Vol.2018-AL-166 No.9
2018/1/29

指定された頂点被覆を根とするクリークマイナー問題

大塚 広夢[1,†1,a)]　北村 啓介[1,†1,b)]　玉木 久夫[1,†1,c)]

概要：指定された頂点被覆を根とするクリークマイナー問題とは，グラフ G とその頂点被覆 X が与えられたとき，G 上に X を根とするクリークマイナーが存在するかを判定する問題である．この問題のクリークマイナー判定は，Bodlaender や Koster によって導入された木幅に対し安全なセパレータの十分条件を発見的に判定するために用いられる．本稿では，この問題の NP 完全性を証明し，この問題に対する固定パラメータアルゴリズムの設計及び実験的評価を行う．

キーワード：半 2 部グラフ，クリークマイナー，カーネル，NP 完全

1. はじめに

グラフを扱った多くの問題は NP 困難であり多項式時間で一般に解くことは難しいが，例えば，グラフの木分解を用いた，グラフの木幅をパラメータとした固定パラメータアルゴリズムの設計が行え，これがより高速な解法として用いられる．

一方で，グラフの最適な木分解を求めることもまた NP 困難な問題であり，一般には多項式時間では求めることはできない．木幅計算における，概念の 1 つとして木幅に対して安全なセパレータ [5] というものがあり，この安全なセパレータが分かればそこでグラフを分割して考えることができ，木幅計算の問題を小さくする上で重要な概念である．

セパレータが安全であるかの十分条件の一つに，セパレータによってできる各連結成分に対しセパレータを根とするクリークマイナーが存在するというものがある．

しかし同様に，グラフが大きい場合のセパレータの安全性の判定は，一般に，短時間で解くことは難しい．解決策として，連結成分の辺の縮約によって，本稿で扱う問題に帰着する方法がある．[1]

本稿では，問題の計算の困難性を示した上で，それに対するアルゴリズムを提示する．

2. 準備

グラフ G，その頂点集合 $V(G)$，辺集合 $E(G)$ とする．G の辺 $e = \{u, v\}$ とする．G における次の操作を v を u に**縮約する**という：u と v を削除し，u と v の隣接頂点を継承した新たな頂点に置き換え，その頂点を u とする．頂点集合 $X \subseteq V(G)$ とする．G の辺 e について，$u \in X$ かつ $v \notin X$ であるとき，v を u に縮約することを，e を X へ**縮約する**という．頂点集合 X が G の**頂点被覆**であるとは，全ての辺 $\{u, v\} \in E(G)$ について，$u \in X$ または $v \in X$ であるときをいう．X 上のグラフ H が G の X を**根とするマイナー**であるとは，G に対する以下の有限回の操作によって H が得られるときをいう．

(1) 辺 $\{u, v\}$ を選び，u を v に，または，v を u に縮約する．
(2) 頂点とその隣接辺の削除．
(3) 辺の削除．

特に，H が完全グラフであるとき，**クリークマイナー**という．

本稿で扱う問題を次のように定義する．

指定された頂点被覆を根とするクリークマイナー問題
入力： グラフ G とその頂点被覆 $X \subseteq V(G)$.
問い： G の X を根とするクリークマイナーは存在するか？

また，問題を考える上で，以下の半二部グラフを用いる．

定義 2.1 **半二部グラフ**とは，グラフ $G = (X, Y, E)$ である．ここで，E は G の辺集合，X と Y は G の頂点集合の 2 分割であり，X は G の頂点被覆である．

半二部グラフ $G = (X, Y, E)$ とする．G の辺 $\{u, v\} \in E(G)$ が**不在辺**であるとは，$u, v \in X$ かつ $\{u, v\} \notin E(G)$ であるときをいう．

グラフ G とその頂点被覆 $X \subseteq V(G)$ が与えられたとき，

1 情報処理学会
IPSJ, Chiyoda, Tokyo 101–0062, Japan
†1 現在，現在，明治大学
a) ohtsuka_yume@cs.meiji.ac.jp
b) k_kitamura@cs.meiji.ac.jp
c) tamaki@cs.meiji.ac.jp

© 2018 Information Processing Society of Japan

G は半二部グラフ $(X, V(G) \setminus X, E(G))$ である．このとき，以下は明らかに成立する．

補題 2.2 グラフ G，その頂点被覆 X とする．このとき，G 上の X を根とするクリークマイナーが存在するならば，かつそのときに限り，半二部グラフ $(X, V(G) \setminus X, E(G))$ において，X と $V(G) \setminus X$ 間に張られるいくつかの辺を X へ縮約することによって X 上の完全グラフが得られる．

