数理解析研究所講究録
第1809巻 2012年 24-33

ユークリッドＴＳＰの近似アルゴリズムについて

秋田大学・工学資源学研究科 石郷岡直輝 (Naoki Ishigouoka)
山村明弘 (Akihiro Yamamura)
Graduate school of Engineering and Resource Science
Akita University

概要

Aroraにより提案されたＴＳＰの近似アルゴリズムについて説明する。このアルゴリズムは与えられたノードを正方形の領域で分割しそれぞれで部分解を求める。次に一段階領域を大きくして部分解を求めるという動作を繰り返して近似解を求める。$(1+\epsilon)$-近似解が計算時間 $O(n^3(\log n)^c)$ で得られる。(ただし $0<\epsilon<1, c=O(\frac{1}{\epsilon})$) また Arora のアルゴリズムを拡張することを検討する。

1 導入

巡回セールスマン問題（ＴＳＰ:Traveling Salesman Problem）とは n 個のノードとすべてのノード間のコストが与えられたとき、コストの和が最小となるハミルトン閉路を求める問題である。この古典的な問題は重要なアルゴリズムの考え方として数十年の間研究され、オペレーションズリサーチや多面体理論、複雑系理論などの分野に影響を与えている。また1970年代にTSPはＮＰ-困難であることが示された。[9]

与えられるノードが三角不等式を満たす場合にはメトリックＴＳＰと呼ぶ。また２次元（または d 次元）上にノードが与えられノード間のコストにユークリッド距離が使われる場合、特にユークリッドＴＳＰと呼ぶ。ユークリッドＴＳＰはメトリックＴＳＰの特別な場合である。

ユークリッドＴＳＰもＮＰ-困難である。[5][11] そこである程度良い解である近似解を求める研究が進められた。Karp は最適値から $1+\epsilon$ 倍 ($1<\epsilon$である任意の定数) 以内のツアーを高い確率で求めるアルゴリズムを提案した。[10] また Christofides は多項式時間ですべてのメトリックＴＳＰのインスタンスに対して最適値の $\frac{3}{2}$ 倍以内のツアーを求める近似アルゴリズムを提案した。[3] 二十年間の研究によりメトリックＴＳＰに対する Christofides のアルゴリズムは改善されている。Held-Karp は $\frac{4}{3}$-近似アルゴリズムを提案している。[6]

ＰＴＡＳ (Polynomial-Time Approximation Scheme) は $1+\epsilon$ 倍 ($1<\epsilon$である任意の定数) 以内の近似解を多項式時間で計算可能なアルゴリズムのことをいう。ＰＴＡＳが存在する問題は非常に少ない。ＰＴＡＳの存在する問題として Subset-Sum[7] や Bin-Packing[4][8] が知られている。

Arora,Mitchell はユークリッドＴＳＰのＰＴＡＳを提案している。本論文では簡単に説明する。$\frac{1}{2}$の確率で $(1+\epsilon)$-近似解が計算時間 $O(n(\log n)^c)$ で得られる。(ただし $0<\epsilon<1, c=O(\frac{1}{\epsilon})$) d 次元の場合は計算時間 $O(n(\log n)^{(O(\sqrt{d}c))^{d-1}})$ で得られる。また決定的アルゴリズムとすることが出来て二次元の場合、計算時間 $O(n^3(\log n)^c)$ で必ず近似解が得られる。ここで示すＰＴＡＳは領域を正方格子で区切った直線とツアーとの交差が定

数回となるようなツアーを動的計画法を用いて求める。正方格子の数は多くても $n\log n$ 個であり1つの正方格子に関して計算時間 $O((\log n)^{O(c)})$ であるのでこのアルゴリズムは $O(n(\log n)^{O(c)})$ で計算可能である。

2 準備

2.1 良い丸め性質について

定義 1 ユークリッドＴＳＰのインスタンス $V \subseteq \mathbb{R}^2 (n = |V|)$ は 下記のすべての条件を満たすとき良い丸め性質をもつという。

