数理解析研究所講究録
第1691巻2010年108-114

最大クリーク抽出の単純な最大時間計算量評価と多項式時間的可解性

中西 裕陽[†], 富田 悦次[†,a], 若月 光夫[†,b]

Worst-case time-complexity and polynomial time solvability for the maximum clique problem

† 電気通信大学 先進アルゴリズム研究ステーション
a 中央大学研究開発機構
b 電気通信大学 情報通信工学科
E-mail: {hironaka, tomita, wakatuki}@ice.uec.ac.jp

あらまし. 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 のグラフに対して，時間計算量が $O(2^{0.33331n})$ となる 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．また，最大クリーク問題の多項式時間的可解性についての新たな1結果も与える．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もなるものである．

キーワード. NP 困難, 最大クリーク, 最大独立節点集合, 時間計算量, 最大次数

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，その計算量が $O(2^{0.3334n})$-時間となることを示す．この値は Shindo-Tomita のオーダ評価結果を改善するものであり，かつ場合分け解析を排除して解析過程を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．

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，その先頭から i 番目の要素を $V[i]$ 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，

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

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．

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

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に示す. このアルゴリズムは基本となる処理である深さ優先探索EXPANDに, 下記の限定操作

- 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. このうち部分森の同一化については[4]に詳しいので, 適宜これを参照されたい.

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.

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

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

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

$|(SUBG^{(1)} - \{u_1, v_1, v_2, , .., v_{i-1}\}) \cap \Gamma(v_i)|$ 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.

このとき次が成立する.

[命題1] 隣接部分 $SUBG_{v_i}$ $(1 \leq i \leq |EXT^{(1)}_{u_1}|)$ がそれ以前に探索された隣接部分

$SUBG^{(2)}_{u_1}$

のいずれかの部分集合であるならば, $SUBG^{(2)}_{v_i}$ からの探索で得られる極大クリークのサイズは,

$SUBG^{(2)}_{u_1}$

の探索により得られる最大クリークのサイズ以下となる.

(証明略)

最大クリーク抽出において[命題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 $SUBG^{(2)}_{v_i}$ を定義した時は, これが $SUBG^{(2)}_{u_1}$ 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

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

の最後尾節点 $v_{|EXT^{(1)}_{u_1}|}$ 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2] $SUBG^{(2)}_{v_{|EXT^{(1)}_{u_1}|}} \subseteq SUBG^{(2)}_{u_1}$ **(証明略)**

3.2 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

$SUBG^{(2)}_{v_i}$ が $SUBG^{(2)}_{u_1}$ 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以下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 $SUBG^{(1)}_{u_0}$ および $SUBG^{(1)}_{v_i}$ $(1 \leq i \leq |EXT^{(1)}_{u_0}|)$ 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 $SUBG^{(1)}_{u_0}$ および $SUBG^{(1)}_{v_i}$ $(1 \leq i \leq |EXT^{(1)}_{u_0}|)$ 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 $|RSUBG \cap \Gamma(v)|$ が最大となる節点を v として, 以下の操作を行う.

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

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.

(2) $F = U_2[i]$ $(1 \leq i \leq |U_2|)$ とする. 即ち F は U_2 の i 番目の要素であり(定義(1))2要素からなる集合である. 隣接部分 $W_2 = \Gamma(F[1]) \cap \Gamma(F[2])$ を考え,

$Q - \{v\} \subseteq W_2$ ならば

$(Q - \{v\}) \cup U_2[i]$ はクリークである. 何故ならば, いま

$Q - \{v\} \subseteq Q$

はクリークであり, $Q - \{v\}$ 中の全ての節点は互いに

procedure MAXCLIQUE(G)
begin
  global $Q := \emptyset$;
  global $Qmax := \emptyset$;
    EXPAND$(V, 0)$
end {of MAXCLIQUE}
  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**
    $EXT^{(d)} := SUBG^{(d)} - \{u_d\} - SUBG^{(d)}_{u_d}$;
    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**
        $v := v_i$;
        $U_1 := EXT^{(1)}_{u_1} \cap \Gamma(v_i)$;
        if $|U_1| \leq 1$ **then** $i := i + 1$ **fi**
       $SUBG^{(d+1)}_{v_i} := \Gamma(v_i) \cap SUBG^{(d)}$;
        if $d = 0$ **then** $RSUBG := SUBG^{(1)}_{v_i}$ **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 1 アルゴリズム MAXCLIQUE

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

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

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, $SUBG^{(2)}_v$ に含まれる全ての極大クリーク Q について

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

かつ

$Q_a - (Q - \{v\}) = \{w_1, w_2\}$ $w_1, w_2 \in REXT$

をみたす Q より 1 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.

