

定数コストの横断辺を持つ梯子状ネットワークの最適化

林 克幸^{1,a)} 松林 昭^{1,b)}

概要：利己的ルーティングの下では、ネットワークの一部のリンクを削除することでナッシュフローのコストが減少する場合がある。一般のネットワークにおいてそのようなリンクを特定するネットワーク最適化問題は NP 困難である。さらに、ナッシュフローのコストを多項式時間で最小化できるような自明でない十分条件でさえも知られていない。本報告では定数コストの横断辺を持つ n ノードの梯子状ネットワークに対して、動的計画法を用いて最小コストのナッシュフローを持つ全域部分ネットワークを求める $O(n^3)$ 時間アルゴリズムを示す。

1. はじめに

近年では PC を接続することによって多くのネットワークが形成されている。また、都市とそれらを結ぶ道路についても、ネットワークという形で表現することができる。このようなネットワークにおいて各ユーザは使用する経路を決定する必要があるが、それらを集中的な制御で行うことはしばしば困難である。そうした場合にはユーザ自身が経路選択を行わなければならない。その時に各ユーザが他のユーザの利益を考えずに自分自身の利益が最大となるような戦略をとることがある。このような戦略に基づく経路選択を利己的ルーティングと呼ぶ。

利己的ルーティングの下ではリンクの追加がネットワークの能力を低下させる場合があることが知られている [1]。すなわち、各ユーザのコスト（例えば経路を通過するのに要する時間）の平均値が、ネットワークにリンクを追加することで増加することがある。この直感に反する現象は、Braess のパラドックスと呼ばれている [1]。与えられたネットワークに対して、能力を低下させているリンクを検出し、利己的ユーザのコストが最小となるような全域部分ネットワークを求める問題（ネットワーク設計問題、またはネットワーク最適化問題）は重要な問題である。

Roughgarden[2] はネットワーク最適化問題が NP 困難であることを示した。また、各リンクのコスト関数が線形である場合、入力されたネットワークをそのまま出力する自明なアルゴリズムが近似比 $\frac{4}{3}$ を持つことを示した。さら

に $P=NP$ でない限り、コスト関数が線形の一般のネットワークに対して $(\frac{4}{3} - \epsilon)$ -近似多項式時間アルゴリズムが存在しないことを示した。

一方で Fotakis ら [3] はパスの数がリンク数の多項式であり各パス長がリンク数の対数多項式となっていて、各リンクのコスト関数が線形であるようなネットワークに対して、準指数時間で最適な部分ネットワークのナッシュフローとそのコストが計算できることを示した。

一般のネットワークを最適化する問題は NP 困難であるが、効率的に最適化可能なネットワークの十分条件を示すことは応用上重要なことである。Frank[4] は、Braess がパラドックスとして示したネットワークの一般化として、線形コスト関数を持つ梯子状ネットワークを考え、ナッシュフローにおいて全てのリンクにフローが存在する場合に、横断リンクのコスト関数を増加させた時にナッシュフローのコストが減少する必要条件と十分条件を示した。本研究では、横断リンクのコスト関数が非負定数であり、その他のリンクのコスト関数が線形であるような梯子状ネットワークを最適化する多項式時間アルゴリズムを示す。この結果は、多項式時間で最適化可能なネットワークの自明でない最初の十分条件を与える。

2. 準備

2.1 梯子状ネットワーク

本研究では、図 1 で示すような点集合 $V = \{v_{i,j} | 1 \leq i \leq 2, 1 \leq j \leq n\}$ 、有向辺集合 $E_1 = \{(v_{1,j}, v_{1,j+1}) | 1 \leq j \leq n-1\}$ と、それを横断する有向辺の集合 $E_2 = \{(v_{1,j}, v_{2,j}) | 1 \leq j \leq n\}$ からなる梯子状ネットワーク $G_n = (V, E)$ を考える。ただし $E = E_1 \cup E_2$ である。辺 $(v_{1,i}, v_{2,i})$ を e_i とする。

