Vol.2014-AL-150 No.17
2014/11/21

split グラフ上の全域木混雑度問題に対する反復丸めを用いた近似アルゴリズム

久保 浩平[1]　山内 由紀子[1]　来嶋 秀治[1]　山下 雅史[1]

概要：グラフの全域木混雑度とは，与えられたグラフ G の全ての全域木の中で最小の混雑度を言う．混雑度は与えられたグラフ（ネットワーク）G をその全域木に置き換えた際の各辺の混雑具合を表す指標である．本稿では split グラフ上の全域木混雑度問題の乱択 2 近似アルゴリズムを提案する．提案アルゴリズムでは，多品種フローを応用した定式化を与え，その LP 緩和に基づいて最小混雑木から近似解をランダムに構成する．また，脱乱択化の試みとして，反復丸めアルゴリズムを応用した近似アルゴリズムを提案する．

キーワード：全域木混雑度，split グラフ，線形計画緩和，近似アルゴリズム，反復丸めアルゴリズム

1. はじめに

全域木混雑度とは，Ostrobskii[5] が定義したグラフパラメータである．ある連結，無向グラフ $H = (U, D)$ に対し，全域木混雑度は以下のように定義される．H の全域木 T に対し，辺 $\{u, v\} \in D$ の迂回路を，T における u–v パスと定義する．$e \in T$ の混雑度を迂回路が e を含む H の辺の本数とし，$\mathrm{cng}_T(e)$ で表す．H における T の**混雑度** (*congestion*) を T の全ての辺の混雑度の最大値と定義し，$\mathrm{cng}_H(T)$ で表す．H の**全域木混雑度** $\mathrm{stc}(H)$ とは H の全ての全域木のうち，最小の混雑度のことを言う．すなわち，$\mathrm{stc}(H) = \min_T\{\mathrm{cng}_H(T)\}$ である．Simonson[6] はこのパラメータをカット幅の近似に用いている．

本稿では split グラフ上の全域木混雑度問題について述べる．split グラフ上の全域木混雑度問題は，岡本らが NP 完全であることを示している [4]．また，全域木混雑度問題については様々な結果が報告されている ([1], [3], [4]) が，整数計画法を用いたアプローチは見当たらない．そこで 3 章では多品種フローを応用した split グラフ上の全域木混雑度問題の定式化について述べる．4 章では 3 章の LP 緩和に基づく単純な乱択 2 近似アルゴリズムを設計する．5 章で反復丸めアルゴリズムを用いた近似アルゴリズムを設計する．4 章の結果とギャップがあるが，ある特殊な場合については決定的に 2 近似アルゴリズムが得られることを示す．

[1] 九州大学
Kyushu University

2. 準備

2.1 記号

$G = (V, E)$ を連結無向グラフとする．ある $S \subseteq V$ に対して，$G(S)$ で S によって誘導される部分グラフを表す．V がクリーク C と独立集合 I に分割できるとき，G は split グラフという．split グラフは弦グラフのサブクラスである．今後，G は split グラフと仮定する．G の頂点 v の隣接頂点の集合を $N(v)$ で表す．v の次数を $\deg(v)$ で表す．V 上の木 T と T の枝 e に対して，T から e を除いたときの 2 つの連結成分を $L_e(T), R_e(T)$ とする．$C_{R_e(T)} = C \cap R_e(T), I_{R_e(T)} = I \cap R_e(T)$ とする．$C_{L_e(T)}, I_{L_e(T)}$ についても同様である．e の混雑度は $L_e(T), R_e(T)$ のカットのサイズと等しくなる．

2.2 最小混雑木

$H = (U, D)$ を任意の連結無向グラフとする．T を U 上の木とする．H の **tree congestion** を

$$\mathrm{tc}(H) = \min\{\mathrm{cng}_H(T) | T \text{ は } U \text{ 上の木 }\}$$

と定義する．T の枝は G の辺とは限らないことに注意されたい．$\mathrm{cng}_H(T^*) = \mathrm{tc}(H)$ となるような木 T^* を H の**最小混雑木** (**minimal congestion tree**) という．$a, b \in U$ に対し，$\mu_H(a, b)$ を H での辺素 $a - b$ パスの最大本数とし，$\mu_H = \max_{a,b \in U} \mu_H(a, b)$ とする．Ostrovskii[5] の結果の一部を次の定理にまとめている．

定理 2.1. ([5]) 任意のグラフ H に対して，
(1) $\mathrm{stc}(H) \geq \mu_H = \mathrm{tc}(H)$.

(2) 次数が μ_H 以下の頂点が葉であるような H の最小混雑木が存在する．

split グラフの最小混雑木に対して，次の補題が成り立つ．

補題 2.2. 任意の *split* グラフ G に対して，任意の $i \in I$ が葉であるような最小混雑木が存在する．

この補題から，今後 split グラフの最小混雑木では任意の $i \in I$ は葉であると仮定する．G の最小混雑木を T とする．$e \in T$ と $v \in L_e(T)$ に対して，$N_e(v) = N(v) \cap R_e(T)$ とする．$v \in R_e(T)$ のときも同様である．各 $v' \in N_e(v)$ に対して，$\{v, v'\}$ の迂回路が e を通るため，$|N_e(v)|$ は v の e の混雑度への寄与分と見ることができる．

3. IP 定式化と LP 緩和

この章では split グラフ上の全域木混雑度問題を多品種フローの容量最大値最小化問題を使った整数計画問題としての定式化を述べる．グラフ $G = (V, E)$ を入力とし，T を G の全域木とする．各辺 $\{s, t\} \in E(G)$ をフローのソースとシンクの組とみなし，s から t に 1 の整数フローを流す．ただし，フローを流すことができるのは T の辺のみである．s から t へのフローの流れが $\{s, t\}$ の迂回路に対応する．$\mathrm{cng}_G(T) = k$ であるかを調べるには，T の各辺の容量を k に設定し，フローを流すことができるかを調べればよい．このことを踏まえて，全域木混雑度問題を以下のように整数計画問題として定式化する．$x_{\{u,v\}}$ は辺 $\{u, v\} \in T$ ならば 1，それ以外なら 0 となる変数である．$f_{\{s,t\}}(u, v)$ は $\{u, v\} \in E(G)$ に対して，u から v へ流れる s, t 間のフローを表す．

$$\text{minimize} \quad k \qquad \text{(IP)}$$

subject to.

