

# 正則グラフの巡回セールスマン問題を解くアルゴリズムの時間計算量

## Time complexities of algorithms for the traveling salesman problem in regular graphs

数学専攻 山本 浩之  
Hiroyuki YAMAMOTO

### 1 はじめに

巡回セールスマン問題 (Traveling Salesman Problem) は, グラフの辺に非負の重みが与えられたとき, すべての頂点を, ちょうど一度ずつ通るハミルトンサイクルのうち, 辺の重みの合計が最小となるものを求める組み合わせ最適化問題のひとつであり, TSP と略される [12].

2007 年に David Eppstein は, 3-正則グラフに対する TSP の時間計算量を  $O(1.260^n)$  に改善する *Forced TSP* アルゴリズムと 4-正則グラフに対する TSP の時間計算量を  $O(1.890^n)$  に改善する確率的アルゴリズムを発表した [8]. その後, 2008 年に Heidi Gebauer により, 4-正則グラフに対する TSP の時間計算量を  $O(1.733^n)$  に改善する *MinHC* アルゴリズムが発表された [9], [10]. また, 2014 年に Maciej Li'skiewicz–Martin R. Schuster によって, 3-正則グラフに対する TSP の時間計算量は  $O(1.2553^n)$  に改善された [13]. 本論文では, 次数が 3 以上の正則グラフに対する TSP の計算時間量を改善する各アルゴリズムを紹介するとともに, 最大次数が 4 である入力グラフのサイズを減らす変形を試みた自身の結果を述べる.

### 2 用語の定義

グラフ (graph)  $G$  とは, 集合  $V$ ,  $E$  とひとつの関数  $\psi$  の 3 つの組  $(V, E, \psi)$  をいう.  $V$  の元を  $G$  の頂点 (vertex) とよび,  $G$  の頂点の個数をその位数 (order) という.  $E$  は  $V$  とは互いに素な集合であり, その元を  $G$  の辺 (edge) という. 両端点と同じ辺をループ (loop), 同じ 2 頂点を接続している複数の辺を平行辺 (parallel edges) という.  $\psi$  は接続関数 (incidence function) とよばれ,  $G$  の各辺に  $G$  の頂点の対を対応させる.  $G$  の各辺  $e$  にひとつの非負の実数を対応させて,  $e$  の重み (weight) という.

頂点  $v$  に接続している辺の本数を  $v$  の次数 (degree) という. ループが接続しているときは, 1 本あたり 2 と数える. また, すべての頂点の次数が  $k$  であるグラフを  $k$ -正則グラフ ( $k$ -regular graph) という. 頂点と辺が交互に並んだ列をウォーク (walk) という. ウォークに含まれている辺の本数をそのウォークの長さ (length) といい, 2 個の頂点を結ぶ最小のウォークの長さをそれら 2 点の距離 (distance) という. すべての頂点が異なるウォークをパス (path) という. また, 始点と終点とが一致するパスをサイクル (cycle) とよび, 長さが  $k$  のサイクルを  $k$ -サイクル ( $k$ -cycle) という. 特にグラフのすべての頂点をちょうど 1 回だけ通るサイクルはハミルトンサイクル (Hamilton cycle) という.

### 3 最大次数が 3 のときの TSP の時間計算量

2007 年に Eppstein によって, 最大次数が 3 の重み付き  $n$  頂点グラフがアルゴリズムに入力されたときの TSP の時間計算量を,  $O(2^{\frac{2}{3}n}) \approx O(1.260^n)$  に改善する, *Forced TSP* アルゴリズムが発表された [8]. このアルゴリズムでは, *forced edge* とよばれる要素をもつ, グラフの辺集合の部分集合  $F$  を考え, 重み付きグラフと  $F$  を入力したとき,  $F$  のすべての要素を含むようなハミルトンサイクルの最小の重みを出力する. はじめに図 1 のような変形を行って入力グラフのサイズを減らした後, 一回の再帰呼び出し時に問題

を 2 つの *subproblem* に分割して再帰的に解くことで、時間計算量の改善を図っている。その後、2014 年に Li'skiewicz-Schuster によって、問題の分割の際、時間計算量に影響を及ぼす最悪のケースを減らすことで、時間計算量は  $O(2^{\frac{1219}{3717}n}) \approx O(1.2553^n)$  に改善された [13].

