2C1-2

疑似独立集合制約と正規化カットを用いたグラフの構造比較

Contrasting Two Graphs under Constraints about Normalized Cut and Pseudo Independent Sets

間澤 直寛[*1] Naohiro Mazawa　ジェイ 泓杰[*1] Hongjie Zhai　原口 誠[*1] Makoto Haraguchi　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
Advanced Algorithms Research Laboratory, The University of Electro-Communications

This paper presents a fast algorithm for finding significant contrast sets of vertices over two graphs. The vertex sets are required to be dense clusters at a target graph and to be pseudo independent sets at a base graph. To realize an efficient detection of such sets, the target graph is transformed to a density-variable planner graph for which a standard maximal clique mining engine is applied, while the constraint about pseudo independentness at the base must be simultaneously checked.

1. はじめに

目まぐるしく変化する情報の世界においては，変化および変化の兆しや地域により注目されるトピックの違いを検出することは重要なタスクであると考えられる．本研究では，グラフ構造 [1] における差異の検出をターゲットとし，異なる地域，イベント発生の前後等，異なるグラフを比較し差異を表す頂点集合を検出する方法を与える．すなわち，ターゲットグラフ G_T で密結合であり，かつ，ベースグラフ G_B では疎結合となる頂点集合を求めるマイニング問題を考察する．文献 [4] では単調性を持たない孤立疑似クリーク枚挙により，グラフの差異を検出する方法論が提案されているが，同論文で定めた疑似クリークが単調性を持たないために高速な検出器の設計が困難であること，かつ，G_B で疎結合であればあるほど，差異の検出の困難性は増すという問題点を有していた．

こうした先行研究が持つ問題点をも踏まえ，[5, 6] では，疑似クリークとしては，ネットワーク解析 [1] においても実際に用いられており，かつ，単調性を有する疑似クリークである k-Plex [9] のクラスを考え，

- G_T で k-Plex かつ G_B で疑似独立集合（補グラフにおける k-Plex）となる頂点集合で極大となり，
- かつ，G_T における孤立性，および G_B における発散性

を満たすものを考察した．頂点集合の G_T における孤立性は，クラスターとして認識できることと等価であり，ベース G_B においてはクラスターとしては認めがたいが，G_T においてはクラスタを形成する頂点集合を求めていると理解して良い．ちなみに，G_T におけるクラスターを先に求め，G_B での非クラスタ性を検証する素朴な方法もあり得るが，出力頂点集合はクラスタリング手法が持つバイアスに強い影響を受け，G_B で疎，G_T で密なものを検出しそこなうリスクは常にある．こうした点を考慮し，[5, 6] および本稿では，特定のクラスタリング手法には依存しない形で，条件を満たす頂点集合を列挙するマイニング問題を追求している．

さて，前掲の [6] では k-Plex に対する制約マイナーとして実装し，Twitter graph から，イベントの前後における顕著な構造的差異の検出に成功したが，より大きな k に対しては，パフォーマンスは急速に低下する点が難点であった．k-Plex における k とは，許容できる非結合な頂点数を表し，大きな k に対しては，非結合の組合せ爆発を伴うからである．変化を表す頂点集合はそれなりに大きくなることもあり，そうした頂点集合の抽出を目的として，本稿では下記を新たに提案する：

- ターゲット G_T における頂点集合の指標としては，クラスターとしての指標となりえる正規化カットを採用する．通常のスペクトル法 [2] に従えば，固有（部分）空間に射影後のベクトルの類似性により頂点の近接性を計量する．しかるに，もとのグラフによっては，近接性の度合いが局所的に異なることも考慮し，本稿では，射影後の各ベクトル毎に近傍パラメータを決める
- 各点ごとのこうした局所密度を反映した変換グラフを求め，変換グラフにおけるクリークを求める．

クリークは非結合な反例を許さず，それゆえに，高速な全列挙が可能なことが既に良く知られている．G_T の変換グラフにおけるこうしたクリーク制約に加え，G_B における同じく単調性を持つ疑似独立集合制約を同時に課すことにより，より大規模なグラフに対しよりサイズ大な変化頂点集合を高速に検出できることを示す．

2. 正規化カット

求めたい頂点集合 A は，ターゲットを表す無向グラフ G_T においては，内部で密結合でかつ外部への接続が少数でなければならない．その指標として正規化カット Ncut を用いる．

$$\mathrm{Ncut}(A, \overline{A}) = \frac{\mathrm{cut}(A, \overline{A})}{\mathrm{vol}(A)}.$$