(ただし $0 < \epsilon < 1, ||v - w||_2 = \sqrt{(v_x - w_x)^2 + (v_y - w_y)^2})$

(1) すべての $(v_x, v_y) \in V$ に対して、v_x, v_y は奇整数である。

(2)$max_{v,w \in V} ||v - w||_2 \leq \frac{64n}{\epsilon} + 16$

(3)$min_{v,w \in V, v \neq w} ||v - w||_2 \geq 8$

以下の命題が成立することが知られているので、以降良い丸め性質をもつインスタンスのみを扱う。

命題 1 良い丸め性質に限定したユークリッドＴＳＰに対する近似アルゴリズムがあるとする。すると、一般のユークリッドＴＳＰに対する近似アルゴリズムが存在する。

2.2 シフトしたグリッドを定義

定義 2 $L = max_{v,w \in V} ||v - w||_2, N = \lceil \log L \rceil + 1$ と定義すると一般性を失うことなくすべての座標は $[0, 2^N] \times [0, 2^N]$ の正方形のなかにあると仮定できる。この正方形を正方格子に分割することを以下のように繰り返す。これをシフトしたグリッドと定義する。$i = 1, ..., N-1$ に対して $G_i^{(a,b)} = X_i^{(b)} \cup Y_i^{(a)}$ とする。ただし $a, b \in \{0, 2, 4, ..., 2^N - 2\}$ は偶整数とする。

$$X_i^{(b)} = \left\{ \left[\left(0, (b + k2^{N-i}) \bmod 2^N\right), \left(2^N, (b + k2^{N-i}) \bmod 2^N\right) \right] | k = 0, 1, 2, ..., 2^i - 1 \right\}$$
$$Y_i^{(a)} = \left\{ \left[\left((a + j2^{N-i}) \bmod 2^N, 0\right), \left((a + j2^{N-i}) \bmod 2^N, 2^N\right) \right] | j = 0, 1, 2, ..., 2^i - 1 \right\}$$

（ただし $[(x, y), (x', y')]$ は (x, y) と (x', y') の間の線分を表す。）

なお $i = N-1$ の時は正方格子の間隔が2であり、a, b は偶整数であるため $G_{N-1} = G_{N-1}^{(a,b)}$ は a, b に依存しない。

線分 l は $l \in G_1^{(a,b)}$ のときレベル1にあるという。$l \in G_i^{(a,b)} \setminus G_{i-1}^{(a,b)} (i = 2, 3, 4, ..., N-1)$ のときはレベル i にあるという。

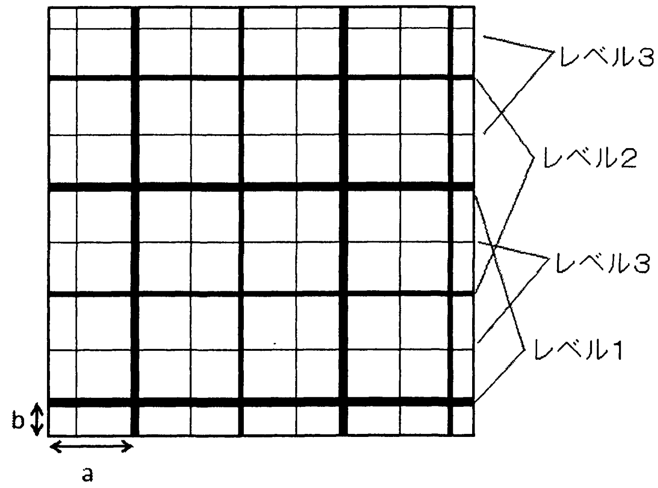

図1　シフトしたグリッド $G_i^{(a,b)}$

グリッド $G_i^{(a,b)}$ の領域は $j,k \in \{0,1,2,...,2^i-1\}$ に対する集合

$$\left\{(x,y) \in [0,2^N)\times[0,2^N) \middle| (x-a-j2^{N-i}) \bmod 2^N < 2^{N-i}, (y-b-k2^{N-i}) \bmod 2^N < 2^{N-i}\right\}$$

