

整数最小極大フローに対するアルゴリズム

吉 永康^{†1} 施 建明^{†2}

ネットワークにおける最大流問題が離散最適化の分野での典型問題であり、幅広い分野への応用性をもつ。しかし、現実世界では、しばしばフローを自由に増減させることができない場合がある。そのとき得られるのは極大流である。最小極大流は、ネットワークにおける非効率性指標の一つと考えられる。最小極大流量値を求める問題を最小極大フロー問題と呼び、本論文ではこれまで提案されたアプローチを基礎とし、最小極大フロー問題を線形計画問題とある種の線形相補性問題に帰着できることを理論的に示し、新たなアルゴリズムを提案する。

An Algorithm for Minimum Maximal Network Flow

YONGKANG JI^{†1} and JIANMING SHI^{†2}

The maximum flow problem is a typical problem in combinatorial optimization, and has been applied in many fields. The optimal value of maximum flow can be attained only on the assumption that a flow on every edge of the given flow can be freely increased and decreased. However, in the real world this assumption can not be satisfied. The minimum maximal flow indicates how inefficiency of the network being utilized. We theoretically reduce the problem into solving a linear programming problem and a linear complementary problem. Finally, we propose a new algorithm based upon the related existing algorithms.

1. はじめに

ネットワークフロー問題は1950-60年代にFordとFulkersonより提出されて以来、50年

以上経っているにもかかわらず、依然として活発に研究がされている。この分野で代表的な問題として最大流問題がある¹⁾。最大流問題に対して、プリフロー・プッシュアルゴリズムは、近年の並列アルゴリズムの視点が、従来の技法とマッチし、初めて生み出されたものである。これらのアルゴリズム群の進展の背景には、古典的アルゴリズムで使われた着想や技法と新しい着想や技法の融合が見られている⁷⁾。

最小極大フロー問題とは、ネットワークのすべての極大フローの中で、最も流量の小さいフローを見つけ出す問題であり、 \mathcal{NP} 困難な問題であることが知られている¹¹⁾。従来のネットワーク最適化理論はフローを自由に増減させ、最適値に達することができるという条件の下で発展したが、最小極大フロー問題は既存フローを削除できない前提で提起された問題である。

最小極大フロー問題に関する研究としては、1997年にShiとYamamotoは最小極大フロー問題を定義し、下界法でこの問題を解決するためのアルゴリズムを提案した¹¹⁾。また2003年にGotoh, ThoaiとYamamotoは大域最適法のアルゴリズムを示した⁵⁾。その後、幾つかの成果も報告された。その一つとして2007年にYamamotoとZenkeは外部近似法でのアルゴリズムを提案した¹³⁾。

下界法では、ネットワークの各辺にフローの下限を加え、全域探索法で最小極大フローを見つける。大域最適法では、最小極大問題を大域最適化問題に変換し、局所探索法と分枝限定法を利用し、最小極大フローを見つける。外部近似法では、ギャップ関数を用いて極大フローの実行可能領域を定義し、対象の問題を $D.C.$ 最適化問題⁶⁾に適応させ、 $D.C.$ 問題に対する外部近似法と局所探索法を用いて最小極大フローを見つける。しかしながら、下界法や外部近似法などの既存研究で計算する場合は、ネットワークにおける枝の本数を次元とする凸多面体の端点を計算する必要があるため、その次元が高くなると、莫大な計算時間がかかる。

本論文では、最小極大フロー問題における目的関数と制約条件をそれぞれ線形関数と非凸多面体を用いて定式化し、対象問題をいくつかの線形計画問題と相補性問題を解くことに帰着させる。更に、これに基づくアルゴリズムを提案する。

2. 最小極大フロー問題

最小極大フロー問題はネットワークにおけるフロー問題の1つであるため、ネットワークについての定義と記号をあらかじめ定める。また、最小極大フロー問題についての定式化と定義を導入する。