$$\sum_{\{s,t\} \in E} \{f_{s,t}(u,v) + f_{s,t}(v,u)\} \le k x_{\{u,v\}} \qquad \forall \{u,v\} \in E, \tag{1}$$

$$f_{s,t}(u,v) + f_{s,t}(v,u) \le x_{\{u,v\}} \quad \forall \{s,t\}, \{u,v\} \in E, \tag{2}$$

$$\sum_{w \in N(s)} f_{s,t}(s,w) = \sum_{w \in N(t)} f_{s,t}(w,t) = 1 \quad \forall s \in V, \tag{3}$$

$$\sum_{w \in N(v)} f_{s,t}(v,w) = \sum_{w \in N(v)} f_{s,t}(w,v) \qquad \forall v \in V \setminus \{s,t\}, \forall \{s,t\} \in E, \tag{4}$$

$$f_{s,t}(w,s) = f_{s,t}(t,w) = 0 \qquad \forall \{s,t\} \in E, w \in V, \tag{5}$$

$$\sum_{e \in E} x_e = |V| - 1, \tag{6}$$

$$x_e, f_{s,t}(u,v) \in \{0,1\} \qquad \forall e, \{s,t\}, \{u,v\} \in E. \tag{7}$$

この定式化において，k を (整数に) 固定すると，目的関数のない線形制約整数計画問題の実行可能性を問う問題になっている．k の最小化には，実行可能性について k に関するバイナリサーチを行う．各制約式の説明を加える．(1) 式は辺 $\mathrm{cng}_{G,T}(\{u, v\}) \le k$ を表す制約式である．ここでは k に $x_{\{u,v\}}$ をかけて条件を厳しくしている．(2) 式は $\{u, v\}$ 上を流れる s, t 間のフローは $x_{\{u,v\}}$ 以下であることを表している．(3) 式は各ソースから出るフローの合計は 1 であり，各シンクに入るフローの合計も 1 であることを表している．(4) 式はフロー保存則を表している．(5) 式はソースに入るフロー，シンクから出るフローは 0 であることを表している．(6) 式は全域木制約である．各辺の端点からもう一方の端点へ 1 のフローを流さなければならないので連結性もたもたれている．(7) 式の整数条件を除くことで LP 緩和が得られる．今後 IP を LP 緩和したものを LP と表記する．

3.1 split グラフ上の全域木混雑度の性質

split グラフ上の全域木混雑度の性質を述べ，その性質を定式化し，LP の制約式に加える．

補題 3.1. G を *split* グラフとする．このとき，次の条件を満たす全域木 T^* が存在する．

(1) 任意の $u \in I$ に対し，u が葉.

(2) $\mathrm{cng}_G(T^*) = \mathrm{stc}(G)$.

証明．G の全域木 T' を $\mathrm{cng}_G(T') = \mathrm{stc}(G)$ であり，葉でないような $u \in I$ が存在するような木とする．u の T' での隣接点を $v_1, v_2, \ldots, v_h$ とする．h は u の T' での隣接点の個数である．T' から条件 1,2 を満たすような木 T^* を構成する．

まず $\{u, v_2\}, \{u, v_3\}, \ldots, \{u, v_h\}$ を T' から除き，$\{v_1, v_2\}, \{v_1, v_3\}, \ldots, \{v_1, v_h\}$ を加える．このとき各 $i (2 \le i \le h)$ に対して，$\mathrm{cng}_{T'}(\{u, v_i\}) = \mathrm{cng}_{T^*}(\{v_1, v_i\})$ が成り立つ．なぜなら $L_{\{u,v_i\}}(T') = L_{\{v_1,v_i\}}(T^*)$ が成り立ち，カットされる辺の集合が等しいためである．

この操作により $\mathrm{cng}_{T^*}(\{u, v_1\}) \le \mathrm{stc}(G)$ が保存されることを示す．次数が 2 以上の $u \in I$ を持つ split グラフ G では，$\mu_G \ge |C|$ である．u の次数は高々 $|C|$ なので，定理 2.1 より，$\mathrm{cng}_{T^*}(\{u, v_1\}) \le |C| \le \mathrm{stc}(G)$ を得る．

この操作を繰り返し行うことで，条件 1,2 を満たすような全域木 T^* を構成できる． □

補題 3.1 より，split グラフ上の全域木混雑度問題の最適解では任意の $u \in I$ は葉であると仮定できる．葉から出ている辺は 1 本のみであり，この辺の混雑度は葉の次数である．よって LP に次の 2 式を加える．

$$\sum_{t \in N(i)} \{f_{i,t}(i,v) + f_{i,t}(v,i)\} \le \deg(i) x_{\{i,v\}} \qquad \forall i \in I, \forall v \in N(i), \tag{8}$$

$$\sum_{v \in N(i)} x_{\{i,v\}} = 1 \qquad \forall i \in I, \tag{9}$$

今後，(8),(9) 式を加えた LP を LP–split と表記する．

4. 乱択 2 近似アルゴリズム

4.1 アルゴリズム

Algorithm1 に本章のアルゴリズムをまとめている．$k^* = \mathrm{tc}(G)$ とする．split グラフ G と k を入力とする LP を $\mathrm{LP}(G,k)$ と表記する．アルゴリズムでは最小混雑木からクリーク内の部分木を得ている．その後，一部の変数が決定された LP を解き，その解に基づいてランダムに解を構成する．補題 3.1 から，クリーク内の部分木は連結であり，クリークなのでこの部分木は G の辺から構成されている．よって，I の頂点を G の辺で接続されるように付け替えれば G の全域木を得ることができる．次節で次の定理を示す.

定理 4.1. *Algorithm 1* の出力する木の混雑度を確率変数 Z であらわす．このとき，$\mathrm{E}[Z] \le 2\mathrm{stc}(G)$.

Algorithm 1 Randomized Algorithm for STC on split

```
1: G の最小混雑木 T を求める.
2: 任意の e ∈ E∩T に対して，x_e = 1 として LP(G, 2k*) を解く.
3: for i ∈ I do
4:    i の T での隣接点を u とする.
5:    確率 x_{i,v} で {i,v} を選び，T := T ∪ {{i,v}} \ {{i,u}} に
      更新する.
6: end for
7: return T
```

4.2 解析

$G' = G \cup T$ とする．明らかに T は G' の全域木である．また，加えた辺は I から出ている辺のみであり，さらに I の頂点は葉であるため，辺を加えたことによる混雑度の増加は加えた辺に対してしか発生しない．$i \in I$ から出ている辺の混雑度は $\deg(i) \le |C| \le \mathrm{tc}(G)$ である．よって $k^* = \mathrm{tc}(G)$ とすると，T は $\mathrm{LP}(G', k^*)$ の実行可能解である．この解を $(\boldsymbol{x^*}, \boldsymbol{f^*})$ で表す．ただし $\boldsymbol{x^*}$ は

$$x^*_e = \begin{cases} 1 & (e \in T) \\ 0 & (e \notin T) \end{cases}$$

とし，$\boldsymbol{f^*}$ は 0/1 解とする．以上の議論を観察としてまとめておく.

