Vol.2010-AL-129 No.8
2010/3/5

最大クリーク問題の多項式時間的可解性の新たな結果

中 西 裕 陽[†1]　富 田 悦 次[†1,†2]　若 月 光 夫[†3]

最大クリーク抽出問題（あるいは，それと双対な最大独立節点集合問題）の時間計算量については，Tarjan-Trojanowski(1977) から Robson(2001) や Fomin ら (2006) に至るまで，長年にわたる逐次的進展がなされているが，それらのほとんどはアルゴリズムあるいは時間計算量解析が非常に複雑である．これに対し本稿では，節点数 n, 最小次数 $n-4$ のグラフに対して, 最大時間計算量が $O(n^4)$ なる多項式オーダーの単純な最大クリーク抽出アルゴリズムと単純な解析を提唱する．本稿の手法は，一般の最大クリーク抽出問題の最大時間計算量評価改善の新しい基礎となるものと考えられる．

Another result on polynomial-time solvability for the maximum clique problem

Hiroaki Nakanishi ,[†1] Etsuji Tomita[†1,†2] and Mitsuo Wakatsuki [†3]

Several improvements have been done for the time-complexity of the maximum clique problem, though it is exponential of the number of vertices. In this note, we present a simple branch-and-bound algorithm for the maximum clique problem. It is based on our preceeding algorithm CLIQUES, which is designed for generating all maximal cliques. Then we give simple proof that its polynomial time-complexity is $O(n^4)$ for a graph with n vertices and whose minimum degree is $n-4$. The result could contribute to improve the upper bound of the time-complexity for finding a maximum clique in a general graph.

1. はじめに

無向グラフ中の最大クリークを抽出する問題は理論, 応用の両面で重要な問題であり, 理論と実験の双方から様々な研究がなされている[2)-14)]. 計算量理論の視点から見れば, この問題は自明な計算量が $O(P(n)2^n)$(n はグラフの節点数, $P(n)$ は n の適当な多項式) という解決困難な問題である．この時間計算量はまず Tarjan ら[2)] によって改善され, これに Robson[3)] が続き, 従来の多項式領域での最良結果は $O(2^{0.288n})$[5)] となっていた．一方，Tomita らは極大クリークを全列挙する $O(3^{n/3}) = O(2^{0.528n})$-時間アルゴリズムを発表しているが[6)]，これを最大クリーク 1 個だけと出力を限定することによ り，当然この計算量は大きく軽減できる．これに従い，Shindo-Tomita はアルゴリズム MAXCLIQUE[4)] において，単純な $O(2^{0.3493n})$-時間アルゴリズムを提唱した．

このアルゴリズムは非常に単純であり，理論的オーダ評価では上記 Tarjan らの結果より若干大きくなるものの，実働結果では Tarjan らの結果と比較して非常に高速であった．

筆者らはこれらの結果を受けて, 先述のアルゴリズム MAXCLIQUE を基にした解析過程の見直しを行い, 計算量を改善した一連の結果[11),12),13)] を発表してきている．但しこの結果は膨大な場合分けの上に立った非常に煩雑な計算量解析となっており, 難解であるという面があった．そこで本稿においては, 上記の各結果の解析過程を簡単化し, また解析結果の更なる改善を可能にすべくアルゴリズム MAXCLIQUE を改良した単純な分枝限定アルゴリズムを提唱し, このアルゴリズムが節点数 n, 最小次数 $n-4$ なるグラフに対して, 最大時間計算量 $O(n^4)$ の多項式アルゴリズムとなることを示す．解析の過程においては場合分けを排除して大幅に簡単化することに成功している．

本稿のアルゴリズム，解析手法は，文献[6),14)] を直接的基盤としているので，適宜それらを参照されたい．

2. 諸定義と記法

(1) 本稿で対象とするグラフは, 自己閉路をもたない無向グラフ $G = (V, E)$ である．ここで, V は**節点**の有限集合, E は相異なる 2 節点の非順序対 (v, w)（これを**辺**と呼ぶ) の集合である．節点 v と w は, $(v, w) \in E$ が成り立つとき**隣接している**という．

†1 電気通信大学 先進アルゴリズム研究ステーション
†2 中央大学研究開発機構
†3 電気通信大学 情報通信工学科

Vol.2010-AL-129 No.8
2010/3/5

集合 V に対して, その要素数を $|V|$ で表す．また，集合 V が順序付き集合である時，その先頭要素を $V[1]$ で表す.

(2) $v \in V$ について, $\Gamma(v)$ を $G=(V,E)$ の内で v に隣接する全ての節点の集合とする. 即ち

$\Gamma(v)=\{w \in V \mid (v,w) \in E\}$ ($\not\ni v$).

$|\Gamma(v)|$ を v の次数と呼ぶ.

(3) 節点の部分集合 $W \subseteq V$ について,

$E(W)=\{(v,w) \in W \times W \mid (v,w) \in E\}$

としたとき, $G(W)=(W,E(W))$ を $G=(V,E)$ の W による**誘導部分グラフ**と呼ぶ.

(4) 与えられた $Q \subseteq V$ の誘導部分グラフ $G(Q)$ に対して, 次が成り立つとき, $G(Q)$ は**完全**であるという.