^{†1} 室蘭工業大学情報電子工学系専攻修士2年生

Muroran Institute of Technology E-Mail: 10054099(at)muroran-it.ac.jp

^{†2} 室蘭工業大学情報電子工学系

Muroran Institute of Technology E-Mail: mathopt(at)gmail.com

2.1 ネットワーク

ネットワーク $N = (V, E, s, t, c)$ は、有向グラフ $G = (V, E)$ の各枝 e に正の実数の容量 $c(e)$ が割り振られているグラフである。ただし、枝 $(u, v) \in E$ は始点 u 、終点 v を表し、 V は $m+2$ 個の端点集合であり、 E は n 個の枝集合であり、 s, t はネットワークのソース、シンクである。

ネットワーク N の枝集合から非負実数集合への関数 $f: E \rightarrow R_+$ が以下の条件を満たすとき、 f を N のフローという。

- (1) 任意の枝 $e \in E$ に対して、 $0 \leq f(e) \leq c(e)$ が成り立つ。
- (2) s, t を除く任意の V の点 v に対して、流量保存則 (流入量=流出量) が成り立つ。数式で書くと、次のようになる。

$$\sum_{e=(u,v) \in E} f(e) = \sum_{e=(v,w) \in E} f(e)$$

また、 s から流出するフローの正味量は t に流入するフローの正味量に等しい。この正味量をフロー f の流量という。

2.2 最小極大フロー問題の定式化

$m \times n$ の接続行列 A の各要素を次のように定義しておく。

$$a_{v,e} = \begin{cases} 1 & (\text{点 } v \text{ から出る枝 } e \text{ の場合}) \\ -1 & (\text{点 } v \text{ に入る枝 } e \text{ の場合}) \\ 0 & (\text{その他}) \end{cases}$$

フローの条件を満たすベクトルはフローであり、次のようにフロー集合 X で表す。

$$X = \{x \in R^n \mid Ax = \mathbf{0}_m, \mathbf{0}_n \leq x \leq c\}$$

ただし、 R^n は n 次元の列ベクトルの集合、 $\mathbf{0}_m, \mathbf{0}_n$ がそれぞれ m 次元と n 次元の列ベクトルである (以降は単に $\mathbf{0}$ と表記する)。 X はフローの実行可能領域である。

本研究では、議論を簡潔するため、与えられたネットワークが次のような仮定を置く。

仮定 1 各枝の容量 $c(e)$ は正の整数である。

仮定 2 $t-s$ 経路が存在しない

さらに、 $d(d \in R^n)$ の各要素を次のように定義しておく。

$$d_e = \begin{cases} 1 & (\text{ソース } s \text{ から出る枝 } e \text{ の場合}) \\ -1 & (\text{ソース } s \text{ に入る枝 } e \text{ の場合}) \\ 0 & (\text{その他}) \end{cases}$$

上記の定義より、従来の最大流問題は次の形式となる。

$$\begin{cases} \text{maximize} & d^\top x \\ \text{subject to} & x \in X \end{cases}$$

ただし、 \top は転置である。

不等式 $z \geq x$ と $z \neq x$ を満たす実行可能フロー $z \in X$ が存在しない時、実行可能フロー x を極大フローといい、その集合を X_E とする。極大フローの集合 X_E 上で最小の流量値を求める問題

$$(MMF) \begin{cases} \text{minimize} & d^\top x \\ \text{subject to} & x \in X_E \end{cases}$$

を最小極大フロー問題 (MMF) という¹⁰⁾。

最小極大フロー問題について次の図 1 を用いて説明する。

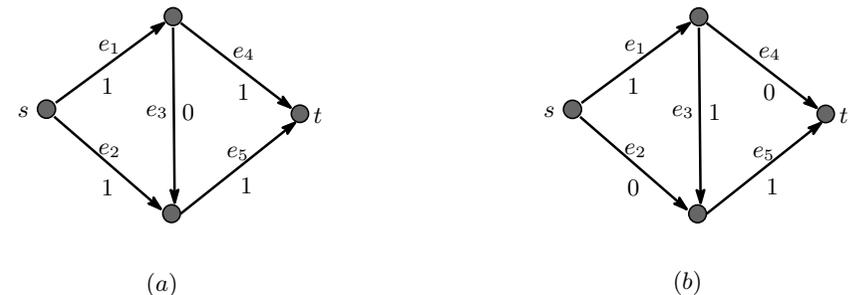


図 1 最小極大フロー問題の例

図 1 では、ネットワークにおけるすべての枝の容量は 1 であり、 s, t はそれぞれネットワークのソースとシンクであり、各枝上の数値は枝の流量である。図 1 の (a) ではこのネットワークの極大フローの一つを示してあり、最大流でもある。(b) ではこのネットワークの別の極大フローを示し、最小極大フローとなっている。最小極大フロー問題と最大流量問題は本質的に異なることがわかる。