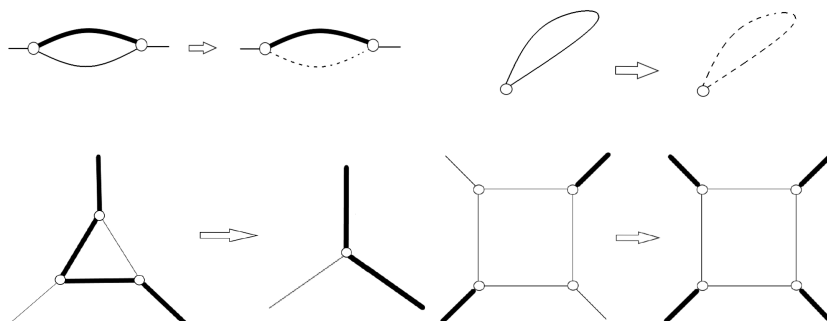


図 1 入力グラフに対する変形の例. *forced edge* を太線で示している.

#### 4 Eppstein による最大次数が 4 のときの TSP の時間計算量の改善

2007 年に Eppstein によって、最大次数が 4 の重み付き  $n$  頂点グラフがアルゴリズムに入力されたときの TSP の時間計算量を、 $O((\frac{3}{2})^n 2^{\frac{n}{3}}) \approx O(1.890^n)$  に改善する、モンテカルロ法による確率的アルゴリズムが発表された [8]. このアルゴリズムでは、次数が 4 の頂点に対して分割処理を行い、最大次数が 3 のグラフに変形した後、表 1 の *Forced TSP* アルゴリズムに適用することで、時間計算量の改善を図っている.

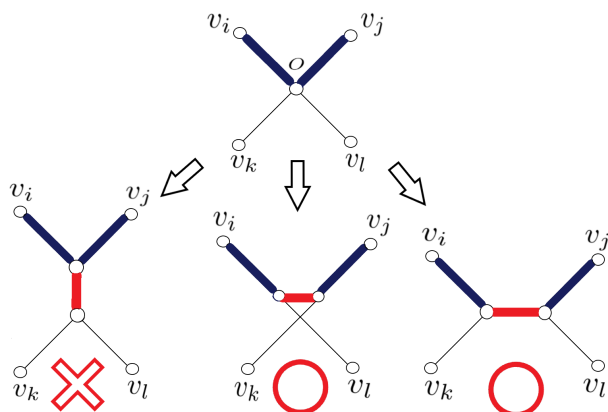


図 2 次数が 4 の頂点  $O$  に対する分割処理の例. 解となるパスが保たれる確率は  $2/3$ .

図 2 のように、次数が 4 の頂点すべてに分割処理を行った後、最適解を見つけられる確率を求めて時間計算量の改善を図っている。アルゴリズムを  $(\frac{3}{2})^f$  回繰り返せば、最適解が見つかる回数の期待値は 1 となるので、 $f \leq n$  より、時間計算量は  $O((\frac{3}{2})^n 2^{\frac{n}{3}}) \approx O(1.890^n)$  となる。さらに、Li'skiewicz-Schuster によって改良された *Forced TSP* アルゴリズムを適用することで、時間計算量を  $O((\frac{3}{2})^n 2^{\frac{1219}{3717}n}) \approx O(1.8829^n)$  に改善できる。

## 5 最大次数が 4 の入力グラフに対する変形

*Forced TSP* アルゴリズムでは最大次数が 3 の入力グラフに対する変形を行うが、特定のパターンが含まれている最大次数が 4 の入力グラフに対しても変形を行うことができると考え、そのサイズを減らすことを試みた、自身の結果について述べる。

*Observation 5.1* 頂点の次数が 3, 3, 4 から成る三角形で、1 個以上の次数が 3 の頂点が  $F$  に含まれている非三角形の辺をもち、かつ次数が 4 の頂点が  $F$  に含まれている非三角形の辺を 1 本以上もつとき、次数が 4 の頂点の向かい側の三角形の辺を  $F$  に追加し、 $F$  に含まれている非三角形の辺をもつ次数が 3 の頂点と次数が 4 の頂点を接続している三角形の辺を取り除く (図 3 左).

