非対称巡回セールスマン問題に対する解法の研究

菊池　健之・藤野　悠（沼田　一道　助教授）

１．はじめに

情報･通信の発達した現代社会でも宅配便や商品配送のような“ 実物 ”の移動は必須であり, 様々な規模で日常的に行われている．それに伴う費用（輸送コスト）は全体として膨大なものであり, それを削減するための工夫・研究が盛んに行なわれている．そこに現れる基本的問題に，巡回セールスマン問題（Traveling Salesman Problem : TSP）と呼ばれるものがある．TSP は，指定された地点をすべて訪問して出発地点に戻る巡回路の中で移動に伴う費用が最小となるものを求める問題である．TSP については，厳密解法と近似解法の両面から非常に多くの研究がなされている．

沼田研究室では，近年非対称型の TSP（Asymmetric TSP : ATSP）に対する近似解法の研究を行なってきている[2,3]．その際，近似解法の精度を絶対評価するために，厳密解を求める必要が生じた．しかし Web 上を探しても，ATSP の厳密解法を求めるソフトウェアは見当たらない．それは，１）厳密解法の基本的枠組みに大きな進展がない　２）ATSP を STSP へ変換することも（理論的には）可能なので STSP の厳密解法の研究に集中している　３）ATSP は STSP と比べデータ量が多く扱い難い　などの理由によるものと思われる．しかし，ATSP の研究は実務上重要であり，手軽に厳密解を求めるソフトの需要は高い．

そこで，本研究では厳密解法の開発をとりあげ，分枝限定法を用いた ATSP の厳密解法についての基本アルゴリズムの理解と実装，ヒューリスティックな部分の工夫，これらを基にした設定時間内に解き終わる解法の提案，その解法の性能評価を行なう．

２．ATSP

２．１．ATSP の必然性

トラックで工場から顧客のもとに製品を届ける際，製品を積み込み地点で満載にし，積み下ろし地点で全て降ろすという輸送形態がよく現れる．トラックは与えられた輸送要求を全て満たし，再び輸送基地へ戻る．その際，積み下ろし地点から次の輸送要求の積み込み地点までは荷物を何も積んでいない状態で走らなければならない．輸送要求を満たす順番によって変化するのは，この空輸送の距離だけなので，空輸送の総距離を最小にする巡回路が問題となる．

そこで，図１のような２つの輸送要求が与えられたとする．輸送要求はそれぞれ１つのまとまりとして考える．このとき，輸送要求 i から j へ行く場合と輸送要求 j から i へ行く場合とでは，トラックが空で走る距離が異なる．つまり，輸送要求iからjへの空輸送距離をc_{ij}とすると $c_{ij} \neq c_{ji}$ である．よって，トラックの総空輸送距離を最小にする巡回路を求める問題に，ATSP がみられる．

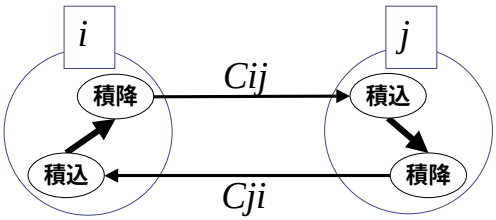

図１　輸送要求間の輸送順序

２．２．ATSP の定式化

都市数をn, $\{1,2,\ldots,n\}$ の都市集合をV, 都市iから都市jへの費用をc_{ij}, 総費用をz, 都市iから都市jへの移動の有無をx_{ij}であらわすとき, 移動する場合は$x_{ij} = 1$, 移動しない場合は$x_{ij} = 0$ とするとATSP は次の（ATSP）のように定式化される．

(ATSP)

$$\text{Minimize} \quad z = \sum_{i \in V} \sum_{j \in V-i} c_{ij} x_{ij}$$

$$\text{Subject To} \quad \sum_{j \in V-i} x_{ij} = 1 \quad (\forall i \in V)$$

$$\sum_{i \in V-j} x_{ij} = 1 \quad (\forall j \in V)$$

$$\sum_{i \in S} \sum_{j \in V-S} x_{ij} \geq 1 \quad (\forall S \subset V \quad , S \neq V, \phi)$$

$$x_{ij} \in \{1,0\} \quad (\forall i, \forall j \in V)$$

(AP)

$$\text{Minimize} \quad \sum_{i \in V} \sum_{j \in V-i} c_{ij} x_{ij}$$

$$\text{Subject To} \quad \sum_{j \in V-i} x_{ij} = 1 \quad (\forall i \in V)$$

$$\sum_{i \in V-j} x_{ij} = 1 \quad (\forall j \in V)$$

$$x_{ij} \geq 0 \quad (\forall i, \forall j \in V)$$

3．ATSP の厳密解法

都市数や費用をランダムに生成した ATSP の問題は最適値が未知であるため，近似解を評価する場合，厳密解が必要になる． ATSP の厳密解を求めるには，本質的に全列挙操作を含まざるを得ないと信じられている．そこで，不要な列挙を特定し省略してできるだけ能率的な列挙を行なう分枝限定法の枠組みに注目し，それを用いた ATSP の厳密解を求めるプログラムを構築する．

