3J3-1

極大 (j,k)-疑似クリークの全列挙に関する一考察

An Algorithm for Enumerating Maximal j-cored Connected k-Plexes

松平 将宜 [*1] Masanobu Matsudaira　原口 誠 [*1] Makoto Haraguchi　大久保 好章 [*1] Yoshiaki Okubo　富田 悦次 [*2] Etsuji Tomita

[*1] 北海道大学大学院情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University

[*2] 電気通信大学先進アルゴリズム研究ステーション
The Advanced Algorithms Research Laboratory,The University of Electro-Communications

A notion of k-Plexes has been originally introduced to define a class of pseudo-cliques with the property of anti-monotonicity. By the anti-monotonicity, a simple enumeration algorithm can be designed just like a maximal clique generator. However, sparse k-Plexes exist in general particularly for larger k. To restrict our consideration to a subclass of more dense pseudo cliques, we consider additional constraints that k-Plex must be connected and a j-core. A vertex set inducing j-cored connected k-Plex is called a (j,k)-PC. Although the class of (j,k)-PC does not satisfy anti-monotonicity, we can present a branch-and-bound algorithm for enumerating all (j,k)-PC by extending standard maximal k-Plex or clique enumerator. Particularly, the branch-and-bound control is used to cut off hopeless vertex sets that cannot grow to some (j,k)-PC.

1. はじめに

グラフやネットワークにおいて，クリークは最も密なコミュニティを表現している．一方で，現実のコミュニティではクリークになるとは限らず，クリークの緩和モデルが多数提案されてきた [3]．本稿では，その中の一つである k-Plex [4] を基にし，新たな疑似クリークを提案し，その列挙法について考察する．

k-Plex とは非接続数上限制約（各頂点に対し非接続な他の頂点は高々 $k-1$ 個以下）を満たす無向グラフの頂点集合であり，特に $k=1$ の場合はクリークとなる．比較的にサイズ大な頂点集合に対しては，許容できる非接続頂点数も一般に増加し，k-Plex は現実に則したモデルとなり得る．一方，k の増加に伴い，k-Plex の総数は指数的に増え，全列挙は困難となる．この問題を克服するための一つの方法は連結な k-Plex に制限することである ([5])．極大クリークや極大 k-Plex の全列挙では，形成過程の頂点集合に，候補と呼ばれる頂点を追加し頂点集合を単調に増加させる方法が標準である．連結な k-Plex では，後述するように，現在の頂点集合に追加可能な候補を，単純な k-Plex のそれよりもさらに絞りこむことができ，対象となる頂点集合の総数を小さくできる利点がある．しかしながら，連結であっても密結合とは言えず，例えば鎖状や環状に接続された頂点集合も 連結 k-Plex として許容されてしまい，密度の高い疑似クリークとは言い難い．

本稿では，一定の密度を持つ連結 k-Plex を得る目的で，j-核性 [3] を第３の制約として課す．j-核性頂点集合とは，頂点毎の接続数下限制約（[1, 3]）を満たすものであり，本稿の抽出目標は極大な j-核性 連結 k-Plex である．すなわち，連結性，接続数下限制約，非接続数上限制約の３つの制約を同時に満たす極大疑似クリークの全列挙器を考える．特に，j-核性連結 k-Plex のクラスは，逆単調性を持たないが，逆単調性を持つ k-Plex の枚挙器に，頂点毎の分枝限定枝刈り規則を搭載することが可能であり，高速な全列挙が期待できる．

連絡先: 原口 誠
北海道大学大学院情報科学研究科
〒 060-0814 札幌市北区北 14 条西 9 丁目
mh@ist.hokudai.ac.jp

2. 用語と基本事項

ここでは，単純無向グラフ $G=(V,E)$ を考え，以下では単にグラフと呼ぶ．頂点 $v \in V$ について，G における v の開隣接頂点集合を $\Gamma(v)$ とする．すなわち $\Gamma(v)=\{u \in V \mid (u,v) \in E\}$ である．簡単のために，$x \notin \Gamma(x)$ を仮定する．また，$|\Gamma(v)|$ を v の次数と呼び，$deg(v)$ とする．グラフ $G=(V,E)$ の頂点集合 $X \subseteq V$ において，$G[X]=(X, E \cap (X \times X))$ で定義されるグラフを X による G の誘導部分グラフと呼ぶ．G の頂点集合 $X \subseteq V$ が誘導する部分グラフが連結であるとき，X を連結，また，その中で極大なものを連結成分と言う．

3. 連結な k-Plex

定義 1 k-Plex
グラフ $G=(V,E)$ の頂点集合 $X \subseteq V$ を考える．ここで，任意の頂点 $x \in X$ に対して，$|X \setminus \Gamma(v)| \leq k$ となる時，$G[X]$（または単に X）を k-Plex と呼ぶ．特に，X が連結なとき，連結 k-Plex と呼ぶ.

