Department of Social Systems and Management

Discussion Paper Series

No. 1231

最大密度部分集合問題と近似２分探索による解法

by

張明超, 高橋里司, 繁野麻衣子

April, 2009

UNIVERSITY OF TSUKUBA
Tsukuba, Ibaraki 305-8573
JAPAN

最大密度部分集合問題と近似２分探索による解法

張明超, 高橋里司, 繁野麻衣子

概 要

本論文では，最大密度部分グラフ問題をセットシステム上に拡張した最大密度部分集合問題を扱う．まず，コミュニティ抽出における最大密度部分集合の妥当性を検証する．そして，最大密度部分集合問題を解く近似 2 分探索法を用いた効率の良いアルゴリズムの提案をする．さらに，提案する近似 2 分探索アルゴリズムの他の問題への適用性についても議論する．

キーワード：アルゴリズム, 組合せ最適化, 密部分グラフ

1 はじめに

ウェブマイニングや社会ネットワーク分析では，対象となるネットワークをグラフで表現し，その構造を特徴づける．特に，対象となるネットワークの中で共通の関心や利害関係をもつ集団であるコミュニティの抽出は重要である．近年では Newman[10] によるモジュラリティというネットワークの分割指標を用いたコミュニティ抽出手法が有効であり注目を浴びている．一方，グラフ上では頂点間の枝が多い密な部分グラフがコミュニティに対応することから，クリーク (clique) やクリークの条件を緩めた部分グラフをみつけるための様々な抽出アルゴリズムが古くから議論されている ([3, 15] など参照)．クリークの条件を緩めた部分グラフとしては，取り出した部分グラフの頂点間の距離を制限した k-クリーク (k-clique) や k-クラン (k-clan)，部分グラフの次数を制限した k-コア (k-core) や k-プレックス (k-plex) などがある（[1] など参照）．また，部分グラフの頂点数を指定して，より枝数の多い部分グラフを取り出す最大枝密度部分グラフ問題（maximum edge subgraph problem) もある．クリークやクリークの条件を緩めたこれらの部分グラフをみつける問題はいずれも NP-困難であることが知られており，近似アルゴリズムが広く研究されている．一方で，平均次数を最大とする部分グラフを取り出す問題は最大密度部分グラフ問題（maximum density subgraph problem) として古くから知られており，多項式時間アルゴリズムがある．

ところで，頂点同士の関係が複数頂点の小グループによって与えられるときは，グラフよりもセットシステム（ハイパーグラフ）で対象ネットワークを表現した方が良いことがある．近年，グラフ上の密な部分集合の概念をセットシステム上に拡張してコミュニティの抽出をおこなう幾つかの研究が行われている [2, 5, 12].

本論文では，密な部分グラフの概念の一つである最大密度部分グラフをセットシステム上に拡張した問題を扱う．次節では，セットシステム上に拡張した最大密度部分集合問題を定義し，コミュニティ抽出における妥当性を検証する．3 節では最大密度部分集合問題を解くためのパ

ラメトリック探索法について議論し，4 節では近似 2 分探索法を用いた効率の良いアルゴリズムを提案する．最後に 5 節で，提案する近似 2 分探索法の他の問題への適用性について述べる．

2 最大密度部分集合問題

有限集合 N と N の部分集合族 $\mathcal{H} \subseteq 2^N$ からなるセットシステム $\Gamma = (N, \mathcal{H})$ において，$S(\subseteq N)$ に対して，$\Gamma(S) = \{J \in \mathcal{H} \mid J \subseteq S\}$ とする．このセットシステム $\Gamma = (N, \mathcal{H})$ 上で，

$$\max_{S \subseteq N, S \neq \emptyset} \frac{|\Gamma(S)|}{|S|}$$

を達成する S を最大密度部分集合といい，最大密度部分集合を求める問題を最大密度部分集合問題という [5]．セットシステム Γ がグラフのとき，すなわち $\mathcal{H} \subseteq N \times N$ のとき，この問題は最大密度部分グラフ問題 (maximum density subgraph problem) として古くから知られており，多項式時間アルゴリズムがある（[4, 6, 11] など）．以下，$n = |N|$, $m_H = |\mathcal{H}|$, $q = \sum_{J \in \mathcal{H}} |J|$ とする．

ここで，セットシステム上の最大密度部分集合とセットシステムをグラフで表現したときの最大密度部分グラフとの違いを示す．N =\{A, B, C, D, E, F, G\} におけるグループ関係が，表 1 のグループ 1〜4 のように与えられているとき，各グループを $\mathcal{H}$ の要素としたセットシステムにおける最大密度部分集合は {A, B, C, D, E} となる．一方，図 1 のように各グループをクリークで表現した多重グラフ上で最大密度部分グラフを求めると，N 全体となる．ここに，グループ 5 を加える．このとき最大密度部分集合は変わらずに {A, B, C, D, E} となるが最大密度部分グラフは {A, B, D, E} となる．

表 1: グループ関係の例

グループ	A	B	C	D	E	F	G
1	○	○	○				
2	○	○		○			
3		○		○	○		
4			○		○	○	○
5	○	○		○	○		

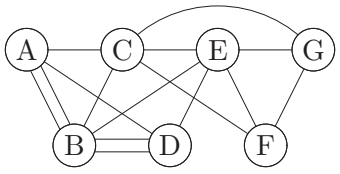

図 1: グラフによるグループ関係の表現

次に実データで、セットシステムとグラフとのモデルの違いを比較する．日本オペレーションズリサーチ学会論文誌（Journal of the Operations Research Society of Japan）と日本オペレーションズ・リサーチ学会和文論文誌（Transactions of the Operations Research Society of Japan）の 38 巻（1995 年）から 52 巻 1 号（2009 年）までに掲載されている論文の共著関係からコミュニティとして最大密度部分集合を抽出する．セットシステムモデルでは，N を著者集合とし，$\mathcal{H}$ を論文の共著者関係を要素とする多重集合族とする．また，グラフモデルでは，著者を頂点とし，共著関係を枝とする多重グラフを構成する．対象となる著者数 (N の要素数) は 570 名であり，対象となる論文数 ($\mathcal{H}$ の要素数) は 344 本である．また，グラフモデルでの枝数は 830 本である．なお，対象期間の単著の論文は除いてある．また，和文誌，英文誌の両方を含めたデータを使用するため，著者名のローマ字表記が一致していれば同一著者とした．結果として、セットシステムにおける最大密度部分集合は $\{$*Ryusuke Hohzaki, Koji Iida, Toru Komiya, Masao Mori*$\}$ となり，グラフモデルにおける最大密度部分集合は $\{$*Ryusuke Hohzaki, Koji Iida*$\}$ となった．