$^{\forall}v,w \in Q$ $(v \neq w)$ に対して $(v,w) \in E$

このとき Q は**クリーク**であるという．またクリークのサイズを $|Q|$ によって定義する.

自分自身を除くグラフ中の異なる任意のクリークの真の部分クリークでないクリークを**極大クリーク**といい, 極大クリークのうちでサイズが最大であるものを**最大クリーク**という.

3. アルゴリズム MAXCLIQUE

アルゴリズム MAXCLIQUE を図 1, 図 2.1 及び図 2.2 に示す．このアルゴリズムは基本アルゴリズム EXPAND に, 以下の 3 つの限定操作

・部分森の同一化

・部分集合森の削減

・近接拡大クリークの抽出

を加えた分枝限定アルゴリズムである.

基本アルゴリズム EXPAND は, 探索の各深さにおいて候補節点集合 $SUBG$ 中の節点と最も多く隣接する節点を選択し, この節点の隣接部分と $SUBG$ との積集合を新たな候補節点集合として EXPAND を行うことで, クリークを抽出していく深さ優先探索である.

MAXCLIQUE による最大クリークの探索過程は深さ優先探索木の集合 (クリーク探索森と呼ぶ) として表現することができる.

限定操作の関係上アルゴリズムには変数 d を用いて再帰の深さを導入している．即ち再帰の一番上＝ 1 段階目を深さ 0 として, 対象となる候補節点集合に関しては, $SUBG^{(0)}$ のように上付き添え字 (0) を付ける．深さ 1 以降も同様とする.

3.1 部分森の同一化

基本アルゴリズムに, 次のような**部分森の同一化**と呼ぶ節点の並べ替え処理を導入する.

(1) 入力節点集合 $SUBG$ 中の最大次数節点 u を先頭に移す.

(2)$SUBG-\{u\}$ を, u に隣接する

$SUBG_u=\Gamma(u) \cap SUBG$ と, 隣接しない

$EXT_u=(SUBG-\{u\})-SUBG_u$ とに分割する.

(3) この順序付き節点集合

$\{u\} \cup EXT_u \cup SUBG_u$

を改めて $SUBG$ とおく.

このような節点の並べ替え処理を行った後に基本アルゴリズムを用いてクリーク探索を行うと, 先頭の u を根とした探索森から u を除いた部分（即ち，u の子節点集合 $SUBG_u$ を根集合とする探索部分森）と, $SUBG$ の最後部の $SUBG_u$ 部分に対する探索木の集合（部分森）とは同一のものとなる．したがって, u を根とする探索木より得られる最大クリークの節点数は, 最後部 $SUBG_u$ 部分の探索から得られる最大クリークの節点数よりも (u の分)1 だけ大きい．そこで最大クリーク抽出においては, 最後部 $SUBG_u$ 部分の探索は行わない.

3.2 部分集合森の削減

MAXCLIQUE による探索においては, 深さ 1 において, 節点集合 $SUBG^{(1)}$ 中の最大次数節点を u_1 として

$SUBG^{(2)}_{u_1}=SUBG^{(1)} \cap \Gamma(u_1)$

および

$EXT^{(1)}=SUBG^{(1)}-\{u_1\}-SUBG^{(2)}_{u_1}$

によって各節点

$u_1, v_1, v_2, ..., v_{|EXT^{(1)}|}$

に対してそれぞれ隣接部分

$SUBG^{(2)}_{u_1}, SUBG^{(2)}_{v_1}, SUBG^{(2)}_{v_2}, ..., SUBG^{(2)}_{v_{|EXT^{(1)}|}}$

が定義される．このとき節点 v_i は $|(SUBG^{(1)}-\{u,v_1,v_2,,..,v_{|EXT_u|-1}\}) \cap \Gamma(v_i)|$ が最大である節点であり

$SUBG^{(2)}_{v_i}:=\Gamma(v_i) \cap SUBG^{(1)}$

である.

これらの隣接部分は上記の順番で探索が行われる．以下深さ 1 を固定して議論を進めるものとし, 各節点集合に関して深さの記述は省略する.

```
procedure MAXCLIQUE(G)
begin
  global Q := ∅;
  global Qmax := ∅;
      EXPAND(V, 0)
end {of MAXCLIQUE}
```

図 1 アルゴリズム MAXCLIQUE

このとき次が成立する.

[命題 1.1] 隣接部分

$SUBG_{v_i}$ $(1 \le i \le |EXT_u|)$

がそれ以前に探索された隣接部分 $SUBG_u$ の部分集合であるならば, $SUBG_{v_i}$ からの探索で得られる極大クリークのサイズは $SUBG_u$ に対する探索により得られる最大クリークのサイズ以下となる.

(証明) 紙数の関係上省略する. □

最大クリーク抽出において [命題 1.1] の条件が成立する場合には探索を行う必要はないので,MAXCLIQUE において $SUBG_{v_i}$ を定義した時は, これが $SUBG_u$ の部分集合でないことを確認し, そうでなければ探索を省く. この限定操作を**部分集合森の削減**と呼ぶ. この操作は EXPAND の任意の深さにおいて実行可能であるが, 後述するもう 1 つの限定操作との関係で実行を深さ 1 の場合だけに限定している.