¹ 金沢大学大学院自然科学研究科電子情報科学専攻
Division of Electrical and Computer Engineering,
Kanazawa University, Kakuma, Ishikawa 920-1192, Japan

a) khayashi@genie.ec.t.kanazawa-u.ac.jp

b) mbayashi@t.kanazawa-u.ac.jp

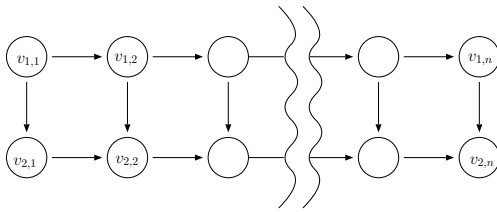


図 1 梯子状ネットワーク

Fig. 1 Ladder Network

すべての辺 $e \in E$ はコスト関数 $c_e(f_e)$ を持つ。ここでは辺 $(v_{1,i}, v_{1,i+1})$ と辺 $(v_{2,i}, v_{2,i+1})$ のコスト関数をそれぞれ $a_i x + b_i$, $a'_i x + b'_i$ ($a_i, a'_i, b_i, b'_i \geq 0$) とし, e_i のコスト関数を非負の定数 d_i とする。コスト関数の集合を c とする。

2.2 フロー

グラフ G と, G の始点 s と終点 t が与えられているとする。本研究では, G_n の始点を $v_{1,1}$, 終点を $v_{2,n}$ と仮定する。以後特に断らない限り, 単にパスと書いたときは, 着目しているグラフの始点と終点を結ぶ有向パスを表わすものとする。パスの集合を $\mathcal{P}$ とし, $P \in \mathcal{P}$ を通るフローの量を f_P で表す。

$f_e = \sum_{P \in \mathcal{P}: e \in P} f_P$ を辺 e を流れるフローの量とする。本研究では単一コモディティ, つまり単一の始点と終点の対のみを考えるため, s から t へ向かう有限で正のフローの量を正規化して 1 とする。 $f_P = \{f_P | P \in \mathcal{P}\}$ とし, $\sum_{P \in \mathcal{P}} f_P = 1$ となる f_P を許容フローと呼ぶ。パス P のコスト $c_P(f_P)$ をパスを構成するすべての辺のコストの総和と定義する。すなわち $c_P(f_P) = \sum_{e \in P} c_e(f_e)$ である。

インスタンス (G, c) に対する許容フロー f_P のコスト $C(f_P)$ は, 各パスのコストとフローの量の積の総和で表され, $C(f_P) = \sum_{P \in \mathcal{P}} c_P(f_P) f_P = \sum_{P \in \mathcal{P}} \sum_{e \in P} c_e(f_e) f_P$ である。

2.3 ナッシュフロー

$f_{P_1} > 0$ である各 $P_1, P_2 \in \mathcal{P}$ に対して $\sum_{e \in P_1} c_e(f_e) \leq \sum_{e \in P_2} c_e(f_e)$ となるような f_P を, ナッシュフローと呼ぶ。これは, フローが流れているパスのコストは他のパスのコスト以下であるということを表している。利己的ルーティングではナッシュフローのとき, 各ユーザーは自分の経路を変更しなくなり均衡状態となる (この状態をナッシュ均衡と呼ぶ)。

任意のインスタンス (G, c) に対してナッシュフローが存在することが知られている [5]。また, (G, c) のナッシュフローのコストは一意的に定まることが知られている [6]。

2.4 ネットワーク最適化

インスタンス (G, c) に対して, G の全域部分ネットワークのなかでナッシュフローのコストが最小となるネット

ワーク H を G の最適ネットワークと呼ぶ。このような全域部分ネットワーク H を構成をネットワーク最適化と呼ぶ。

3. アルゴリズム

入力インスタンスを (G_n, c) とし, 最適ネットワークを出力するアルゴリズムを示す。

このアルゴリズムは G_n の横断辺の部分集合を削除することにより最適ネットワークを構成する。これは, $(v_{1,i}, v_{1,i+1})$ が無いネットワークは $e_{i+1}, e_{i+2}, \dots, e_n$ が無いネットワークとフローが存在する辺が等しくなり, 同様に $(v_{2,i}, v_{2,i+1})$ が無いネットワークは $e_1, \dots, e_i$ が無いネットワークとフローが存在する辺が等しくなるからである。

