手法で高速に確定することが可能となるからである。

## 6 結論

本研究ではタブー探索の枠組の中で車両配送問題を効率的に解く近似アルゴリズムを設計するに当たって指針となるような探索の性質を系統的な実験により調べあげた。具体的には、設計戦略の鍵となる1) 初期解生成、2) 近傍設計、3) タブーリストそれぞれについて実験的知見を得た。1) 初期解生成についてはセービング法がかなり良い性能を示すが、計算力に余裕があればランダム他スタート方式を試す価値はあるといえる。2) 近傍設計については shift, interchange 近傍をより一般化した可変近傍は、より良い解に到達できる力を持つので、これも計算力があれば採用した方がよい。そして3) タブーリストについては、リスト長と得られる解の値との間に原則的に凸形の関係が成り立つので、適切なリスト長は比較的容易に確定することができるといえる。

## 7 今後の課題

今回の研究はタブー探索のアルゴリズムの設計要素が得られる解に与える影響を系統的な実験で調べた。これらの指標は最終的に効率の良い近似アルゴリズムを実装するための土台となる知見である。アルゴリズムの効率性を考えた時には今まで議論してきた定性的な要素以外に定量的な要素、特に時間の要素を考慮しなければならない。例えば可変近傍や枝近傍は近傍の大きさを大きくとれるため探索の力は強力にできる可能性を持っているが、その反面単純に設計すると1回の近傍探索に時間がかかるという欠点を持つ。この欠点を克服する方法としては、近傍の中で良く使われるものだけを調べて大部分の近傍探索を省略する方式が考えられる。これらの方式が効率のかどうかを議論するためには、さらなる実験的裏付けが

必要となろう。現実的な近似アルゴリズムをデザインするならばこのような側面もバランス良く考えていかなければならない。これらの実際的な問題を検証するために、今後もより詳細な実験を重ねていきたい。

## 参考文献

- [1] F. Glover, "A user's guide to tabu search," *Annals of Operations Research* vol.41 1993, p.3-28
- [2] I. H. Osman, "Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem," *Annals of Operations Research* vol.41 (1993), p.421-451
- [3] É. Taillard, "Parallel Iterative Search Methods for Vehicle Routing Problems," *Networks* Vol.23 (1993), p.661-673
- [4] C. A. Hjorring "The Vehicle Routing Problem and Local Search Metaheuristics," Ph.D. dissertation submitted to Dept. of Engineering Science, The University of Auckland 1995
- [5] G. Clarke, J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Operations Research* vol.12 p.568-581
- [6] N. Christofides, A. Mingozzi, P. Toth "Combinatorial Optimization", Wiley 1979
- [7] G. Laporte, Y. Nobert "Exact Algorithms for the Vehicle Routing Problem" *Annals of Discrete Mathematics* 31 (1987) p.147-184
- [8] M. L. Fisher "Optimal Solution of Vehicle Routing Problems Using Minimum K-trees" *Operations Research* Vol.42 No.4 p.626-642