(6) クリーク Q に対し, クリーク Q_a が条件

$|Q_a| = |Q| + 1$

をみたし, かつ

$Q - \{q\} = Q_a - \{q_1, q_2\}$

なる $q \in Q$ および $q_1, q_2 \in Q_a$ が存在するとき, Q_a を Q の近接拡大クリークという.

部分集合森の削減によって, もし $SUBG_w^{(2)} \subseteq SUBG_v^{(2)}$ であれば $SUBG_w^{(2)}$ 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 $SUBG_w^{(2)}$ が $SUBG_v^{(2)}$ に含まれない節点を1個でも含む場合, 即ち

$|SUBG_w^{(2)} \cap SUBG_v^{(2)}| < |SUBG_v^{(2)}|$

なるときには適用できない. しかしながら上記近接拡大クリークの抽出を用いれば, $SUBG_w^{(2)}$ が $SUBG_v^{(2)}$ に含まれない節点をただ1個含む場合にも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 3] $SUBG_v^{(2)}$ 中の最大クリークを Q_v とする.

いま $|SUBG_v^{(2)}| \geq |SUBG_w^{(2)}|$

かつ $|SUBG_w^{(2)} \cap SUBG_v^{(2)}| = |SUBG_v^{(2)}| - 1$

であるとき, $SUBG_w^{(2)}$ 中の最大クリーク Q_{w_1} のサイズは Q_v 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 $SUBG_w^{(2)}$ 中には $SUBG_v^{(2)}$ に含まれない節点がただ1個存在するので, その節点を u とすると

$SUBG_w^{(2)} - \{u\} \subseteq SUBG_v^{(2)}$

である. 従ってここで $SUBG_w^{(2)} - \{u\}$ 中の最大クリークを Q'_w とおけば, [命題 1] により

$|Q'_w| \leq |Q|$

もし節点 u が Q'_w 中の節点に隣接しているならば, $Q'_w \cup \{u\} = Q_w$ であり, そうでないならば $Q'_w = Q_w$ である. よっていずれの場合にも

$|Q_w| \leq |Q| + 1 = |Q_a|$ である. **(証明終)**

近接拡大クリークの抽出を行うと, $SUBG_{v_i}^{(2)}$ が $EXT_{u_1}^{(2)}$ 中の節点を2個以上含まない場合には探索を省略できる.

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 $|SUBG_u \cap SUBG_v|$ が小さい, すなわち $SUBG_v$ 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 $|SUBG_u \cap SUBG_v|$ が大き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.

アルゴリズム上でこの処理を実現するためには MAX-CLIQUE の 3400 - 3500 行間に図2に示す 3401 - 3412 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, $SUBG_v^{(d)} \subseteq V$ に対する EXPAND$(SUBG_v^{(d)}, d)$ での最大時間計算量を $t(|SUBG_v^{(d)}|)$ とすると, 次の関係式が成り立つ.

$$
\begin{aligned}
&t(|SUBG_v^{(d)}|) \\
&\leq t(|SUBG_{u_d}^{(d+1)}|) + \textstyle\sum_{i=1}^{|EXT_{u_d}^{(d+1)}|} t(|SUBG_{v_i}^{(d+1)}|) \\
&\qquad + C(max\{\Delta, |SUBG_v^{(d)}|\})^2 \quad \text{(T1)}
\end{aligned}
$$

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,

$$
\begin{aligned}
t(n) &\leq t(|SUBG_{u_0}^{(1)}|) \\
&\quad + \textstyle\sum_{i=1}^{|EXT_{u_0}^{(1)}|} t(|SUBG_{v_i}^{(1)}|) + Cn^2 \quad \text{(T1-0)}
\end{aligned}
$$