このようにグループ関係をグラフで表すと部分的な情報しかもたずに，セットシステムと異なる結果を導く．従って，小グループの関係が与えられるときは，セットシステムとして扱った方がよい場合もあり，グラフ上の密な部分グラフの概念をセットシステム上に拡張し，それらを抽出するための効率的なアルゴリズムの開発は重要であるといえる．次節以降で，最大密度部分集合問題に対するアルゴリズムについて述べる．

3 最大密度部分集合問題に対するパラメトリック探索法

最大密度部分集合問題は，分数計画問題の一つであり，パラメータ λ を導入した補助問題:

$$\mathrm{P}(\lambda) : \max_{S \subseteq N, S \neq \emptyset} (|\Gamma(S)| - \lambda |S|)$$

を用いて解ける．$\mathrm{P}(\lambda)$ の最適値を $z(\lambda)$ とする．以下は分数計画問題でよく知られている性質である．

補題 3.1 $z(\lambda^*) = 0$ のとき，λ^* は最大密度部分集合問題の最適値であり，$\mathrm{P}(\lambda^*)$ の最適解が最大密度部分集合となる．∎

以下，最大密度部分集合問題の最適値を λ^* と書く．$z(\lambda)$ は λ に関して凸な減少関数なので，ニュートン法や 2 分探索法で最適な λ^* を見つけることができる．

補助問題 $\mathrm{P}(\lambda)$ は，最小カット問題として解くことができる．ここで，本節と次節で用いるネットワーク流の基本的用語について述べる．連結な有向グラフ $G = (N, A)$ が入口頂点 $s \in N$ と出口頂点 $t \in N$ をもち，各枝の容量 $c : A \to \mathbb{R}_+$ が与えられているとき，頂点の非空な真部分集合 Y に対し $\sum\{c(a) \mid a = (i, i') \in A, i \in Y, i' \notin Y\}$ を Y のカット容量といい，$\kappa_c(Y)$ で表す．部分集合 $Y(\subset N)$ が $s \in Y, t \notin Y$ であるとき，Y を s-t カットといい，最小のカット容量をもつ s-t カットを最小カットという．非負のパラメータ μ が与えられたとき，s-t カット Y が任意の s-t カット X に対して $\kappa_c(Y) \leq \kappa_c(X) + \mu$ を満たすとき Y を μ-近似カットという．一方，$\varphi : A \to \mathbb{R}$ に対して，$\partial\varphi(i) := \sum_{a=(i,j)\in A, j\in N} \varphi(a) - \sum_{a=(k,i)\in A, k\in N} \varphi(a)$ とする．$\varphi : A \to \mathbb{R}$ が各枝 a の容量制約 $0 \leq \varphi(a) \leq c(a)$ を満たし，各頂点 $i \in N \setminus \{s, t\}$ で $\partial\varphi(i) = 0$ を満たすと

き，φ を実行可能流という．$\partial\varphi(s)$ を最大とする実行可能流 φ を最大流といい，任意の実行可能流 ψ に対して，$\partial\varphi(s) \geq \partial\psi(s) - \mu$ を満たす実行可能流 φ を μ-近似流という．実行可能流 φ に対する残余ネットワークを $(G_\varphi = (N, A_\varphi), c_\varphi)$ とする．実行可能流 φ が μ-近似流であることの必要十分条件は φ に対する残余ネットワーク $(G_\varphi = (N, A_\varphi), c_\varphi)$ 上に $\kappa_{c_\varphi}(Y) \leq \mu$ を満たす s-t カット Y が存在することである．$\kappa_{c_\varphi}(Y) = \kappa_c(Y) - \partial\varphi(s)$ であり，任意の s-t カット X に対して，$\partial\varphi(s) \leq \kappa_c(X)$ なので，$\kappa_{c_\varphi}(Y) \leq \mu$ を満たす s-t カット Y は μ-近似カットである．よって，μ-近似カットや最小カットは最大流を求めるアルゴリズムによって得ることができる．

補助問題 $\mathrm{P}(\lambda)$ は，以下の容量付き 2 部グラフ D 上の最小カット問題として解くことができる．Georgakopoulos–Politopoulos[5] では，$\mathrm{P}(\lambda)$ を線形計画問題として定式化し，双対変数を用いて最小カット問題を解くための容量付き 2 部グラフを定義している．ここでは，双対変数を用いずに容量付き 2 部グラフを定義する．D は頂点集合を $N \cup \mathcal{H} \cup \{s, t, d\}$，有向枝集合を $A_\mathrm{F} \cup A_\mathrm{B} \cup A_s \cup A_t \cup A_d \cup A_e \cup \{(s, d)\}$ とする．ただし，$A_\mathrm{F} = \{(i, J) \mid J \in \mathcal{H}, i \in J\}$, $A_\mathrm{B} = \{(J, i) \mid (i, J) \in A_\mathrm{F}\}$, $A_s = \{(s, i) \mid i \in N\}$, $A_t = \{(i, t) \mid i \in N\}$, $A_d = \{(d, J) \mid J \in \mathcal{H}\}$, $A_e = \{(J, d) \mid J \in \mathcal{H}\}$ とする．ここで，$h_{\max} = \max\{|J| \mid J \in \mathcal{H}\}$ とする．各枝 a の容量は λ によって，

$$c_\lambda(a) = \begin{cases} 1/h_{\max} & (a \in A_\mathrm{F}) \\ (h_{\max} - |J|)/h_{\max} & (a = (d, J) \in A_d) \\ \infty & (a \in A_\mathrm{B} \cup A_e) \\ \sum_{J \ni i} 1/h_{\max} & (a = (s, i) \in A_s) \\ \sum_{J \in \mathcal{H}} (h_{\max} - |J|)/h_{\max} & (a = (s, d)) \\ \lambda & (a \in A_t) \end{cases}$$