よって，以降では，特に断りがない限り，G とその頂点被覆 X が与えられたときに，G の X を根とするクリークマイナーの存在を考えるときは，半二部グラフ $G = (X, Y, E)$ において，X と Y の間に張られるいくつかの辺の X への縮約によって X 上の完全グラフが得られるかを考えることとする．

問題の計算の困難性を示すために，以下で定義する 3SAT を用いる．

3SAT
入力： 3CNF 論理式 ϕ.
問い： ϕ は充足可能か？

ここで，節におけるリテラルの数がちょうど 3 である連言標準形の論理式のことを 3CNF 論理式という．

また，以下の木分解を用いる．

定義 2.3 G の**木分解** $(T, \{X_t\}_{t \in V(T)})$ とは，木 T と T の各ノード $t \in V(T)$ によって定義される頂点集合 $X_t \in V(G)$ の族で，以下の条件を満たすものである．

(1) $\bigcup_{t \in V(T)} X_t = V(G)$,
(2) 任意の辺 $\{v, w\} \in E(G)$ に対して，$v, w \in X_t$ を満たす $t \in V(T)$ が存在し，
(3) 任意の頂点 $v \in V(G)$ に対して，ノードの集合 $\{t \mid v \in X_t\}$ は T において連結である．

ここで，族 $\{X_t\}_{t \in V(T)}$ の各要素を**バッグ**と呼ぶ．木分解 T の**幅**とは，全ての T のノード $t \in V(T)$ における X_t の大きさの最大値から 1 減じた値であり，G の**木幅** $\mathrm{tw}(G)$ とは，G の全ての木分解の幅の最小値である．

3. NP 完全性の証明

指定された頂点被覆を根とするクリークマイナー問題の NP 完全性を示す．この問題は，自明に NP に属するので，3SAT からの帰着を示し，この問題の NP 困難性を証明する．

定理 3.1 ([6]) 3SAT は NP 完全である．

ただし，3SAT について，ある論理変数とその否定を表すリテラルが，同一の節に含まれることはないとする．また，この仮定は，NP 困難性の証明において，一般性を損なわない．

3CNF 論理式 ϕ が与えられたとき，半二部グラフ $G = (X, Y, E)$ を定義する．

3.1 3SAT からの帰着

以降の帰着で定義される半 2 部グラフ $G = (X, Y, E)$ では，X に属する頂点間の任意の辺は，以降で定義する不在辺以外は全て E に属すものととする．

帰着の概略は次の通りである．論理式 ϕ における変数の割当が頂点の縮約に対応するように不在辺を定義し，各節が部分グラフに対応するように，グラフ全体を構成する．各節に対応する部分グラフでは，連言標準形の論理和に対応する工夫を加え，縮約の結果としてグラフ全体で X を根とするクリークマイナーが存在することの必要十分条件が，縮約の結果として個々の節に対応する部分グラフで，X に属する頂点で構成される頂点集合を根とするクリークマイナーが存在することであるように全体のグラフを構成する．

3.2 変数割当の表現

3SAT における，変数および変数の割当を半 2 部グラフ上で表現するために，論理式 ϕ の各変数および各リテラルについて，次のように半 2 部グラフ上の頂点と辺を定義する．

(1) 論理式 ϕ における各変数 x_i について，変数頂点 $x_i, \bar{x}_i \in X$ を定義する．
(2) 論理式 ϕ における各リテラル l_i について，リテラル頂点 $l_i \in X$ を定義する．
(3) 任意の変数頂点 x_i と，その否定に当たる変数頂点 $\bar{x}_i$ について，不在辺 $\{x_i, \bar{x}_i\}$ を定義する．
(4) 任意のリテラル頂点 l_i と，論理的に等価な変数頂点 x_j について，不在辺 $\{l_i, x_j\}$ を定義する．
(5) 任意の変数頂点 x_i と，その否定に当たる変数頂点 $\bar{x}_i$ の組について，真理値割当頂点 $T_{x_i} \in Y$ を定義する．
(6) 任意の真理値割当頂点 T_{x_i} と対応する変数頂点 $x_i, \bar{x}_i$ を辺 $\{T_{x_i}, x_i\}, \{T_{x_i}, \bar{x}_i\}$ で結ぶ．
(7) 任意のリテラル頂点 l_i について，ある変数頂点 x_j との間に不在辺 $\{l_i, x_j\}$ が定義されているとき，真理値割当頂点 T_{x_j} を辺 $\{T_{x_j}, l_i\}$ で結ぶ．