部分集合森の削減を導入すると, $EXT_u = \{v_1, v_2, ..., v_{|EXT_u|}\}$ の最後尾節点 $v_{|EXT_u|}$ の隣接部分は探索が削減されることがわかる. 即ち以下が成立する.

[命題 1.2]$SUBG_{v_{|EXT_u|}} \subseteq SUBG_u$

(証明) 節点 $v_{|EXT_u|}$ の隣接部分が探索されるのは節点 $u, v_1, v_2, ..., v_{|EXT_u|-1}$ の隣接部分が全て探索された後であるから, これらの節点はすでに探索から除外されている. ここから

$SUBG_{v_{|EXT_u|}} = \subseteq SUBG_u$

である. **(証明終)**

3.3 近接拡大クリークの抽出

命題 1 に示したことから, 部分集合森の削減は, 隣接部分 $SUBG_{v_i}$ $(1 \le i \le |EXT_u|)$ がそれ以前に探索された隣接部分 $SUBG_u$ の部分集合でない場合には探索の効率化を期待できない. しかし下記に示すように, この状況を逆に積極的に利用することで効果を発揮する次のような限定操作を導入することができる.

再帰深さ 1 における各候補節点集合 $SUBG_{u_0}^{(1)}$ および $SUBG_{v_i}^{(1)}$ $(1 \le i \le |EXT_{u_0}^{(1)}|)$ のそれぞれに対する探索において, ある極大クリーク Q が抽出されたとき, 以下を確認する. まずそれぞれの極大クリークに対してその根集合は $SUBG_{u_0}^{(1)}$ および $SUBG_{v_i}^{(1)}$ $(1 \le i \le |EXT_{u_0}^{(1)}|)$ のいずれか 1 つであるから, その集合を $RSUBG$ として深さ 1 において保持しておく.

次に $RSUBG$ 中で $|RSUBG \cap \Gamma(u_1)|$ が最大となる節点を u_1 として, 以下の操作を行う.

(1)$REXT = RSUBG - \{u_1\} - (RSUBG \cap \Gamma(u_1)) = \{v_1, v_2, ..., v_{|REXT|}\}$

として

$U_2 := \{^{\forall}\{s,t\} \subseteq REXT | (s,t) \in E\}$ とする. 即ち U_2 は $REXT$ 中で互いに隣接する任意の 2 節点の組の集合である. 隣接部分 $W_2 = \Gamma(U_2[i][1]) \cap \Gamma(U_2[i][2])$ について,

$Q - \{u_1\} \subseteq W_2$ ならば

$(Q - \{u_1\}) \cup U_2[i]$

はクリークである. 何故ならば, いま

$Q - \{u_1\} \subseteq Q$

はクリークであり, $Q - \{u_1\}$ 中の全ての節点は互いに隣接する 2 節点の集合 $U_2[i]$ の全ての節点に隣接しているからである.

クリーク $(Q - \{u_1\}) \cup U_2[i]$ は Q よりサイズが 1 大きいため, 最大クリーク抽出においては Q を保持する必要はない. そこで Q を新たに

$(Q - \{u_1\}) \cup U_2[i]$

によって書き換える.

(2)U_2 から順に節点の組 $U_2[i]$ $(1 \le i \le |U_2|)$ を取り出し, 集合

$U_3 := \{^{\forall}\{s,t,u\} \subseteq REXT | (s,t), (t,u), (u,s) \in E\}$

を構成する. 即ち U_3 は任意の 2 節点の組が隣接する $REXT$ のサイズ 3 の部分集合を全て

Vol.2010-AL-129 No.8
2010/3/5

procedure EXPAND($SUBG^{(d)}$, d)
begin
 if $SUBG^{(d)} \neq \emptyset$ **then** $u_d :=$ a vertex in $SUBG^{(d)}$ that maximizes $|SUBG^{(d)} \cap \Gamma(u_d)|$;
  if $d = 1$ **then** $v := u_1$ **fi**
  $Q := Q \cup \{u_d\}$;
  $SUBG^{(d+1)}_{u_d} := \Gamma(u_d) \cap SUBG^{(d)}$;
  if $d = 0$ **then** $RSUBG := SUBG^{(1)}_{u_0}$ **fi**
  EXPAND($SUBG^{(d+1)}_{u_d}$, $d+1$)
  $Q := Q - \{u_d\}$;
  if $d = 1$ **then** $RSUBG := RSUBG - \{u_1\}$ **fi**
  $EXT^{(d)} := SUBG^{(d)} - \{u_d\} - SUBG^{(d+1)}_{u_d}$;
  for $i := 1$ **to** $|EXT^{(d)}|$ **do**
   $v_i :=$ a vertex in $EXT^{(d)}$ that maximizes $|SUBG^{(d)} \cap \Gamma(v_i)|$;
   if $d = 1$ **then**
    $U := \Gamma(v_i) \cap EXT^{(1)}_{u_1}$;
    if $|U| \leq 2$ **then** $i := i + 1$ **fi fi**
   $SUBG^{(d+1)}_{v_i} := \Gamma(v_i) \cap SUBG^{(d)}$;
   if $d = 0$ **and** $i = 1$ **then** $RSUBG := SUBG^{(1)}_{v_1}$ **fi**
   if $d = 0$ **and** $i \neq 1$ **then** $RSUBG := \emptyset$ **fi**
   if $d = 1$ **then**
    if $SUBG^{(2)}_{v_i} \subseteq SUBG^{(2)}_{u_1}$ **then** $i := i + 1$ **fi fi**
   $Q := Q \cup \{v_i\}$;
   EXPAND($SUBG^{(d+1)}_{v_i}$, $d+1$);
   $Q := Q - \{v_i\}$;
   $EXT^{(d)} = EXT^{(d)} - \{v_i\}$;
   if $d = 1$ **then** $RSUBG := RSUBG - \{v_i\}$ **fi**
  od
 else $\{i.\ e.\ SUBG^{(d)}_v = \emptyset\}$
  if $|Q| \geq |Qmax|$ **then** $Qmax := Q$
  fi
end {of EXPAND}