と定める．

補題 3.2 (D, c_λ) 上の最小カットを Y とすると，$Y \cap N = \emptyset$ のとき，$z(\lambda) \leq 0$ を得る．$Y \cap N \neq \emptyset$ のとき，$Y \cap N$ は $\mathrm{P}(\lambda)$ の最適解となる．さらに，$\kappa_{c_\lambda}(Y) = m_H - z(\lambda)$ が成り立ち，$z(\lambda) \geq 0$ を得る．

証明　はじめに，(D, c_λ) 上の最小カット Y に対して，$d \in Y$ であり，かつ，$\Gamma(Y \cap N) = Y \cap \mathcal{H}$ と仮定できることを示す．$d \notin Y$ のとき，ひとつでも $J \in Y \cap \mathcal{H}$ が存在すると，枝 (J, d) の容量が ∞ であるので Y のカット容量は ∞ となる．しかし，(D, c_λ) には有限な容量のカット，例えば $\{s\}$，が存在するので，Y は最小カットと成り得ない．そこで，$d \notin Y$ かつ $Y \cap \mathcal{H} = \emptyset$ の場合を考える．このとき，

$$\kappa_{c_\lambda}(Y \cup \{d\}) - \kappa_{c_\lambda}(Y) = \sum_{(d,J) \in A_d} c_\lambda(d, J) - c_\lambda(s, d) = 0$$

を得るので，Y が最小カットならば，$Y \cup \{d\}$ も最小カットとなる．以上より，$d \in Y$ を仮定できる．

次に，$\tilde{J} \in \Gamma(Y \cap N) \setminus (Y \cap \mathcal{H})$ であると，$\tilde{J}$ から出る枝 $(\tilde{J}, i) \in A_\mathrm{B}$ では $i \in Y$ であり，$d \in Y$ も仮定していたので，

$$\kappa_{c_\lambda}(Y) - \kappa_{c_\lambda}(Y \cup \{\tilde{J}\}) = \sum_{\{(i,\tilde{J}) \in A_\mathrm{F} \mid i \in N\}} c_\lambda(i, \tilde{J}) + c_\lambda(d, \tilde{J}) \geq 0$$

となり，Y が最小カットならば，$Y \cup \{\tilde{J}\}$ も最小カットとなる．よって，$\Gamma(Y \cap N) \subseteq Y \cap \mathcal{H}$ を仮定できる．逆に，$\tilde{J} \in Y \cap \mathcal{H} \setminus \Gamma(Y \cap N)$ のとき，$i \in \tilde{J}$ かつ $i \notin Y$ である $i \in N$ が存在する．ところで，$(\tilde{J}, i) \in A_{\mathrm{B}}$ の容量が ∞ であるので，Y のカット容量は ∞ となり最小カットとは成り得ない．以上より，$\Gamma(Y \cap N) = Y \cap \mathcal{H}$ が示せた．

さて，$d \in Y$ かつ $\Gamma(Y \cap N) = Y \cap \mathcal{H}$ である Y のカット容量 $\kappa_{c_\lambda}(Y)$ は

$$
\begin{aligned}
& \sum_{i \in Y \cap N} \sum_{\substack{(i,J) \in A_{\mathrm{F}} \\ J \notin \Gamma(Y \cap N)}} c_\lambda(i,J) + \sum_{\{(s,i) \in A_s | i \in N \setminus Y\}} c_\lambda(s,i) + \sum_{\{(d,J) \in A_d | J \notin \Gamma(Y \cap N)\}} c_\lambda(d,J) + \sum_{\{(i,t) \in A_t | i \in Y \cap N\}} c_\lambda(i,t) \\
= & \sum_{i \in Y \cap N} \sum_{\substack{J \ni i \\ J \notin \Gamma(Y \cap N)}} \frac{1}{h_{\max}} + \sum_{i \in N \setminus Y} \sum_{J \ni i} \frac{1}{h_{\max}} + \sum_{J \in \mathcal{H} \setminus \Gamma(Y \cap N)} \frac{h_{\max} - |J|}{h_{\max}} + \lambda |Y \cap N| \\
= & \sum_{i \in Y \cap N} \sum_{\substack{J \ni i \\ J \notin \Gamma(Y \cap N)}} \frac{1}{h_{\max}} + \left(\sum_{i \in N} \sum_{J \ni i} \frac{1}{h_{\max}} - \sum_{i \in Y \cap N} \sum_{J \ni i} \frac{1}{h_{\max}} \right) + \sum_{J \in \mathcal{H} \setminus \Gamma(Y \cap N)} \frac{h_{\max} - |J|}{h_{\max}} + \lambda |Y \cap N| \\
= & \sum_{i \in Y \cap N} \sum_{\substack{J \ni i \\ J \notin \Gamma(Y \cap N)}} \frac{1}{h_{\max}} + \sum_{J \in \mathcal{H}} \frac{|J|}{h_{\max}} - \sum_{i \in Y \cap N} \sum_{J \ni i} \frac{1}{h_{\max}} + \sum_{J \in \mathcal{H} \setminus \Gamma(Y \cap N)} \frac{h_{\max} - |J|}{h_{\max}} + \lambda |Y \cap N| \\
= & \sum_{J \in \mathcal{H}} \frac{|J|}{h_{\max}} - \sum_{i \in Y \cap N} \sum_{\substack{J \ni i \\ J \in \Gamma(Y \cap N)}} \frac{1}{h_{\max}} + \sum_{J \in \mathcal{H} \setminus \Gamma(Y \cap N)} \frac{h_{\max} - |J|}{h_{\max}} + \lambda |Y \cap N| \\
= & \sum_{J \in \mathcal{H}} \frac{|J|}{h_{\max}} - \sum_{J \in \Gamma(Y \cap N)} \frac{|J|}{h_{\max}} + \sum_{J \in \mathcal{H} \setminus \Gamma(Y \cap N)} \frac{h_{\max} - |J|}{h_{\max}} + \lambda |Y \cap N| \\
= & |\mathcal{H} \setminus \Gamma(Y \cap N)| + \lambda |Y \cap N|
\end{aligned}
\tag{1}
$$