定義から，特にサイズが k 以下の任意のグラフは k-Plex であり，連結性は保証されないことがわかる．k-Plex は逆単調 (anti-monotone) である．すなわち，X が k-Plex ならば，その部分集合 Y ならびに $x \in X-Y$ に対する $Yx = Y \cup \{x\}$ はともに k-Plex となる．x は Y に対する （k-Plex）候補と呼ばれる．極大 k-Plex X は部分 k-Plex X_j にその候補 x_{j+1} を追加する操作を繰り返すことにより構成できる．極大 k-Plex 枚挙器は，k-Plex 列 $\{X_j\}_j$ を探索パスとして持つ探索木の展開により，全ての極大 k-Plex を重複なく列挙する．

$$\emptyset = X_0 \subset X_1 = X_0 x_1 \subset X_2 = X_1 x_2 \subset \cdots \subset X_n = X_{n-1} x_n = \{x_1, ..., x_n\} \quad k\text{-Plex 増加列}$$

一方，連結 k-Plex X の場合は，X は特に連結集合であり，連結部分集合列の構成法が基本となる．

$\emptyset = X_0 \subset X_1 = X_0 x_1 \subset X_2 = X_1 x_2 \subset$
$\cdots \subset X_n = X_{n-1} x_n = \{x_1, ..., x_n\}$ 連結集合増加列
ただし，x_{j+1} は X_j に隣接，すなわち，$\Gamma(x_{j+1}) \cap X_j \neq \emptyset$.

連結集合増加列において，X が k-Plex の場合は，列中の連結集合 X_j は全て k-Plex である．すなわち，x_{j+1} は X_j の k-Plex 候補でもある．よって，連結 k-Plex の探索においては，k-Plex の候補で X_j と直接隣接したものだけを選択・追加するだけで良い．これは k が大な場合，サイズが小な初期の X_j に対して膨大な候補を試みなければならない 極大 k-Plex 探索と比較すると，探索枝が大幅に削減されることは明らかであろう．

3.1 直接隣接しない k-Plex 候補

連結な k-Plex X_j の k-Plex としての候補は，前節の議論から明らかなように，下記の２つの種類の候補に分類できる：

隣接候補 ： X_j と直接隣接した k-Plex 候補

非隣接候補 ： X_j と直接隣接していない k-Plex 候補

非隣接候補は，隣接候補の追加の後に，隣接候補となりえるものと，隣接候補になる可能性がないものに，さらに分類される．すなわち，X_j の k-Plex 候補集合を $C(X_j)$ と記したとき，$X_j \cup C(X_j)$ は複数の連結成分に分解され，X_j を含まない連結成分に属する $x \in C(X_j)$ は将来追加されることは決してない．上記の考察に基づき，連結 k-Plex における候補を，X_j を含む連結成分における k-Plex 候補の集合として定め，これを $Cand(X_j)$ と表記する．

定義 2 連結 k-Plex の候補
連結な k-Plex X に対し，候補集合 $Cand(X)$ を下記で定める
$\{x \in C(X) \mid Xx$ は $X \cup C(X)$ の連結成分の部分集合 $\}$

j-核性 k-Plex の検出・構成手法においては，このようにして絞りこまれた候補を想定したうえで，接続可能数を評価し，接続数下限値に満たない場合は枝刈りを行う．したがって，連結性条件を考慮した上記の候補の定義は接続数下限制約を実装する段階で本質的な役割を演じることに注意したい．

4. j-核性連結 k-Plex

k の増加に対して，密なものを抽出するために制約を考える．ここでは，グラフの統計的性質，細かい次数の分布を知る手段の一つとして用いられている j-Core [7] を考える [*1].

定義 3 j-Core
グラフ $G = (V, E)$ の頂点集合 $X \subseteq V$ を考える．ここで，任意の頂点 $v \in X$ に対して，$deg_{G[X]}(v) \geq j$

ここで，j-Core の要素数は j 以上であるため，k よりも大きい値を設定することによって密なものを得ることができる．

定義 4 j-核性連結 k-Plex
グラフ $G = (V, E)$ の連結集合 $X \subseteq V$ を考える．ここで，$G[X]$ が k-Plex かつ j-Core である時，j-核性 連結 k-Plex と呼び，(j, k)-PC と表記する．