この変数の表現で定義した頂点と辺を図 1 に示す．ただし，X の頂点は着色されていない頂点であり，着色されている頂点は Y の頂点である．また，以降の図についても同様である．

この変数の表現では，X を根とするクリークマイナーが存在するとき，不在辺の本数を減らすために，任意の真理値割当頂点 $T_{x_i} \in Y$ が $x_i, \bar{x}_i \in X$ のいずれかに縮約されることを期待する．この $T_{x_i} \in Y$ の $x_i, \bar{x}_i \in X$ への縮約が，元の論理式 ϕ において，$x_i = true, \bar{x}_i = true (x_i = false)$ とする変数の割当に対応する．

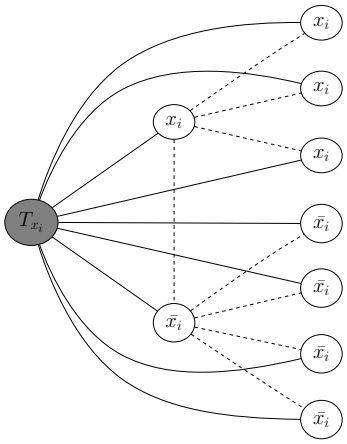

図 1 変数の表現

3.3 節の表現

3SAT における節および，論理式の連言標準形を表現するために，既に定義した変数頂点，リテラル頂点，割当頂点とは別に，論理式 ϕ の各節について，次のように新たな割当頂点を定義する．

(1) 論理式 ϕ の各節 c_i について，離接割当頂点 $o^0_{c_i}, o^1_{c_i} \in Y$ を定義する．

(2) 論理式 ϕ の各節の 3 つのリテラル $l^0_{c_i}, l^1_{c_i}, l^2_{c_i}$ を表現するリテラル頂点と論理的に等価な変数頂点 $B^0_{c_i}, B^1_{c_i}, B^2_{c_i}$，及び，離接割当頂点 $o^0_{c_i}, o^1_{c_i}$ を，辺 $\{\{o^n_{c_i}, l^m_{c_i}\} \mid n = 0,1$ かつ $m = 0,1,2\}$，$\{\{o^n_{c_i}, B^m_{c_i}\} \mid n = 0,1$ かつ $m = 0,1,2\}$ で結ぶ．

既述の変数頂点，リテラル頂点および真理値割当頂点と合わせて，定義した離接割当頂点を図 2 に示す．

この節の表現では，X を根とするクリークマイナーが存在するとき，各節に対応する G の部分グラフについて，少なくとも 1 つ以上の変数頂点及びリテラル頂点が真理値割当頂点の縮約先となり，残りの 2 つ以下の変数頂点及びリテラル頂点が離接割当頂点の縮約先となることを期待し，これが，連言標準形における節の論理和を表現する．

また，それぞれの節に対応する G の部分グラフにおいて、異なる G の部分グラフのリテラル頂点間に不在辺を定義しないことによって，個々の G の部分グラフにおいて，X に属する頂点で構成される頂点集合を根とするクリークマイナーが存在することが，半 2 部グラフ全体において，X を根とするクリークマイナーが存在することの必要十分条件とする意図があり，実際にこれが成立することを以降で示す．

以上で示した帰着の方法で，論理式 $\phi = (x_1 \vee \bar{x_2} \vee x_4) \wedge (x_2 \vee x_5 \vee x_4) \wedge (x_2 \vee \bar{x_3} \vee \bar{x_5})$ を半 2 部グラフに変換したものを，図 3 に示す．

3.4 帰着の妥当性

帰着の妥当性，すなわち，3SAT の論理式 ϕ が充足することの必要十分条件が，ϕ から既述の帰着で作成した半 2 部グラフ G に X を根とするクリークマイナーが存在することであることを示す．そのために，幾つかの補題を用いる．

