最小k-部分木問題の$\mathcal{NP}$困難性・貪欲的下界値算法・近似解法

01107880 防衛大学校 *片岡 靖詞 KATAOKA Seiji
01700900 防衛大学校 山田 武夫 YAMADA Takeo
02301780 防衛大学校 高橋 秀雄 TAKAHASHI Hideo

1 はじめに

点集合Vと枝集合Eからなる無向連結グラフ$G=(V,E)$において，根となる点$r\in V$と整数k ($1\leq k\leq |V|-1$) が与えられているとする．このとき，k-部分木とは，rを根として枝の数が丁度k本であるような連結部分木のことをいう．

本研究で扱う**最小k-部分木問題** (Minimum k-Subtree Problem：以下k-ST)[1] とは，各枝にコストが与えられているとき，枝のコストの総和が最小のk-部分木を求める問題である．

本稿では，k-ST の$\mathcal{NP}$困難性と貪欲算法による下界値の導出法および近似解法について議論する．

2 $\mathcal{NP}$困難性

ここでは，k-ST を高々$|V|$回解くことにより，すでに$\mathcal{NP}$困難性が示されているスタイナー木問題を解くアルゴリズムを構築し，k-ST の$\mathcal{NP}$困難性を導く．ここでk-ST には，枝のコストに非負制約がないことを注意しておく．またスタイナー木問題には，枝のコストの非負性を仮定する．

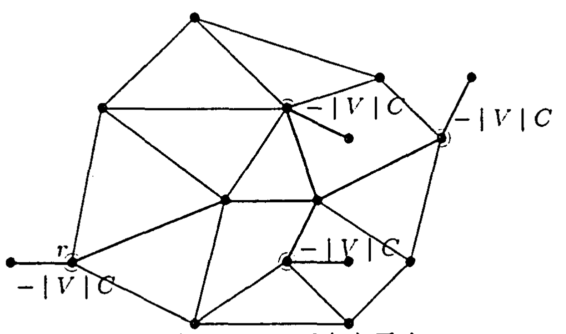

図 1: $(K+|S|)$-ST と スタイナー木問題の等価性

スタイナー木問題のスパニング点の集合を$S(\subset V)$とする．図1のように，各スパニング点$i\in S$に対してダミーの点i'を加え，枝(i,i')のコストを十分小さな値，たとえばCを枝のコストの最大値とするとき，$-|V|C$に設定する．また，もとのグラフにおけるスタイナー木問題の最適解で選ばれる枝の数をKとする．スパニング点の任意のひとつを根とすれば，変換されたグラフにおける$(K+|S|)$-ST の最適解は，もとのグラフにおけるスタイナー木問題の最適解を与える．Kの値は未知であるが，$|S|-1\leq K\leq |V|-1$であることは分かっているので，次のアルゴリズム Steiner のように，k-ST を最大$|V|-|S|+1$回解くことによってKを決定することができる．

アルゴリズム Steiner

```
for (k = 2|S|-1; k < |V|+|S|; ++k) {
  k-ST を解く；
  すべての (i,i') (i∈S) が解に含まれていれば
    K = k として，停止する；
}
```

このときk-ST が多項式オーダで解くことができれば，スタイナー木問題も多項式オーダで解けることになるので矛盾が生じる．したがって，k-ST は$\mathcal{NP}$困難な族に属する問題であることが示された．

3 貪欲的下界値算法

k-ST の下界値を求める初歩的な算法のひとつとして，Kruskal による最小木のアルゴリズムを，第kステップ目で停止させる方法 (KLB) が考えられる．ここでは KLB よりも，より理知的で強い下界を得る貪欲算法 (GLB) を提案する．アルゴリズムの説明に先立って，次を定義する．

定義:枝の歩数 グラフ$G=(V,E)$において，すべての枝の長さを1とし，根節点rから点qまでのある最短路を$r,\ldots,p,q$ (pはr-q最短路におけるqの直前の点) とするとき，その最短距離を枝(p,q)の歩数とする．