*Observation 5.2* 頂点の次数が 3, 4, 4 から成る三角形で、次数が 3 の頂点の向かい側の三角形の辺が  $F$  に含まれているとき、次数が 3 の頂点と接続している非三角形の辺を  $F$  に追加する (図 3 中央).

*Observation 5.3* *unforced edge* から成る 4-サイクルで、対角に向かい合う 2 個の頂点の次数が 3 同士と 4 同士であり、次数が 4 の頂点の外側で隣接する辺が 1 本ずつ  $F$  に含まれているとき、次数が 3 の頂点の外側で隣接する *unforced edge* をすべて  $F$  に追加し、次数が 4 の頂点の外側で隣接する *unforced edge* をすべて取り除く (図 3 右).

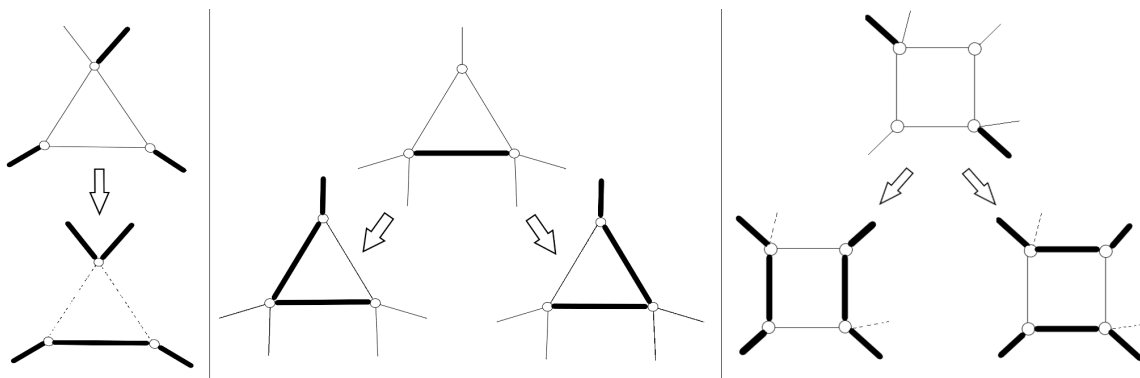


図 3 *Observation 5.1–5.3* の例

## 6 Gebauer による最大次数が 4 のときの TSP の時間計算量の改善

2008 年に Gebauer によって、 $n$  が偶数である最大次数が 4 の重み付き  $n$  頂点グラフがアルゴリズムに入力されたときの TSP の時間計算量を、 $O(3^{\frac{n}{2}} \cdot \text{poly}(n)) \approx O(1.733^n)$  に改善する、*MinHC* アルゴリズムが発表された [9]。このアルゴリズムでは、与えられた 2 個の頂点  $v_1, v_m (2 \leq m \leq n)$  に対して、最小の重み付きサイクルである  $(v_1, v_m)$ -サイクルを見つけることを考える。

アルゴリズムの最初の処理では、頂点  $v_1$  から出発して、まだ訪れていないすべての近傍の各頂点を再帰的に訪問していくことで、長さ  $\frac{n}{2}$  のパスを得るためのアルゴリズムとして考えることができる。2 頂点のペア  $(v_1, v_m)$  の組み合わせは  $n - 1$  通りなので、 $(v_1, v_m)$ -パスの長さは  $\frac{n}{2}$  より、パスに含まれる頂点

