

「システム論」 S1 ターム 火2・金2時限
教員: 福永アレックス

最終レポート課題

Steiner tree problem in graph theory

J4-190507 理科一類 19組 2年 木下裕太

November 24, 2020

本レポートではグラフ G に対しその頂点集合を $V(G)$ 、辺集合を $E(G)$ と表記し、特に断りのない限りは単純グラフ¹のみを扱うものとする (本質的に非単純である必要がないため)。考えているグラフが明らかなきはこれらを省略して単に V, E と表記する。また以下の議論でしばしば使う性質として、グラフ T が Tree (木)²ならば $V(T) = E(T) + 1$ が成り立つというものがある。

1 Steiner tree problem

Steiner tree problem (シュタイナー木問題) は発案した数学者 Jakob Steiner に由来して名付けられた、組合せ最適化の分野におけるある種の問題群の総称である。

1.1 Statement

上で説明したように Steiner tree problem と呼ばれる問題にはいくつかの設定が存在する。ここでは本レポートで主に考察の対象とする、グラフ理論における 1 つの version について判定問題として定式化したものを述べる。

INSTANCE:

- unweighted な無向グラフ G
- 頂点の部分集合 $S \subseteq V$
- 正の整数 k

QUESTION:

G の部分木 $T \subseteq G$ であって次の 2 つを満たすものが存在するか判定せよ。

- $S \subseteq V(T)$
- $|E(T)| = |V(T)| - 1 \leq k$

QUESTION の 1 つ目の条件を満たす部分木 T を (G における) S を張る Steiner tree という。この判定問題と等価な最適化問題として、“ S を張る Steiner tree であって頂点数最小のもの (Steiner minimum tree) を求める” と言い換えた形でも知られている。なお特に $S = V(G)$ の場合は Steiner tree

¹multiple edge や self loop を持たないグラフのこと。

²cycle をもたない無向グラフのこと。

は Spanning tree (全域木) と同義になり、グラフが連結であるかの判定は $O(|V|^2)$ などと比較的高速に行える。また $|S| = 2$ の場合は Shortest path problem と等価で、こちらも幅優先探索などのアルゴリズムにより Linear time $O(|V|)$ で解を得ることが可能である。

1.2 Variations

本来 Steiner tree problem という呼称は次の問題を指して使われていたようである。

平面上に n 個の点を与えられる。それら全てが連結となるようにするために必要な線分の長さの和を最小化せよ。用いる線分の端点は必ずしも与えられた点である必要はない。

現在ではこの問題は Euclidean Steiner tree problem あるいは Geometric Steiner problem と呼ばれている。最適な線分の組は互いに交わらずかつ閉路を作らないため、それらの端点を頂点とした木だとみなせる。これを Steiner minimum tree と呼ぶ。またこの木の頂点であって、与えられた n 点のいずれとも異なるものは Steiner points (シュタイナー点) と呼ばれる。

例として $n = 3$ の場合の Euclidean Steiner tree problem について見てみる。

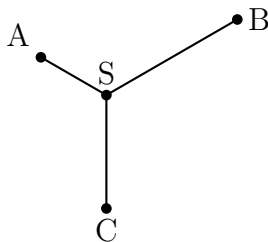


Figure 1: Steiner minimum tree in case of $n = 3$.

Figure 1 において A, B, C が与えられた 3 点であり、S は Steiner point である。ここで 3 本の線分 SA, SB, SC が互いになす角は 120° となっている。このように $n = 3$ の場合、Steiner minimum tree は与えられた 3 点を作る三角形の Fermat 点と各頂点を結ぶ 3 つの線分からなることが知られている。

このコンテキストでの Steiner minimum tree に関する現在未解決の予想として次のものが知られている。

Conjecture(Gilbert & Pollack, 1968)

Euclid 平面上の 2 つ以上の異なる点からなる集合 P に対し、

$L_{SMT}(P)$: Steiner minimum tree の合計の長さ

$L_{MST}(P)$: P を頂点集合とする Minimum spanning tree の合計の長さ

とする。このとき次の等式が成り立つ。

$$\inf \left\{ \frac{L_{SMT}(P)}{L_{MST}(P)} \mid P \right\} = \frac{\sqrt{3}}{2}$$

既に n を小さい値に限定した主張については証明されており、一般の場合についても証明できたという旨の論文 [1] が 1992 年に出されたが、2012 年には彼らの議論に gap があることを主張する論文 [2] が出版されている。