3. ギャップ関数による (MMF) の再定式化

すべての要素が 1 である n 次元の列ベクトルを e とし、ギャップ関数 $r: R^n \rightarrow (-\infty, \infty)$ を

$$r(x) := \max\{e^\top(y - x) \mid y \geq x, y \in X\} \quad (1)$$

と定義する。定義により、 $r(x)$ は区分的に線形な凹関数であり、任意の $x \in X$ に対して $r(x) \geq 0$ である。

定理 1 $r(x) = 0$ が成立する必要十分条件は $x \in X_E$

証明 :まず、必要性を証明する。

$x \notin X_E$ と仮定する。即ち、フロー x は極大フローではない。極大フローの定義により、 $y \geq x$ を満たす X の実行可能解 y が存在する。即ち、 $y - x \neq 0$ が成り立つ。よって、フロー y は $r(x)$ 関数 (式 3.1) の実行可能領域に含まれる。 $r(x)$ 関数の値は $e^\top(y - x) \neq 0$ であることが成り立つ。即ち、 $r(x) \neq 0$ が成り立つ。背理法より、定理の必要性が成立する。同様に、十分性は明らかである。定理 1 により、(MMF) 問題を次の形式に書き直すことができる。

$$(MMF) \begin{cases} \text{minimize} & d^\top x \\ \text{subject to} & x \in X, r(x) = 0 \end{cases} \quad (2)$$

系 1 (MMF) 問題において、 $d^\top x$ が最小になるのは $r(x) = 0$ のときに限ることは明らかである。即ち、 $r(x)$ は (MMF) 問題の極大フローであるかを判断する関数である。ある実数 $t > 0$ に対して、次のペナルティ関数 $P(t)$ を定義する。

$$P(t) := \min\{d^\top x + t \cdot r(x) \mid x \in X\} \quad (3)$$

$P(t)$ に対して、次の定理 2 が成り立つ。

定理 2 十分に大きい t に対して、もし $x^* \in \arg \min P(t)$ ならば、 $r(x^*) = 0$ が成り立つ。

証明: $r(x^*) \neq 0$ と仮定する。 $r(x) \geq 0$ より、 $r(x^*) > 0$ が成り立つ。また、ある $y \in X_E$ が

存在すると仮定する。定理 2 より、 $r(y) = 0$ が成り立つ。

$t = (\max\{d^\top x \mid x \in X\} + 1)/r(x^*)$ とおく。明らかに $t > 0$ が成り立つ。よって、

$$\begin{aligned} \{d^\top x^* + t \cdot r(x^*)\} &= d^\top x^* + \max\{d^\top x \mid x \in X\} + 1 \\ &> \max\{d^\top x \mid x \in X\} \\ &\geq d^\top y \\ &= d^\top y + t \cdot r(y) \end{aligned}$$

上記の式より、 x^* が $P(t)$ の最適解ではないことになる。背理法より、定理が成立する。□
パラメーター $\alpha > 0$ とおく。次の実行可能領域を $X(\alpha)$ で定義する。