また, 以下では $a_i = a'_i = 0$ となる $1 \leq i \leq n$ が存在しないと仮定する。 $a_i = a'_i = 0$ であるような辺 $(v_{1,i}, v_{1,i+1})$, $(v_{2,i}, v_{2,i+1})$ が存在する場合を考える。この時 $(v_{1,i}, v_{1,i+1})$, $(v_{2,i}, v_{2,i+1})$, e_i , e_{i+1} を通るためのコストは, これらの辺を流れるフローに関係なくそれぞれ b_i , b'_i , d_i , d_{i+1} である。 $v_{1,i}$ から $v_{1,i+1}$ を通り $v_{2,i+1}$ に向かう場合にかかるコストは $b_i + d_{i+1}$ であり, $v_{1,i}$ から $v_{2,i}$ を通り $v_{2,i+1}$ に向かう場合にかかるコストは $d_i + b'_i$ である。 $b_i + b_{i+1} > d_i + b'_i$ の場合, $v_{1,i}$ から $v_{1,i+1}$ を通り $v_{2,i+1}$ まで行くコストが $v_{1,i}$ から $v_{2,i}$ を通り $v_{2,i+1}$ まで行くコストを常に上回る。利己的ルーティングではより小さいコストのパスが存在する場合, 大きなコストのパスは使用されないため, ナッシュフローにおいて e_{i+1} を含むパスのフローは 0 である。一方で $b_i + b_{i+1} < d_i + b'_i$ の場合, $v_{1,i}$ から $v_{1,i+1}$ を通り $v_{2,i+1}$ まで行くコストが $v_{1,i}$ から $v_{2,i}$ を通り $v_{2,i+1}$ まで行くコストを常に下回る。この場合はナッシュフローにおいて e_i を含むパスのフローは 0 である。そして $b_i + d_{i+1} = d_i + b'_i$ の場合には $v_{1,i}$ から $v_{1,i+1}$ を通り $v_{2,i+1}$ まで行くコストが $v_{1,i}$ から $v_{2,i}$ を通り $v_{2,i+1}$ まで行くコストと等しくなる。この時, e_i を含むパスのコストと e_{i+1} を含むパスのコストは一致する。そのため e_i を含むパスのフローが 0 となるようなナッシュフローが存在する。ナッシュフローにおいて使用されない辺は削除してもナッシュフローに影響を与えないため, $b_i + d_{i+1} \geq d_i + b'_i$ の場合は e_{i+1} を, $b_i + d_{i+1} < d_i + b'_i$ の場合は e_i を削除したネットワークを考えれば十分である。

このアルゴリズムは $i = 1, \dots, n$ に対して G_i の全域部分ネットワークのうち, $1 \leq j < i$ の各 j に対して $f_{e_j}, f_{e_i} > 0$ となるナッシュフローが存在し, $e_{j+1}, \dots, e_{i-1}$ が存在しないものの中でナッシュフローのコストが最小となるネットワーク $H_{i,j}$ を動的計画法によって求める。

形式的な定義を以下に示す。

Step 1: $i = 1, 2, \dots, n$ に対して, 1 から順番に以下を行う。

- G_i から $e_1, \dots, e_{i-1}$ を削除して得られるネットワークを $H_{i,0}$ とし, $p_{i,0} = \sum_{j=1}^{i-1} (a_j + b_j) + d_i$, $q_{i,0} = 1$ とする.
- $1 \leq j < i$ に対して, $H_{j,0}, H_{j,1}, \dots, H_{j,j-1}$ のうち $q_{j,k} > \frac{d_j - d_i + \sum_{l=j}^{i-1} (a'_l + b'_l - b_l)}{\sum_{l=j}^{i-1} (a'_l + a_l)}$ であり, 最小の $p_{j,k}$ を持つ $H_{j,k}$ ($1 \leq k \leq j-1$) を探す. そして j 段目までは $H_{j,k}$ に等しく, $e_{j+1}, \dots, e_{i-1}$ を削除した G_i の全域部分ネットワークを $H_{i,j}$ とする. ただし該当するネットワークが存在しない, あるいは $\frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a'_k + a_k)} < 0$ の場合は何もしない. $H_{i,j}$ は存在しないこともある. $p_{i,j} = p_{j,k} + \sum_{l=j}^{i-1} (a'_l(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)}) + b_l)$, $q_{i,j} = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$ とする.