観察 4.2. $(\boldsymbol{x^*}, \boldsymbol{f^*})$ は $\mathrm{LP}(G', k^*)$ の実行可能解である．

$(\boldsymbol{x^*}, \boldsymbol{f^*})$ から $\mathrm{LP}(G, 2k^*)$ の小数実行可能解を構成する．次の補題は k' を $\mathrm{LP}(G,k')$ が実行可能であるように選んだときの，小数実行可能解の性質に関するものである．この補題から $\mathrm{LP}(G, 2k^*)$ に実行可能解が存在することを示す.

補題 4.3. k' を，任意の $e \in E \cap T$ に対して，$x_e = 1$ としたときの $\mathrm{LP}(G,k')$ が実行可能解を持つような整数とする．この実行可能解を $(\boldsymbol{x}, \boldsymbol{f})$ とし，$\{a,b\} \in E \cap T$ とする．各 $i \in I$ に対して，

$$\textstyle\sum_{j \in N(i)} \{f_{i,j}(a,b) + f_{i,j}(b,a)\} \le 2 \sum_{j \in N(i)} \{f^*_{i,j}(a,b) + f^*_{i,j}(b,a)\}.$$

証明．$j \in N(i)$ は任意とする．$i \in I$ をソース，j をシンクとしたとき，(3) 式より，i から j へ 1 のフローを流している．(2) 式より，任意の $m \in N(i)$ に対して $f_{i,j}(i,m) \le x_{\{i,m\}}$ が成り立つ．$(\boldsymbol{x}, \boldsymbol{f})$ が LP の実行可能解ならば，任意の $m \in N(i)$ に対してこの不等式が等号で成り立つことを背理法で示す．$f_{i,j}(i,m') < x_{\{i,m'\}}$ となる $m' \in N(i)$ が存在すると仮定する．(3) 式より $\sum_{w \in N(i)} f_{i,j}(i,w) = 1$ である．一方 (9) 式より $\sum_{w \in N(i)} x_{\{i,w\}} = 1$ であるため $\sum_{w \in N(i)} f_{i,j}(i,w) = \sum_{w \in N(i)} x_{\{i,w\}}$ が成り立つ．仮定より $f_{i,j}(i,m') < x_{\{i,m'\}}$ となる m' が存在するので，$f_{i,j}(i,m^*) > x_{\{i,m^*\}}$ となる $m^* \in N(i) (m^* \ne m')$ が存在する．これは (2) 式に矛盾する．以上から任意の $m \in N(i)$ に対して $f_{i,j}(i,m') = x_{\{i,m'\}}$ が成り立つ．任意の $j' \in N(i)$ をシンクとしたときも同様のことが言えるので，各 $\{i,m\} \in E$ について，

$$\sum_{j \in N(i)} f_{i,j}(i,m) = \deg(i) x_{\{i,m\}} \tag{10}$$

が成り立つ．(8),(10) 式より，i 以外の $i' \in I$ から流れるフローは i を経由せず，$T \cap G$ の辺上を流れる．よって $i' \ne i$ を満たす $i' \in I$ についても各 $j' \in N(i')$ に対して，$x_{\{i',j'\}}$ を固定すれば i と独立に同様の議論ができる.

$N_e(i)$ が空でないような辺 $e = \{a,b\}$ を考える．$j \in N_e(i)$ をシンクとするとき，$N_e(i)$ を経由して流れるフローは e を通らない．それ以外の $N(i)$ の点を経由して流れるフローは全て e を通る．よって，e を流れるフローの合計は

$$1 - \sum_{v \in N_e(i)} x_{\{i,v\}} \tag{11}$$

である．$j \in N(i) \setminus N_e(i)$ をシンクとするとき，$N_e(i)$ の点を経由するフローのみが e を通る．よって，e を流れるフローの合計は

$$\sum_{v \in N_e(i)} x_{\{i,v\}} \tag{12}$$

以上から，

$$\begin{aligned}
&\sum_{j \in N(i)} \{f_{i,j}(a,b) + f_{i,j}(b,a)\} \\
&= |N_e(i)|(1 - \sum_{v \in N_e(i)} x_{\{i,v\}}) + (\deg(i) - |N_e(i)|) \sum_{v \in N_e(i)} x_{\{i,v\}} \\
&= |N_e(i)| + (\deg(i) - 2|N_e(i)|) \sum_{v \in N_e(i)} x_{\{i,v\}} \qquad (13)
\end{aligned}$$

が成り立つ．ここで解の一つとして各 $j \in N(i)$ に対して

$x_{\{i,j\}} = \frac{1}{\deg(i)}$ と置くと，(13) 式は

$$
\begin{aligned}
&|N_e(i)| + (\deg(i) - 2|N_e(i)|) \sum_{v \in N_e(i)} x_{\{i,v\}} \\
&= |N_e(i)| + (\deg(i) - 2|N_e(i)|) \frac{|N_e(i)|}{\deg(i)} \\
&\leq 2|N_e(i)|
\end{aligned}
$$

$(\boldsymbol{x}^*, \boldsymbol{f}^*)$ において，j から $N_{\{a,b\}}(i)$ の頂点への迂回路は全て $\{a,b\}$ を通るので，
$\sum_{j \in N(i)} \{f^*_{i,j}(a,b) + f^*_{i,j}(b,a)\} = |N_{\{a,b\}}(i)|$ である．よって

$$
\begin{aligned}
&|N_e(i)| + (\deg(i) - 2|N_e(i)|) \frac{|N_e(i)|}{\deg(i)} \\
&\leq 2 \sum_{j \in N(i)} \{f^*_{i,j}(a,b) + f^*_{i,j}(b,a)\}
\end{aligned}
$$

