

汎用 MIP ソルバによる 巡回セールスマン問題の求解 —多項式オーダ本数の部分巡回路除去制約—

沼田 一道

1. はじめに

巡回セールスマン問題 (Traveling Salesman Problem; TSP) を汎用の混合整数計画問題ソルバ (以下, MIP ソルバ) で手軽に解く方法を紹介する。もちろん小規模 (100 都市程度まで) の問題しか解けないが、厳密解が求まる。厳密解といえば、多面体アプローチによる分枝カット法が他の追随を許さないレベルで最強である。それを実装した TSP の専用ソルバも公開されているが、それは (大規模の) 純粋な TSP の求解専用であり、OR 的な諸制約を考慮した巡回活動の最適化に使おうとすると敷居が高い。そのような場合には、プログラムレスで興味ある問題を手軽に扱える汎用 MIP ソルバにそのまま入力できるモデルにも意味がある。この種の定式化については近年の参考書・文献等ではあまり取り上げられていないので、ここで改めて紹介する次第である。まず、部分巡回路除去制約を多項式オーダの制約式で与える定式化を示し、それを用いて TSPLIB[11] の問題例をいくつか解いた結果を示す。解情報利用 (釘付けテスト) の可能性についても触れる。

2. TSP とその解法

TSP (巡回セールスマン問題) とは、「訪問すべき点集合と点間の移動コストが与えられたとき、すべての点を 1 度ずつ訪問して出発点に戻る巡回路 (ハミルトニアン閉路) の中で、総移動コスト (移動する枝の長さの和) が最小のものを求める」問題である。組合せ最適化分野の代表的問題として、さまざまなアプローチによる解法の研究が行われてきた[7][1][13]。巡

ぬまた かずみち
東京理科大学 工学部
〒162-8601 新宿区神楽坂 1-3

回活動を伴う実際問題への応用もさることながら、理解しやすい単純な問題であるのに、厳密な最適解を能率よく求める方法が知られていないことが人々を引き付ける理由の一つといわれている。以下では、都市間の移動コストが移動方向によらない「対称 TSP (Symmetric TSP; STSP)」を想定して話を進めるが、提示する方法は非対称な TSP (Asymmetric TSP; ATSP) にもそのまま適用可能である。

STSP 対する解法として有力なのは、Lin-Kernighan 法をベースにした発見的解法[8][9][6][2]と、多面体アプローチに基づく分枝カット法[12][1]である。例えばウェブサイト[3]には両解法のプログラム (Lin-kern と Concorde) が公開されている¹。

TSP は MIP ソルバが通常採用している手法では扱い難い構造を持つことが知られているので、専用解法の研究開発が盛んに行われ、上記ウェブサイト[3]の home page などで公開されている。しかしながら、TSP の変種である一方通行や複数巡回路を含む等の制約を持つ問題を解くには専用解法のプログラム変更が必要となる。これらの変更は、数理と実装技術に関する専門的な知識・経験がないと難しく、公開プログラム (TSP ソルバ) を独自の最適化問題に適用するのは容易ではない。よって、巡回活動を含む独自の最適化問題を解く際は、通常、問題に即した発見的解法を準備することになる。それ自体興味深い作業であるが、それなりの工夫とプログラミング技術を要するので、問題 (のみ) に興味のある人が手軽に行えるというほど自明な作業でもない。また、得られる解の精度 (真の最適解との相違) も気になる。

¹ このサイトでは、85,900 “都市” の STSP が Concorde で厳密に解かれたこと、100,000~200,000 “都市” の STSP 問題例およびそれに対する現時点での最良巡回路長などが報告されている。

このとき、相対的に小規模の問題例であっても、より手軽に厳密な最適解を求めることができると便利である。例えば、問題の記述（定式化）を直接汎用 MIP ソルバに入れプログラムレスで解を得られれば、問題そのものに専念できる。定評ある商用ソルバを用いれば、100 都市程度の巡回を含む問題は充分実用的な時間内で解けるし、付加的に加わる制約条件にも柔軟に対処することができる。