ここで $\mathrm{vol}(A)$ は A 中の頂点の次数の総和であり，また，$\mathrm{cut}(A, \overline{A})$ は，A とその外部を接続する辺の総数である．内部

連絡先: ジェイ 泓杰
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
zhaihj@kb.ist.hokudai.ac.jp

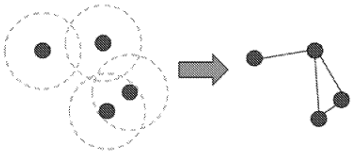
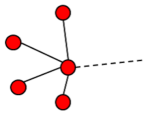


図 1: 単位円グラフ



	1	2	3	4	5	6	
1	0	0	0	0	1	0	
2	0	0	0	0	1	0	
3	0	0	0	0	1	0	
4	0	0	0	0	1	0	
5	1	1	1	1	0	0	
6	0	0	0	0	0	0	

図 2: スター構造グラフの例

で密結合で外部との接続数が小なほど，Ncut$(A, \overline{A})$ は小さくなる傾向を持ち，この理由により，G_T においては，頂点集合 A を Ncut で評価する．

Ncut に基づく近接性は，正規化ラプラシアン L_{norm} の正の固有値の固有ベクトルが張る部分空間において，頂点次数の平方根の逆数で補正したベクトルの近接性で測れることが良く知られている．本稿においては，こうした近接性の議論と，ターゲット G_T において，孤立疑似クリークとなるものが近似的に単位円グラフ内の点に写像されることを経験則として用いる．すなわち，単位円グラフにおける極大クリークを枚挙する．ただし，固定の円半径を用いるのではなく，点毎に半径を依存させ，さらに，ベース G_B における疑似独立集合制約を設ける．これらについては，次節以降で説明する．

2.1 単位円グラフ

単位円グラフ (Unit Disk Graph) とは，平面上に配置された頂点から半径 r の単位円を展開し円が交差，または接していれば辺を張ることで作られるグラフである．単位円グラフは主にアドホックネットワークのモデルとしてつ知られている．[8] アドホックネットワークとは基地局やアクセスポイントなどを用いずに端末のみで構成されるネットワークある．各端末の通信半径が全て等しいと仮定したときに得られるネットワークが単位円グラフとなる．L_{norm} の最小固有値を距離空間に射影し，一定距離以内の頂点同士で辺を張ったグラフを作る．

固有ベクトルを距離パラメータとした場合，単純な距離パラメータでは取り出しにくいクラスタがある．それが，2 のようなスターグラフに近い構造の場合である．この場合，スター構造周辺の固有ベクトルは非常に近い値となるが，スター構造の中心の固有ベクトルとは多少離れる．この距離は中心につながる数や固有ベクトルの大きさに左右され一様な距離で近さを計れない．距離空間上で頂点の辺を張る距離を一定の基準にすると，距離の値が小さい場合，スター構造周辺の点しか得られず部分集合内部にほとんど辺をもたない Ncut が小さくない集合が検出されることがある．しかし，辺を張る距離を大きくすると別の問題が発生する．距離を大きく取ると部分集合内部で全体が疎だが，局所的に密である集合が検出される．クリークやそれに近い密な集合は，つながっている頂点がほとんど同じ

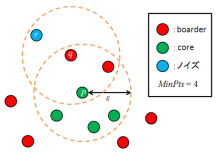


図 3: DBSCAN の例

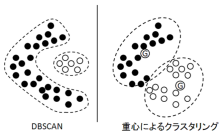


図 4: クラスタリングの比較

になり，似た固有ベクトルの値を持つ．辺を張る距離が大きい場合，密な集合に対してわずかに辺を張っているだけの頂点が大量に追加されるという問題が起こる．そのため以下の手段を取る．

1. DBSCAN でというクラスタリング手法で密度が一定以上の集合を取り出す．
2. 得られたクラスタの各頂点からクラスタ外の最近傍の頂点集合 (以降 Cand) を取り出し，辺を張る．

これにより，クラスタごとに動的な近さの基準を作る．追加される頂点 Cand の数が制限されるので，クリークのような密な集合に Cand が追加されたとしても，全体の密度はそれほど下がらない．