$N_e(i)$ が空のときは I から流れるフローが e を通ることはないので，

$$
\begin{aligned}
&\sum_{j \in N(i)} \{f_{i,j}(u,v) + f_{i,j}(v,u)\} \\
&= \sum_{j \in N(i)} \{f^*_{i,j}(u,v) + f^*_{i,j}(v,u)\} = 0.
\end{aligned}
$$

□

補題 4.3 から直ちに次の補題が導ける．

補題 4.4. LP$(G, 2k^*)$ に対して，実行可能解が存在する．

証明． $S \subseteq V$ に対して，$E(S)$ で S によって誘導される辺集合とする．$T \cap G$ の各辺 $e = \{u,v\}$ に対して，

$$
\begin{aligned}
&\sum_{\{s,t\} \in E} \{f_{s,t}(u,v) + f_{s,t}(v,u)\} \\
&= \sum_{\{s,t\} \in E(C)} \{f_{s,t}(u,v) + f_{s,t}(v,u)\} \\
&\quad + \sum_{i \in I} \sum_{j \in N(i)} \{f_{i,j}(u,v) + f_{i,j}(v,u)\} \\
&\leq \sum_{\{s,t\} \in E(C)} \{f^*_{s,t}(u,v) + f^*_{s,t}(v,u)\} \\
&\quad + 2 \sum_{i \in I} \sum_{j \in N(i)} \{f^*_{i,j}(u,v) + f^*_{i,j}(v,u)\} \quad (\text{補題 4.3 より}) \\
&\leq 2 \sum_{\{s,t\} \in E} \{f^*_{s,t}(u,v) + f^*_{s,t}(v,u)\} \\
&\leq 2k^*. \qquad ((1) \text{ 式より})
\end{aligned}
$$

□

定理 4.1 の証明． 各 $e \in T', g \in E$ に対して，W^g_e を g の迂回路に e が含まれているなら 1，そうでないなら 0 となる確率変数とする．e を含む迂回路の本数，すなわち混雑度を $Y_e = \sum_{g \in E} W^g_e$ とする．$E(C)$ の迂回路はすでに決定されているので，$E(I)$ の迂回路について見ていく．e を含む $E(I)$ の迂回路の本数を $Y^I_e = \sum_{i \in I} \sum_{j \in N(i)} W^{\{i,j\}}_e$ とする．ある $i \in I$ と $e = \{a,b\}$ を固定し，$\sum_{j \in N(i)} W^{\{i,j\}}_e$ を考える．$N_e(i) = \emptyset$ のとき $\sum_{j \in N(i)} W^{\{i,j\}}_e = 0$ である．$N_e(i) \neq \emptyset$ のときを考える．$j \in N_e(i), j' \in N(i) \setminus N_e(i)$ とする．全域木の辺として $\{i,j\}$ が選ばれたとき，各 $\{i,j'\}$ の迂回路が e を通るので，

$$
\sum_{h \in N(i)} W^{\{i,h\}}_e = \deg(i) - |N_e(i)|
$$

となる．$\{i,j'\}$ が全域木の辺として選ばれたとすると，$\{i,j\}$ の迂回路が e を通るので，

$$
\sum_{h \in N(i)} W^{\{i,h\}}_e = |N_e(i)|
$$

となる．よって，

$$
\begin{aligned}
&\mathrm{E}\left[\sum_{h \in N(i)} W^{\{i,h\}}_e\right] \\
&= (\deg(i) - |N_e(i)|) \sum_{h \in N_e(i)} x_{\{i,h\}} \\
&\quad + (1 - \sum_{h \in N_e(i)} x_{\{i,h\}})|N_e(i)| \\
&= \sum_{h \in N(i)} \{f_{i,h}(a,b) + f_{i,h}(b,a)\}. \qquad ((13) \text{ より}) \qquad (14)
\end{aligned}
$$

$i' \in I (i' \neq i)$ から出るフローが i を経由することはないので，I の点から辺を選ぶ操作は独立である．よって (14) 式は任意の I の点に対して成り立つ．よって，

$$
\begin{aligned}
\mathrm{E}[Y^I_e] &= \sum_{i \in I} \mathrm{E}\left[\sum_{h \in N(i)} W^{\{i,h\}}_e\right] \\
&= \sum_{i \in I} \sum_{h \in N(i)} \{f_{i,h}(a,b) + f_{i,h}(b,a)\} \\
&\leq 2 \sum_{i \in I} \sum_{h \in N(i)} \{f^*_{i,h}(a,b) + f^*_{i,h}(b,a)\}. \quad (\text{補題 4.3 より})
\end{aligned}
$$

以上から任意の $e \in T'$ に対して，

$$
\begin{aligned}
\mathrm{E}[Y_e] &= \mathrm{E}\left[\sum_{g \in E} W^g_e\right] \\
&= \mathrm{E}\left[\sum_{g \in E(C)} W^g_e + Y^I_e\right] \\
&\leq \sum_{g \in E(C)} \{f^*_{s,t}(u,v) + f^*_{s,t}(v,u)\} \\
&\quad + 2 \sum_{i \in I} \sum_{h \in N(i)} \{f^*_{i,h}(a,b) + f^*_{i,h}(b,a)\} \\
&\leq 2k^*.
\end{aligned}
$$

が成り立つ．$k^* = \mathrm{tc}(G) \leq \mathrm{stc}(G)$ より，$\mathrm{E}[Z] = \mathrm{E}[\max_{e \in T} Y_e] \leq 2\mathrm{stc}(G)$. □

定理 4.1 では期待値が高々 $\mathrm{tc}(G)$ の 2 倍以内に収まることしか示していない．次章で本章のアルゴリズムの脱乱択化を目指し，反復丸めアルゴリズムを用いた近似アルゴリズムを与える．