なので，

$$\kappa_{c_\lambda}(Y) = m_H - (|\Gamma(Y \cap N)| - \lambda |Y \cap N|)$$

であり，Y は最小カットなので，任意の非空な $S \subseteq N$ に対し，

$$\kappa_{c_\lambda}(Y) \leq \kappa_{c_\lambda}(S \cup \Gamma(S) \cup \{s,d\}) = m_H - (|\Gamma(S)| - \lambda |S|)$$

である．$Y \cap N = \emptyset$ のとき，$\kappa_{c_\lambda}(Y) = m_H$ なので，$|\Gamma(S)| - \lambda |S| \leq 0$ であり $z(\lambda) \leq 0$ を得る．$Y \cap N \neq \emptyset$ のとき，$|\Gamma(Y \cap N)| - \lambda |Y \cap N| \geq |\Gamma(S)| - \lambda |S|$ が成り立つので，$Y \cap N$ が $\mathrm{P}(\lambda)$ の最適解となり，$\kappa_{c_\lambda}(Y) = m_H - z(\lambda)$ を得る．さらに，$\kappa_{c_\lambda}(\{s\}) = m_H \geq \kappa_{c_\lambda}(Y)$ より $z(\lambda) \geq 0$ が成り立つ．∎

従って，パラメトリック最大流問題に対する Gallo-Grigoriadis-Tarjan のアルゴリズム [4] を用いれば $\mathrm{O}((n + m_H) q \log \frac{(n+m_H)^2}{q})$ 時間で最大密度部分集合を得ることができる．

一方，Georgakopoulos–Politopoulos[5] では 2 分探索法に Goldberg-Rao の最大流アルゴリズム [7] を適用している．まず，2 分探索法による正確な計算量を算定しよう．2 分探索法では最適なパラメータ λ^* を含む探索区間 $[\mathrm{LB}, \mathrm{UB}]$ を維持する．任意の非空な $S \subseteq N$ に対して $0 \leq |\Gamma(S)|/|S| \leq m_H$ であるので，初期探索区間は $[0, m_H]$ とできる．$z(\lambda)$ が減少関数なので，$z(\lambda) > 0$ ならば LB を λ に更新し，$z(\lambda) \leq 0$ ならば UB を λ に更新して，探索区間を縮小する

ことができる．$\lambda = (\mathrm{LB} + \mathrm{UB})/2$ として補助問題 $\mathrm{P}(\lambda)$ を解いて $z(\lambda)$ の正負を調べれば，常に探索区間を半減できる．2 分探索の終了条件は以下の補題で与えられる．

補題 3.3 $\mathrm{UB} - \mathrm{LB} \leq 1/n^2$ のとき，$\mathrm{P}(\mathrm{LB})$ の最適解は最大密度部分集合である．

証明　最大密度部分集合を S^* とする．最大密度部分集合ではない任意の非空な $S \subseteq N$ に対して，$\frac{|\Gamma(S)|}{|S|} \neq \frac{|\Gamma(S^*)|}{|S^*|}$ なので，

$$\frac{|\Gamma(S^*)|}{|S^*|} - \frac{|\Gamma(S)|}{|S|} = \frac{|\Gamma(S^*)| \cdot |S| - |\Gamma(S)| \cdot |S^*|}{|S^*| \cdot |S|} > 1/n^2 \geq \mathrm{UB} - \mathrm{LB}$$

であり，$\mathrm{UB} \geq \frac{|\Gamma(S^*)|}{|S^*|}$ より，$\mathrm{LB} > \frac{|\Gamma(S)|}{|S|}$ を得る．一方，$\mathrm{LB} \leq \frac{|\Gamma(S^*)|}{|S^*|}$ なので，

$$|\Gamma(S^*)| - \mathrm{LB}|S^*| \geq 0 > |\Gamma(S)| - \mathrm{LB}|S|$$

となり，S は $\mathrm{P}(\mathrm{LB})$ の最適解にならない．∎

実際には，$\mathrm{LB} = \lambda$ のとき，(D, c_{LB}) の最小カットから $\mathrm{P}(\mathrm{LB})$ の最適解が得られないかもしれない．そこで，$\mathrm{UB} - \mathrm{LB} \leq \frac{1}{2n^2}$ まで探索区間を縮小し，$(D, c_{\mathrm{LB}-\frac{1}{2n^2}})$ の最小カットを求めれば，$\lambda > \mathrm{LB} - \frac{1}{2n^2}$ なので，$P(LB - \frac{1}{2n^2})$ の最適解を得ることができる．従って，2 分探索で $\mathrm{P}(\lambda)$ を解く回数は $\lceil \log(n^2 m_H) \rceil + 2$ となる．補助問題 $\mathrm{P}(\lambda)$ を解く際に Goldberg-Rao の最大流アルゴリズムを適用するには容量をすべて整数にしなくてはいけない．補題 3.3 より，パラメータ λ は $1/(2n^2)$ の整数倍としてよい．よって，A_t の枝の容量を $\lfloor 2\lambda n^2 \rfloor/(2n^2)$ とできる．また，A_{B} および A_e の枝の容量は ∞ であるが，D 上で任意の $J \in \mathcal{H}$ に入る枝の容量の和が 1 なので，1 よりも大きければ（例えば 2 で）十分である．このとき，すべての枝の容量を $2n^2 h_{\max}$ 倍すれば容量はすべて整数となり，各枝の容量は $2n^2 h_{\max} m_H = \mathrm{O}(n^3 m_H)$ を超えない．よって，Goldberg-Rao の最大流アルゴリズムは (D, c_λ) 上の最小カット問題を $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\} q \log \frac{(n+m_H)^2}{q} \log(n m_H))$ 時間で解き，最大密度部分集合は 2 分探索法により $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\} q \log \frac{(n+m_H)^2}{q} \log^2(n m_H))$ 時間で求められる．次節では，2 分探索法を高速化する近似 2 分探索法を適用したアルゴリズムを構築する．