が成立する. また $d = 1$ とすれば

$$
\begin{aligned}
&t(|SUBG_{u_0}^{(1)}|) \leq t(|SUBG_{u_1}^{(2)}|) \\
&\quad + \textstyle\sum_{i=1}^{|EXT_{u_1}^{(2)}|} t(|SUBG_{v_i}^{(2)}|) + C(\Delta+1)^2 \quad \text{(T1-1A)} \\
&t(|SUBG_{v_i}^{(1)}|) \leq t(|SUBG_{u_1}^{(2)}|) \\
&\quad + \textstyle\sum_{j=1}^{|EXT_{v_j}^{(2)}|} t(|SUBG_{v_j}^{(2)}|) + C(\Delta+1)^2 \\
&\qquad\qquad (1 \leq i \leq |EXT_{u_0}^{(1)}|) \quad \text{(T1-1B)}
\end{aligned}
$$

がそれぞれ成立する.

これらの式末尾の Cn^2 および $C(\Delta+1)^2$ 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 $SUBG_{u_d}^{(d+1)}$ および $SUBG_{v_i}^{(d+1)}$ $(1 \leq i \leq |EXT_{u_d}^{(d+1)}|)$ を常に空集合で与える処理を 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.

```
REXT := RSUBG − {v} − (RSUBG ∩ Γ(v));
if |REXT| ≥ 3 then
   U2 := ∅;
   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 fi od od
   for i := 1 to |U2| do
      F := U2[i];
      W1 := Γ(F[1]) ∩ Γ(F[2]);
      if Qmax − {v} ⊆ W1 then Qmax := (Qmax − {v}) ∪ U2[i]} fi od
fi
```

図 2 近接拡大クリークの抽出

$SUBG_{u_d}^{(d+1)} = \emptyset$ 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 w_1 に隣接する節点の集合 U を構成する. その後 U から順に節点を取り出しクリークのチェックおよび, 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

$(Qmax - \{v\}) \cup \{U_2[i]\}$

がクリークかどうかの判定, およびクリーク書き換えに要する計算量であるから $O(|SUBG_v^{(d)}|)$ で可能である. 条件から

$$|REXT| = |RSUBG - \{v\} - SUBG_v| < |SUBG_{u_1}^{(2)}| \leq \Delta$$

となるので, $REXT$ 各節点に関する操作に要する手数は $O(\Delta^2)$ である. そこでいまそのような上界 $C_3\Delta^2$ 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

$$C_1|SUBG_v^{(d)}|^2 + C_2|SUBG_v^{(d)}| + C_3\Delta^2 < (C_1 + C_2 + C_3)(max\{\Delta, |SUBG_v^{(d)}|\})^2$$

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 $O((max\{\Delta, |SUBG_v^{(d)}|\})^2)$ 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 $O(n^2)$ および $O(\Delta^2)$ 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. 即ち次が成立する.

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

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 2 つ減少させることができる.

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

$1 \leq k \leq |EXT_v^{(1)}|$ $(v \in \{u_1, v_1, v_2, \ldots, v_{|EXT_{u_1}^{(1)}|}\})$

において, $|EXT_v^{(1)}| \geq 2$ ならば

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

なる $SUBG_{v_j}^{(2)}$ $(1 \leq j \leq |EXT_v^{(1)}|)$ が少なくとも 2 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 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1つ題意を満たす $SUBG_{v_j}$ が存在することを示す.

いま $SUBG_{v_1}, SUBG_{v_2}, ..., SUBG_{v_{|EXT_{u_1}|-2}}$ のうち1個以上に対して限定操作が適用されるならば, 題意は成立する. そこで
$SUBG_{v_1}, SUBG_{v_2}, ..., SUBG_{|EXT_{u_1}|-2}$ には全て限定操作が適用されないとする.

このとき隣接部分 $SUBG_{v_{|EXT_{u_1}|-1}}$ を考える. もし $SUBG_{v_{|EXT_{u_1}|-1}}$ が EXT_{u_1} 中の節点を1個も含まないならば

$$SUBG_{v_{|EXT_{u_1}|-1}} \subseteq SUBG_{u_1}$$

であるから, 題意は成立する. また $SUBG_{v_{|EXT_{u_1}|-1}}$ が EXT_{u_1} 中の節点を含むとしても, この時点で節点 $v_1, v_2, ..., v_{|EXT_{u_1}|-2}$ は探索から除外されているから,

$$|EXT_{u_1} - \{v_1, v_2, ..., v_{|EXT_{u_1}|-1}\}| = |\{v_{|EXT_{u_1}|}\}| = 1$$