5. 反復丸めを用いた近似アルゴリズム

この章では 4 章のアルゴリズムの脱乱択化を目指して，決定的アルゴリズムを設計する．本章のアルゴリズムとその解析は [2] を参考にしている．
G の最小混雑木を T とする．アルゴリズムの方針は 4 章と同じで，T からクリーク内の部分木を決定したあと，全域木を得るために I の頂点を G の辺で接続されるように付け替える．クリーク内の部分木を $T(C)$ とする．$T(C)$ の枝で，葉から出ているものの集合を C_L，それ以外の $T(C)$ の枝の集合を C_{IN} とする．本章のアルゴリズムでは付け替える際に生じる問題を線形計画問題として定式化し，反復丸めアルゴリズムを適用することで $2\alpha+2$ 近似アルゴリズムを得る．ただし $\alpha = \lceil \frac{|C|-1}{|C_L|} \rceil$ である．
付け替えが行われる際の $e \in T(C)$ の混雑度の変化について考察する．$u \in I_{L_e(T)}$ としても一般性を失わない．u が $v \in C_{R_e(T)}$ の点へ付け替えられたとき，u の e への寄与分は $\deg(u) - |N_e(u)|$ となる．T での u の e への寄与分は $|N_e(u)|$ だったので，付け替えたことによる寄与分の増加は $\deg(u) - 2|N_e(u)|$ である．u が $C_{L_e(T)}$ の点へ付け替えられたとき，u の e への寄与分は T のときと変わらない．ここで生じる問題は，全域木に作り変える際，各辺の混雑度が 4 章で導出した期待値を超えないように I の頂点を付け替えることである．この問題を以下のように定式化した．任意の $u \in I$ と $v \in N(u)$ に対して，$x_{\{u,v\}}$ を $\{u,v\}$ が全域木の辺となるなら 1，そうでないなら 0 をとる変数とする．各 $e \in T(C)$ に対して，$B_e = \sum_{u \in I} |N_e(u)|$ とする．端点の一方が C に属しており，もう一方が I に属しているような辺の集合を F とする．

$$
\begin{aligned}
&\text{minimize} \quad 0 && \text{(LP–split2)} \\
&\text{subIect to.} && (15) \\
&\sum_{v \in N(u)} x_{\{u,v\}} = 1 \qquad \forall u \in I, && (16) \\
&\sum_{u \in I_{R_e(T)}} \sum_{v \in N_e(u) \cap C_{L_e(T)}} (\deg(u) - 2|N_e(u)|) x_{\{u,v\}} \\
&\quad + \sum_{u \in I_{L_e(T)}} \sum_{v \in N_e(u) \cap C_{R_e(T)}} (\deg(u) - 2|N_e(u)|) x_{\{u,v\}} \\
&\qquad \le B_e \qquad \forall e \in T(C), && (17) \\
&x_f \ge 0 \qquad \forall f \in F. && (18)
\end{aligned}
$$

この定式化では期待値以内の解を見つけることに重点を置いているため，目的関数は特に定めない．
補題 5.1. *LP–split2* は実行可能解を持つ．

証明． 任意の $u \in I$ と任意の $v \in N(u)$ に対して，$x_{\{u,v\}} = \frac{1}{\deg(u)}$ と置く．この解が LP–split2 の実行可能解であることを示す．明らかに，任意の $f \in F$ に対して (18) 式を満たす．任意の $u \in I$ に対して

$$
\begin{aligned}
\sum_{v \in N(u)} x_{\{u,v\}} &= \deg(u) \frac{1}{\deg(u)} \\
&= 1
\end{aligned}
$$

となるので (16) 式を満たす．また任意の $e \in C_{IN}$ に対して

$$
\begin{aligned}
&\sum_{u \in I_{R_e(T)}} \sum_{v \in N_e(u)} \frac{\deg(u) - 2|N_e(u)|}{\deg(u)} \\
&+ \sum_{u \in I_{L_e(T)}} \sum_{v \in N_e(u)} \frac{\deg(u) - 2|N_e(u)|}{\deg(u)} \\
&\le \sum_{u \in I_{R_e(T)}} \sum_{v \in N_e(u)} 1 + \sum_{u \in I_{L_e(T)}} \sum_{v \in N_e(u)} 1 \\
&= \sum_{u \in I_{L_e(T)}} |N_e(u)| + \sum_{u \in I_{R_e(T)}} |N_e(u)| \\
&= \sum_{u \in I} |N_e(u)| = B_e
\end{aligned}
$$

となるので (17) も満たす． □

一般的な線形計画問題に対して成り立つ次の補題を導入する．
補題 5.2. 制約式の本数が m，変数の個数が n であるような線形計画問題の実行可能解を z とする．z が端点解であるための必要十分条件は n 本の線形独立な制約式を等式で満たすことである．

補題 5.2 から，今後の議論で重要な LP の端点解の性質に関する補題を導くことができる．
補題 5.3. *x を全ての要素が非零な LP–split2 の端点解とする．C^*_L, C^*_{IN} をそれぞれ (17) 式を等号でみたす葉からでている枝の集合と，それ以外の枝の集合とする．このとき，$|F| = |I| + |C^*_L| + |C^*_{IN}|$.*

証明． 補題 5.2 から，x が端点解であるとき，$|F|$ 本の制約式を等号で満たしている．任意の $f \in F$ に対して $x_f > 0$ なので，等号を満たしている制約式は $|I| + |C^*_L| + |C^*_{IN}|$ 本のみである．よって $|F| = |I| + |C^*_L| + |C^*_{IN}|$. □

5.1 反復丸めアルゴリズム

Algorithm2 に反復丸めアルゴリズムを載せる．$\deg_F(v), N_F(v)$ で辺集合が F のときの v の次数と，v の隣接点の集合を表す．アルゴリズムでは，各反復の最初に LP–split2 を解き，この LP 解に整数変数があるならばそれを解の一部に採用する．整数辺が無い場合は $T(C)$ の葉に 2α 個の I' の点が F' の辺で接続されていることが保証される (補題 5.6) ため，それらの辺を全て解として採用す

る．途中，制約式を違反した場合，違反分をその制約式の右辺に足し，実行可能性を保つようにする．このアルゴリズムに対して，次の補題が成り立つ．次節で証明する．

補題 5.4. $e \in T(C)$ は任意とする．x^* をアルゴリズムから得られた *LP–split2* の整数解とする．このとき，

$$\begin{aligned}&\sum_{u \in I_{R_e(T)}} \sum_{v \in N_e(u)} (\deg(u) - 2|N_e(u)|) x^*_{\{u,v\}} \\ &+ \sum_{u \in I_{L_e(T)}} \sum_{v \in N_e(u)} (\deg(u) - 2|N_e(u)|) x^*_{\{u,v\}} \\ &\leq B_e + (2\alpha + 1)|C_{R_e(T)}||C_{L_e(T)}|.\end{aligned}$$

この補題から，ただちに次の定理を得る．

定理 5.5. 任意の *split* グラフに対して，$\mathrm{stc}(G) \leq (2\alpha + 2)\mathrm{tc}(G)$.