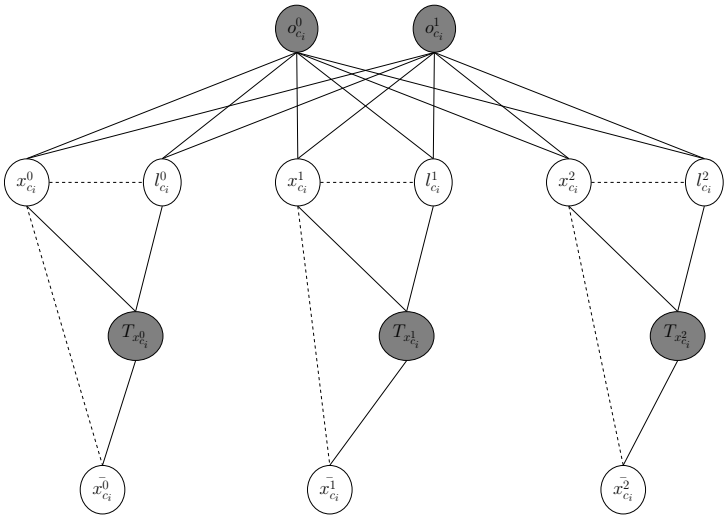

図 2 節の表現

補題 3.2 半 2 部グラフにおいて，X を根とするクリークマイナーが存在するとき，X 上の完全グラフを得る有限回の縮約操作では，頂点間に不在辺が存在するような，ある 2 つの頂点 $v_i, v_j \in X$ について，v_i, v_j の両点に隣接する Y 上の頂点が v_k ただ 1 つのとき，v_k は必ず，v_i, v_j のいずれか 1 つに縮約される．

証明． v_k が v_i, v_j 以外の頂点に縮約されると仮定する．このとき，v_i, v_j の両点に隣接する Y 上の頂点は v_k 以外存在しないので，どんな縮約でもってしても，不在辺 $\{v_i, v_k\}$ は縮約後も存在する．よって矛盾する． □

この補題から，真理値割当頂点について，次の補題が成立する．

補題 3.3 上述の帰着で作成した半 2 部グラフにおいて，X を根とするクリークマイナーが存在するとき，X 上の完全グラフを得る有限回の縮約操作では，任意の変数頂点の組 $(x_i, \bar{x}_i)$ について，任意の真理値割当頂点 T_{x_i} は $x_i, \bar{x}_i$ のいずれか 1 つに必ず縮約される．

証明． 論理式の任意の節において，リテラル x_i と $\bar{x}_i$ が同一の節に現れない限り，変数頂点 $x_i, \bar{x}_i \in X$ の両点に同一の離接割当頂点が隣接することはなく，$x_i, \bar{x}_i$ の両点に隣接する Y 上の点は T_{x_i} ただ 1 つのみである．よって，補題 3.2 より，これが成立する． □

したがって，真理値割当頂点は，対応する変数頂点のいずれかに必ず縮約されることとなり，これが変数の割当に対応する．また，節の表現についても，いくつかの補題を示す．まず，節の表現について次のように定義する．