である。図2のように $i < N-1$ に対して $G_i^{(a,b)}$ は2個または4個の長方形からなる連結でない領域をとることもある。

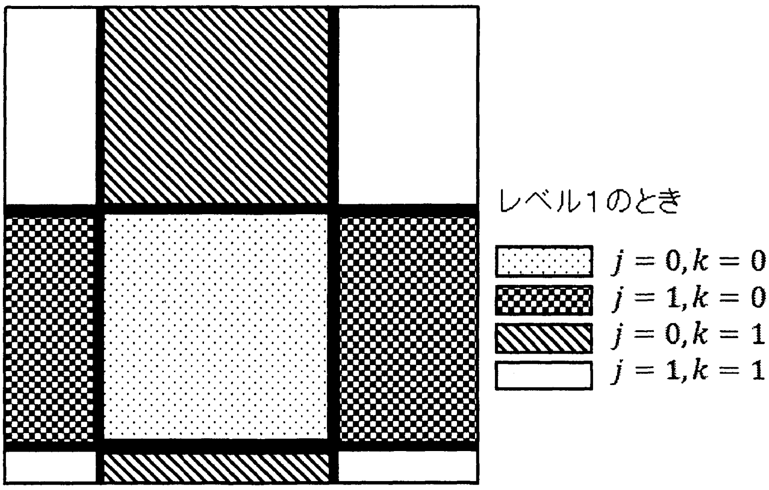

図2　連結でない領域の例

2.3 ポータルを定義

定義 3 グリッド上に等間隔に配置する点の集合をポータルと定義する。具体的には $C = 7 + \lceil \frac{36}{\epsilon} \rceil, P = N\lceil \frac{6}{\epsilon} \rceil$ とし、$l = [(0, (b + k2^{N-i}) \bmod 2^N), (2^N, (b + k2^{N-i}) \bmod 2^N)]$ がレベル i の水平な直線があればそのポータルの集合を

$$\{((a + \frac{h}{P}2^{N-i}) \bmod 2^N, (b + k2^{N-i}) \bmod 2^N) | h = 0, 1, 2, ..., P2^i\}$$

と定義する。グリッドの一辺を等間隔に P 分割するように P + 1 個のポータルを配置する。垂直な直線に対しても同様にポータルを定義する。

定義 4 $V \subseteq [0, 2^N] \times [0, 2^N]$ をユークリッド TSP の良い丸め性質をもつインスタンスとする。偶整数である $a, b \in \{0, 2, ..., 2^N - 2\}$ が与えられたとする。V を含む直線のツアーでグリッドの各直線との交差がポータルの部分集合となるようなものをシュタイナーツアーという。シュタイナーツアーは各 i と $G_i^{(a,b)}$ の各領域に対してツアーが領域の各辺と高々 C 回しかないとき軽いと呼ばれる。

2.4 キーとなる定理

定理 1 (Arora[1998])$V \subseteq [0, 2^N] \times [0, 2^N]$ をユークリッド TSP の良い丸め性質をもつインスタンスとする。偶整数である a, b が $\{0, 2, ..., 2^N - 2\}$ からランダムに選ばれるとすると少なくとも $\frac{1}{2}$ の確率で長さが高々 $(1 + \epsilon)OPT(V)$ の軽いシュタイナーツアーが存在する。

3 Arora のアルゴリズムについて

3.1 アルゴリズムについて

定理 1 を用いて Arora のアルゴリズムを示す。このアルゴリズムは動的計画法を用いる。小さい領域から順番に部分解を求める。次に一段階領域を大きくし、前回求めた部分解を使いながら同じ方法で部分解を求める。これを繰り返し最終的に軽いシュタイナーツアーを求める。最後にもともと存在するノードのみを通るツアーを構成する。

Arora のアルゴリズム
入力：ユークリッド TSP の良い丸め性質をもつインスタンス $V \subseteq [0, 2^N] \times [0, 2^N]$ と実数 $0 < \epsilon < 1$
出力：最適ツアーの $(1 + \epsilon)$ 倍以内のツアー