*1 元論文では k-core と表記されるが，k-Plex の k と区別するために，ここでは j-core と記す．

ここで，注意すべき点として，極大な j-核性連結 k-Plex は必ずしも極大連結 k-Plex とは限らないことである．このことは，j-核性制約により，連結 k-Plex に追加可能な候補がさらに絞りこまれることを示唆しており，次節で詳しく論じる．

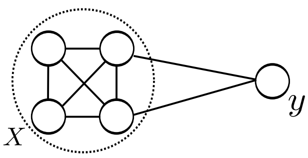

図 1: 極大連結 k-Plex でない 極大連結 (j, k)-PC の例

図 1 における X は $(3, 3)$-PC であり，$X \cup \{y\}$ は 連結 3-Plex だが，$(3, 3)$-PC ではない．よって，X は極大な $(3, 3)$-PC だが，極大な連結 3-Plex ではない．

j-核性 k-Plex の非逆単調性

連結 k-Plex の場合，極大連結 k-Plex を構成する頂点集合列 $\{X_n\}_n$ における X_n は全て連結 k-Plex であった．一方，j-核性を加味した (j, k)-PC の場合は，特に集合列の初期段階で出現する X_n はサイズが小であり，j-核性は成立しない．また，次の図で例示されるように，極大でない (j, k)-PC に頂点を追加する際，途中で出現する連結 k-Plex は必ずしも j-核性を持つとは限らない．このことにより，極大 (j, k)-PC を定める連結 k-Plex の構成列においては，候補の追加操作により将来 (j, k)-PC に成長できる可能性のある候補を追加する必要があり，逆に，そうした候補のみを追加すれば十分であることもわかる．

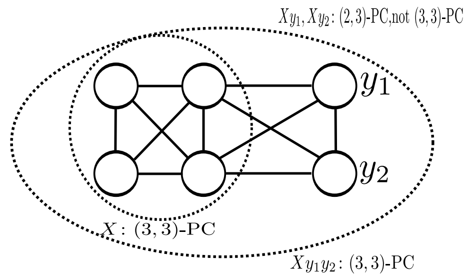

図 2: (j, k)-PC の非逆単調の例： $X \subset Y \subset Z$ に対して，X, Z が (j, k)-PC であるとする．Y は必ずしも (j, k)-PC ではない．

5. 極大な j-核性連結 k-Plex の全列挙法

連結 k-Plex の列挙法を土台に (j, k)-PC の列挙法を提案する．(j, k)-PC の構成においては，頂点集合 X に対する候補として，将来 (j, k)-PC に成長する可能性のある候補のみを保持し，その中で特に，X と直接隣接した候補のみを追加する形で実現できる．別の表現をすれば，将来 (j, k)-PC に成長する可能性のない候補は枝刈りされる．

5.1 探索中に満たすべき条件

連結 k-Plex とは異なり，探索中の頂点集合内のみを確認するだけでは (j, k)-PC の場合不十分である．すなわち，探索中の頂点集合 X と k-Plex の候補頂点集合 $Cand(X)$ との間に次の条件が必要である．この条件が成立しない場合は，X に

候補を追加しても j-核性は成立することはなく，よって，X 以降の探索（候補の追加）は無意味であり，X 自体が枝刈り・棄却される．

任意の $x \in X$ に対して，$|\Gamma(x) \cap (X \cup Cand(X))| \geq j$.

X に対して，拡張した後にこの条件を満たすように候補頂点集合を次のように定義する．

定義 5 $(j,\, k)$**-PC の候補頂点集合**

$$j\text{-}Cand(X) =$$
$$\{\ y \in Cand(X)\ \mid\ \forall z \in Xy,\ |N(z) \cap (Xy \cup Cand(Xy))| \geq j\}$$

極大 (j,k)-PC は，j-$Cand$ 候補の追加により達成される：

定理 1
(j,k)-PC X と連結 k-Plex の構成列 $\{X_i\}_i$ を考える．$X = \{x_1, ..., x_n\}$，$X_0 = \emptyset$，$X_i = X_{i-1}x_i$ である．このとき，$x_{i+1} \in j$-$Cand(X_i)$ が成立する．さらに，X が極大ならば j-$Cand(X) = \emptyset$ である．

すなわち，空集合から探索を開始し，j-$Cand$ 中の候補を順次追加することにより，極大 (j,k)-PC を求めることができる，

連結 k-Plex 列の生成は j-$Cand$ が空集合のとき停止する．j-$Cand(X_n) = \emptyset$ とは，(j,k)-PC に追加される可能性のある候補集合がないことを意味していた．よって，X_n が j-核性を満たしている場合は 極大 (j,k)-PC であり，逆にそうでない場合は失敗探索ノードとして枚挙器は親探索ノード（連結 k-Plex 列における一つ前の連結 k-Plex X_{n-1}）にバックトラックし，j-$Cand(X_{n-1})$ の別の候補の追加を試みる．