次のアルゴリズム GLB において，歩数iの枝の集合をE_i，枝eのコストを$c(e)$，枝eの歩数を$d(e)$とする．

命題1 アルゴリズム GLB により，Lはk-ST の下界値を得る．

証明 省略（詳細は seiji@jpnnda.bitnet まで）

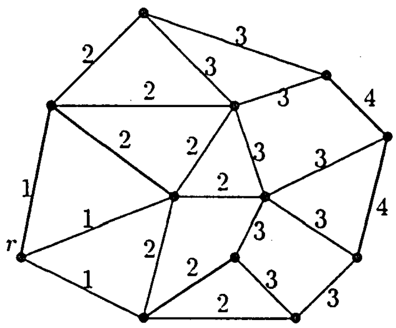

図 2: 枝の歩数と 5-ST に GLB を適用した Fの例

アルゴリズム GLB

$F = \emptyset$; $i = 1$; $L = 0$;
while ($i \leq k$) {
  $c(e_i) = \min\{c(e) \mid e \in \bigcup_{j=1}^{i} E_j\}$;
  if ($F \cup \{e_i\}$ has no cycle) {
    $F = F \cup \{e_i\}$;
    $L = L + c(e_i)$;
    $i = i + 1$;
  }
  $E_{d(e_i)} = E_{d(e_i)} \setminus \{e_i\}$;
}

4 近似解法

k-ST の実行可能解を得る方法としては，Prim による最小木のアルゴリズムを，第kステップ目で停止する方法 (PUB) が考えられる．本節では，PUB をもとに逐次改善を施すことにより，より解精度の優れた近似解を得る方法 (HUB) を提案する．逐次改善における近傍としては，次の点に留意して２通りの手法を考えた．

- PUB では，選ばれた点における最小木を構築しているので，木で張られる点の交換を主体とする
- 点を交換した後に構成される木もまた，張られている点の中では最小木を構築するようにする

特に第２点は，k-ST の最適解が，張られている点の中で最小木になっている性質を重視している．

現在の実行可能解の点集合を P，枝集合を Tとする．

手法 1　カット $(P, V \setminus P)$ から最小コストの枝 e を取り込み，木 $T \cup \{e\}$ から最大コストの葉を取り除く．

手法 1 を繰返すだけでも，アニーリング法などのように解の改悪を許せば，すべての k-部分木に移行することが理論上可能ではあるが，実際には局所解に陥り，なかなか抜け出せなくなる．そこで，葉だけの交換にすぎない手法 1 に加えて，次の手法 2 を考えた．

手法 2　カット $(P, V \setminus P)$ から最小コストの枝 e を取り込み，木 $T \cup \{e\}$ から次数 2 の関節点に接続する 2 本の枝を取り除き，生成される 2 連結成分で構成されるカットの中から，コスト最小の枝を取り入れる．

命題 2　手法 1，2 とも，点交換した後の木は，張られた点における最小木になっている．

次の例題および計算機実験では，どちらの手法とも，目的関数値が最も大きく下がるように点の交換を行っており，解の改善ができなくなった時点で終了している．

5 計算機実験（例示）および結果

二川ら [2] で記述している例題を用い，kをパラメータとして実験を行った．

PUB Prim の最小木アルゴリズムを第 kステップ目で停止して得られた解の値

HUB 本稿第 4 節で提示した近似解の値

GLB 本稿第 3 節で提示した貪欲的下界値

KLB Kruskal の最小木アルゴリズムを第 kステップ目で停止して得られた解の値

なお，これらの値には，KLB $\leq$ GLB $\leq$ HUB $\leq$ PUB という関係が成立しており，図 3 では HUB を 1 としたときの各評価値の割合を累積グラフにしている．

この結果から，kが小さいときは，GLB が効果的に働くことが分かる．また，kが大きくなるにしたがって，下界と上界の差が小さくなっていくことも分かる．

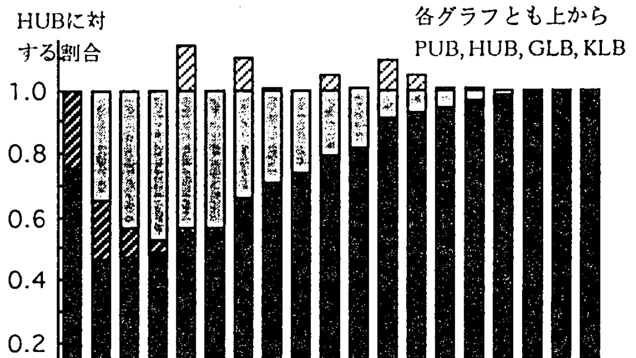

図 3: kをパラメータにした各評価値

[1] 片岡，高橋，山田 OR 学会アブストラクト 1994 秋
[2] 二川，山田，片岡 OR 学会アブストラクト 1995 春
(本研究を進めるにあたり，NTT の伊藤氏，東大の皆川氏には貴重なコメントを頂きましたことを感謝致します．)