1．$\{0, 2, ..., 2^N - 2\}$ から a と b をランダムに選ぶ
$R_0 = \{([0, 2^N] \times [0, 2^N], V)\}$ とする。
2．$For \quad i = 1 \quad to \quad N - 1 \quad do:$
$G_i^{(a,b)}$ を構成する。$R_i = \emptyset$ とする。
$For |V_r| \geq 2$ を満たす各 $(r, V_r) \in R_{i-1} \quad do:$
$G_i^{(a,b)}$ の４つの領域 $r_1, r_2, r_3, r_4 (r_1 \cup r_2 \cup r_3 \cup r_4 = r)$ 構成し
$(r_1, V_r \cap r_1), (r_2, V_r \cap r_2), (r_3, V_r \cap r_3), (r_4, V_r \cap r_4)$ を R_i に加える。
3．$For \quad i = N - 1 \quad to \quad 1 \quad do:$
For 各領域 $r \in R_i \quad do:$
r に付随する部分問題を以下のようにしてすべて解く
$If \quad |V_r| \leq 1 \quad then$ 　すぐに解ける
$else$ 　すでに計算済みの４つの部分領域に対する部分問題の最適解を使う
4．R_0 の４つの部分領域に対する部分問題の最適解を用い V の最適な軽いシュタイナーツアーを計算する。
最後にシュタイナー点を除去しツアーを求める。

部分問題を以下に示す。
$1 \leq i \leq N - 1$ となるグリッド $G_i^{(a,b)}$ の１つの領域 r の境界辺上にある偶数個のポータルの集合を A とする。A での完全グラフの完全マッチング M で求める。マッチングした点を始点、終点とし領域内に存在するノードを通る最短パスを部分解とする。

グリッドのレベルが一番大きいとき間隔は２であり、良い丸め性質よりノード間の間隔は少なくとも８以上あるため一番小さい領域にはノードが多くても一つしか存在しない。上記の部分解は領域の小さいものから求めていくため始めは一意的に最短パスが求まる。それ以降は以前に求めた部分解を利用すれば求めることができる。
具体的には図３の点線上のポータルを任意個選んだ集合を A' とする。A' を選ぶと一意的にパスが求まる。この集合 A' の選び方すべてに対してのパスを求め一番短いものを部分解とする。また同じ集合 A の選びかたでもマッチングの違いによって得られるパスが異なるため、考えられるすべてのマッチングを考える必要がある。

3.2 計算量について

まず１つの領域に存在する部分問題の個数を求める。領域 r の境界上の４つ辺にはポータルがそれぞれ P 個ずつ存在する。１つの領域 r の境界辺上にある偶数個のポータルの集合を A とし、各辺で A に属するポータルの個数をそれぞれ i, j, k, l とする。(ただし i, j, k, l の総和は偶数, $i, j, k, l \leq C$) 集合 A の選び方は $\binom{P}{i}\binom{P}{j}\binom{P}{k}\binom{P}{l}$ 通りある。
また完全グラフ K_{2x} 上の完全マッチングの個数を $PM(2x)$ とする。$PM(2) = 1$ は自明。