4　最大密度部分集合問題に対する近似 2 分探索アルゴリズム

近似 2 分探索法は，2 分探索によりパラメトリック探索を行うアルゴリズムを高速化する常套手段であり，Zemel[16] によって基本概念が提案された．最小平均閉路問題 [13]，最大平均カット問題 [8, 9]，分数型割当問題 [14] に対して，近似 2 分探索法による高速なアルゴリズムが構築されている．最大平均カット問題に対する近似 2 分探索によるアルゴリズムではパラメトリックな最大流問題を繰り返し解いている．しかし，最大平均カット問題で解く最大流問題ではすべての枝の上限容量にパラメータ値が加えられ，すべての枝の下限容量からパラメータ値が引かれている．この性質を利用して近似 2 分探索を組み込んだアルゴリズムとなっているため，最大密度部分集合問題に直接適用することはできない．本節では，近似 2 分探索に Goldberg-Rao

の最大流アルゴリズム [7] を組み込んだ，より適用範囲が広く，効率の良いアルゴリズムを提案する．

近似 2 分探索では，2 分探索と同様に最大密度部分集合の最適値 λ^* を含む探索区間 [LB, UB] を維持する．近似 2 分探索では，固定したパラメータ λ に対して，補助問題 $\mathrm{P}(\lambda)$ を近似的に解く．以下が近似 2 分探索法の枠組みである．

ステップ **0** [LB, UB]= $[0, m_H]$ とする．

ステップ **1** $\mu = (\mathrm{UB} - \mathrm{LB})/4$，$\lambda = (\mathrm{UB} + \mathrm{LB})/2$ とする．(D, c_λ) での μ-近似カット Y を求める．

ステップ **2** $Y \cap N = \emptyset$ のとき UB を $\lambda + \mu$ に更新する．そうでないとき，LB を $\lambda - \mu$ に更新する．

ステップ **3** $\mathrm{UB} - \mathrm{LB} < \frac{1}{2n^2}$ のとき，$\mathrm{P}(\mathrm{LB} - \frac{1}{2n^2})$ の最適解を出力して終了．そうでないときは，ステップ 1 へ．

まず，近似 2 分探索によるアルゴリズムの正当性を示そう．

補題 4.1 近似 2 分探索法の各繰り返しにおいて LB$\leq \lambda^* \leq$UB を満たす．さらに，出力される解は最大密度部分集合である．

証明　得られた μ-近似カット Y が $Y \cap N = \emptyset$ のときを考える．$(D, c_{\lambda+\mu})$ での最小カットを X とする．Y が μ-近似カットなので，$\kappa_{c_\lambda}(Y) \leq \kappa_{c_\lambda}(X) + \mu$ が成り立つ．ここで，$X \cap N \neq \emptyset$ と仮定すると，$\kappa_{c_\lambda}(X) \leq \kappa_{c_{\lambda+\mu}}(X) - \mu$ である．一方，$\kappa_{c_\lambda}(Y) = \kappa_{c_{\lambda+\mu}}(Y)$ なので，$\kappa_{c_{\lambda+\mu}}(Y) \leq \kappa_{c_{\lambda+\mu}}(X)$ となり，Y も $(D, c_{\lambda+\mu})$ での最小カットとなる．よって，補題 3.2 より $z(\lambda + \mu) \leq 0$ を得る．

次に，μ-近似カット Y が $Y \cap N \neq \emptyset$ のときを考える．このとき，$\kappa_{c_{\lambda-\mu}}(Y) \leq \kappa_{c_\lambda}(Y) - \mu$ である．$(D, c_{\lambda-\mu})$ での最小カットを X' とすると，Y が μ-近似カットなので，$\kappa_{c_\lambda}(Y) \leq \kappa_{c_\lambda}(X') + \mu$ が成り立つ．$X' \cap N = \emptyset$ と仮定すると，$\kappa_{c_{\lambda-\mu}}(X') = \kappa_{c_\lambda}(X')$ なので，$\kappa_{c_{\lambda-\mu}}(Y) \leq \kappa_{c_{\lambda-\mu}}(X')$ を得る．よって，Y も最小カットとなり，補題 3.2 より $z(\lambda - \mu) \geq 0$ を得る．

以上より，近似 2 分探索法で探索区間 [LB, UB] が更新されても，LB$\leq \lambda^* \leq$UB を満たす．ステップ 3 で出力される解の正当性は補題 3.3 による．∎

さて，近似 2 分探索によるアルゴリズムにおいて μ-近似カットが効率よく求められれば，2 分探索によるアルゴリズムを高速化できる．そのために，本アルゴリズムでは，各繰り返しで (D, c_{LB}) での μ-近似流 φ を維持する．

補題 4.2 近似 2 分探索法の各繰り返しにおいて，(D, c_{LB}) での μ-近似流 φ は，(D, c_λ) において $(2n+1)\mu$-近似流となる．

証明　$\lambda = \mathrm{LB} + 2\mu$ より，(D, c_λ) では A_t の枝容量が (D, c_{LB}) よりも 2μ 増加する．よって，φ は (D, c_λ) でも実行可能流である．ここで，(D, c_{LB})，(D, c_λ) における最小カットをそれぞれ，Y_{LB}, Y_λ とすると，最大流最小カット定理と μ-近似流の定義より $\kappa_{c_{\mathrm{LB}}}(Y_{\mathrm{LB}}) \leq \partial\varphi(s) + \mu$