2.2 DBSCAN

DBSCAN とは，ある点 $p_i (i = 1, ..., n)$ から一定の距離 d 内にある点の数を数え，規定値 m 個以上の点が d 内に含まれていれば p_i をクラスタの core，core から距離 d 内にあり core の条件を満たさない点をクラスタの boarder として，core と boarder を合わせた集合を一つのクラスタとするクラスタリング手法．3 のように，規定値 $m = 4$ とすると，点 p から距離 d 以内に五つの点が含まれるので点 p は core となる．点 q は距離 d 以内に三つの点しかもっていないため core にはなれない．しかし，core である点 p から距離 d 以内に含まれているため boader としてクラスタに含まれる．点 r のような core ではなく，別の core にも含まれない点はどのクラスタにも含まれないノイズとして破棄される．

DBSCAN は一定距離内の点を追うようにクラスタリングできるので複雑な形のクラスタを取り出すことができる．また，計算量は $O(n\log\ n)$ と非常に高速である．一般的な重心を用いたクラスタリングでは円または楕円形のクラスタとなるが，DBSCAN では複雑な形のクラスタを取り出せる．4 のようなデータセットを二つに分ける場合，左側の細長く折れ曲がったデータの集まり (以降クラスタ A) と右側の小さな集まり (以降クラスタ B) で分けたい．重心を用いたクラスタリングではクラスタ A に囲まれるようにクラスタ B があるため，4 右側のようにクラスタ B とクラスタ A の一部を結合させることになり，二つ以上に分けられるとしてもクラスタ A を分割しなければならない．しかし，DBSCAN は近くにある点を追うようにクラスタリングするためクラスタ A のような複雑な形でも一つのクラスタとして取り出せる．

DBSCAN によりクラスタの core となる点は一定の距離 d 以内に必ず m 個以上の頂点を持っており，単位円グラフの半径 $r \geq d/2$ ならば少なくとも m 本以上の辺を張ることができ，クリークを枚挙する際にある程度大きな頂点集合に絞り込むことができる．

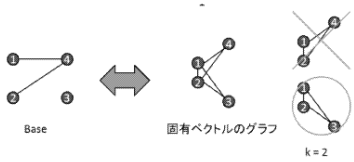


図 5: 補グラフの k-plex 制約の例

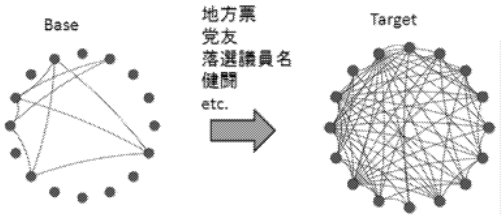


図 6: 得られた結果

3. 疑似独立集合

本稿では，グラフの比較を行うため一方のグラフ (以降 Target) には 2 節のノーマライズカットによる制約を設け，そして，もう一方のグラフ (以降 Base) には疑似独立集合による制約を課する．独立集合とは集合内部に一切辺を持たない集合であり，疑似独立集合は集合内の辺がほとんどつながっていない集合を指す．補グラフに k-plex 制約を要請することで疑似独立集合を検出する．

3.1 k-plex

k-plex とはクリークの緩和モデルの一つで，各頂点ごとに非隣接頂点数を制限している．k-plex は自信をのぞいた非隣接頂点数は高々k-1 までの条件を満たす頂点集合．独立集合とクリークは逆の概念であり，クリークの補グラフが独立集合である．よって，本稿ではクリークの緩和モデル，疑似クリークである k-plex の補グラフを疑似独立集合として扱う．

3.2 グラフ比較のための制約

3 節の Target から得られた固有ベクトルを距離空間に射影し，距離が近い頂点同士で辺を張ったグラフに対して極大クリークを枚挙する．その際，Base で同じ頂点集合の部分グラフを作り，その補グラフが k-plex 制約を満たすか確認する．k-plex 制約を満たすか確認する方法は非常に簡単であり，部分集合内の頂点の次数を数えるだけで組み合わせなどは一切使わない．Target から得たクリークが Base において k-plex 制約を満たさなければ，Target においては内側に密な閉じた集合であるが，Base において疑似独立的ではなく，Target・Base の差異が小さい集合とする．図 5 のように，頂点 1〜4 のグラフを Target の固有ベクトルから得られたとする．そこから極大クリークを枚挙すると，頂点 $C_1 = \{1, 2, 3\}$ と $C_2 = \{1, 2, 4\}$ の集合が得られる．集合 C_1 は Base において頂点の次数がすべて 0 でありクリークとして成立する．しかし，集合 C_2 は Base において頂点 4 が次数 2 であるため，k-plex の制約を満たさないため集合 C_2 は破棄される．補グラフが k-plex 制約を満たすクリークは，Target において内側に密な閉じた集合であり，Base において疑似独立な集合を取り出せたことになり，それが最終的な出力となる．