Step 2: $1 \leq i < n$ に対して, $H_{i,0}, \dots, H_{i,i-1}$ の中で $p_{i,j}$ が最小のネットワーク $H_{i,j}$ を探す. そして i 段目までは $H_{i,j}$ と等しく, $e_{i+1}, \dots, e_n$ を削除した G の全域部分ネットワークを H_i とし, $p_i = p_{i,j} + \sum_{k=j}^{i-1} (a'_k + b'_k)$ とする.

Step 3: $\{H_i | i = 1, \dots, n-1\} \cup \{H_{n,i} | i = 0, \dots, n-1\}$ の中から p_i または $p_{n,i}$ が最小のネットワークを出力する.

4. アルゴリズムの正当性

アルゴリズムが多項式時間で正しい解を出力することを証明する. $1 \leq j < n$ と G_n の全域部分ネットワーク S に対して, 点集合 $\{v_{k,l} | k \in \{1, 2\}, 1 \leq l \leq j\}$ で誘導される S の全域部分ネットワークを $\tilde{S}^j$ で表す. $\tilde{S}^j$ を単に S の j 段目までの部分ネットワークと呼ぶことがある.

補題 1. $1 \leq k \leq j < i \leq n$ に対して, $H_{i,j}$ のナッシュフローにおいて, $v_{1,k}$ から $v_{2,i}$ へ向かう経路のうち, フローが存在する経路のコストは一致する.

証明. 背理法を用いる.

$H_{i,j}$ のナッシュフローにおいて, $v_{1,k}$ から $v_{2,i}$ へ向かう異なる経路のうち, フローが存在する2つの経路 P_A, P_B で異なるコスト p_A, p_B ($p_A > p_B$) が掛かると仮定する.

s から $v_{1,j}$ へ向かうコストを p とすると, 経路 P_A のコストは $p + p_A$ であり, 経路 P_B のコストは $p + p_B$ となる.

フローが存在する2の経路のコストを比較すると, 仮定より $p + p_A > p + p_B$ となるが, これはナッシュフローの定義と矛盾する. したがって異なるコストを持つ経路は存在しない. そのため, この補題が成立する. $\square$

補題 2. $1 \leq j < i \leq n$ に対して, G_i の全域部分ネットワークにおいて, $f_{e_i}, f_{e_j} > 0$, $f_{e_{j+1}} = \dots = f_{e_{i-1}} = 0$ であるナッシュフローが存在するならば, $f_{e_i} = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$ となるナッシュフローが存在する.

証明. ナッシュフローにおいて $v_{1,j}$ から $v_{1,i}$ を通り $v_{2,i}$ に向かう経路のコストは $\sum_{k=j}^{i-1} (a_k f_{e_i} + b_k) + d_i$ である. そして $v_{1,j}$ から $v_{2,j}$ を経由して $v_{2,i}$ へ向かう経路のコストは $\sum_{k=j}^{i-1} (a'_j(1 - f_{e_i}) + b'_k) + d_j$ であり, 補題1からこの2つの経路のコストは一致している. よって

$$d_i + \sum_{k=j}^{i-1} (a_k f_{e_i} + b_k) = d_j + \sum_{k=j}^{i-1} (a'_k(1 - f_{e_i}) + b'_k)$$

であるため,

$$f_{e_i} = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$$

が得られる. $\square$

補題 3. $1 \leq j < i \leq n$ に対して, $e_{j+1}, \dots, e_{i-1}$ を持たない G_i の全域部分ネットワーク S_i が $f_{e_i}, f_{e_j} > 0$, となるナッシュフローを持つ必要十分条件は $\tilde{S}_i^j$ が e_j に $\frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$ より大きなフローが流れるナッシュフローを持つことである.