極大 (j,k)**-PC 探索での右候補制御**

オーバーラップするクリークや疑似クリークが多数存在するグラフの場合，重なり部分の探索枝を回避する手法は，探索の高速化に多大に寄与することが報告されている [2, 6]．本稿では，重なり部分の候補頂点を右候補と呼び，クリーク探索同様に右候補枝刈りが可能なことを示せたので，最後に報告しておく．

現在の連結 k-Plex とその候補 $u \in j$-$Cand(X)$ に隣接した候補 $y \in j$-$Cand(X)$ で条件

$$|\Gamma(y) \cap X| - 1 \leq |\Gamma(u) \cap X|.$$

を満たすものを右候補と呼ぶ．右候補と X から構成される (j,k)-PC $Z \subseteq j$-$Cand(X)$ に対し，XuZ は (j,k)-PC となり，よって，$X \cup Z$ は極大 (j,k)-PC になることはない．よって，少なくとも極大な (j,k)-PC の探索においては，右候補を探索枝として使う必要はない．

6. 実験

頂点数 2851，辺数 15093 のベンチマークグラフ [8] を用いた予備実験を行った．最大次数は 17 のグラフである．連結 k-Plex と (j,k)-PC との出力数，及び計算時間について比較する．
4-Plex:5397572(369)
(4,4)-PC:92530(44), (5,4)-PC:1631(20), (6,4)-PC:781(7)
ただし，出力数 (実行時間 [秒]) と表記した．以上から，出力数は大きく減少し計算時間の短縮に寄与できたといえる．

7. おわりに

本稿では，連結 k-Plex を基に新たに (j,k)-疑似クリークと，完全性を保つ列挙法を提案した．

疑似クリークの定義は様々なものがあるが，枝密度を用いたものが主流であろう．本研究における (j,k)-PC は，枝の数に関する制約を組み合わせたものであり，枝密度は疑似クリークのサイズを用いて間接的に表現することになる．すなわち，抽出目標とするサイズを大まかに想定し，枝密度から接続数下限と非接続数上限制約を設定することである．抽出目標としては，高次数頂点の小規模の疑似クリークを求めたいのか，あるいは逆に，低次数頂点の比較的大きめの疑似クリークを求めたいかによりパラメータ設定は当然変わる．高次数頂点の組み合わせに関しては，次数分布が power law に従うことが多く，高次数頂点数がそもそも少数なことから，十分に高速に動作することが期待できよう．一方，低次数頂点の数は一般に膨大であり，大きめの k に対しては非力であった k-Plex の手法を，連結性と j に基づく候補削減効果によりどれだけ改善できるのかが実際の検証課題となる．その効果を示す予備的な結果は実験の節で軽く述べたが，本格的な実験については口頭発表時に報告したい．

参考文献

[1] Seidman, S. B.: Network Structure and Minimum Degree, Social Networks, 5, pp. 269 - 287, 1983.

[2] Tomita, E., Tanaka, A. and Takahashi, H.: The Worst-Case Time Complexity for Generating All Maximal Cliques andComputational Experiments, Theoretical Computer Science 363(1), pp. 28 - 42, Elsevier, 2006.

[3] Pattillo, J., Youssef, N. and Butenko, S.: Clique Relaxation Models in Social Network Analysis, Thai, M. T. and Pardalos, P. M. (eds.), Handbook of Optimization in Complex Networks: Communication and Social Networks, Springer Optimization and Its Applications 57, pp. 143 - 162, Springer, 2012.

[4] Seidman, S. B. and Foster, B. L.: A Graph Theoretic Generalization of the Clique Concept, Journal of Mathematical Sociology 6, pp. 139 - 154, Taylor and Francis, 1978.

[5] Wu, B. and Pei, X.: A Parallel Algorithm for Enumerating All the Maximal k-Plexes, Proc. of the PAKDD 2007 Workshops, LNAI-4819, pp. 476 - 483, 2007.

[6] Okubo, Y., Haraguchi, M., and Tomita, E.: Structural Change Pattern Mining Based on Constrained Maximal k-Plex Search, Proceedings of the 15th International Conference on Discovery Science - DS'12, LNAI 7569, pp. 284 - 298, 2012.

[7] Batagelj, V. and Zaversnik, M.: An $O(m)$ algorithm for cores decomposition of networks, Advances in Data Analysis and Classification, Vol 5, pp. 129-145, Springer, 2003.

[8] Bader, D. A., Meyerhenke, H., Sanders, P., and Wagner, D.: 10th DIMACS Implementation Challenge: Graph Partitioning and Graph Clustering, 2011.