証明． 任意の $e \in T'(C)$ について，$\mathrm{cng}_T(e) = B_e + |C_{R_e(T)}||C_{L_e(T)}|$ である．定理 5.4 より，

$$\begin{aligned}\mathrm{stc}(G) &\leq \mathrm{cng}_{T'}(e) \\ &\leq \mathrm{cng}_T(e) + B_e + (2\alpha + 1)|C_{R_e(T)}||C_{L_e(T)}| \\ &= 2B_e + (2\alpha + 2)|C_{R_e(T)}||C_{L_e(T)}| \\ &\leq (2\alpha + 2)\mathrm{cng}_T(e) \\ &\leq (2\alpha + 2)\mathrm{tc}(G).\end{aligned}$$

□

Algorithm 2 Iterative Algorithm for STC on split

1: G の最小混雑木 T を求める．
2: $T' := T(C), I' := I, F' := F, C' := C$
3: **while** $I' \neq \emptyset$ **do**
4: $G' = (C \cup I', F')$ 上で LP–split2 を解く．
5: 各 $x_f = 0$ となる辺を全て除く．
6: 各 $u \in I'$ と $v \in N(u)$ に対して，$x_{\{u,v\}} = 1$ ならば
$T' := T' \cup \{u, v\}, I' := I' \setminus \{u\}$,
混雑度が増加する全ての辺 e に対して，
$B_e := B_e - (\deg(u) - 2|N_e(u)|)$ に更新．
7: $\deg_{F'}(v) = 0$ であるような $v \in I' \cap C'_L$ を全て除く．
8: 各 $v \in C'_L$ に対して，$\deg_{F'}(v) \leq 2\alpha$ ならば
$I' := I' \setminus N_{F'}(v), C' := C' \setminus \{v\}$,
T' に v から出ている F' の辺を全て加える．
制約を違反した辺に対して，超過分を対応する制約式の右辺に足す．
9: **end while**
10: **return** T'

5.2 アルゴリズムの解析

この章では Algorithm2 の解析を行う．まず，アルゴリズムの停止性を保障する次の補題を示す．

補題 5.6. x を全ての要素が小数値であるような *LP–split2* の端点解とする．このとき，$\deg_F(v) \leq 2\alpha$ となるような頂点 $v \in C_L$ が存在する．

証明． 全ての $v \in C_L$ に対して，$\deg_F(v) > 2\alpha$ と仮定し，矛盾を導く．各 $u \in I$ に対して，$\sum_{v \in N(u)} x_{\{u,v\}} = 1$ であり，仮定から $x_f = 1$ となる変数は存在しない．よって各 $u \in I$ に対して

$$\deg_F(u) \geq 2 \tag{19}$$

が成り立つ．補題 5.3 より，$|F| = |I| + |C^*_L| + |C^*_{IN}|$ が成り立つ．よって

$$\begin{aligned}&|I| + |C^*_L| + |C^*_{IN}| \\ &= |F| \\ &= \frac{\sum_{u \in I} \deg_F(u) + \sum_{v \in C} \deg_F(v)}{2} \qquad \text{(握手補題より)} \\ &> |I| + |C_L|\alpha + \frac{\sum_{v \in C \setminus C_L} \deg_F(v)}{2} \\ &\geq |I| + |C| - 1 \\ &\geq |I| + |C_L| + |C_{IN}| \\ &\geq |I| + |C^*_L| + |C^*_{IN}|\end{aligned}$$

となり，矛盾． □

この補題から，整数変数が存在しないときはアルゴリズムの 8 行目の処理に入ることが保証される．よって各反復の開始前に比べて開始後は必ず入力のサイズが真に小さくなっている．このことから任意の split グラフに対してアルゴリズムは停止する．定理 5.4 の証明に入る前に次の補題を証明する．

補題 5.7. *Algorithm2* の最初の反復において，*7* 行目の終了時点で $|I'| \leq |C| - 1$ が成り立つ．

証明． 最初の反復での LP–split2 の端点解を x^1 とする．変数の数は $|F|$ 個，制約式の数は $|I| + |C| - 1$ 本なので，補題 5.2 より，少なくとも $|F| - |I| - |C| + 1$ 個の x^1 の要素の値が 0 である．$|I| > |C| - 1$ のとき，I_β を小数辺の端点であるような I の頂点の集合とし，I_γ を整数辺 ($x_f = 1$) の端点であるような I の頂点の集合とする．各 $u \in I_\beta$ は (16) 式より，少なくとも 2 本の辺を持つ．各 $u \in I_\gamma$ はちょうど 1 本の辺を持つ．$|I_\beta| + |I_\gamma| = |I|$ かつ $|I| + |C| - 1 \geq 2|I_\beta| + |I_\gamma|$ なので，$|I_\gamma| \geq |I| - |C| + 1$ が成り立つ．I_γ は全てアルゴリズムの 4 行目で除かれるため，$|I'| = |I| - |I_\gamma| \leq |C| - 1$. □

補題 *5.4* の証明． $e \in T(C)$ は任意とする．一般性を失わず $|C_{R_e(T)}| \leq |C_{L_e(T)}|$ とする．アルゴリズムの 6 行目では制約式を破ることはないので，6 行目の操作による混雑度の増加は高々 B_e である．

8 行目のときの混雑度の増加を考える．$u \in I_{L_e(T)}$ とする．アルゴリズムによって u が $C_{R_e(T)}$ の頂点に付け替えられたとする．このとき，

Vol.2014-AL-150 No.17
2014/11/21

$$
\begin{aligned}
\deg(u) - 2|N_e(u)| &= |N(u) \cap C_{L_e(T)}| \\
&\quad + |N(u) \cap C_{R_e(T)}| - 2|N_e(u)| \\
&= |N(u) \cap C_{L_e(T)}| + |N_e(u)| - 2|N_e(u)| \\
&\leq |C_{L_e(T)}| - |N_e(u)| \\
&\leq |C_{L_e(T)}|.
\end{aligned}
$$

よって e の混雑度は高々 $|C_{L_e(T)}|$ 増加する．$u \in I'_{R_e(T)}$ のときも同様である．8 行目での e の混雑度の増加の総和は高々

$$
\begin{aligned}
&\sum_{u \in I'_{L_e(T)}} |C_{L_e(T)}| + \sum_{u \in I'_{R_e(T)}} |C_{R_e(T)}| \\
&= |I'_{L_e(T)}||C_{L_e(T)}| + |I'_{R_e(T)}||C_{R_e(T)}|
\end{aligned}
$$