証明. まず, 必要性を示す. 補題1より, S_i のナッシュフローにおいて, s から $v_{2,i}$ までの経路のうちフローが存在する全ての経路のコストが一致している. そして横断辺のコスト関数は定数であるため, 横断辺を通るためにかかるコストはフローの量と関係なく一定である. そのため S_i のナッシュフローで e_i を通るフローを e_j に流しても, s から $v_{2,j}$ までの経路のコストは変化しない. すなわち, S_i 上の補題の条件を満たすナッシュフロー f に対して $f'_P = f_P$ ($P \ni e_k, 1 \leq k \leq j-1$), $f'_P = f_P + f_{P'}$ ($P \ni e_j, P' \ni e_i$) からなる $\tilde{S}_i^j$ 上のフロー f' を構成すると, このフローは全フローが s から $v_{2,j}$ に流れ, フローのある全パスのコストが一致する. この時, $\tilde{S}_i^j$ の全ての辺に対して, 通るコストは $H_{i,j}$ のナッシュフローにおける各辺を通るコストと一致している. したがって, $\tilde{S}_i^j$ 上でより小さなコストを持つパスは存在しないため, $\tilde{S}_i^j$ のナッシュフローとなる. 補題2より $f'_P = f_P + f_{P'} > f_{P'} = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)} > 0$ ($P \ni e_j, P' \ni e_i$) である.

次に十分性を示す. $\tilde{S}_i^j$ 上の補題の条件を満たすナッシュフロー f を考える. S_i 上のフロー f' を $f'_P = f_P$ ($P \ni e_k, 1 \leq k < j$), $f'_P = f_P - \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)} > 0$ ($P \ni e_j$), $f'_P = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$ ($P \ni e_i$) からなるフローとする. この時, f' は S_i 上の許容フローである. そして, $\tilde{S}_i^j$ のナッシュフローのコストを p とすると, e_i を通る経路のコストは

$$p - d_j + \sum_{k=j}^{i-1} \left(a_k \frac{d_j - d_i + \sum_{l=j}^{i-1} (a'_l + b'_l - b_l)}{\sum_{l=j}^{i-1} (a_l + a'_l)} + b_l \right) + d_i$$

であり, フローが存在する $(v_{2,i-1}, v_{2,i})$ を通るパスのコストは

$$p + \sum_{k=j}^{i-1} \left(a'_k \left(1 - \frac{d_j - d_i + \sum_{l=j}^{i-1} (a'_l + b'_l - b_l)}{\sum_{l=j}^{i-1} (a_l + a'_l)} \right) + b'_k \right) \quad (1)$$

である. この2式が等しいことは容易に確かめられる. よって f' は $H_{i,j}$ のナッシュフローであり, $f'_P, f'_{P'} > 0$ ($P \ni e_j, P' \ni e_i$) である. $\square$

補題 4. $0 \leq j < i \leq n$ に対して, $p_{i,j}$ は $H_{i,j}$ のナッシュフローのコストに等しく, $q_{i,j} = f_{e_i}$ である. さらに $i \geq 2$ のとき, $f_{e_i}, f_{e_j} > 0$ である.

証明. i に関する帰納法で示す.

$i = 1$ の時, $H_{1,0}$ にはパスが1つしかないため, e_i に流れるフローの量は $p_{i,0} = 1$ に等しく, ナッシュフローのコストは $p_{i,0} = d_1$ に等しい.

$i \geq 2$ として, $i - 1$ 以下で補題が成立していると仮定する. アルゴリズムの定義より, $\tilde{H}_{i,j}^j = H_{j,k}$ となるような k ($0 \leq k < j$) が存在する. 帰納法の仮定から, $p_{j,k}$ は $H_{j,k}$ のナッシュフローのコストであり, $q_{j,k}$ は $H_{j,k}$ のナッシュフローが e_j に流すフローの量に等しい. アルゴリズムの定義より, $q_{j,k} > \frac{d_j - d_i + \sum_{l=j}^{i-1} (a'_l + b'_l - b_l)}{\sum_{l=j}^{i-1} (a_l + a'_l)}$ であるので, 補題3の十分性の証明より, $p_{i,j} = p_{j,k} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right)$ は式(1)と等しいことから $p_{i,j}$ は $H_{i,j}$ のナッシュフローのコストに等しい. また, 補題3の十分性より, $H_{i,j}$ には $f_{e_i}, f_{e_j} > 0$ であるようなナッシュフローが存在する. したがって補題2より, $f_{e_i} = \frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a_k + a'_k)}$ であり, この値は $q_{i,j}$ に等しい.