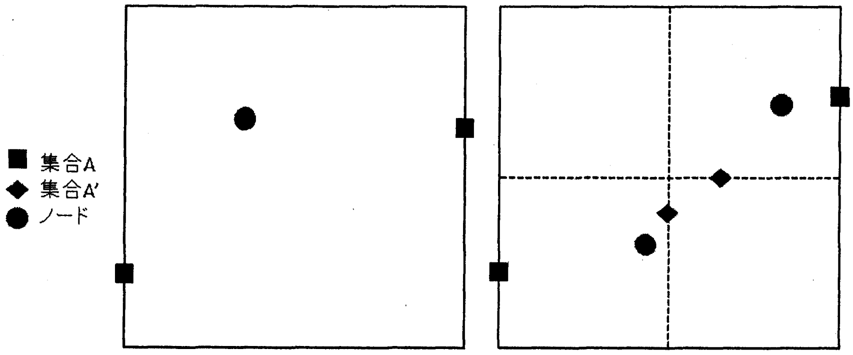

図3　部分解について

K_{2x} 上の完全マッチングは一辺を固定すると K_{2x-2} 上の完全マッチングとなるので以下の漸化式が成り立つ。

$$PM(2x) = (2x-1)PM(2x-2) = \frac{(2x)!}{2^x x!}$$

A 上の完全グラフの完全マッチングの個数は $PM(i+j+k+l)$ となる。よって部分問題の個数は以下の式で表すことができる。

$$\sum_{0 \le i,j,k,l \le C \text{ かつ } i+j+k+l \text{ は正の偶数}} \binom{P}{i}\binom{P}{j}\binom{P}{k}\binom{P}{l} PM(i+j+k+l)$$

ここで $\binom{n}{m} \le n^m, PM(2x) \le (2x)!, i+j+k+l \le 4C$ より

$$\le \sum_{0 \le i,j,k,l \le C} P^i P^j P^k P^l (4C)!$$

$$\le (4C)!\Big(\sum_{0 \le i \le C} P^i\Big)^4$$

$$\le (4C)!\Big(\sum_{0 \le i \le C} \binom{C}{i} P^i\Big)^4$$

二項定理より

$$= (4C)!(P+1)^{4C}$$

次に部分解の計算が必要な領域の個数を求める。これは領域内にノードが存在する領域の個数と等価である。領域内のノードの個数が一つ以下の場合は定数時間で解ける。よって複数個のノードが存在する領域の個数を求める。そこで以下のような有向木Bを考える。

根は R_0 にある領域である。各領域は $r \in R_i$ は0または4個の子をもつ。S をこの木Bの点で4個の葉を子としてもつものの集合とする。これらの領域の内部同士は互いに素であり、領域内のノードの数が1個以下になるまで分割を繰り返すため集合 S の各領域はノードを少なくとも2個含んでいる。したがって $|S| \leq \frac{n}{2}$ である。この木 B の深さは N であり B の各点は葉か少なくとも1個の S の祖先なので、葉でない領域が高々 $N\frac{n}{2}$ 個となり合わせて高々 $\frac{5}{2}Nn$ 個の領域しかない。

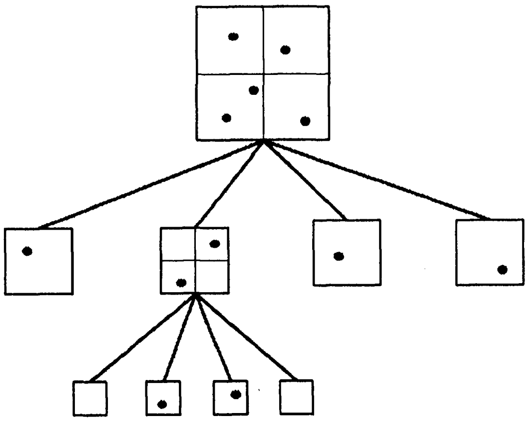

図4　有向木Bの例

領域内に複数ノードが存在するときの部分解を求める計算量は、部分領域間の4本の辺上にあるポータルの選び方とポータルを訪れるすべての可能な順序が考慮される。これは高々 $(P+2)^{4C}(8C)!$ 個ある。