図 2 手続き EXPAND

集めたものである. 隣接部分 $W_3 = \Gamma(U_3[i][1]) \cap \Gamma(U_3[i][2]) \cap \Gamma(U_3[i][3])$ について

$Q - \{u_1\} \subseteq W_3$ ならば

$(Q - \{u_1\}) \cup U_3[i]$

は (2) と同様にクリークであり, Q よりサイズが 2 大きい. そこで Q を新たに $(Q - \{u_1\}) \cup U_3[i]$ によって書き換える.

EXPAND によって極大クリークが抽出されたときに実行される上記の処理を, **近接拡大クリークの抽出**と呼ぶ.

この操作によって, $SUBG^{(2)}_{u_1}$ に含まれる全ての極大クリーク Q について

$Q_a \subseteq SUBG^{(2)}_{v_i}$ $(1 \leq i \leq |EXT^{(1)}_{u_1}|)$

かつ

$Q_a - (Q - \{u_1\}) = \{w_1, w_2\}$ $w_1, w_2 \in SUBG^{(2)}_{v_i}$

または

$Q_a - (Q - \{u_1\}) = \{w_1, w_2, w_3\}$ $w_1, w_2, w_3 \in SUBG^{(2)}_{v_i}$

をみたす Q より 1 ないし 2 だけサイズが大きいクリークを全て抽出することができる. いまこのようなクリークを**近接拡大クリーク**とよぶ事にし, 以下のように定義する.

(5) 集合 A, B に対して, 条件

$|B| = |A| + |W|$

のもとに

$A - \{a\} = B - W$

なる $a \in A$ および $W \subseteq B$

が存在するとき, B を A の拡大幅 $|W|$ の**近接拡大集合**という.

クリーク Q に対し, クリーク Q_a が条件

$|Q_a| = |Q| + |Q_w|$

をみたし, かつ

$Q - \{q\} = Q_a - Q_w$

なる $q \in Q$ および $Q_w \subseteq Q_a$ が存在するとき, Q_a を Q の拡大幅 $|W|$ の近接拡大クリーク

という．即ち Q_a はクリークであり, Q の拡大集合である．

部分集合森の削減によって, もし $SUBG^{(2)}_{v_i} \subseteq SUBG^{(2)}_{u_1}$ であれば $SUBG^{(2)}_{v_i}$ についての探索は省略できるのであるが, この限定操作は $SUBG^{(2)}_{v_i}$ が $SUBG^{(2)}_{u_1}$ に含まれない節点を1個でも含む場合には適用できない．しかしながら上記近接拡大クリークの抽出を用いれば, $SUBG^{(2)}_{v_i}$ が $SUBG^{(2)}_{u_1}$ に含まれない節点を2個以下含む場合には探索を省略できることがわかる．この省略は次の命題により保証される．

[命題 2.2]$SUBG^{(2)}_{u_1}$ 中の最大クリークを Q_{u_1} とする．いま条件

$|SUBG^{(2)}_{u_1}| \geq |SUBG^{(2)}_{v_i}|$

かつ

$|SUBG^{(2)}_{u_1} - SUBG^{(2)}_{v_i} \cap SUBG^{(2)}_{u_1}| = W$

が成立するとする．このとき $SUBG^{(2)}_{v_i}$ 中の最大クリーク Q_{v_i} のサイズは Q_{u_1} の拡大幅 $|W|$ の近接拡大クリーク Q_a のサイズ以下である．

(証明) 条件から, $SUBG^{(2)}_{v_i}$ の節点のうち $SUBG^{(2)}_{u_1}$ に含まれない節点の集合を W とすると

$SUBG^{(2)}_{v_i} - W \subseteq SUBG^{(2)}_{u_1}$

である．従って $SUBG^{(2)}_{v_i} - W$ 中の最大クリークを Q'_{v_i} とおけば, [命題 1.1] により

$|Q'_{v_i}| \leq |Q|$

ここで集合 W 中の最大クリークを Q_w とおけば, $Q_w \subseteq W$ であるから

$|Q_{v_i}| \leq |Q'_{v_i} \cup Q_w| = |Q_{v_i}| + |Q_w| \leq |Q| + |W| = |Q_a|$

である．**(証明終)**

$SUBG^{(2)}_{v_i}$ が $SUBG^{(2)}_{u_1}$ に含まれない節点を2個以下しか含まないとき, 上記の近接拡大クリーク抽出を $SUBG^{(2)}_{u_1}$ の任意の極大クリークに対して実行しておけば, $SUBG^{(2)}_{v_i}$ の探索は省略できる．そこで $SUBG^{(2)}_{v_i}$ については, $EXT^{(2)}_{u_1}$ 中の節点が3個以上含まれている場合にのみ探索を実行し, 3個未満の場合にはそれ以上再帰を行わない．