$$X(\alpha) := X \cap \{x \in R^n \mid d^\top x \geq \alpha\} \quad (4)$$

また、 $X(\alpha)$ を用いた問題 $Q(\alpha)$ を次のように定義する。

$$Q(\alpha) := \min\{d^\top x \mid x \in X(\alpha)\} \quad (5)$$

$Q(\alpha)$ 関数の役割はアルゴリズムの各段階において制約される領域の中で、最小流量フローを見つけるということである。

4. アルゴリズムの提案

MMF に対するアルゴリズム

ステップ 0: $r(0)$ を解き、 $r(y^1) = 0$ のような y^1 を得る。

パラメーター $\varepsilon > 0$ を設定し、 $k := 1$ とおく。

ステップ 1: $\alpha_k := d^\top y^k - \varepsilon$ とおく。 $Q(\alpha_k)$ を解き、 q^k を得る。

ステップ 2: もし $r(q^k) = 0$ ならば、 $y^{k+1} := q^k, k := k + 1$ とおき、

ステップ 1 に戻る。

ステップ 3: $\max\{e^\top w \mid d^\top w = d^\top q^k, w \in X\}$ を解き、 w^k を得る。

ステップ 4: もし $r(w^k) = 0$ ならば、 $y^{k+1} := w^k, k := k + 1$ とおき、

ステップ 1 に戻る。

ステップ 5: $H_{\alpha_k} := \{y \mid d^\top y = \alpha_k\} \cap X$ の上に極大フローが

存在するかどうかを判断する。

ステップ 6: もし極大フローが存在しなければ、終了する。

(q^{k-1} は最適解である).

ステップ 7: 極大フロー存在すれば, 求めた極大フローを y^{k+1} とする.

$k := k + 1$ とおき, ステップ 1 に戻る.

提案するアルゴリズムによって見つけれられる最小極大フローを x^* とおく. 次に, 2次元のイメージ図 2 を用い, 最小極大フローを見つける手順を説明する.

まず, 初期値を設定しておく. X はフローの実行可能領域であり, X_E (太線) は極大フローの実行可能領域である. また, 最大フロー y^1 を見つける (図 2 の (a)). 次に, $d^T y^1$ の値を ε だけ減少させ, 新しいフローと極大フロー領域 $X(\alpha_1)$ を得る. $Q(\alpha_1)$ 関数の最小の点を見つけ, $r(x)$ 関数でこの点は極大フローであるかを判断する (図 2 の (b)). 極大フローであるならば, 同様に繰り返していく (図 2 の (c) と (d)). 極大フローでないならば, $\max\{e^T w \mid d^T w = d^T q^k, w \in X\}$ でこのフローの最大点 x^* を見つけ, x^* が極大フローであるかを確認する (図 2 の (e)). $d^T x^*$ を ε だけ減少させた後のフローに極大フローが含まれなくなるまで続け, フローに極大フローが存在しないならば, その値以下には極大フローが存在しないことになるので, 直前の q^{k-1} を x^* とする (図 2 の (f)). 以上の動作より, フロー x^* は MMF 問題の最小極大フローとして得られる.

定理 3 もし x^* が MMF 問題の最適解であれば, x^* も $Q(d^T x^*)$ の最適解である.

証明: x^* は (MMF) 問題の最適解とする. このとき, ステップ 1 の $Q(d^T x^*)$ にて探索領域は $X(d^T x^*) = X \cap \{x \in R \mid d^T x \geq d^T x^*\}$ である. よって領域内の任意の x に対しても, $d^T x \geq d^T x^*$ が成立する. x^* 自身も領域内に含まれるため, $Q(d^T x^*) = \min\{d^T x \mid x \in X(d^T x^*)\} = d^T x^*$ となる. よって, x^* も $Q(d^T x^*)$ の最適解となる. \square

続いて, 提案するアルゴリズムの正当性と収束性について証明する.

定理 4 提案するアルゴリズムがステップ 5 での計算が有限回で終了するならば, アルゴリズム全体が有限回で (MMF) 問題の最適解を見つけることができる.

証明: (正当性) アルゴリズムの最終段階のステップ 5 からステップ 7 の中で, フロー q^k に対して, もしステップ 4 での $r(w^k) \neq 0$ ならば, 流量が $d^T q^k$ であるフローの中に, 極大フローが存在するかを確認する. もしそのようなフローが存在しないならば, フローの整数性と X_E の連結性より, 明らかに前段階の $r(x) = 0$ を満たすフロー q^{k-1} が最適解となる.