以上より1つの領域ですべて部分解を求めるのに必要な計算量は $O((P+2)^{8C}(4C)!(8C)!)$ である。

$N = O(\log \frac{n}{\epsilon}), C = O(\frac{1}{\epsilon}), P = O(\frac{N}{\epsilon})$ なので全体の計算量は以下のようになる。

$$O\Big(\frac{5}{2}Nn(P+2)^{8C}(8C)^{12C}\Big) = O\Big(n(\log n)^{O(\frac{1}{\epsilon})}O(\frac{1}{\epsilon})^{O(\frac{1}{\epsilon})}\Big)$$

$$= O\Big(n(\log n)^{c}\Big)$$

ただし $c = O(\frac{1}{\epsilon})$ である。したがってより良い解を得ようと ϵ を小さく設定すると計算量が増加する。

Arora のアルゴリズムはシフトしたグリッドで使用した a, b のすべての組合せを試すことで決定的アルゴリズムにすることができる。これは計算時間を $O(\frac{n^2}{\epsilon^2})$ 倍するので最終的に以下の計算量が得られる。

$$O\Big(n^3(\log n)^{c}\Big)$$

4 Arora のアルゴリズムの拡張

今まで取り扱ってきたユークリッドＴＳＰで与えられるノードは平面上であった。これを円筒の側面上に拡張することを考える。側面上を移動することを考えれば任意の２点 v, w の辺は２通り存在する。ここでは短い方を v, w 間の辺とする。すべてのノード間でこの動作を行ってできたグラフをＧとする。ここで二通りの場合分けを考える。

(1) グラフＧと垂直な直線 l との交差がひとつもない l が存在するとき

(2) グラフＧと垂直な直線 l との交差がひとつもない l が存在しないとき

4.1 グラフＧと垂直な直線 l の交差がひとつもない l が存在するとき

この場合は Arora のアルゴリズムをそのまま適用できる。直線 l で展開するとユークリッドＴＳＰに帰着できるからである。

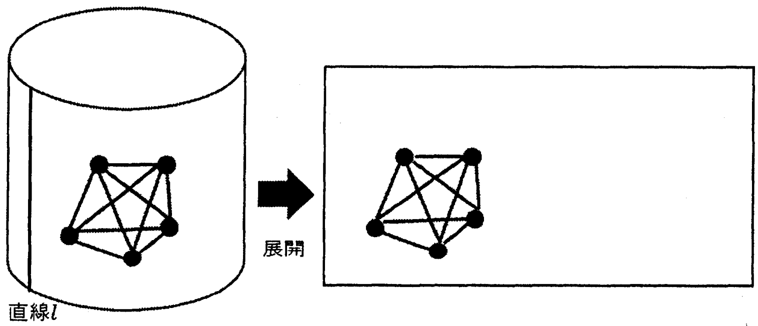

図５ グラフＧと直線 l の例

4.2 グラフＧと垂直な直線 l の交差がひとつもない l が存在しないとき

この場合は以下のように場合分けをして考える。

(a) 通常のツアーを求める

(b) 円柱を一周するようなツアーを求める

(a)(b) で得られたツアーで短い方のツアーを解とする。

(a) については Arora のアルゴリズムを適用して求める。しかし得られるツアーは円柱の展開の方法に依存するので考えられるすべての展開の方法に関して求める必要がある。与えられるノードの数を n とすると多くて n 通りの展開の方法が存在する。

(b) についてはまずＧと l の交差が最小な部分で展開する。まず図６のようにＧと l の交差が一回のみの場合を考える。交差した辺を e_{vw} とするとノード v, w を始点終点とするコ

ストが最小のハミルトンパスを求める。求めたハミルトンパスと辺 e_{vw} をつなげることでツアーが求まる。この最小ハミルトンパス問題を解くアルゴリズムとして Arora のアルゴリズムが利用できるかを現在検討中である。求めるパスのノード v,w の次数が 1 、それ以外のノードの次数が 2 となる。これを考慮して以下の条件を Arora のアルゴリズムに加えることで解けるのではないかと考える。