を得る．従って，(D, c_λ) での任意の実行可能流 ψ に対して，

$$\partial\psi(s) \leq \kappa_{c_\lambda}(Y_\lambda) \leq \kappa_{c_\lambda}(Y_{\mathrm{LB}}) \leq \kappa_{c_{\mathrm{LB}}}(Y_{\mathrm{LB}}) + 2\mu n \leq \partial\varphi(s) + \mu + 2\mu n$$

が成り立ち，$\partial\varphi(s) \geq \partial\psi(s) - (2n+1)\mu$ を得る．▮

補題 4.3 近似 2 分探索法の k 回目の繰り返しにおける，LB, μ をそれぞれ LB^k, μ^k と書く．このとき，(D, c_{LB^k}) での μ^k-近似流 φ は，$(D, c_{\mathrm{LB}^{k+1}})$ において $\frac{4}{3}(n+1)\mu^{k+1}$-近似流となる．

証明　$\mathrm{LB}^{k+1} \leq \mathrm{LB}^k + \mu^k$ であるので，$(D, c_{\mathrm{LB}^{k+1}})$ では A_t の枝容量が (D, c_{LB^k}) よりも高々 μ^k 増加する．よって，$(G, c_{\mathrm{LB}^{k+1}})$ での任意の実行可能流 ψ に対し，補題 4.2 と同様に，$\partial\psi(s) - \partial\varphi(s) \leq (n+1)\mu^k$ を得る．さらに，$\mu^{k+1} = 3\mu^k/4$ であるので，$\partial\varphi(s) \geq \partial\psi(s) - \frac{4}{3}(n+1)\mu^{k+1}$ が成り立つ．▮

ここで，Goldberg–Rao[7] による 2μ-近似流から μ-近似流を求める $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\}q \log \frac{(n+m_H)^2}{q})$ 時間の解法を $\mathrm{O}(\log n)$ 回適用すれば，(G, c_{LB^k}) での μ^k-近似流から，(G, c_λ) での μ^k-近似流を得て，さらに，$(G, c_{\mathrm{LB}^{k+1}})$ での μ^{k+1}-近似流を求められる．μ-近似流を求める過程で μ-近似カットも得られるので，近似 2 分探索の各繰り返しは $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\}q \log \frac{(n+m_H)^2}{q} \log n)$ 時間で実行できる．近似 2 分探索法の繰り返し回数は $\mathrm{O}(\log(nm_H))$ なので以下の定理を得る．

定理 4.1 近似 2 分探索法により $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\}q \log \frac{(n+m_H)^2}{q} \log n \log(nm_H))$ 時間で最大密度部分集合は得られる．▮

セットシステムにおいては，m_H が n に比べて非常に大きい場合もあり，このときは 2 分探索法よりも近似 2 分探索法の方が効率が良いと言える．

5　おわりに

本稿では，まず，コミュニティ抽出における最大密度部分集合問題の特徴を議論した．グラフ上での密な部分グラフの定義をセットシステム上に拡張してアルゴリズムを議論することは重要と思われる．

さらに，最大密度部分集合を求めるパラメトリック探索を用いたアルゴリズムを紹介し，近似 2 分探索法により理論的に効率の良いアルゴリズムが構築できることを示した．提案する近似 2 分探索法の枠組みは他のパラメトリック最大流問題にも有効である．最大密度部分集合問題において，$\mathcal{H}$ の各要素 J が正の重み $w(J)$ を持っているときに，

$$\max_{S \subseteq N, S \neq \emptyset} \frac{\sum_{J \in \Gamma(S)} w(J)}{|S|}$$

を求める重み付き最大密度部分集合問題に対しては，補助問題を解くための容量付き 2 部グラフの容量を

$$c_\lambda(a) = \begin{cases} w(J)/h_{\max} & (a = (i, J) \in A_{\mathrm{F}}) \\ w(J)(h_{\max} - |J|)/h_{\max} & (a = (d, J) \in A_d) \\ \infty & (a \in A_{\mathrm{B}} \cup A_e) \\ \sum_{J \ni i} w(J)/h_{\max} & (a = (s, i) \in A_s) \\ \sum_{J \in \mathcal{H}} w(J)(h_{\max} - |J|)/h_{\max} & (a = (s, d)) \\ \lambda & (a \in A_t) \end{cases}$$

とすれば，同様のパラメトリック探索で解くことができる．重み w がすべて整数のとき，$W = \max_{J \in \mathcal{H}} w(J)$ とすると，重み付き最大密度部分集合問題は近似 2 分探索法により $\mathrm{O}(\min\{(n+m_H)^{2/3}, q^{1/2}\} q \log \frac{(n+m_H)^2}{q} \log n \log(n m_H W))$ 時間で解ける．

提案する近似 2 分探索法の枠組みは最大平均カット問題にも適用できる．最大平均カット問題は連結な有向グラフ $G = (N, A)$ と各枝の上限容量 $u : A \to \mathbb{Z}$，下限容量 $\ell : A \to \mathbb{Z}$ が与えられたときに，

$$\max_{S \subset V, S \neq \emptyset} \frac{\sum_{a \in \Delta^+ S} \ell(a) - \sum_{a \in \Delta^- S} u(a)}{|\Delta^+ S| + |\Delta^- S|}$$

を求める問題である．ただし，$\Delta^+ S = \{(i, j) \in A \mid i \in S, j \in N \setminus S\}$，$\Delta^- S = \Delta^+(N \setminus S)$ であり，各枝 a で $\ell(a) \leq u(a)$ が成り立ち，$\sum_{a \in \Delta^+ S} \ell(a) - \sum_{a \in \Delta^- S} u(a) > 0$ となる S が存在すると仮定している．ここで，$n = |N|$, $m = |A|$, $B = \max\{|\ell(a)|, |u(a)| \mid a \in A\}$ とする．最大平均カット問題に対する既存の近似 2 分探索アルゴリズム [8, 9] の計算量は $\mathrm{O}(nm \log(nB))$ であるが，一方，本論文で示した近似 2 分探索の枠組みを適用すると，付録に示すように $\mathrm{O}(\min\{n^{2/3}, m^{1/2}\} m \log \frac{n^2}{m} \log n \log(nB))$ 時間のアルゴリズムを構築できる．以上のように，本論文で示した近似 2 分探索の枠組みは計算量の優れたアルゴリズムを構築することができ，また，最大流問題を補助問題にもつパラメトリック探索を行う問題への適用性にも優れているといえる．

謝辞

本研究は科研費（19510137) の助成を受けて行われた．

参考文献

[1] B. Balasundaram, S. Butenko, I. V. Hicks, and S. Sachdeva: Clique relaxations in social network analysis: The maximum k-plex problem, 未発表原稿.