したがって, 補題は成り立つ. $\square$

以下の補題は自明に成り立つ:

補題 5. $1 \leq i \leq n$ に対して, $H_{i,0}$ は, $f_{e_i} > 0$ であり, $f_{e_1} = \dots = f_{e_{i-1}} = 0$ であるような一意的なナッシュフローを持つ G_i の全域部分ネットワークである.

補題 6. $1 \leq j < i \leq n$ に対して, $H_{i,j}$ は $f_{e_i}, f_{e_j} > 0$ であり $f_{e_{j+1}} = \dots = f_{e_{i-1}} = 0$ であるナッシュフローが存在する G_i の全域部分ネットワークのうち, ナッシュフローのコストが最小のネットワークである.

証明. i による帰納法で示す.

$i = 2$ のとき $H_{2,1}$ は G_2 であり, $f_{e_1}, f_{e_2} > 0$ となる部分ネットワークは他に存在しないため, ナッシュフローのコストが最小の全域部分ネットワークである.

$i \geq 3$ とし, $i - 1$ 以下で補題が成り立っていると仮定する. 背理法を用いて $H_{i,j}$ で補題が成立することを示すため, $0 \leq j < i$ に対して, $f_{e_i}, f_{e_j} > 0$, $f_{e_{j+1}} = \dots = f_{e_{i-1}} = 0$ であり, $H_{i,j}$ より小さなコストのナッシュフローを持つ, G_i の全域部分ネットワーク $H'_{i,j}$ が存在すると仮定する.

$\tilde{H}_{i,j}^j$ のナッシュフローにおいて, e_j 以外の横断辺にフ

ローが存在する場合は, $f_{e_j}, f_{e_k} > 0$, $f_{e_{k+1}} = \dots = f_{e_{j-1}} = 0$ となる k ($1 \leq k < j$) が存在する. 帰納法の仮定と, 補題2より, ナッシュフローにおいて e_j を流れるフローの量は $H_{j,k}$ と $\tilde{H}_{i,j}^j$ で等しくなるため, アルゴリズムの定義より, $p_{i,j}$ は高々 $p_{j,k} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right)$ である. 一方で補題3(式(1))より, $H_{i,j}$ のナッシュフローのコストは $\tilde{H}_{i,j}^j$ のナッシュフローのコスト p' を用いて $p' + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right)$ となる. 帰納法の仮定より, $p' \geq p_{j,k}$ であるため, $p' + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right) \geq p_{j,k} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right) \geq p_{i,j}$ となる. 補題4より, $p_{i,j}$ は $H_{i,j}$ のナッシュフローのコストであるため, この不等式は背理法の仮定と矛盾する.

また, $\tilde{H}_{i,j}^j$ のナッシュフローで e_j 以外の横断辺にフローが存在しない場合, $\tilde{H}_{i,j}^j$ のナッシュフローは $H_{j,0}$ と等しくなるため, ナッシュフローのコストも等しくなる. アルゴリズムの定義より, $p_{i,j}$ は高々 $p_{j,0} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right)$ である. 一方で補題4より, $p_{j,0}$ は $H_{j,0}$ のナッシュフローのコストであるため, $H'_{i,j}$ のナッシュフローのコストは $p_{j,0} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right)$ となる. よって $p_{j,0} + \sum_{l=j}^{i-1} \left(a'_l \left(1 - \frac{d_j - d_i + \sum_{m=j}^{i-1} (a'_m + b'_m - b_m)}{\sum_{m=j}^{i-1} (a_m + a'_m)} \right) + b_l \right) \geq p_{i,j}$ である. 補題4より, $p_{i,j}$ は $H_{i,j}$ のナッシュフローのコストであるため, 背理法の仮定と矛盾する.

したがって, $H_{i,j}$ よりもコストの小さい $H'_{i,j}$ は存在せず, i で補題が成立する. $\square$

定理 1. アルゴリズムは梯子状ネットワークを最適化する. その際の時間計算量は $O(n^3)$ であり, 空間計算量は $O(n^3)$ である.

証明. 補題6より, $f_{e_n} > 0$, かつ $f_{e_{j+1}} = \dots = f_{e_n} = 0$, $f_{e_j} > 0$ となるような $1 \leq j < n$ が存在する最小のコストのナッシュフローを持つ G_n の全域部分ネットワークは $H_{n,j}$ である. したがって, 補題5と合わせて $f_{e_n} > 0$ であるような最小のコストのナッシュフローを持つ G_n の全域部分ネットワークは, $\{H_{n,i} | 0 \leq i < n\}$ の中にある.