(収束性) アルゴリズムのステップ 0(初期化) で, $r(y^1) = 0$ のようなフロー y^1 を見つける. フロー y^1 が $y^1 \leq c$ を満たすため, y^1 の値は有限であることは自明である. ステップ

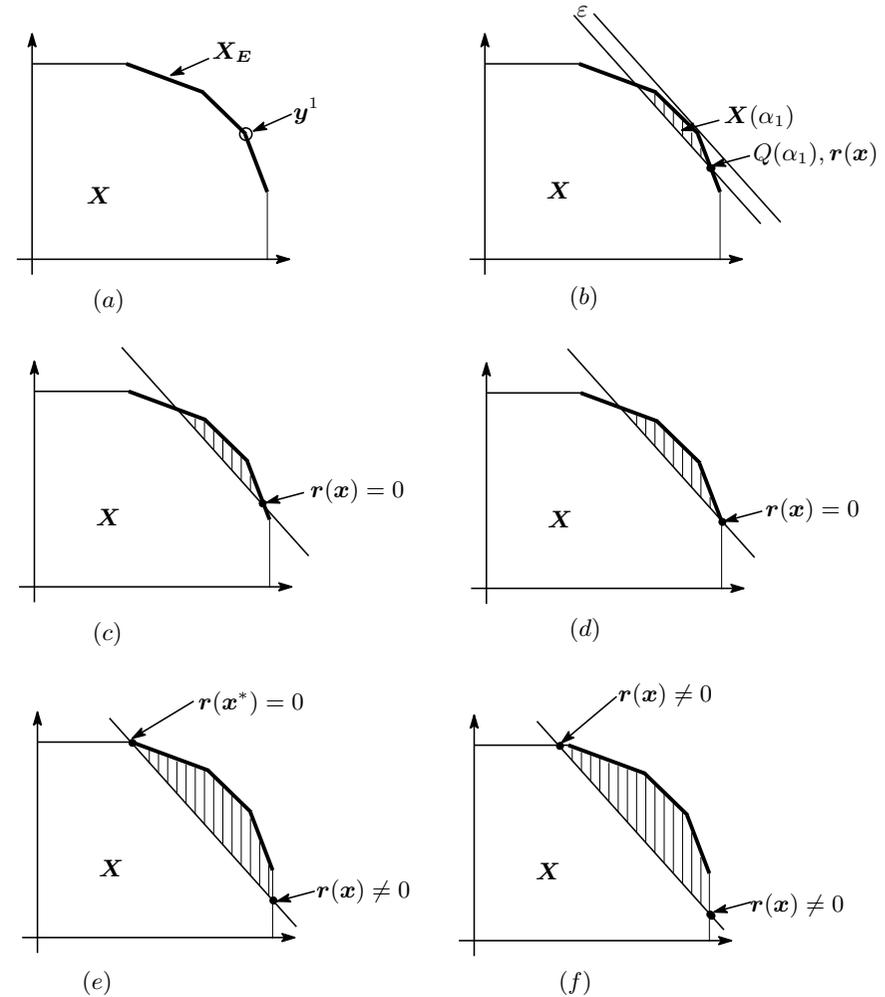


図 2 最小極大フローを見つける手順

1で、 y^1 の値は整数 $\varepsilon > 0$ ずつ減少していく。また、最小のフローの値は0となることとフローの整数性より、極大フローの判断回数は多くとも $\lceil d^T y^1 / \varepsilon \rceil$ 回を超えない。ステップ5での極大フローが存在するかどうかの判断を有限回で終了できるなら、有限回で最適解を見つけることができる。□

5. 極大フローの存在判断

提案したアルゴリズムのステップ5では $H_\gamma := \{y \mid d^T y = \gamma\} \cap X$ の上に極大フローが存在するかどうかを判断する必要がある。この極大フローの存在判断問題が問題(6)に解の存在判断問題に帰着できる³⁾

$$\begin{cases} \min & v^T z_3 \\ \text{s.t.} & A^T z_1 - A^T z_2 + z_3 - z_4 = e \\ & d^T v = d^T c - \gamma, Av = Ac, 0 \leq v \leq c \\ & z_i \geq 0, i = 1, \dots, 4. \end{cases} \quad (6)$$

定理5 問題(6)に対して、もし最適値が0であれば、 H_γ には極大フローが存在する、最適値が0でないとき、極大フローが存在しない。

6. 線形相補性問題

問題(6)が線形相補性問題であり、相補ピボットアルゴリズムを使って解くとする。

問題(6)に対して、 $M = (A^T, -A^T, I, -I)$, $z = (z_1^T, z_2^T, z_3^T, z_4^T)^T$,

$$N = \begin{pmatrix} d^T & 0 \\ A & 0 \\ I & I \end{pmatrix}, \quad u = \begin{pmatrix} v \\ s \end{pmatrix}, \quad p = \begin{pmatrix} d^T c - \gamma \\ Ac \\ c \end{pmatrix}.$$