3. 部分巡回路除去制約

TSP を汎用 MIP ソルバで解くためには、問題の表現（定式化）を与えるなくてはならない。まず、訪問対象の n 個の点（都市）を $V = \{1, 2, \dots, n\}$ 、点間の移動の可能性を表す枝集合を $E = \{ij | i, j \in V, i \neq j\}$ 、枝 ij の長さを d_{ij} とする。STSP については、枝 ij と ji を区別せず（同一視し）、 $n(n-1)/2$ 本の無向枝の採否で巡回路を表現することが多いが²、ここでは、セールスマントラベルスマンが枝 ij 上を i から j に移動する(1)か否(0)かを表す 0-1 決定変数 x_{ij} で巡回路を表現する。これは ATSP, STSP のいずれにも対応できる表現である。多くの文献は TSP の基本表現として、つぎと等価な定式化を与えており[7][13][1]、

$$\min. \sum_{i \in V} \sum_{j \in V \setminus \{i\}} d_{ij} x_{ij} \quad (1)$$

$$\text{s. t. } \sum_{h \in V \setminus \{i\}} x_{hi} = 1, \quad i = 1, 2, \dots, n, \quad (2)$$

$$\sum_{j \in V \setminus \{i\}} x_{ij} = 1, \quad i = 1, 2, \dots, n, \quad (3)$$

$$\sum_{i, j \in S} x_{ij} \leq |S| - 1, \quad (4)$$

$$x_{ij} \in \{0, 1\}, \quad i, j (\neq i) = 1, 2, \dots, n. \quad (5)$$

(1)式は、目的関数（巡回路長）の最小化を意味する。(2), (3)式は各点 i において、セールスマントラベルスマンが、 i 以外の点から枝上を移動して i に至り、 i から枝上を移動して i 以外の点へ移ることを制約している。すなわち、セールスマントラベルスマンの移動が巡回路を構成する必要条件である。しかし、これだけだと点の部分集合に対する巡回路の集まりでも条件を満たしてしまうので、(4)式によりそのような部分巡回路の存在を禁止する。これが部分巡回路除去制約であり、 $2^n - n - 2$ 本の不等式からなる。この部分巡回路除去制約は、無向枝変数を用いた STSP 定式化の場合 $\mathbf{R}^{\frac{n(n-1)}{2}}$ における STSP 多面体²のファセットを構成し多面体アプローチの出発点となるものであるが、組合せ数オーダーの制約式はこのままでは扱えない。汎用 MIP ソルバに入力するため

には、コンパクトな（多項式オーダーの制約式からなる）定式化が必要となる。

既存のコンパクトな表現として、Miller ら[10]によるものがある。これは、部分巡回路が存在すると“不能な”不等式が導かれるような重みを各点に割り付けながら（矛盾が生じないように枝を選びながら）、巡回路長を最小化するものである。この定式化では（暗黙のうちに）巡回路の方向を考慮するので、各枝を有向として扱い、 x_{ij} と x_{ji} の両方を用いる。まず、部分巡回路ができる場合には、(2), (3)式の制約から、任意の特定の点（例えば点“1”）を含まない部分巡回路が（少なくとも 1 つは）できることに注意する。そして、点 1 (home base) 以外の点 i に割り付ける値（重み）を変数 (w_i) として導入する。この w_i を枝の選択変数 x_{ij} と関係させて次式のように制約する。

$$w_i - w_j + nx_{ij} \leq n - 1, \quad i, j (\neq i) = 2, 3, \dots, n. \quad (6)$$

構成点数 k の部分巡回路が存在するとき、その構成枝 ($x_{ij}=1$) に沿って(6)式を辺々加え合わせると、 $nk \leq (n-1)k$ となって矛盾する。この制約式を用いた TSP の定式化 (MTZ) は以下のようになる、

$$(MTZ) \quad \begin{aligned} & \min. \sum_{i \in V} \sum_{j \in V \setminus \{i\}} d_{ij} x_{ij} \\ & \text{s. t. } \sum_{h \in V \setminus \{i\}} x_{hi} = 1, \quad i = 1, 2, \dots, n, \\ & \quad \sum_{j \in V \setminus \{i\}} x_{ij} = 1, \quad i = 1, 2, \dots, n, \\ & \quad x_{ij} \in \{0, 1\}, \quad i, j (\neq i) = 1, 2, \dots, n, \\ & \quad w_i \in \mathbf{R}, \quad i = 2, 3, \dots, n. \end{aligned} \quad (6)$$