の個数は  $\frac{n}{2} + 1$  個となる。そのため、 $n$  の多項式を  $poly(n)$  で表したとき、 $(v_1, v_m)$ -パスの本数は高々  $(n-1) \cdot 4 \cdot 3^{\frac{n}{2}-1} \approx 3^{\frac{n}{2}} \cdot poly(n)$  本であり、 $PL$  に納められる要素の最大の個数も同じ式で制限できる。したがって、Step 1 の時間計算量は  $O(3^{\frac{n}{2}} \cdot poly(n))$  である。

その後の処理では、 $PL$  に対するソートと探索の実行にとどまるため、アルゴリズム全体の時間計算量には影響が及ばない。したがって、アルゴリズム全体の時間計算量は、 $O(3^{\frac{n}{2}} \cdot poly(n)) \approx O(1.733^n)$  である。

また、 $n$  が奇数のときは、頂点  $v_1, v_m$  が距離  $\frac{n}{2}$  をもつ  $(v_1, v_m)$ -サイクルの代わりに、頂点  $v_i$  と辺  $e$  がひとつずつ与えられ、 $e$  の両端点から  $v_i$  までの距離がどちらも  $\frac{n-1}{2}$  をもつハミルトンサイクル、 $(v_i, e)$ -サイクルを考えることで、 $n$  が偶数のときと同様にアルゴリズムを実行できるため、時間計算量は変わらない。

## 7 最大次数が 5 以上のときの TSP の時間計算量

Gebauer の *MinHC* アルゴリズムに入力するグラフを  $k$ -正則グラフとしても、アルゴリズム自体に変更が発生しないため、4-正則グラフのときと同様に TSP の時間計算量を求めることができる。したがって、 $k$ -正則グラフの TSP の時間計算量は  $O((k-1)^{\frac{n}{2}} \cdot poly(n)) \approx O((k-1)^{\frac{n}{2}})$  で求められる。

## 参考文献

- [1] A. Björklund, T. Husfeldt, P. Kasaki, M. Koivisto. The Travelling Salesman Problem in Bounded Degree Graphs Proc. 35th International Colloquium on Automata, Languages, and Programming, Part I. Lecture Notes in Computer Science, vol. 5125, ICALP pp.198–209, 2008.
- [2] J. A. Bondy, U. S. R. Murty 著, 立花 俊一, 奈良 知恵, 田澤 新成 共訳. グラフ理論への入門 共立出版, 1991.
- [3] J. A. Bondy, U. S. R. Murty. Graph Theory. Springer, 2000.
- [4] W. J. Cook 著, 松浦 俊輔 訳. 驚きの数学 巡回セールスマン問題 青土社, 2013.
- [5] S. Dasgupta, C. H. Papadimitriou, U. V. Vazirani. Algorithms. McGraw-Hill, 2006.
- [6] R. Diestel. Graph Theory. Springer, 2000.
- [7] R. Diestel 著, 根上 生也, 太田 克弘 訳. グラフ理論, 丸善出版, 2012.
- [8] D. Eppstein. The Traveling Salesman Problem for Cubic Graphs. Journal of Graph Algorithms and Applications. <http://jgaa.info/> vol. 11, no. 1, pp.61-81, 2007.
- [9] H. Gebauer. On the number of Hamilton cycles in bounded degree graphs, in: Proc. 5th Workshop on Analytic Algorithmics and Combinatorics, ANALCO, pp.241-248, 2008.
- [10] H. Gebauer. Finding and enumerating Hamilton cycles in 4-regular graphs. Theoretical Computer Science 412(35) pp.4579–4591, 2011.
- [11] K. Iwama, T. Nakashima. An Improved Exact Algorithm for Cubic Graph TSP. International Computing and Combinatorics Conference COCOON 2007: Computing and Combinatorics pp.108–117, 2007.
- [12] 久保 幹雄 著. 巡回セールスマン問題への招待 I. オペレーションズ・リサーチ, 39. pp.25–31, 1994.
- [13] M. Li'skiewicz, M. R. Schuster. A new upper bound for the traveling salesman problem in cubic graphs: J. Discrete Algorithms 27 pp.1–20, 2014.
- [14] S. Lipschutz 著, 成島 弘 訳. マグロウヒル大学演習 離散数学 コンピュータサイエンスの基礎数学, オーム社, 1995.