・点 v,w につなげる辺は一本にする。

・領域内にノード v,w のどちらか一つ存在する場合には選ぶ外側のポータルを奇数個にする。これ以外は通常通り偶数個にする。

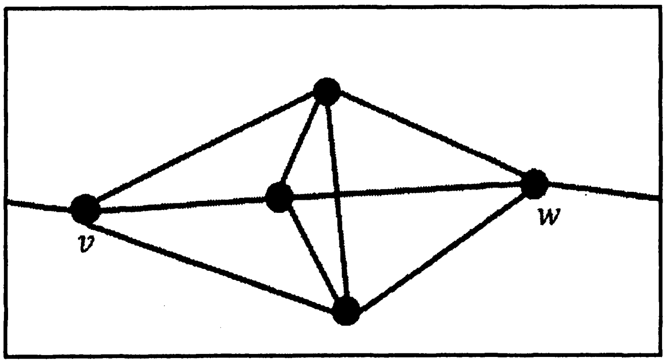

図6　G と l の交差が辺 e_{vw} との一回のみの例

また図 7 のように交差する辺が複数ありすべて同一のノードとつながっている場合は v, w_1 と v, w_2 を始点終点とする最小ハミルトンパス問題を解く。よってノードの個数を n とすると多くて $n-1$ 回の場合分けを考えればよい。

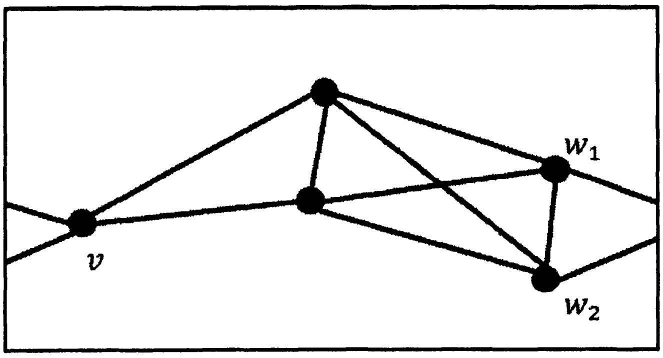

図7　G と l の交差が複数あり同一のノードとつながっている例

また交差が複数あり、複数のノードとつながっている場合については交差している辺からツアーとして選ぶ辺の本数を考えられるすべての場合について考える必要がある。詳細については現在検討中である。

参考文献

[1] B．コルテ/J．フィーゲン、組合せ最適化 理論とアルゴリズム第２版、シュプリンガージャパン、(2009)601-610

[2] Arora.S, 1998,Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems.Journal of the ACM 45,p.753-782

[3] Christofides,N. 1976. Worst-case analysis of a new heuristic for the traveling salesman problem.In symposium on New Directions and Recent Results in Algorithms and Complexity,J.F.Traub,ed.Academic Press,Orlando,Fla.,p.441.

[4] Fernandez de la Vega,W.,and Lueker,G.S.1981. Bin packing can be solved within $1+\epsilon$ in linear time.Combinatorica 1,4,349-355.

[5] Garey,M.R.,Graham,R.L.,and Johnson,D.S. 1976. Some NP-complete geometric problems.In Proceedings of the 8th Annual ACM Symposium on Theory of Computing(Hershey,Pa.,May 3-5).ACM,New York,pp.10-22.

[6] Goemans,M 1995. Worst-case comparison of valid inequalities for the TSP.Math.Prog 69,336-349 Ibarra,O.H.,and Kim,C.E. 1975. Fast approximation algorithms for the knapsack and sum of subsets problems.J.ACM22,4(Oct.)463-468.

[7] Ibarra,O.H.,and Kim,C.E. 1975.Fast approximation algorithms for the knapsack and sum of subsets problems. J.ACM 22,4(Oct.)463-468.

[8] Karmarkar,N.,and Karp,R.M. 1982. An efficient approximation scheme for the onedimensional bin-packing problem.In Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science.IEEE Computer Society Press,Los Alamitos,Calif.,pp.312-320.

[9] Karp,R.M. 1972. Reducibility among combinatorial problems.In Complexity of Computer Computations,R.E.Miller and J.W. Thatcher,eds.Advances in Computer Research,Plenum Press,New York,pp.85-103.

[10] Karp,R.M. 1977.Probabilistic analysis of partitioning algorithms for the TSP in the plane.Math.Oper.Res.2,209-224.

[11] Papadimitriou,C.H.1977. Euclidean TSP is NP-complete.Theoret.Comput.Sci.4,237-244.