また類似した別の version として、Euclid 距離を Manhattan 距離で置き換えたものも知られており、こちらは Rectilinear Steiner tree と呼ばれる。さらに次元や距離関数を一般化した問題についても研究されている。

1.1 で述べた version を一般化した問題として、重み付きグラフについての以下のものも知られている。

INSTANCE:

- weighted な無向グラフ G とそのコスト関数 $c :$
 $E \rightarrow \mathbb{R}_+$
- 頂点の部分集合 $S \subseteq V$
- 正の実数 $k \in \mathbb{R}_+$

QUESTION:

G の部分木 $T \subseteq G$ であって次の 2 つを満たすものが存在するか判定せよ。

- $S \subseteq V(T)$
- $\sum_{e \in E(T)} c(e) \leq k$

コスト関数 c が定数関数であればこれは 1.1 で述べたものと等価であるため、確かに一般化になっている。

1.3 Applications

有名な実用例としてはネットワークの通信経路策定や VLSI 回路の配線設計などが挙げられる。前者は weighted graph に関する version の適用例であり、後者は Euclidean plane に関する version の適用例である。

2 Complexity analysis

Steiner tree problem は Richard Karp's 21 NP-complete problems[3] の 1 つであり、前章で紹介したいずれの version も NP 完全な問題であることが知られている [4]。この事実は非常に興味深いものであるが、本章では最初に述べた unweighted graph についての問題のみ、その NP-completeness を示すことにする³。

Steiner tree problem がクラス NP に属することは明らかであるから、NP-hard であることを示せばよい。[5] で述べられている Exact Cover by 3-Sets (NP 完全問題の 1 つ) からの reduction による証明にヒントを得て、本レポートでは NP 完全問題として有名な Vertex cover problem からの多項式時間還元による証明を試みる。

2.1 Vertex cover problem(VC)

Vertex cover problem (頂点被覆問題) とは判定問題として次のように定式化されるものである。

INSTANCE:

- 無向グラフ G
- 正の整数 k

QUESTION:

G の 部分グラフ $H \subseteq G$ であって次の 2 つを満たすものが存在するか判定せよ。

- $E(H) = E(G)$
- $|V(H)| \leq k$

³Steiner tree problem のどの version についてもその NP 完全性を示している文献はあまり見当たらなかった。

QUESTION の 1 つ目の条件を満たす部分グラフ H (あるいはその頂点集合 $V(H)$) を G の Vertex cover (頂点被覆) という。Steiner tree problem と同様に Minimum vertex cover を求める最適化問題としてもよく知られ、やはり Richard Karp's 21 NP-complete problems[3] の 1 つでもある。

2.2 Polynomial time reduction from VC

今 Vertex cover problem X の instance として無向グラフ G と正の整数 k が与えられているとする。それに対し新たに無向グラフ G' を次のように定める⁴。

- $V(G') = \{\alpha\} \cup V(G) \cup E(G)$
- $E(G') = \{\{\alpha, v\} | v \in V(G)\} \cup \{\{e, v\} | e \in E(G) \wedge v \in e\}$

また G' に加えて頂点集合 S' および正の整数 k' を

$$\begin{aligned} S' &= \{\alpha\} \cup E(G) = V(G') \setminus V(G) \\ k' &= k + |E(G)| \end{aligned}$$

と定め、この (G', S', k') を instance とする Steiner tree problem Y と変換 $f: X \mapsto Y$ を考える。この変換 f が所望の還元になっていることを以下の 3 ステップで示す。

1. f is executable in polynomial time.
2. $X \text{ is True} \Rightarrow f(X) \text{ is also True.}$
3. $f(X) \text{ is True} \Rightarrow X \text{ is also True.}$

Step 1

G は単純グラフなので $|E(G)| \in O(|V(G)|^2)$ であり、ゆえに $|V(G')|, |E(G')|$ についても $|V(G)|$ の多項式オーダーである。よって変換 f は多項式時間で実現可能であることが分かる。

⁴定義から分かるように G における辺も G' では頂点として扱っている。また頂点 α は G の辺に対応する頂点同士が G' において互いに連結になるために便宜上加えたものである。