である．8 行目ではクリークの頂点 1 個につき高々 2α 個の独立集合の頂点が接続されるので，

$$|I'_{R_e(T)}| \leq 2\alpha|C_{L_e(T)}|, \tag{20}$$

$$|I'_{L_e(T)}| \leq 2\alpha|C_{R_e(T)}| \tag{21}$$

が成り立つ．補題 5.7 から，

$$|I'_{R_e(T)}| + |I'_{L_e(T)}| \leq |C| \tag{22}$$

である．以上から，

$$
\begin{aligned}
&|I'_{L_e(T)}||C_{L_e(T)}| + |I'_{R_e(T)}||C_{R_e(T)}| \\
&\leq |I'_{L_e(T)}||C_{L_e(T)}| + (|C| - |I'_{L_e(T)}|)|C_{R_e(T)}| \\
&\qquad\qquad (\text{(22) 式より}) \\
&= |I'_{L_e(T)}|(|C_{L_e(T)}| - |C_{R_e(T)}|) + |C||C_{R_e(T)}| \\
&\leq 2\alpha|C_{R_e(T)}|(|C_{L_e(T)}| - |C_{R_e(T)}|) + |C||C_{R_e(T)}| \\
&\qquad\qquad (\text{(21) 式より}) \\
&= (2\alpha+1)|C_{L_e(T)}||C_{R_e(T)}| - (2\alpha-1)(|C_{R_e(T)}|)^2 \\
&\leq (2\alpha+1)|C_{L_e(T)}||C_{R_e(T)}|.
\end{aligned}
$$

最後から 2 番目の等号は $|C| = |C_{R_e(T)}| + |C_{L_e(T)}|$ から導かれる．以上の議論から，8 行目で e の混雑度は高々 $(2\alpha+1)|C_{R_e(T)}||C_{L_e(T)}|$ だけ増加する．よってアルゴリズム全体での e の混雑度の増加は高々 $B_e + (2\alpha + 1)|C_{R_e(T)}||C_{L_e(T)}|$. □

5.3 split グラフ上の全域木混雑度の上界

この節では，前節と同様の手法を用いたアルゴリズムを設計し，split グラフの全域木混雑度の上界を導出する．$\deg(v_1) \geq \deg(v_2) \geq, ..., \geq \deg(v_{|V|})$ とする．次の定理を証明する．

定理 5.8. 任意の *split* グラフに対して，$\mathrm{stc}(G) \leq 2\deg(v_2)$.

本節で設計するアルゴリズムでは最小混雑木の代わりに v_1 を中心とするスターを求め，I の各点を全域木となるように付け替える．スターの場合の I の付け替えの問題を定式化する．T がスターのとき，各 $u \in I$ の $e = \{v_1, v\}$ への寄与分 $|N_e(u)|$ は

$$
|N_e(u)| = \begin{cases} 1 & (v \in N(u)) \\ 0 & (v \notin N(u)) \end{cases} \tag{23}
$$

である．よって $B_e = \sum_{u \in I} |N_e(u)| = |N(v) \cap I|$ である．e の混雑度は v に I の頂点が付け替えられるとき，またそのときに限り高々 $\deg(u) - 2$ 増加する．T がスターのときの定式化は以下の通りである．

$$
\begin{aligned}
&\text{minimize} \quad 0 && \text{(LP–split2–star)} \\
&\text{subIect to} && \\
&\sum_{v \in N(u)} x_{\{u,v\}} = 1 && \forall u \in I, \quad (24) \\
&\sum_{u \in I} (\deg(u) - 2) x_{\{u,v\}} \leq B_{\{v,v_1\}} && \forall v \in C \setminus \{v_1\}, \quad (25) \\
&x_f \geq 0 && \forall f \in F. \quad (26)
\end{aligned}
$$

次の補題が成り立つ．

補題 5.9. x を全ての要素が非零な *LP–split2–star* の端点解とする．C^* を *(25)* を等号でみたす枝の集合とする．このとき，$|F| = |I| + |C^*|$.

証明． 補題 5.3 と同様. □

Algorithm3 に本節のアルゴリズムを載せている．アルゴリズムが正しく動作することを保証するための補題を証明する．方針はほぼ補題 5.6 と同様である．

Algorithm 3 Iterative Algorithm for STC on split(star)

1: v_1 を中心とするスターを構成する．
2: $T' := T(C), I' := I, F' := F, C' := C$
3: **while** $I' \neq \emptyset$ **do**
4: $G' = (C' \cup I', F')$ 上で LP–split2–star を解く．
5: 各 $x_f = 0$ となる辺を全て除く．
6: 各 $u \in I'$ と $v \in N(u)$ に対して，$x_{\{u,v\}} = 1$ ならば $T' := T' \cup \{u, v\}, I' := I' \setminus \{u\}$, $B_{\{v,v_1\}} := B_{\{v,v_1\}} - (\deg(u) - 2)$ に更新.
7: $\deg_{F'}(v) = 0$ となる点を全て除く．
8: 「$\deg_{F'}(v) = 2$ かつ $\sum_{u \in I} x_{\{u,v\}} \geq 1$」または「$\deg_{F'}(v) = 1$」を満たす $v \in C' \setminus \{v_1\}$ が存在するなら $I' := I' \setminus N_{F'}(v), C' := C' \setminus \{v\}$ とし，T' に v から出ている F' の辺を全て加える．
9: **end while**
10: **return** T'

補題 5.10. x を全ての要素が小数値であるような *LP–split2–star* の端点解とする．このとき，次の条件のどちらか 1 つを満たすような頂点 $v \in C \setminus \{v_1\}$ が存在する．

(1) $\deg_{F'}(v) = 1$,

(2) $\deg_{F'}(v) = 2$ かつ $\sum_{u \in I} x_{\{v,u\}} \geq 1$.