いま隣接部分 $SUBG_u$ 探索終了後 $SUBG_v$ が探索されるとする．このとき $SUBG_v$ に対する探索は一般に $SUBG_v$ が $SUBG_u$ に含まれない節点を多く含むほど困難であることが予想できる．逆に節点数が少ないほど探索は容易であるといえるが, ここに示した近接拡大クリーク抽出はそのような容易な部分問題を排除するための操作であるといえる．

排除されなかった部分問題については, その問題自身は困難であるが, 逆にその問題を処理することで既に探索済みの集合がより広い集合になると言える．探索済みの集合が大きくなることはさきに与えた部分集合森の削減が適用される可能性を増加させることに繋がるので, 近接拡大クリークの抽出は容易な部分問題の排除を行うと同時に, 部分集合森の削減の効果を補強するものである．

アルゴリズム上でこの処理を実現するためには MAXCLIQUE の 3400 - 3500 行間に図3に示す 35 - 3524 行を挿入すればよい．

近接拡大クリークの抽出は再帰深さ1の各集合に対してのみ実行される処理である．この処理は一般に全深さにおいて定義することも可能であるが, 全深さでこの処理を追加した場合, 計算量の増加が指数オーダーとなることが予想される．そのためこの操作は実行を深さ1に限定してある．

4. 最大時間計算量評価

節点集合 V で $|V| = n$, 最大次数 $\Delta \leq n-1$ であるグラフについて, いま一般に再帰深さ d $(0 \leq d \leq n-1)$ とするとき, $SUBG^{(d)}_v \subseteq V$ に対する EXPAND($SUBG^{(d)}_v$, d) での最大時間計算量を $t(|SUBG^{(d)}_v|)$ とすると, 次の関係式が成り立つ．

$$t(|SUBG^{(d)}_v|) \leq t(|SUBG^{(d+1)}_{u_d}|) + \sum_{i=1}^{|EXT^{(d+1)}_{u_d}|} t(|SUBG^{(d+1)}_{v_i}|) + C|SUBG^{(d)}_v|^3 \quad \text{... (T1)}$$

但し, ここでは計算量評価においては一般深さで (T1) 式を用いることはせず, 全体の計算量を決定する深さ0および近接拡大クリークの抽出が適用される深さ1においての (T1) 式の関係を考慮する．即ち (T1) 式において $d = 0$ とすれば,

$$t(n) \leq t(|SUBG^{(1)}_{u_0}|) + \sum_{i=1}^{|EXT^{(1)}_{u_0}|} t(|SUBG^{(1)}_{v_i}|) + Cn^3 \text{... (T1-0)}$$

が成立する．また $d = 1$ とすれば

$$t(|SUBG^{(1)}_{u_0}|) \leq t(|SUBG^{(2)}_{u_1}|) + \sum_{i=1}^{|EXT^{(2)}_{u_1}|} t(|SUBG^{(2)}_{v_i}|) + C(\Delta+1)^3 \text{... (T1-1A)}$$

$$t(|SUBG^{(1)}_{v_i}|) \leq t(|SUBG^{(2)}_{u_1}|) + \sum_{j=1}^{|EXT^{(2)}_{v_j}|} t(|SUBG^{(2)}_{v_j}|) + C(\Delta+1)^3 \quad (1 \leq i \leq |EXT^{(1)}_{u_0}|) \text{... (T1-1B)}$$

がそれぞれ成立する．

これらの式末尾の Cn^3 および $C(\Delta+1)^3$ はこのような問題分割に必要な全ての処理に要する多項式オーダーの計算量の上界であり, ここで定数 $C > 0$ は以下の様に定義する．

いま EXPAND の再帰深さを $d \geq 1$, すでに得られている最大クリーク $Qmax \subseteq V$, 近接拡大クリークの抽出に用いる集合 $RSUBG \subseteq V$ とそれぞれおく．

この条件のもと, EXPAND($SUBG^{(d)}$, d) に対し, 次の再帰深さの候補節点集合となる $SUBG^{(d+1)}_{u_d}$ および $SUBG^{(d+1)}_{v_i}$ $(1 \leq i \leq |EXT^{(d+1)}_{u_d}|)$ を常に空集合で与える処理を

Vol.2010-AL-129 No.8
2010/3/5

```
REXT := RSUBG − {v} − (RSUBG ∩ Γ(v));
if |REXT| ≥ 3 then
    U2 := ∅;
    U3 := ∅;
    for i = 1 to |REXT| do
        for j = i + 1 to |REXT| do
            U := {REXT[i], REXT[j]};
            if (REXT[i], REXT[j]) ∈ E then U2 := U2 ∪ U;
        od
    od
    for i := 1 to |U2| do
        W2 := Γ(U2[i][1]) ∩ Γ(U2[i][2]);
        if Qmax − {v} ⊆ W2then Qmax := (Qmax − {v}) ∪ U2[i]} fi
    for i := 1 to |U2| do
        for j = i + 1 to |U2| do
            if |U2[i] ∪ U2[j]| = 3 then U := U2[i] ∪ U2[j];
        if (U[1], U[2]) ∈ E ∧ (U[2], U[3]) ∈ E ∧ (U[3], U[1]) ∈ E then U3 := U3 ∪ U fi
        od
    od
    for i := 1 to |U3| do
        W3 := Γ(U3[i][1]) ∩ Γ(U3[i][2]) ∩ Γ(U3[i][3]);
        if Qmax − {v} ⊆ W3 then Qmax := (Qmax − {v}) ∪ U3[i]} fi
    od
fi
```