ここで、 $0 \leq s \leq c$ 、問題(12)が

$$\begin{cases} \min & v^T z_3 \\ \text{s.t.} & Mz = e \\ & Nu = p \\ & z, u \geq 0, \end{cases} \quad (7)$$

と書き換えることができる。

つまり、この問題が $Mz = e, Nu = p, v_i^T z_{3i} = 0, (i = 1, \dots, n), z, u \geq 0$ を満たす解を求める線形相補性問題になる。このような問題を解くためにLemke法⁴⁾をベースにして下記のアルゴリズムを提案する。

線形相補性問題へのアルゴリズム

ステップ0: シンプレックスタブローで $Nu = p$ を表し、

$u \geq 0$ を満たす一つの解を探す。解がなければ終了する。

ステップ1: v の中に0になった要素の添字を記録して、それと同じ添字の

z_3 の要素を0とする。 $Mz = e$ を満たす z があれば、終了する。

ステップ2: ピボットでタブローを更新する

ステップ3: 停止条件を満たせば、終了する。でなければ、ステップ1に戻る。

ステップ3での停止条件がLemke法の停止条件と類似であろうと思われる。⁴⁾

7. 結 論

本論文では最小極大流フロー問題を線形計画問題と双線形問題に変換し、更にある種の相補性問題に帰着できることを理論的に示した。これまで提案されたアプローチを基礎とし、新たなアルゴリズムを提案した。アルゴリズムの中では相補性問題を解く必要があり、Lemke法をベースにしたアルゴリズムを提案した。数値実験が必要であるが、今後の課題としてさせて頂く。

参 考 文 献

- 1) 浅野孝夫: 情報の構造ネットワークアルゴリズムとデータ構造, 日本評論社 (1994).
- 2) 陳文傑: 最小極大流フロー問題に対する効率になアルゴリズムについて, 室蘭工業大学修士学位論文 (2009).
- 3) 吉永康: 第85回数理モデル化と問題解決研究発表会, 室蘭工業大学 (2011年9月16日).
- 4) Bazaraa, M.S., Sherali, H.D. and Shetty, C.M.: Chapter 11, NONLINEAR PROGRAMMING: THEORY AND ALGORITHMS, John Wiley & Sons, NY (1979).
- 5) Gotoh, J., Thoai, N. and Yamamoto, Y.: Global optimization method for solving the minimum maximal flow problem, *Optimization Methods and Software*, Vol.18, No.4, pp.395-415 (2003).
- 6) Horst, R. and Tuy, H.: *Global Optimization Deterministic Approaches*, Springer Verlag (1993).

- 7) Lawler, E.: COMBINATORIAL OPTIMIZATION NETWORKS AND MATROIDS, Dover publications, Mineola, NY (2001).
- 8) Marey, K.G.: *Linear Complementary and Nonlinear Programming Sigma Series in Applied Mathematics 3*, Heldermann Verlag, Berlin (1988).
- 9) Mangasarian, O.L.: NONLINEAR PROGRAMMING, McGraw-Hill, NY (1969).
- 10) Shigeno, M., Takahashi, I. and Yamamoto, Y.: Minimum maximal flow problem: an optimization over the efficient set, *Journal of Global Optimization*, Vol.25, No.1, pp.425–443 (2003).
- 11) Shi, J. and Yamamoto, Y.: A global optimization method for minimum maximal flow problem, *ACTA Mathematica Vietnamica*, Vol.22, No.1, pp.271–287 (1997).
- 12) White, D.J.: A linear programming approach to solving bilinear programmes, *Mathematical Programming*, Vol.56, No.1, pp.45–50 (1992).
- 13) Yamamoto, Y. and Zenke, D.: Outer approximation method for the minimum maximal flow problem, *Journal of Operations Research Society of Japan*, Vol.50, No.1, pp.14–30 (2007).