[2] M. Brinkmeier, J. Werner, and S. Recknagel: Communities in graphs and hypergraphs, *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management* (2007), 869–872.

[3] G. W. Flake, K. Tsioutsiouliklis, and L. Zhukov: Methods for mining web communities: Bibliometric, spectral, and flow, *Web dynamics: adapting to change in content, size, topology and use*, M. Levene, A. Poulovassilis(編)(Springer, 2004).

[4] G. Gallo, M. G. Grigoriadis, and R. E. Tarjan: A fast parametric maximum flow algorithm and applications, *SIAM Journal on Computing*, **18** (1989), 30-55.

[5] G. F. Georgakopoulos, and K. Politopoulos: Max-density revisited: A generalization and a more efficient algorithm, *The Computer Journal*, **50** (2007), 348-356.

[6] A. V. Goldberg: Finding a maximum density subgraph, UC Berkeley report, UCB/CSD/84/171, (1984).

[7] A. V. Goldberg, and S. Rao: Beyond the flow decomposition barrier, *Journal of the ACM*, **45** (1998), 783-797.

[8] K. Iwano, S. Misono, S. Tezuka, and S. Fujishige: A new scaling algorithm for the maximum mean cut problem, *Algorithmica*, **11** (1994), 243-255.

[9] S.T. McCormick: Approximate binary search algorithms for mean cuts and cycles. *Operations Research Letters*, **14** (1993), 129-132.

[10] M. E. Newman: Fast algorithm for detecting community structure in networks, *Physical Review*, **E. 69** (2004), 066133.

[11] J. C. Picard, and M. Queyranne: Selected applications of minimum cuts in networks, *INFOR*, **20** (1982), 394-422.

[12] R. Qian, W. Zhang, and B.Yang: Community detection in scale-free networks based on hypergraph model, *Lecture Notes in Computer Science*, **4430** (2007), 226-231.

[13] J. B. Orlin and R. K. Ahuja: New scaling algorithms for the assignment and minimum mean cycle problems. *Mathematical Programming*, **54** (1992), 41–56.

[14] M. Shigeno, Y. Saruwatari and T. Matsui: An algorithm for fractional assignment problems. *Discrete Applied Mathematics*, **56** (1995), 333–343.

[15] 湯田聴夫:コミュニティ抽出法とその展望, オペレーションズ・リサーチ, **53** (2008), 529-535.

[16] E. Zemel: A linear time randomizing algorithm for search ranked functions. *Algorithmica* **2** (1987), 81–90.

付録. 最大平均カット問題に対する近似 2 分探索アルゴリズム

連結な有向グラフ $G=(N,A)$ と各枝の上限容量 $u:A\to\mathbb{Z}$，下限容量 $\ell:A\to\mathbb{Z}$ が与えられたときに，

$$\rho^* := \max_{S\subset V, S\neq\emptyset} \frac{\sum_{a\in\Delta^+S}\ell(a)-\sum_{a\in\Delta^-S}u(a)}{|\Delta^+S|+|\Delta^-S|}$$

を達成する S を最大平均カットという．ただし，$\Delta^+S=\{(i,j)\in A\mid i\in S, j\in N\setminus S\}$, $\Delta^-S=\Delta^+(N\setminus S)$ であり，各枝 a で $\ell(a)\le u(a)$ が成り立ち，$\sum_{a\in\Delta^+S}\ell(a)-\sum_{a\in\Delta^-S}u(a)>0$ となる S が存在すると仮定する．以下，$n=|N|$, $m=|A|$, $B=\max\{|\ell(a)|,|u(a)|\mid a\in A\}$ とする．

最大平均カット問題も分数計画問題の一つであるのでパラメータ ρ を導入した補助問題：

$$\begin{aligned} Q(\rho):\quad & \max_{S\subset V,S\neq\emptyset}\left\{(\sum_{a\in\Delta^+S}\ell(a)-\sum_{a\in\Delta^-S}u(a))-\rho(|\Delta^+S|+|\Delta^-S|)\right\} \\ &= -\min_{S\subset V,S\neq\emptyset}\left\{\sum_{a\in\Delta^-S}(u(a)+\rho)-\sum_{a\in\Delta^+S}(\ell(a)-\rho)\right\} \end{aligned}$$

を用いて解ける．補題 3.1 と同様に，$Q(\rho)$ の最適値が 0 のとき $\rho=\rho^*$ であり，$Q(\rho)$ の最適解は最大平均カットであることがいえる．$Q(\rho)$ の最適値を $\tilde{z}(\rho)$ とすると，$\tilde{z}(\rho)$ は ρ に関して減少関数なので，$\tilde{z}(\rho)\le 0$ ならば $\rho^*\le\rho$ であり，$\tilde{z}(\rho)>0$ ならば $\rho^*>\rho$ である．$\tilde{z}(\rho)$ が非正かどうかは，以下の補助ネットワーク $\tilde{D}$ 上の最小カット問題を解くことで判断できる．補助ネットワーク $\tilde{D}$ は頂点集合を $N\cup\{s,t\}$，枝集合を $A_{\rm F}\cup A_{\rm B}\cup A_s\cup A_t$ とする．ただし，$A_{\rm F}=A$, $A_{\rm B}=\{(i,j)\mid(j,i)\in A\}$，$A_s=\{(s,i)\mid i\in N,\partial\ell(i)<0\}$, $A_t=\{(i,t)\mid i\in N,\partial\ell(i)>0\}$ である．各枝 a の容量は ρ によって，

$$\tilde{c}_\rho(a)=\begin{cases} u(a)-\ell(a)+\rho & (a\in A_{\rm F})\\ \rho & (a\in A_{\rm B})\\ -\partial\ell(i) & (a=(s,i)\in A_s)\\ \partial\ell(i) & (a=(i,t)\in A_t)\end{cases}$$