次に, $f_{e_n} = \dots = f_{e_{i+1}} = 0$ かつ $f_{e_i} > 0$ となるような $1 \leq i < n$ が存在する最小のコストのナッシュフローを持つ G_n の全域部分ネットワークを考える. このとき全て経路は $v_{2,i}$ から $v_{2,n}$ へ向かい, その部分に要するコストはこの経路上のコスト関数によってのみ定まる. したがって, このネットワークの i 段目までの部分は, $H_{i,0}, \dots, H_{i,i-1}$ の中でナッシュフローのコストが最小のネットワークに等しいため, ゆえにこのネットワークは H_i である.

よって $\{H_i | i = 1, \dots, n-1\} \cup \{H_{n,i} | i = 0, \dots, n-1\}$

の中に G_n を最適化した全域部分ネットワークが含まれている。

時間計算量に関しては Step 1 で $2 \leq i \leq n$ の各 i に対して i 個のネットワークを計算して求める。よって $\sum_{i=2}^n i = O(n^2)$ 個のネットワークを計算する。そして 1 つのネットワークを求めるために $1 \leq j \leq i-1$ の各 j で $\frac{d_j - d_i + \sum_{k=j}^{i-1} (a'_k + b'_k - b_k)}{\sum_{k=j}^{i-1} (a'_k + a_k)}$ を計算するため、加算と除算に $O(n)$ 時間を要する。各ネットワークにおける $q_{i,j}$ ($1 \leq j < i \leq n$) は、そのネットワークを求めるための計算で得られる。 $p_{i,j}$ ($1 \leq j < i \leq n$) の計算は $q_{i,j}$ とコスト関数を用いて $O(n)$ 回の計算で求められる。そして Step 2 で $n-1$ 個のネットワークを求めるが、このときに行われる計算は各ネットワークの p_i ($1 \leq i < n$) を求めるだけのため、各ネットワークについて $O(n)$ である。Step 3 では最小の p_i または $p_{i,j}$ ($1 \leq j < i \leq n$) が最小のネットワークを探すために $O(n)$ 時間かかる。

このことから Step 1 で $O(n^3)$ 、Step 2 で $O(n^2)$ 、Step 3 で $O(n)$ 回の計算を行うため、時間計算量は $O(n^3)$ である。

空間計算量に関しては $1 \leq j \leq n$ に対して $H_{j,0}, \dots, H_{j,j-1}$ を記憶しておく必要がある。そして $H_1, \dots, H_{n-1}$ の $n-1$ 個のネットワークも記憶しておく必要がある。よって $\sum_{i=1}^n i + n - 1 = O(n^2)$ 個のネットワークを記憶しなければならない。そして各ネットワークは各横断辺の有無についての情報のみがあればよいので、 i 段のネットワークに i ビット必要になる。そして $q_{i,j}$ とそのネットワークの $p_{i,j}$ または p_i ($1 \leq j < i \leq n$) を記憶しておくため、定数ビット必要である。

よって空間計算量は $O(n^3)$ である。 $\square$

参考文献

- [1] Braess, D.: Über ein Pradoxon aus der Verkehrsplanung, *Unternehmensforschung*, Vol. 12, pp. 258–268 (1968).
- [2] Roughgarden, T.: On the Severity of Braess's Paradox: Designing Networks for Selfish Users is Hard, *Journal of Computer and System Sciences*, Vol. 72, pp. 922–953 (2006).
- [3] Fotakis, D., Kaporis, A. C. and Spirakis, P. G.: Efficient Methods for Selfish Network Design, in *Automata, Languages and Programming*, Vol. 5556 of *Lecture Notes in Computer Science*, pp. 459–471 (2009).
- [4] Frank, M.: Cost-deceptive Links on Ladder Networks, *Methods of Operations Research*, Vol. 45, pp. 75–86 (1983).
- [5] Roughgarden, T.: *Selfish Routing and the Price of Anarchy*, The MIT Press (2006).
- [6] Beckmann, M. J., McGuire, C. B. and Winsten, C. B.: *Studies in the Economics of Transportation*, Yale University Press (1956).