定義 3.4 節の表現

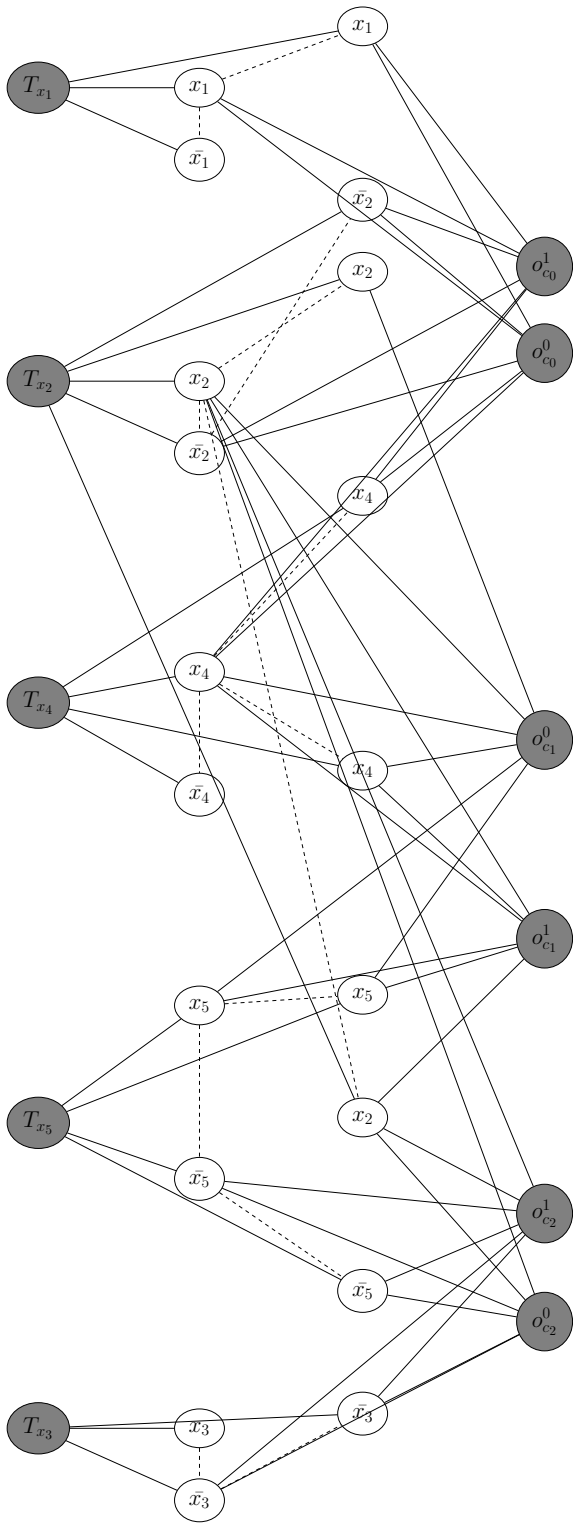

図 3 論理式 $\phi = (x_1 \vee \bar{x_2} \vee x_4) \wedge (x_2 \vee x_5 \vee x_4) \wedge (x_2 \vee \bar{x_3} \vee \bar{x_5})$ の半 2 部グラフへの変換

3CNF 論理式 ϕ の任意の節 $c_i = (l_0 \vee l_1 \vee l_2)(i = 0, 1, \ldots, n-1)$ に対応する節の表現は，帰着で作成される半 2 部グラフ G の部分グラフ G_{c_i} で定義される．

c_i の 3 つのリテラル l_0, l_1, l_2 について，帰着によって定義される 3 つのリテラル頂点で構成される頂点集合を，リテラル頂点集合 $L_{c_i} = \{l_0, l_1, l_2\}$ とする．

また，L_{c_i} 上のリテラル頂点のいずれかとの間に不在辺が定義されている変数頂点とその否定の変数頂点で構成される頂点集合を，変数頂点集合 $B_{c_i} = \{x_0, \bar{x}_0, x_1, \bar{x}_1, x_2, \bar{x}_2\}$ とする．

同様に，L_{c_i} 上のリテラル頂点のいずれかとの間に不在辺が定義されている真理値割当頂点と，辺が定義されている離接割当頂点で構成される頂点集合を，真理値割当頂点集合 $T_{c_i} = \{T_{x_0}, T_{x_1}, T_{x_2}, O_0, O_1\}$ とする．

このとき，G_{c_i} は $B_{c_i} \cup L_{c_i} \cup T_{c_i}$ に誘導される，G の部分グラフとなる．

このとき，次のことが言える．

補題 3.5 3CNF 論理式 ϕ より上述の帰着によって，半 2 部グラフ $G = (X, Y, E)$ を作成したと仮定する．

G について，X を根とするクリークマイナーが存在するとき，3CNF 論理式 ϕ の任意の節 $c_i (i = 0, 1, \ldots, n-1)$ に対応する，G の部分グラフ G_{c_i} について，L_{c_i} 上のリテラル頂点と不在辺が定義されている B_{c_i} 上の変数頂点の少なくとも 1 つは，X 上の完全グラフを得る有限回の縮約操作では，T_{c_i} 上の真理値割当頂点の縮約先となる．

証明． 帰着で作成された半 2 部グラフ G において，節の表現 G_{c_i} では，B_{c_i} 上の変数頂点と L_{c_i} 上のリテラル頂点の間に 3 本の不在辺が定義されている．この 3 本の不在辺のいずれかにおける 2 端点と隣接している離接割当頂点 G 全体においても 2 つのみである．

よって，2 個の離接割当頂点すべてを，変数頂点または，リテラル頂点に縮約しても，G_{c_i} において，少なくとも 1 本の不在辺が残り，この不在辺の端点に縮約可能な頂点は，T_{c_i} 上の真理値割当頂点のみである．

また，補題 3.3 より，真理値割当頂点は必ず，変数頂点に縮約される． □

補題 3.6 3CNF 論理式 ϕ より上述の帰着によって，半 2 部グラフ $G = (X, Y, E)$ を作成したと仮定する．G において X を根とするクリークマイナーが存在することの必要十分条件は，ϕ の任意の節 $c_i (i = 0, 1, \ldots, n-1)$ に対応する，G の部分グラフ G_{c_i} について，任意の i で，G_{c_i} において $B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在することである．

証明． まず，必要性，すなわち，「G において X を根とするクリークマイナーが存在する $\Rightarrow$ 任意の i で，G_{c_i} において $B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在する」を