つぎに示す部分巡回路除去制約は、Gavish and Graves[4]で提案された flow formulation と呼ばれる制約である。(MTZ) と同様、点 1 を特別な点“デポ”とする。そして、デポを出発したセールスマントラベルスマンが $2, 3, \dots, n$ の各点から 1 単位ずつのモノを回収し、デポに $n-1$ 単位のものを持ち帰る収集経路問題を考える。0-1 変数 x_{ij} でセールスマントラベルスマンが点 i から点 j へ直接移動する(1)か否(0)かを表す。また変数 y_{ij} で点 i から j へ移動中のセールスマントラベルスマンが保持しているモノの単位数を表す。移動の向きを考慮するので、 x_{ij} と y_{ij} はそれぞれ $n(n-1)$ 個の変数となる。 i から j へ移動が行われない場合の y_{ij} は 0 とする。また、 y_{ij} は $n-1$ を超えない、

$$y_{ij} \leq (n-1)x_{ij}, \quad i, j (\neq i) = 1, 2, \dots, n. \quad (7)$$

² 正しい巡回路をあらわす $\frac{n(n-1)}{2}$ 次元 0-1 ベクトル $(x_{12}, \dots, x_{ij}, \dots, x_{n-1n})$ をすべて含む最小の凸多面体。

1以外の点を通過するとセールスマンの保持しているモノの量は1単位増える,

$$\sum_{j \in V \setminus \{i\}} y_{ij} - \sum_{h \in V \setminus \{i\}} y_{hi} = 1, \quad i=2, \dots, n. \quad (8)$$

デボを通過するときには、保持しているモノの量は $n-1$ 単位減る,

$$\sum_{j \in V \setminus \{1\}} y_{1j} - \sum_{h \in V \setminus \{1\}} y_{h1} = -(n-1). \quad (9)$$

また、デボを出るときの保持量を“0”とする,

$$\sum_{j=2}^n y_{1j} = 0. \quad (10)$$

以上の設定の下で、点1を通らない部分巡回路は変数 y_{ij} に反映されたモノの保存則により禁止される。この制約式を用いた TSP の定式化 (ff) は以下のようになる,

$$(ff) \left| \begin{array}{l} \min. \sum_{i \in V} \sum_{j \in V \setminus \{i\}} d_{ij} x_{ij} \\ \text{s. t. } \sum_{h \in V \setminus \{i\}} x_{hi} = 1, \quad i=1, 2, \dots, n, \\ \sum_{j \in V \setminus \{i\}} x_{ij} = 1, \quad i=1, 2, \dots, n, \\ (7), (8), (9), (10) \\ x_{ij} \in \{0, 1\}, y_{ij} \geq 0, \\ i, j (\neq i) = 1, 2, \dots, n. \end{array} \right.$$

4. 求解実験

前節で示した定式化 (MTZ) と (ff) を用いて TSPLIB の問題例をいくつか解いた。計算にはノート PC (Core2Duo : 2 GHz, RAM : 1 GB, OS : Windows/Xp) と汎用 MIP ソルバ gurobi (version 4.00) を使用した。gurobi は学術用無償版[5]を標準的にインストールしたものを利用した³。

表1のi-name欄に問題例（いずれも STSP）の名称を示す。名称末尾の2~3桁の数字は、各問題例の

表1 定式化による求解時間の違い

i-name	MTZ	ff	ff_peg
att48	1395	153	42 (53%)
eil51	5	12	26 (72%)
pr76	34774	1378	1143 (26%)
kroA100	—	1927	1202 (33%)
eil101	613	95	69 (81%)
ch150	—	1863	1648 (67%)
kroA200	—	17393	92615 (28%)
gil262	—	—	— (13%)

³ 現在は有償版 (version 4.01) を購入している。

都市数 (n) を表している。MTZ, ff の列には、それぞれの定式化を用いた場合の求解に要した時間(秒)を示す。“—”はメモリ不足による例外終了を表す。

ff_peg の列は、(ff) の定式化で釘付けテストにより問題サイズを縮小した場合の所要時間(秒；釘付けテストに要した時間は除く)である。「釘付けテスト」とは、最適性を損なわずに、0-1 計画問題のサイズを縮小する一般的な方法である。詳細については、例えば、文献[12]の 96~97 頁などを参照されたい。釘付けテストの際の上界値としては最適値を用いた⁴。カッコ内は釘付けされた変数の割合である（すべて“0”へ固定）。