3．1．分枝限定法を用いた ATSP の厳密解法

分枝限定法には分枝操作と限定操作がある．分枝操作は元の問題の特定の変数を固定して子問題に分けることである．限定操作は緩和問題で得られた解が部分巡回路であり，緩和問題の下界値を z_i，暫定値を z' としたとき，$z_i \geqq z'$ となっている場合に分枝操作を停止することである．ただし，下界値とは最適値が取る可能性のある最小の目的関数値であり，暫定値とは過去に求まった最良の実行可能解に対応する目的関数 z の値である．限定操作を行なうには下界値が必要となるが，その下界値は緩和問題を解くことで簡単に得られる．ATSPの場合，緩和問題は割当問題（Assignment Problem ： AP）となるので，例えば逐次最短路法などを用いて能率よく解くことができる．APの定式化は上記（AP）の通りである．APはATSPにおける部分巡回路除去制約を除いたものであるため，一般に部分巡回路を含む．このような分枝・限定操作を繰り返し，巡回路が得られたときその下界値が暫定値より小さければ暫定値を更新する．その後再び未分枝の問題を選び分枝していき，総費用が最小となる巡回路を探索していく．この操作を未分枝の問題がなくなるまで行ない，そのときの暫定値が最適値となる．

3．2． 実装の概略

作成したプログラムの実装における詳細を次に説明する．

初期暫定値　　限定操作に必要な初期暫定値は近似解法（3-opt法）を 500 回繰り返した中で最小となる値とした．そのことにより，最適値に近い値で下界値と比較することになるため，早い段階から限定操作が行なわれる可能性が高くなり実行時間の短縮につながることを期待した．

緩和問題の解き方

前述のように緩和問題は逐次最短路法を用いて解いた．そのプログラム[2]は研究室にある既存のものを用いた．

分枝操作　　APを解いて部分巡回路が含まれている場合，部分巡回路を構成する枝の中から１本の枝（p,q）を選び，その枝を使う・使わないによって問題を右子問題・左子問題に分枝する．これは実行可能領域を２つに分けることになる．右子問題は，それ以降枝（p,q）を必ず使うとした ATSP，左子問題は，枝（p,q）を使わないとした ATSP になる．

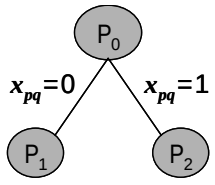

図２　分枝操作

その際，使う枝を X の集合に，使わない枝を Y の集合に含めるものとし，それぞれの問題ごとにその情報を保存する．子問題は親問題の情報を引き継ぎ，それに新たな情報を加えていく．

枝の選び方　　問題を左右の子問題に分枝するとき，左子問題の最適値の下界値がなるべく大きくなるように枝を選んだ．こうすると，左子問題の目的関数の劣化度が大きくなり，左子問題が早い段階で見切られる可能性が大きくなる[1].

問題プール　　分枝してできた問題をためておく場所を問題プールと呼ぶことにする．それぞれの問題には，X ， Yの情報やその問題の下界値，緩和問題の解を保存する．ただし，一度分枝した問題は問題プールから削除する。

費用の書き換え　　枝ijを使う場合，c_{ji}を∞とし，都市 i から出ていく ij 以外の枝の費用と都市 j に入ってくる ij 以外の枝の費用を全て∞とする．こうすることで枝 ij が必ず選ばれることになる．逆に枝 ij を使わない場合は，c_{ij}を∞とすることで枝ijが選ばれないことになる．

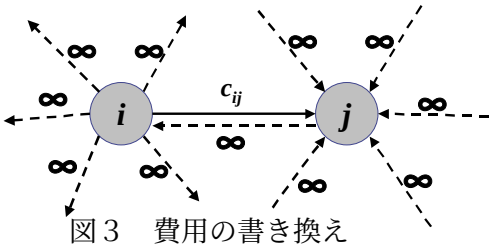

図３　費用の書き換え

限定操作をした問題の処理　　限定操作をした場合，それ以降は最適値が見つからないためその問題を問題プールから削除する．そのことで無駄な問題を比較することがなくなり，計算時間を短縮することになる．

未分枝の問題の選択　　巡回路となる解が見つかった場合，次に分枝を始める未分枝の問題を選ばなければならない. その際, 問題プールに残っている問題の中から下界値が一番小さい問題を選び, その問題から分枝を開始し巡回路を探索するものとした.