図 3 近接拡大クリークの抽出

EXPAND$_0$ とする．このとき EXPAND$_0$ の時間計算量を考える．

まず最大次数節点 u_d の選択および, 集合 $SUBG_{u_d}^{(d+1)}$ と $EXT_{u_d}^{(d+1)}$ の定義に要する手数が $O(|SUBG_v^{(d)}|^2)$ となる．ここでそのような上界 $C_1|SUBG_v^{(d)}|^2$ を与える定数を C_1 とする．

$SUBG_{u_d}^{(d+1)} = \emptyset$ であるから, $SUBG_{u_d}^{(d+1)}$ に関してアルゴリズムは空集合性の判定を行った後, 近接拡大クリークの抽出を実行する．

まず $d \geq 1$ であるから, この時点ですでに定義されている集合 $RSUBG$ から最大次数節点 v を選択し, 集合 $SUBG_v$ および $REXT$ を構成する． ここまでに要する手数は $O(|SUBG_v^{(d)}|^2)$ となるから, いまそのような上界 $C_2|SUBG_v^{(d)}|^2$ を与える定数を C_2 とする．

各集合定義後アルゴリズムはまず $REXT$ から先頭節点 w_1 を選択し, $REXT$ 中で集合 U_2 および U_3 を構成する．この U_2 および U_3 を用いて拡大幅 1 および 2 の近接拡大クリークの抽出および, クリークの更新があれば更新を行うが, このチェックの手数は $REXT$ の節点 1 個節点につき, 拡大幅 1 ならば

$(Qmax - \{v\}) \subseteq \Gamma(U_2[i][1]) \cap \Gamma(U_2[i][2])$

かどうかの判定, 拡大幅 2 については

$(Qmax - \{v\}) \subseteq \Gamma(U_3[i][1]) \cap \Gamma(U_3[i][2]) \cap \Gamma(U_3[i][3])$

かどうかの判定に要する手数である．これらの操作のうち最も手数を要するのは集合 U_2, U_3 の構成であり, その手数は $O((|REXT|^3))$ であるから, これらの操作は $O(|SUBG_v^{(d)}|^3)$ で可能である．

条件から

$|REXT| = |RSUBG - \{v\} - SUBG_v| < |SUBG_v^{(d)}|$

となるので, $REXT$ 各節点に関する操作に要する手数は $O(|SUBG_v^{(d)}|^3)$ である．そこでいまそのような上界 $C_3|SUBG_v^{(d)}|^3$ を与える定数を C_3 とする．

以上から EXPAND$_0$ の実行に要する時間計算量は

$C_1|SUBG_v^{(d)}|^2 + C_2|SUBG_v^{(d)}| + C_3|SUBG_v^{(d)}|^3 < (C_1 + C_2 + C_3)|SUBG_v^{(d)}|^3$ である．

ここで定数 C を

$C = C_1 + C_2 + C_3$ によって定義すると, EXPAND$_0$ は $O(|SUBG_v^{(d)}|^3)$ の処理である．

上記の定数を $d = 0$ および $d = 1$ の場合に適用すれば, それぞれ $O(n^3)$ および $O(\Delta^3)$ の上界を得る．そこでこの定数 C を (T1), (T1-0), (T1-1A) および (T1-1B) 式の C とする[6]．

また EXPAND の計算量は, その性質上対象とする節点集合のサイズに関して単調である．即ちある集合 $SUBG$ について, $|SUBG| = \Delta$ とするとき,

$t(\Delta) \geq t(\Delta - 1) \geq t(\Delta - 2) \geq ... \geq t(0)$ (T2)

である．

あるグラフが与えられたとき, MAXCLIQUE(G) はそのグラフに対応したクリーク探索森を形成する. すなわち MAXCLIQUE(G) の計算量とはそのようなクリーク探索森を形成するために必要な計算量である.

全体の計算量評価を行うにあたり, まずは本稿において導入した限定操作の効果について評価を行う.

いま (T1-1A), (T1-1B) における親節点集合 $SUBG_{u_1}^{(2)}$ および $SUBG_{v_i}^{(2)}$ 中の最大次数節点の次数について考える.

これらの次数は親節点集合のサイズより必ず 1 以上減少するから, これを $k \geq 1$ を用いて $\Delta - k$ と表すと, (T1-1A) または (T1-1B) 式から分割される部分問題の総数は, 部分森の同一化によって

$\Delta - ((\Delta - k) - 1) = k - 1$

個であるが, 限定操作の導入によりこの部分問題数は少なくとも 3 つ減少させることができ, これにより計算量上界を改善することができる.

即ち, 以下の補題が成立する.