Step 2

あるサイズ k 以下の Vertex cover $W \subseteq V(G)$ (X の解の 1 つ) が存在したとする。定義より各 $e \in E(G)$ について $e \cap W \neq \emptyset$ だから、その内のある元を対応させる写像 $h : E(G) \rightarrow W$ がとれる。このとき S' を張る Steiner tree T' であって頂点数が k' 以下のもの (Y の解の 1 つ) が次のように構成できる。

- $V(T') = \{\alpha\} \cup W \cup E(G) \subseteq V(G')$
- $E(T') = \{\{\alpha, v\} | v \in W\} \cup \{\{e, h(e)\} | e \in E(G)\} \subseteq E(G')$

実際にこれは S' を張る G' の部分木であり、また $|E(T')| \leq |W| + |E(G)| \leq k + |E(G)| = k'$ が成り立っている。

Step 3

ある辺数 k' 以下の Steiner tree $T' \subseteq G'$ が存在したとする。 G' の定義より、各 T' の頂点⁵ $e \in E(G) \subseteq V(T')$ に対し、ある $v \in V(T') \cap V(G)$ が存在して $\{e, v\} \in E(T') \subseteq E(G')$ 、つまり $v \in e$ となる⁶。グラフ G に立ち返って考えると、これは $V(T') \cap V(G)$ が G の Vertex cover であることに他ならない。さらに $|V(T')| = |E(T')| + 1 \leq k' + 1 = k + |S'|$ かつ $S' \subseteq V(T')$ なので、 $|V(T') \cap V(G)| = |V(T')| - |S'| = |E(T')| \leq k$ も成り立つ。

Conclusion

以上の議論から変換 f が VC から Steiner tree problem への多項式時間還元になっていることが示され、Steiner tree problem が NP-complete であることの証明が完結した。

2.3 Parameterized complexity

Weighted graph についての Steiner tree problem を解く手法の 1 つとして定番のものが Dreyfus-Wagner 法 [6] である。このアルゴリズムは後で示すように $O(3^{|S|}|V| + 2^{|S|}|V|^2)$ 時間で動作するため、Steiner tree problem は FPT(fixed-parameter tractable) な問題としても知られる。

⁵とみなした G の各辺

⁶そもそも e は $V(G)$ の 2 元集合であった。 $v \in e$ とはグラフ G において頂点 v が辺 e の端点であることを意味する。

3 Solutions for the problem

3.1 Exact solution

前章の最後に触れた Dreyfus-Wagner 法 [6] は Dinamic programming (動的計画法) による解法であり、その概要は以下に示した疑似コード⁷の通りである。

Algorithm 1 A pseudocode of the Dreyfus-Wagner algorithm

Require:

G : a simple and undirected graph.

S : a subset of the vertices in G .

Ensure:

Return value is the number of edges in Steiner minimum tree, and it would be ∞ if impossible to construct.

```
1: function FIND-SMT( $G, S$ )
2:   for  $R \subseteq S$  and  $v \in S$  do                                ▷ initialize the DP table.
3:      $dp[S, v] \leftarrow 0$ 
4:   for  $R \subseteq S$  and  $v \in V \setminus S$  do
5:      $d[S, v] \leftarrow \infty$ 

6:   for  $R \subseteq S$  do                                              ▷ smaller  $R$  must come earlier.
7:     for  $v \in V$  do
8:       for  $X \subseteq R$  do
9:          $dp[R, v] \leftarrow \min\{dp[R, v], dp[X, v] + dp[R \setminus X, v]\}$ 

10:    for  $|V| - |R|$  times do                                       ▷ like Bellman-Ford algorithm.
11:      for  $\{u, v\} \in E$  do
12:         $dp[R, v] \leftarrow \min\{dp[R, v], dp[R, u] + 1\}$ 
13:         $dp[R, u] \leftarrow \min\{dp[R, u], dp[R, v] + 1\}$ 

14:   take  $\forall s \in S$ .
15:   return  $dp[S, s]$ 
16: end function
```

⁷本レポートに記載している疑似コードにおいて、ループや条件分岐のスコープは python 等の言語と同様にインデントに従って決まるものとする。

各 $R \subseteq S$ と $v \in V$ について $dp[R, v] = (\text{頂点集合 } R \cup \{v\} \text{ を張る Steiner minimum tree の辺数})$ となるように $|R|$ の小さい方から順に更新していく⁸。このとき求めたいのはある $s \in S$ に対する $dp[S, s]$ の値 (s の取り方に依らない) である。またループの回数の合計をカウントすることにより、計算量が $O(3^{|S|}|V| + 2^{|S|}|V|^2)$ に収まることが分かる。C 言語による実装では $|S| = 15, |V| = 20$ 程度の INSTANCE で実行に約 1 sec かかることから、現実的にはせいぜい $|S|$ が 2 桁以下の Case にしか適用できないだろう。