即ち $SUBG_{v_{|EXT_{u_1}|-1}}$ が含むことのできる節点はただ1個である. ここで更に

$$|SUBG_{v_{|EXT_{u_1}|-1}}| \leq |SUBG_{u_1}|$$

であるから

$$SUBG_{v_{|EXT_{u_1}|-1}} \subseteq SUBG_{u_1} \cup \{v_{|EXT_{u_1}|}\}$$

である. つまり $SUBG_{v_{|EXT_{u_1}|-1}}$ は $SUBG_{u_1}$ の近接拡大集合である. 従って [命題3] により, $SUBG_{v_{|EXT_{u_1}|-1}}$ 中の最大クリークは $SUBG_{u_1}$ に対する近接拡大クリークの抽出により既に抽出されているから, $SUBG_{v_{|EXT_{u_1}|-1}}$ に対しては限定操作が適用され, 探索が省略される.

以上から題意の $SUBG_{v_j}$ の存在が示せた. **(証明終)**

補題1のような集合 $SUBG_{v_{|EXT_v|}}$ および $SUBG^{(2)}_{v_j}$ に対しては, 部分集合森の削減が適用され,
$t(|SUBG_{v_{|EXT_v|}}|) = t(|SUBG^{(2)}_{v_j}|) = 0$ 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 $\Delta \geq 0$ なる任意のグラフにおいて, EXPAND($SUBG^{(1)}_{u_0}$, 1) および EXPAND($SUBG^{(1)}_{v_i}$, 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, このとき定数 $C' = 250C$ とすると

$$t(\Delta) \leq C' 2^{0.33338\Delta} (\Delta + 1)^2$$

である.

(証明) 以下ではまず (T1-1A) 式を用いて EXPAND($SUBG^{(1)}_{u_0}$, 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, $SUBG^{(2)}_{u_1} = \emptyset, SUBG_{v_i} = \emptyset$ $(1 \leq i \leq |EXT^{(1)}_{u_1}|)$
であるから, 定数 C の定義により

$$t(\Delta) \leq C(\Delta + 1)^2$$
$$< C' \cdot 1 \cdot (\Delta + 1)^2 = C' \cdot 2^{0.33338 \cdot 0} \cdot (\Delta + 1)^2$$

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

$$t(|SUBG^{(2)}_{u_1}|) \leq C' 2^{0.33338\Delta} (\Delta + 1)^2$$

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 $\Delta + 1$ 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

$$t(\Delta + 1) \leq t(\Delta - k) + \sum_{i=1}^{|EXT^{(2)}_{u_1}|-1} t(|SUBG^{(2)}_{v_i}|) + C(\Delta + 2)^2$$

ただし補題1により, $|EXT^{(2)}_{u_1}| \geq 2$, 即ち $k \geq 2$ ならば $SUBG^{(2)}_{v_i}$ $(1 \leq i \leq |EXT^{(2)}_{u_1}| - 1)$ のうち少なくとも1つ, および $SUBG_{v_{|EXT_{u_1}|}}$ は探索されないから, ある $1 \leq j \leq |EXT^{(2)}_{u_1}| - 1$ において $t(|SUBG_{v_j}|) = 0$ としてよい. 即ち

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

である.

節点 u_1 は $SUBG^{(1)}_{u_0}$ 中最大の次数の節点であるから, $v_1, v_2, ..., v_{|EXT^{(1)}_{u_1}|-1}$ の $SUBG^{(1)}_{u_0}$ における次数は u_1 以下である. ここからいま $|SUBG^{(2)}_{v_i}|$ $(1 \leq i \leq |EXT^{(2)}_{u_1}| - 1)$ のサイズを $0 \leq k_i \leq \Delta - k$ を用いてそれぞれ

$$\Delta - k - k_1, \Delta - k - k_2, ..., \Delta - k - k_{|EXT^{(2)}_{u_1}|-1}$$

と表すと

$$\begin{aligned} t(\Delta + 1) &\leq t(\Delta - k) + t(\Delta - k - k_1) \\ &+ ... + t(\Delta - k - k_{j-1}) + t(\Delta - k - k_{j+1}) \\ &+ ... + t(\Delta - k - k_{|EXT^{(2)}_{u_1}|-1}) + C(\Delta + 2)^2 \end{aligned}$$