4. 実験

本稿での提案手法を Java および Mathematica で実装し，CPU：Intel(R) Xeon(R) CPU E55200(2.37GHz)，主記憶：24GB の PC 上で，新聞記事の単語を頂点とする無向グラフを対象とした構造の差異の検出を試みた．

4.1 Twitter ユーザの関係グラフ

某新聞社の 2006 年 9 月の新聞記事において，某政党に関する記事を集め，それらに出現する単語を頂点，単語同士の共起を辺として頻度で重みを付けたものをグラフとする．具体的には，単語 A と単語 B が一つの記事内に同時に出現した場合，辺 (A;B) を張ることで単語間の関係グラフを作成する．共起した回数により辺に重みを付ける．全国版の記事から抽出した単語間の関係グラフをグラフ G_{base}，北海道版からのそれをグラフ G_{target} とし，そこで観測される構造差異の検出を試みる．

4.2 パラメータ設定

本節では，実験で設定するパラメータについて説明する．実験データとして，頂点数 749(G_{base}, G_{target} 共通)，辺数 65463(G_{base})，87226(G_{target}) のグラフを用いる．他のどの頂点とも辺をもたない孤立した頂点がある場合はスペクトル解析できないため，全国版の新聞記事において出現する単語のみで G_{target}, G_{Base} を構成する．DBSCAN のパラメータは $\epsilon = 2.0 \times 10^{-4}$，$MinPts = 5$ とする．k-plex 制約は $k = 5$ とする．d は全固有ベクトルの最近傍までの距離を昇順に並べ，最初にあらわれた閾値を参考に決める．m は特に大きすぎる，もしくは小さすぎない限り問題はない．($6 \geq m \geq 3$ 程度が現実的) DBSCAN で得たクラスタ内の単位円グラフの半径 $r = d$ とする．G_{base} における k-plex の補グラフによる疑似独立集合の制約は k=5 とする．G_{target} において極大クリークを列挙し，そのサイズ s の度数分布表を作る．サイズ s 以上の極大クリークの度数と全体の度数との割合がある程度小さくなる s の値を k とする．G_{target} において部分集合内で次数 s の頂点は有意に密とみなし，G_{base} において，部分集合内で次数 s 以上の頂点を破棄する．

4.3 解集合

得られたパターンの最大のサイズは 29，パターンの総数は 289 個その一例として，地方票・党友・某議員 A 氏・健闘などといった 16 の単語群が検出された．2006 年 9 月は某政党の総裁選があり，全国的には当選した議員の圧勝であったが，北海道では他の議員も健闘していたという記事が見られた．北海道では A 氏が党友からの支持を受けて，当選議員には劣るものの北海道の地方票を獲得しており，それについて多くの記事が書かれていた．一方，全国版では，得られたパターン内の単語はいくつか出てきているがほとんど共起していなかった．このように，本構造差異検出手法によって地域により差異のある単語のグループを検出し，その地域で注目されている話題を観測することができる．

4.4 計算時間

本手法において，いくつか設定するパラメータがあるが，そのどれも計算時間には大きく関わらず，単純に関係グラフの大きさの 3 乗に比例する．本手法は計算過程で固有値分解をしており，計算時間のほとんどはそれに費やされる．計算のうち固有値分解は Java から Mathematica を呼び出して行われ，それ以外は Java プログラムにより計算される．本手法の特徴は

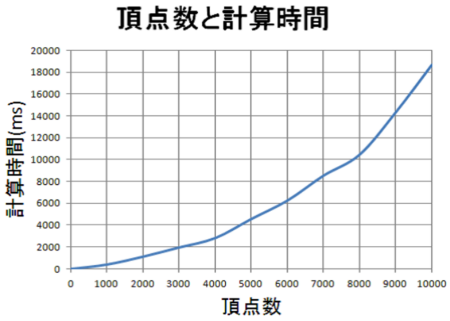


図 7: グラフの頂点数と計算時間

表 1: グラフの密度と計算時間

元のグラフの密度	射影後のグラフの密度	計算時間 (ms)
0.182035	0.016356	2375
0.045933	0.018725	2656

組み合わせ計算が必要な極大クリーク枚挙の計算時間が関係グラフの大きさや密度に対して，大きな影響を受けないことである．G_{target} においてグラフの大きさや密度を変え，本手法の有用性を示す．