３．３．提案する処理方法

厳密解法では最適値は求まるが，問題によっては最適値が得られるまでに非常に時間がかかる．そこで実行時間が一定時間を超えても最適値が得られない場合，必ず巡回路を得られるようにするために次のような処理方法を提案する．実行時間が 60 秒（1 分）を超えたとき分枝操作を中止し，その時点で残っている未分枝の問題の中から下界値の小さい順に 100 個の問題を選ぶ．そしてそれぞれの問題で部分巡回路があればそれらをつなげて巡回路を作る．部分巡回路を巡回路にする方法は分枝限定法における分枝操作を用いる．選ばれた枝を必ず使うとする右子問題を分枝していき巡回路が得られるまで分枝を行なう．このようにして得られた 100 個の巡回路の中で最小となる巡回路の総費用と暫定値を比較し，値の小さい方を最適値とした．

４．実験結果と考察

どのくらいの都市数の問題が，どのくらいの計算時間で厳密解を導き出すのかを確認するため，作成したプログラムにデータを入力，実行し ATSP の厳密解を求める．また，提案した処理方法の性能評価を行なう．実験の結果，それぞれのデータの都市数，得られた最適値，3-opt 法により得られた初期暫定値，提案した処理方法により得られた値，生成された子問題数，計算時間を下記の表１に示す．データは TSPLIB のサイトにある ATSP のデータを使用した．また，費用をランダムに発生させ都市数 50，100，200，300 それぞれの問題の厳密解を求め，その結果を表２に示す．プログラムは Boland 社の Delphi6 を用いて作成した．計算時間の単位は sec，使用した計算機は Windows XP Home Edition，CPU は Intel Pentium4　3.06GHz，メモリ 512MB である．

実験の結果，ベンチマーク問題については都市数が多くても短い計算時間で厳密解が求まるものや，都市数が少なくても計算時間が長くなるものもあることがわかった．よって，計算時間は都市数によらず，費用によって著しく異なるといえる．また，生成子問題数は計算時間に比例していないこともわかる．しかし，実行中に問題プールに残しておく未分枝の問題数は計算時間に直接関係してくると考えられる．これは問題プールに残る問題数が多くなれば比較や計算に時間がかかるためである．

表１　実験結果（ベンチマーク問題）

データ名	都市数	最適値	初期暫定値	提案した処理方法による値	生成子問題数	計算時間 [秒]
br17	17	39	39	40	1643594	762 (約13分)
ftv33	34	1286	1286	－	4482	9
ftv35	36	1473	1473	－	3150	7
ftv38	39	1530	1532	－	2454	7
p43	43	5620	5620	5637	428752	1486 (約24分)
ftv44	45	1613	1613	－	1958	8
ftv47	48	1776	1776	1784	15090	73
ry48p	48	14422	14466	14884	693588	3518 (約59分)
ft53	53	6905	6905	7188	498900	3234 (約54分)
ftv55	56	1608	1608	1629	71970	534（約9分）
ftv64	65	1839	1854	1851	142428	1633（約27分）
ft70	70	38673	38721	38997	190282	2691（約45分）
ftv70	71	1950	1966	2077	619424	9276(約2時間35分)

費用をランダムに発生させた問題では，300 都市までの問題は厳密解を求めることができた．結果より，ベンチマーク問題と費用をランダムに発生させた問題とで計算時間が大きく異なるが，その要因は，費用行列の特徴にあると思われる．これはベンチマーク問題の費用行列は同じ行や列に同じ値のものがある，つまり費用の同じ枝が多くあるため，目的関数値が似通った未分枝の問題が多くなり，計算に時間がかかると考えられるからである．

表２　実験結果（ランダム）

都市数	回数	最適値	初期暫定値	計算時間[秒]
50	1	173	193	2
	2	199	209	3
	3	200	216	5
100	1	246	246	26
	2	204	249	56
	3	224	257	67
200	1	275	357	626
	2	370	370	327
	3	287	364	737
300	1	331	433	2042
	2	343	446	1128
	3	331	440	989

5．まとめ

本研究では，分枝限定法を用いた ATSP の厳密解を求めるプログラムを作成した．これによって近似解法を評価する基準となるものが得られ,研究室の近似解法の研究に役立つことが期待される．そしてこの実験により，ベンチマーク問題集において現在知られている最良値が最適値であることがわかった．また本研究では時間に制約をかけた処理方法を提案した．この処理方法により得られた値を最適値と比較するとほぼ等倍～1.1 倍であることから，計算時間を考慮すると良い結果が得られたと考えられる．

提案した処理方法では時間に制約をかけた後，部分巡回路を巡回路にする方法として分枝限定法における分枝操作を用いたが，この操作に他の方法を用いた場合と精度を比較検討することが望まれる．また厳密解法においては分枝し生成された子問題の格納方法を改善し，大きな規模の問題を解くこと，計算時間の高速化が重要な課題である．

6．参考文献

[1]今野 浩，鈴木 久敏：整数計画法と組合せ最適化，日科技連，1982． [2]吉村 央紀：「非対称巡回セールスマン問題に対する近似解法の提案」平成 14 年度東京理科大学大学院工学研究科経営工学専攻修士論文，2003． [3]升岡 慎吾:「非対称巡回セールスマン問題における近似解法の評価」平成 14 年度東京理科大学工学部経営工学科卒業論文，2003．
[4]松井 知己：「組合せ最適化問題入門」**http://www.misojiro.t.u-tokyo.ac.jp/~tomomi/comb-opt/intro.ppt　P 17,18**
[5]杉原 厚吉，茨木 俊秀，浅野 孝夫，山下 雅史：アルゴリズム工学-計算困難問題への挑戦-，共立出版,2001．