2つの定式化のサイズを比べると、(MTZ) が (変数: n^2-1 , 制約式: $2n^2-2n+2$), (ff) が (変数: $2n^2-2n$, 制約式: $2n^2+n+1$) で、(ff) の方が若干大きい。しかし、所要時間は、eil51 を除いて、(ff) の方が断然小さい⁵。これは、gurobi の自動的カット生成機能が、抽象的な（訪問順序の整合性に関係する）変数 w_i を用いた (MTZ) の制約条件より、意味（構造）のある変数 y_{ij} を用いた (ff) の制約条件に対しての方が、有効に働くからだと思われる。実際、自動生成されたカットの個数は (ff) の方が格段に多い。

釘付けテストの効果は想像したほど大きくない。(0に) 固定される変数 (x_{ij}) はかなりの割合に達するが、計算時間はあまり短縮されない。サイズが小さくなる反面、“分枝限定の意味”で難しくなるのではないかと思われる⁶。ただし、釘付けテストの性能は、扱う問題のデータに大きく依存するので、注意が必要である。

5. まとめ

本報告では、汎用 MIP ソルバを用いて TSP を解くための定式化を与え、200 都市程度までの TSPLIB の問題例をいくつか解いた。実験の結果は、100 都市程度までの問題であれば、許容時間内に厳密な最適解が求められることを示している。この求解能力はソルバの性能に負うものであるが、定式化によっても求解時

⁴ この程度のサイズの問題例だと、多くの場合、多スタート Lin-Kernighan 法などで最適値が求まる。

⁵ eil51 の場合は逆である。このとき、釘付け/問題縮小により所要時間が増えるという現象も起きている。短時間で解ける“易しい”問題なので、実行時間に対する分枝/子問題選択等の影響が強く出た偶然の結果だと推測する。

⁶ eil51 と kroA200 の場合には逆転している。

間は大きく変わる。

本報告では純粋な TSP を対象としたが、同様の定式化は巡回活動を含む多くの最適化問題に適用可能である。さまざまな制約を有する独自の問題をとりあえず考察する場合などには本報告のアプローチが役に立つと考える。制約が複雑になると、求解時間が減る可能性もある。また、実験ではソルバ動作の調整（チューニング）は一切行ってないが、それを行うことにより求解時間が改善される可能性もある。

参考文献

- [1] D. Applegate, R.E. Bixby, V. Chvátal and W.J. Cook : *The Traveling Salesman Problem—A Computational Study*, Princeton University Press, 2006.
- [2] D. Applegate, W.J. Cook and A. Rohe : Chained Lin-Kernighan for large traveling salesman problems ; *INFORMS Journal on Computing*, Vol. 15, pp. 82–92 (2003).
- [3] Concorde TSP Solver : <http://www.tsp.gatech.edu/concorde.html> (最終閲覧日 2011/3/1).
- [4] B. Gavish and S.C. Graves : The traveling salesman problem and related problems ; Working Paper OR-078-78, Operations Research Center, MIT, Cambridge, MA (1978).
- [5] Gurobi Optimizer 4.0 : <http://www.gurobi.com/html/> academic.html (最終閲覧日 2011/3/1).
- [6] K. Helsgaun : An effective implementation of the Lin-Kernighan traveling salesman heuristic ; *European Journal of Operational Research*, Vol. 126, pp. 106–130 (2000).
- [7] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys (ed.) : *The Traveling Salesman Problem*, JOHN WILEY & SONS, 1985.
- [8] S. Lin and B.W. Kernighan : An effective heuristic algorithm for the traveling-salesman problem ; *Operations Research*, Vol. 21, pp. 498–516 (1973).
- [9] K.-T. Mak and A.J. Morton : A modified Lin-Kernighan traveling-salesman heuristic ; *Operations Research Letters*, Vol. 13, pp. 127–132 (1993).
- [10] C.E. Miller, A.W. Tucker and R.A. Zemlin : Integer programming formulations and travelling salesman problems ; *Journal of the ACM*, Vol. 7, pp. 326–329 (1960).
- [11] TSPLIB : <http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/> (最終閲覧日 2011/3/1).
- [12] 茨木俊秀, 福島雅夫 : 最適化の手法, 情報数学講座 14, 共立出版, 1993.
- [13] 山本芳嗣, 久保幹雄 : 巡回セールスマン問題への招待, 朝倉書店, 1997.