と定める．$(\tilde{D},\tilde{c}_\rho)$ 上の任意の s-t カット X のカット容量 $\kappa_{\tilde{c}_\rho}(X)$ は，$X'=X\cap N$ としたとき，

$$\kappa_{\tilde{c}_\rho}(X)=\sum_{\{i\in N\mid\partial\ell(i)>0\}}\partial\ell(i)+\sum_{a\in\Delta^+X'}(u(a)+\rho)-\sum_{a\in\Delta^-X'}(\ell(a)-\rho)$$

である．よって，$(\tilde{D}_\rho,\tilde{c}_\rho)$ の最小カット X が $X\neq\{s\}$ かつ $X\neq N\cup\{s\}$ のとき，$X\cap N$ は $Q(\rho)$ の最適解となる．さらに，$\kappa_{\tilde{c}_\rho}(X)\le\kappa_{\tilde{c}_\rho}(\{s\})=\sum_{\{i\in N\mid\partial\ell(i)>0\}}\partial\ell(i)$ なので，$\tilde{z}\ge 0$ を得る．逆に，$X=\{s\}$ あるいは $X=N\cup\{s\}$ のとき，$\kappa_{\tilde{c}_\rho}(X)=\sum_{\{i\in N\mid\partial\ell(i)>0\}}\partial\ell(i)$ であり，任意の非空な $Y\subset N$ に対して，$\kappa_{\tilde{c}_\rho}(X)\le\kappa_{\tilde{c}_\rho}(Y\cup\{s\})$ なので，$\sum_{a\in\Delta^+Y}(u(a)+\rho)-\sum_{a\in\Delta^-Y}(\ell(a)-\rho)\ge 0$ となり $\tilde{z}(\rho)$ が非正となることが分かる．従って，最大平均カット問題は，パラメータ探索において最大流問題を繰り返し解けばよい．最大平均カットの定義より $0<\rho^*\le B$ なのでパラメー

タの初期探索区間 $[\mathrm{LB}, \mathrm{UB}] = [0, B]$ とできる．また，補題 3.3 と同様に $\mathrm{UB} - \mathrm{LB} < 1/m^2$ のとき $Q(\mathrm{LB})$ での最適解が最大平均カットとなる．

以下が近似 2 分探索アルゴリズムの枠組みである．

ステップ **0** [LB, UB]= $[0, B]$ とする．

ステップ **1** $\mu = (\mathrm{UB} - \mathrm{LB})/4$，$\rho = (\mathrm{UB} + \mathrm{LB})/2$ とする．$(\tilde{D}_\rho, \tilde{c}_\rho)$ での μ-近似カット Y を求める．

ステップ **2** $Y = \{s\}$ あるいは $Y = N \cup \{s\}$ のとき，UB を $\rho + \mu$ に更新する．そうでないとき，LB を $\rho - \mu$ に更新する．

ステップ **3** $\mathrm{UB} - \mathrm{LB} < \frac{1}{2m^2}$ のとき，$Q(\mathrm{LB} - \frac{1}{2m^2})$ の最適解を出力して終了．そうでないときは，ステップ 1 へ．

近似 2 分探索アルゴリズムの正当性は補題 4.1 と同様に示すことができる．

最後に近似 2 分探索アルゴリズムの計算量を算定する．最大密度部分問題に対する近似 2 分探索法と同様に μ-近似カットを効率よく求めるために，$(\tilde{D}, \tilde{c}_{\mathrm{LB}})$ での μ-近似流 φ を維持する．このとき，各繰り返しで，Goldberg–Rao[7] の 2μ-近似流から μ-近似流を求める $\mathrm{O}(\min\{(n + m_H)^{2/3}, q^{1/2}\} q \log \frac{(n+m_H)^2}{q})$ 時間の解法を $\mathrm{O}(\log m) = \mathrm{O}(\log n)$ 回適用すれば，$(\tilde{D}, \tilde{c}_{\mathrm{LB}})$ での μ-近似流 φ から $(\tilde{D}, \tilde{c}_\lambda)$ での μ-近似流と μ-近似カット Y を得ることができる．さらに，ステップ 2 で LB が更新されたときも更新後の LB に対する $(\tilde{D}, \tilde{c}_{\mathrm{LB}})$ での μ-近似流を同様に求めることができる．近似 2 分探索の繰り返し回数は $\mathrm{O}(\log(nB))$ なので，以下の結果を得る．

定理 5.1 近似 2 分探索アルゴリズムは $\mathrm{O}(\min\{n^{2/3}, m^{1/2}\} m \log \frac{n^2}{m} \log n \log(nB))$ 時間で最大平均カットを得る．

A maximum density subset problem and its algorithm with approximate binary search

ZHANG Mingchao, TAKAHASHI Satoshi, and SHIGENO Maiko
University of Tsukuba

Abstract This paper deals with a problem of finding maximum density subsets on a set system, which is a generalization of a maximum density subgraph problem. To find dense subgraphs is worthy in analyzing communities on either web graphs or social networks. Some examples show that maximum density subsets are proper than maximum density subgraphs as communities. By combining approximate binary search and a maximum flow algorithm, an efficient algorithm for finding maximum density subsets is developed. We also discuss how a framework of the proposed approximate binary search algorithm can be applied for a weighted version of the problem and for a maximum mean cut problem.