[補題 1] 節点集合が V で, $|V| = n$, 最大次数が $\Delta \geq 0$ なる任意のグラフにおいて, $u_0 \in V$ を最大次数節点とする. $SUBG_{u_0}^{(1)} = V \cap \Gamma(u_0)$ として, (T1-1A) および (T1-1B) 式による問題分割が発生するとき, ある

$1 \leq k \leq |EXT_{u_1}^{(1)}|$ において

$SUBG_{v_k}^{(2)} \subseteq SUBG_{u_1}^{(2)}$

又は

$|SUBG_{v_k}^{(2)} \cap EXT_{u_1}^{(1)}| \leq 2$

なる k が少なくとも 3 つ存在する.

(証明) 以下全て深さ 1 に関する議論であるから, 特に必要のない限り集合表記時 $^{(d)}$ の表示は省略する.

[命題 1.2] により $SUBG_{v_{|EXT_v|}}$ は常に $SUBG_{u_1}$ の部分集合であるから, 題意の $SUBG_{v_k}$ のうち 1 つを $SUBG_{v_{|EXT_v|}}$ としてよい. そこで次に $SUBG_{v_{|EXT_v|}}$ 以外にも少なくとも 2 つ題意を満たす $SUBG_{v_k}$ が存在することを示す.

いま各隣接部分 $SUBG_{v_1}, SUBG_{v_2}, ..., SUBG_{v_{|EXT_{u_1}|-3}}$ の全てに限定操作が適用されなかったとする. このとき EXT_{u_1} から節点 $v_1, v_2, ..., v_{|EXT_{u_1}|-3}$ は既に探索から除外されているのであるから, $SUBG_{v_{|EXT_{u_1}|-2}}$ が含んでいる可能性のある EXT_{u_1} の節点は $v_{|EXT_{u_1}|-1}$ および $v_{|EXT_{u_1}|}$ の 2 個である. したがって $SUBG_{v_{|EXT_{u_1}|-2}}$ に対しては限定操作が適用される. また同様に $SUBG_{v_{|EXT_{u_1}|-1}}$ が含んでいる可能性のある EXT_{u_1} の節点は $v_{|EXT_{u_1}|}$ ただ 1 個であるから $SUBG_{v_{|EXT_{u_1}|-1}}$ に対しても限定操作が適用される.

もし $SUBG_{v_1}, SUBG_{v_2}, ..., SUBG_{v_{|EXT_{u_1}|-3}}$ のうち少なくとも 1 個以上に対して限定操作が適用されると仮定すれば, 同様にして 3 個以上題意をみたす k が存在することを示せる. **(証明終)**

補題 1 のような集合 $SUBG_{v_{|EXT_v|}}$ および $SUBG_{v_j}^{(2)}$ に対しては, 部分集合森の削減が適用され, $t(|SUBG_{v_{|EXT_v|}}|) = t(|SUBG_{v_j}^{(2)}|) = 0$ となる.

主要定理証明に先立ち, まず最大次数 Δ をパラメーターとして用いた EXPAND による各子問題の時間計算量評価結果を以下に与える.

[補題 2] 節点集合が $SUBG$ で, 最大次数が $\Delta \geq 0$ かつ最小次数が $n - 4$ なる任意のグラフにおいて, EXPAND($SUBG_{u_0}^{(1)}$, 1) および EXPAND($SUBG_{v_i}^{(1)}$, 1) の最大時間計算量上界はそれぞれ $t(\Delta)$ によって与えられるが, このとき

$t(\Delta) \leq C(\Delta + 1)^4$

である.

(証明) 以下ではまず (T1-1A) 式を用いて EXPAND($SUBG_{u_0}^{(1)}$, 1) についての題意の成立を示す. 証明は最大次数 Δ に関する数学的帰納法に

よる.

先ず, $\Delta = 0$ の場合, $SUBG_{u_1}^{(2)} = \emptyset, SUBG_{v_i} = \emptyset$ $(1 \leq i \leq |EXT_{u_1}^{(1)}|)$

であるから, 定数 C の定義により

$t(\Delta) \leq C(\Delta + 1)^3 \leq C(\Delta + 1)^4$

である. 従って題意は成立する.

次に, ある $\Delta \geq 0$ 以下の全ての Δ において

$t(|SUBG_{u_1}^{(2)}|) \leq C(\Delta + 1)^4$

が成立すると仮定する. この仮定のもとで, 節点数 n, 最大次数 $\Delta + 1$ および最小次数が $n - 4$ であるグラフについての計算に要する計算量を考える.

この節点の子節点の最大次数は Δ 以下であるから, そのような最大次数子節点 u_1 の次数を $\Delta - k (0 \leq k \leq \Delta)$ とおく. このとき $t(\Delta + 1)$ は, (T2-1) 式により

$t(\Delta + 1) \leq t(\Delta - k) + \sum_{i=1}^{|EXT_{u_1}^{(2)}|} t(|SUBG_{v_i}^{(2)}|) + C(\Delta + 2)^3$

であるが, ここでこのグラフの最小次数は $n - 4$ であるから, $|EXT_{u_1}^{(2)}| \leq 3$ である. 従って

$t(\Delta+1) \leq t(\Delta-k) + \sum_{i=1}^{3} t(|SUBG_{v_i}^{(2)}|) + C(\Delta+2)^3$

ここで補題 1 により, $SUBG_{v_i}^{(2)}$ $(i=1,2,3)$ のうち 3 つが探索されないのであるから

$t(|SUBG_{v_1}|) = t(|SUBG_{v_2}|) = t(|SUBG_{v_3}|) = 0$

としてよい. 即ち

$t(\Delta+1) \leq t(\Delta-k) + C(\Delta+2)^3$

となる. したがって帰納法の仮定により

$t(\Delta+1) \leq C(\Delta-k)^4 + C(\Delta+2)^3$

$< C(\Delta+1)^4 + C((\Delta+1)+1)^3$

$< C((\Delta+1)^4 + 4\cdot(\Delta+1)^3 + 6\cdot(\Delta+1)^2 + 4\cdot(\Delta+1)+1) = C((\Delta+1)+1)^4$

である.