となる. (T2) 式を用いると

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

であるから, 帰納法の仮定により

$$\begin{aligned} t(\Delta + 1) &\leq ((\Delta + 1) - (\Delta - k) - 1 - 1) \cdot C' 2^{0.33338(\Delta - k)} (\Delta + 2)^2 + C(\Delta + 2)^2 \\ &= (k - 1) C' 2^{0.33338(\Delta - k)} (\Delta + 2)^2 + C(\Delta + 2)^2 \\ &= C' 2^{0.33338\Delta} \left(\frac{k-1}{2^k} + \frac{1}{250 \cdot 2^{0.33338\Delta}}\right) (\Delta + 2)^2 \\ &\leq C' 2^{0.33338\Delta} \left(\frac{k-1}{2^{0.33338k}} + 0.001\right) (\Delta + 2)^2. \end{aligned}$$

$k \geq 0$ において $\frac{k-1}{2^{0.33338k}} < 1.258$ であるから

$$\begin{aligned} t(\Delta + 1) &\leq C' 2^{0.33338\Delta} (1.258 + 0.001) (\Delta + 2)^2 \\ &= C' 2^{0.33338\Delta} \cdot 1.259 \cdot (\Delta + 2)^2 \\ &< C' 2^{0.33338\Delta} \cdot 2^{0.33338} \cdot (\Delta + 2)^2 \end{aligned}$$

$< C'2^{0.33338(\Delta+1)}((\Delta+1)+1)^2$
である．

上記は $|EXT_{u_1}^{(2)}| \geq 2$ なる場合を考えたが，もし $|EXT_{u_1}^{(2)}| \leq 1$ であるならば，u_0 の子節点のうち探索されるのは u_1 のみであるから

$$\begin{aligned} t(\Delta+1) &\leq t(|SUBG_{u_1}^{(2)}|) + C(\Delta+2)^2 \\ &\leq C'2^{0.33338\Delta} + C(\Delta+2)^2 \\ &\leq C'2^{0.33338\Delta}(1+\tfrac{1}{250\cdot 2^{0.33338\Delta}})(\Delta+2)^2 \\ &\leq C'2^{0.33338(\Delta+1)}(\Delta+2)^2 \end{aligned}$$

であり，やはり題意は成立する．

以上と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 のグラフにおいて $t(n) = O(2^{0.3334n})$ である．

(証明) (T1) 式に対して [補題 2] を用いれば

$$\begin{aligned} t(|V|) &\leq (n-\Delta-1)C'2^{0.33338\Delta}(\Delta+1)^2 + Cn^2 \\ &\leq n\cdot C'2^{0.33338n}n^2 + Cn^2 \end{aligned}$$

いま定数 $C'' = 2.5\cdot 10^9$ とすると，任意の n において $n^2 < C''2^{0.00001n}$ が成立するから

$$\begin{aligned} t(n) &< n\cdot C'C''2^{0.33338n}\cdot 2^{0.00001n} + Cn^2 \\ &= C'C''n\cdot 2^{0.0.33339n} + Cn^2 \\ &\leq C'C''n^2\cdot 2^{0.0.33339n} + Cn^2 \\ &\leq (1+\tfrac{C}{C'C''})C'C''2^{0.0.33339n}n^2 \\ &< (1+\tfrac{C}{C'C''})C'C''\cdot C''2^{0.0.33339n}\cdot 2^{0.00001n} \\ &= (1+\tfrac{1}{C'C''})C''2^{0.3334n} \end{aligned}$$

従って，$t(n) = O(2^{0.3334n})$ である． **(証明終)**

また補題 2 からは次の系も導かれる．

[系] あるグラフの最大次数 Δ について，Δ が条件

$\Delta \leq 2.9993d\lg n \ (d \leq 1)$

をみたすとき，$t(n)$ は

$t(n) = O(n^{1+d})$ なる多項式オーダーとなる．

一般に節点数 n のグラフの操作の最大時間計算量として $O(n^2)$ を要することから，系において $d = 1$ とした場合の計算量 $O(n^2)$ は最大クリーク抽出の最適な時間計算量オーダであると考えられる．

5 むすび

一般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 $O(2^{0.3334n})$-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], を大幅に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．

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．

参考文献

[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 $O(2^{0.24915n})$ 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).