図 7 はスケールを 1000 として頂点数 1000〜10000 までの計算時間を表にしたものである．組み合わせの全列挙であるにもかかわらず計算時間の増加が抑えられていることが分かる．

表 1 は Target の密度と計算時間の関係を示した表である．頂点数は約 3000 で Target の密度が高い場合と低い場合で距離空間に射影後の固有ベクトルのグラフの密度と計算時間を比較したものである．元のグラフの密度は約 4 倍の差があるが，固有ベクトルを距離空間に射影し，距離が近い頂点で辺を張ったグラフはの密度は表の下のグラフの方が若干高くなっており，計算時間も若干遅くなっている．

図 8 は Target の頂点数が約 3000 のグラフで Base において補グラフの k-plex 制約を課す際，k の値を変化させたときの計算時間である．k の値を大きくすると組み合わせを考慮しなくてよい頂点が多くなり，計算時間が短縮されていることが分かる．

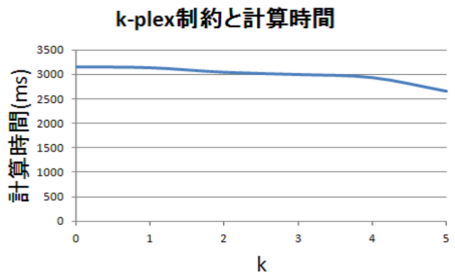


図 8: k-plex 制約の k の値と計算時間

5. まとめと今後の展望

本研究では，二つのグラフの差異を高速に検出し，なおかつ，以前の手法より大きな部分集合を取り出しやすくするために，一方のグラフで正規化グラフラプラシアンをスペクトル解析し，固有値を距離空間に射影した際の距離で単位円グラフを用いてクラスタリングする手法を提案した．

実験により，大きなサイズのグラフでも密度に係わらずクリークを枚挙する計算時間を抑えられていることが確認できた．また，クラスタの大きさも 30 近いものがとれ，大規模なグラフの差異にも適応できる．

構造の差異を検出するために，Ncut や k-plex を用いる本研究は，一方のグラフにおいて頂点集合から外部への辺が少なく内部で多いこと，および，もう一方のグラフにおいて内部での辺が少ないことを制約として用いた．この条件の場合，内部が変わらず疎のままで，外部への辺が極端に減った集合を検出してしまうことがある．この現象は Ncut 制約でクラスタを取り出した後のチェックを Base のみから，Base・Target の両方を見ることで改善が期待できる．

また，単位円グラフからクリークを枚挙しているが，それを k-plex にすることでさらに大きな頂点集合を得られる可能性がある．DBSCAN で検出されるクラスタはデータ点の密度が低い部分をノイズといて切り捨てるので，検出されるクラスタは少なく，クラスタのサイズもそれほど大きくならないため組み合わせの計算にはそれほど時間はかからないと考えられる．

参考文献

[1] B. Furht (ed.), Handbook of Social Network Technologies and Applications, Springer, 2010.

[2] Ulrike von Luxburg, A tutorial on spectral clustering, Statistics and Computing, 17 (4), 2007.

[3] Martin Ester et al.: A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, KDD-96, 1996.

[4] Komusiewicz, C., Hu”ffner, F., Moser, H. and Niedermeier, R.: Isolation Concepts for Efficiently Enumerating Dense Subgraphs, Theoretical Computer Science 410, pp. 3640 - 3654, Elsevier, 2009.

[5] エラウィンディ サラ et al.: クリーク全列挙に基づく構造変化検出アルゴリズム，情報処理学会研究報告，Vol. 2011-MPS-087 No. 32, 2012.

[6] Okubo, Y. et al.: Structural Change Pattern Mining Based on Constrained Maximal k-Plex Search, DS'12, LNAI 7569, pp. 284 - 298, 2012.

[7] Brent N. CLARK and Charles J. COLBOURN, Unit Disk Graphs, December 1990, Pages 165-177, 1990.

[8] Fabian Kuhn, Roger Wattenhofer and Aaron Zollinger, Ad-Hoc Networks Beyond Unit Disk Graphs, DIALM-POMC '03 Proceedings of the 2003 joint workshop on Foundations of mobile computing Pages 69-78, 2003.

[9] Bin Wu and Xin Pei, A Parallel Algorithm for Enumerating All the Maximal k-Plexes, PAKDD 2007 Workshops, LNAI 4819, pp. 476-483, 2007.