従って, 帰納法による帰結により, 任意の $\Delta \geq 0$ において補題の関係が成立する.

上記は EXPAND($SUBG_{u_0}^{(1)}$, 0) についての題意の成立を示したが, (T1-1B) 式を用いれば EXPAND($SUBG_{v_i}^{(1)}$, 0) についても全く同様にして題意の成立を示すことができる.**(証明終)**

これより, 次が成立する.

[定理] 節点数 n, 最小次数 $n-4$ のグラフにおいて

$t(n) = O(n^4)$

である.

(証明) (T1) 式により, いま

$t(|V|) \leq t(\Delta) + \sum_{i=1}^{2} t(|SUBG_{v_i}^{(1)}|) + Cn^3$

である. ここで [補題 2] を用いれば

$t(|V|) \leq 4C(\Delta+1)^4 + Cn^3$

$\leq 4Cn^4 + Cn^4 = 5Cn^4$ が成立する. 従って, $t(n) = O(n^4)$ である. **(証明終)**

5. むすび

節点数 n, 最小次数 $n-4$ なるグラフに対する最大クリーク抽出問題が, 単純なアルゴリズムによって $O(n^4)$ なる多項式時間で解決できることを示した.

一般の最大クリーク抽出問題の最大時間計算量に対して, 筆者らは従来の評価[2)-5)], を改善する 多項式計算領域における結果を発表してきたが[11),12),13)], 本稿における手法は, その一般結果をより単純明快に証明する新しい基礎ともなるものと考えられる.

謝辞 ご支援・協力をいただいた電気通信大学先進アルゴリズム研究ステーション長 西野哲朗 教授, 高橋治久 教授, 他関係者に感謝いたします. なお, 本研究は科研費基盤研究 (B), (C), および総務省戦略的情報通信研究開発推進制度 (SCOPE) による研究助成を受けている.

References

1) M. R. Garey, D. S. Johnson, “ Computers and Intaractability: A Guide to the Theory of NP-Completeness, ”W. H. Freeman & Co. (1979).
2) R. E. Tarjan, A. E. Trojanowski, “ Findig a maximum independent set, ”SIAM J. on Computing 6, 537-546 (1977).
3) J. M. Robson: “Algorithms for maximum independent sets,” J. of Algorithms, 7, pp.425-440 (1986).
4) M. Shindo, E. Tomita, “ A simple algorithm for finding a maximum clique and its worst-case time complexity, ”Systems and Computing in Japan 21, 1-13 (1990).
5) F. V. Fomin, F. Grandoni, D. Kratsch, “ Measure and conquer: A simple $O(2^{0.288n})$ independent set algorithm, ”Proc. ACM-SIAM Symp. On Discrete Algorithms, 18-25 (2006).
6) E. Tomita, A. Tanaka, H. Takahashi, “ The worst-case time complexity for generating all maximal cliques and computational experiments, ” Theoretical Computer Science 363, 28-42 (2006).
7) E. Tomita, T. Kameda, “ An efficient branch-and-bound algorithm for finding a maximum clique with computational experiments, ”Journal of global Optimization 37, 95-111 (2007).
8) E. Tomita, “ The maximum clique problem and its applications (Invited lecture), ” IPSJ SIG Tech. Rep., 2007-MPS-67, 21-24 (2007).
9) E. Tomita, Y. Sutani, T. Higashi, S. Takahashi, M. Wakatsuki, “ A simple and faster branch-and-bound algorithm for finding a maximum clique, ”WALCOM 2010, LNCS 5942, 191–203 (2010).
10) 中西裕陽, 富田悦次, “ 最大クリークを抽出する単純なアルゴリズムの最大次数 4 のグラフにおける計算量,” 信学技報, COMP2007-18, 1-7 (2007).
11) 中西裕陽, 富田悦次, “ 最大クリークを抽出する計算量 $O(2^{0.24945n})$ の多項式領域アルゴリズム, ” 信学技報, COMP2007-46, 33-40 (2007).
12) 中西裕陽, 富田悦次, “ 最大クリークを抽出する計算量 $O(2^{0.19669n})$ の多項式領域アルゴリズム, ” 情処研報, 2007-AL-115, 17-24 (2007).
13) 中西裕陽, 富田悦次, “ 最大クリークを抽出する計算量 $O(2^{0.19171n})$ の多項式領域アルゴリズム,” 情処研報, 2008-AL-116, 15-22 (2008).
14) 中西裕陽, 富田悦次, “ 最大クリーク問題の多項式時間的可解性の一結果,” 電子情報通信学会論文誌 D (2010 年 4 月出版予定).