証明. (1),(2) の両方を満たさないと仮定し，矛盾を導く．
(i)「任意の $v \in C \setminus \{v_1\}$ に対して，$\deg_F(v) \neq 1$ かつ $\deg_F(v) \neq 2$」
$\deg_F(v) = 0$ となる $v \in C \setminus \{v_1\}$ はアルゴリズムにより除かれるので，任意n $v \in C \setminus \{v_1\}$ に対して，$\deg_F(v) > 2$ と仮定する．24 式を満たし，かつ任意の f に対して $x_f < 1$ なので，任意の $u \in I$ に対して $\deg_F(u) \geq 2$ が成り立つ．補題 5.9 から，$|F| = |I| + |C^*|$ である．よって

$$
\begin{aligned}
&|I| + |C^*| \\
&= |F| \\
&= \frac{\sum_{u \in I} \deg_F(u) + \sum_{v \in C} \deg_F(v)}{2} && \text{(握手補題より)} \\
&> |I| + |C| - 1 && \text{((19) 式と背理法の仮定より)} \\
&\geq |I| + |C^*|
\end{aligned}
$$

となり，矛盾．
(ii)「任意の $v \in C \setminus \{v_1\}$ に対して $\deg_F(v) \neq 1$ かつ $\sum_{u \in I} x_{\{v,u\}} < 1$」
(i) と同様の理由で任意の $v \in C \setminus \{v_1\}$ に対して $\deg_F(v) \geq 2$ と仮定できる．また，任意の $u \in I'$ に対して，$\deg_{F'}(u) \geq 2$ が成り立つ．補題 5.9 より，

$$
\begin{aligned}
&|I| + |C^*| \\
&= |F| \\
&= \frac{\sum_{u \in I} \deg_F(u) + \sum_{v \in C} \deg_F(v)}{2} && \text{(握手補題より)} \\
&\geq |I| + |C| - 1 && \text{((19) 式と背理法の仮定より)} \\
&\geq |I| + |C^*|.
\end{aligned}
$$

よって全ての不等式が等号で成り立つ．すなわち任意の $w \in I \cup C \setminus \{v_1\}$ に対して，$\deg_F(w) = 2$ であり，$\deg_F(v_1) = 0$ である．$|F| = 2|I|$ かつ $|F| = 2|C|$ なので，$|I| = |C|$ である．(24) 式より，$\sum_{f \in F} x_f = \sum_{u \in I} \sum_{v \in N(u)} x_{\{u,v\}} = |I|$ である．一方，任意の $v \in C \setminus \{v_1\}$ に対して，$\sum_{u \in I} x_{\{v,u\}} < 1$ なので，$\sum_{v \in C} \sum_{u \in I} x_{\{v,u\}} < |C|$ である．これは $|I| = |C|$ に矛盾する． □

Algorithm3 が出力する全域木の混雑度が高々 $2\deg(v_2)$ であることを証明する．

補題 5.11. T はスターとする．$v \in C \setminus \{v_1\}$ は任意とする．このとき，*Algorithm3* の出力 T' の各 $e = \{v_1, v\}$ の混雑度は高々 $\deg(v) + B_e + |C| - 2 \leq 2\deg(v)$.

証明. 定理 5.4 のときと同様，アルゴリズムの 6 行目では制約式を違反しないので，混雑度の増加は高々 B_e である．$\deg_{F'}(v) = 1$ となるような $v \in C' \setminus \{v_1\}$ が存在するとき，v に高々 1 つ I の頂点が付け替えられ，これ以上 e の混雑度が増加することはない．$\deg(u) \leq |C|$ なので，Algorithm2 による e の混雑度の増加は高々 $B_e + \deg(u) - 2 \leq B_e + |C| - 2$ である．
$\deg_{F'}(v) = 2$ かつ $\sum_{u \in I} x_{\{v,u\}} \geq 1$ となる頂点 $v \in C \setminus \{v_1\}$ が存在するとき，v に 2 つの頂点が付け替えられる．この頂点を u_1, u_2 とする．u_1, u_2 が v に付け替えられる直前までに v に付け替えられた頂点の集合を I_v^* とする．アルゴリズムから $B_{\{v_1,v\}} \geq \sum_{u \in I_v^*} (\deg(u) - 2) + \sum_{i=1,2} (\deg(u_i) - 2) x_{\{u_1,v\}}$ である．u_1, u_2 の付け替え後は高々

$$
\begin{aligned}
&\sum_{u \in I_v^*} (\deg(u) - 2) + \sum_{i=1,2} (\deg(u_i) - 2) \\
&\leq B_{\{v,v_1\}} + \sum_{i=1,2} (\deg(u_i) - 2)(1 - x_{\{u_i,v\}}) \\
&\leq B_{\{v,v_1\}} + (2 - \sum_{i=1,2} x_{\{u_i,v\}})(|C| - 2) \\
&\leq B_{\{v,v_1\}} + |C| - 2.
\end{aligned}
$$

最後の不等式は $\sum_{i=1,2} x_{\{u_i,v\}} \geq 1$ より導かれる．$\deg(v) = B_e + |C| - 1$ なので，補題が成り立つ． □

定理 *5.8* の証明. T は v_1 を中心とするスターなので，$\mathrm{cng}_G(T) = \deg(v_2)$．補題 5.11 より，アルゴリズムの出力 T' と任意の $v \in C \setminus \{v_1\}$ に対して，$\mathrm{cng}_{T'}(\{v_1, v\}) \leq 2\deg(v) \leq 2\deg(v_2)$．よって $\mathrm{stc}(G) \leq \mathrm{cng}_G(T') \leq 2\deg(v_2)$． □

6. おわりに

本稿では split グラフ上の全域木混雑度問題についてのべた．まず LP 緩和を行い，それに基づく乱択丸め近似アルゴリズムを設計した．次に反復丸めアルゴリズムを用いた近似アルゴリズムを設計した．今後の課題は近似アルゴリズムの改善と，今回の反復丸めアルゴリズムを別のグラフに応用できるかどうかを調査することなどが挙げられる．

参考文献

[1] K. Kozawa, Y. Otachi, K. Yamazaki *On spanning tree congestion of graphs*, Discrete Mathematics, 309 (2009), pp. 4215–4224.
[2] L. C. Lau, R. Ravi, M. Singh, *Iterative methods in combinatorial optimization*, Cambridge University Press (2011).
[3] C. Lowenstein, D. Rautenbach, F. Regen, *On spanning tree congestion*, Discrete Mathematics, 310 (2010), pp. 4653–4655.
[4] Y. Okamoto, Y. Otachi, R. Uehara, T. Uno, *Hardness results and an exact exponential algorithm for the spanning tree congestion problem*, Lecture Notes in Computer Science, 6648 (2011), pp. 452–462.
[5] M. I. Ostrovskii, *Minimal congestion trees*, Discrete Mathematics, 285 (2004), pp. 219–226.
[6] S. Simonson, *A variation on the min cut linear arrangement problem*, Mathematical System Theory, 20 (1987), pp. 235–252.