3.2 Approximate solution

次に近似アルゴリズムとして最も広く使用されている KMB 法 [7] を紹介する。この方法では以下の 3 ステップで近似解を得る。

1. All pairs shortest path を the Floyd-Warshall algorithm により求める。
2. 頂点集合 R の重み付き完全グラフ K であって、各辺 $\{u, v\}$ の重みが $u - v$ 間の shortest path の長さに等しいものを考える。
3. K の Minimum spanning tree を Prim's algorithm などを用いて求める。

いずれのステップも頂点数 $|V|$ の多項式時間で実行でき、全体の計算量は $O(|V|^3)$ となる。明らかに厳密解よりも真に小さい解が得られることはない。また疎グラフについてはほぼ正確な結果が得ることができる。実際に頂点数が 20 程度のランダムに生成したグラフでは、いずれも 3.1 で実装したものと同一結果を得ることができた。次のページに上のステップを辿って実装した疑似コードを載せて本レポートを締めたいと思う。

⁸前ページの疑似コードに示された DP テーブルの更新が正しく動作することの詳細な説明は [6] に譲るが、 v と R のつながり方で場合分けすると納得しやすい。

Algorithm 2 An example of approximate solutions

Require:

G : a simple and undirected graph.
 S : a subset of the vertices in G .

Ensure:

Return value is the number of edges in *approximate* Steiner minimum tree, and it would be ∞ if impossible to construct.

```
1: function FIND-APPROX-SMT( $G, S$ )
2:   for  $u \in V$  do                                 $\triangleright$  initialize the pairwise distance table.
3:      $d[u, u] \leftarrow 0$ 
4:     for  $v \in V \setminus \{u\}$  do
5:       if  $\{u, v\} \in E$  then
6:          $d[u, v] \leftarrow 1$ 
7:       else
8:          $d[u, v] \leftarrow \infty$ 

9:   for  $i \in V$  do                                 $\triangleright$  the Floyd-Warshall algorithm.
10:    for  $u \in V$  do
11:      for  $v \in V$  do
12:         $d[u, v] \leftarrow \min\{d[u, v], d[u, i] + d[i, v]\}$ 

13:   take  $\forall t \in S$ .                                 $\triangleright$  Prim's algorithm on  $K$ .
14:   for  $s \in S$  do  $cut[s] \leftarrow \infty$ 
15:    $T \leftarrow \emptyset$ ,  $cost \leftarrow 0$ ,  $mincut \leftarrow 0$ 
16:   loop
17:      $T \leftarrow T \cup \{t\}$ ,  $cost \leftarrow cost + mincut$ 
18:     if  $T = S$  then
19:       return  $cost$ 
20:      $mincut \leftarrow \infty$ 
21:     for  $s \in S \setminus T$  do
22:        $cut[s] \leftarrow \min\{cut[s], d[s, t]\}$ 
23:       if  $cut[s] < mincut$  then
24:          $mincut \leftarrow cut[s]$ ,  $t \leftarrow s$ 
25: end function
```

References

- [1] Du, D. Z., & Hwang, F. K. (1992). A proof of the Gilbert-Pollak conjecture on the Steiner ratio. *Algorithmica*, 7(1-6), 121-135.
- [2] Ivanov, A. O., & Tuzhilin, A. A. (2012). The steiner ratio gilbert - pollak conjecture is still open. *Algorithmica*, 62(1-2), 630-632.
- [3] Karp, R. M. (1972). Reducibility among combinatorial problems. In Complexity of computer computations (pp. 85-103). *Springer, Boston, MA*.
- [4] Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability (Vol. 174)*. San Francisco: freeman.
- [5] Santuari, A. (2003). Steiner tree NP-completeness proof. *Technical report, University of Trento*.
- [6] Dreyfus, S. E., & Wagner, R. A. (1971). The Steiner problem in graphs. *Networks*, 1(3), 195-207.
- [7] Kou, L., Markowsky, G., & Berman, L. (1981). A fast algorithm for Steiner trees. *Acta informatica*, 15(2), 141-145.