示す．

これは自明であり，ある i について，部分グラフ G_{c_i} において，$B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在しないならば，定義より，有限回の縮約操作で完全グラフを得ることはできず，必ず 1 本以上の不在辺が残る．よって，グラフ全体の G において X を根クリークマイナーが存在すること，すなわち，G 上の不在辺の本数が 0 であることの仮定に明らかに矛盾する．

次に，十分性，すなわち，「任意の i で，G_{c_i} において $B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在する $\Rightarrow G$ において X を根とするクリークマイナーが存在する」を示す．

G_{c_i} における，不在辺の集合を M_{c_i} とする．同様に，G における，不在辺の集合を M とする．このとき，$M = (\cup_{i=0}^{n-1} M_{c_i})$ である．

したがって，任意の i で，G_{c_i} において，$B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在するとき，有限回の縮約操作によって M_{c_i} に属する不在辺の本数を 0 とすることができ，このとき G 全体における不在辺の本数も 0 となる．よって，命題は成立する． □

以上より，帰着で作成した半 2 部グラフにおいて，クリークマイナーが存在するのは，任意の節に対応する G の部分グラフ G_{c_i} で $B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在する場合であり，$B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在するには，L_{c_i} 上のリテラル頂点と不在辺が定義されている B_{c_i} 上の変数頂点の少なくとも 1 つは，T_{c_i} 上の真理値割当頂点の縮約先となる必要があり，これが，論理式の連言標準形に対応する．

以上の補題を用いて最終的に，上述の帰着の妥当性を証明する．

補題 3.7 上述の方法によって，3SAT の問題を，指定された頂点被覆 X を根とするクリークマイナー問題に帰着したと仮定する．

このとき 3CNF 論理式 ϕ から生成される，半 2 部グラフ $G = (X, Y, E)$ について，ϕ が充足することの必要十分条件は，G において，指定された頂点被覆 X を根とするクリークマイナーが存在することである．

証明． まず，必要性，すなわち，「ϕ が充足する $\Rightarrow G$ において X を根とするクリークマイナーが存在する」を示す．

ϕ が充足するならば，任意の節 $c_i (i = 0, 1, \ldots, n-1)$ で少なくとも 1 つのリテラルが真理値の割当によって，真となる．縮約操作において，この真となるリテラルを表すリテラル頂点と論理的に等価な変数頂点を，真理値割当頂点の縮約先とし，それ以外の変数頂点もしくは，リテラル頂点を離接割当頂点の縮約先とすれば，任意の節 c_i に対応する G の部分グラフ G_{c_i} において，不在辺の本数は 0 となる．

$V(G_{c_i}) \backslash T_{c_i} = B_{c_i} \cup L_{c_i}$ であるから，$B_{c_i} \cup L_{c_i}$ を根とするクリークマイナーが存在することとなり，補題 3.6 より，G において X を根とするクリークマイナーが存在することとなる．

次に，十分性，すなわち，「G において X を根とするクリークマイナーが存在する $\Rightarrow \phi$ が充足する」を示す．

補題 3.5 より，G において X を根とするクリークマイナーが存在するとき，任意の節 c_i に対応する G の部分グラフ G_{c_i} において，L_{c_i} 上のリテラル頂点と不在辺が定義されている B_{c_i} 上の変数頂点の少なくとも 1 つは縮約操作では，T_{c_i} 上の真理値割当頂点の縮約先となる．したがって，真理値割当頂点の縮約先の変数頂点が表す論理変数に真の真理値を割り当てれば，論理式 ϕ は充足する． □

よって結論として，次を得る．

定理 3.8 指定された頂点被覆 X を根とするクリークマイナー問題は NP 完全である．

4. アルゴリズム

以降では，既に NP 完全性を示した，指定された頂点被覆 X を根とするクリークマイナー問題について，その厳密アルゴリズムを示す．アルゴリズムの概要は次の通り．不在辺に着目した固定パラメータアルゴリズムであり，不在辺の本数が少ないときに計算量が小さくなる．カーネル化後は不在辺に対して，縮約元の頂点についてのバックトラック法を用いることによって，X を根とするクリークマイナーの存在の有無を厳密に判定する．

4.1 還元規則

インスタンスをカーネル化するに当たり，還元規則を導入する．以下の 3 つの規則は，自明である．

規則 1 半二部グラフ (X, Y, E) とする．$v \in X$ について，全ての $w \in X \setminus \{v\}$ について $\{v, w\} \in E(G)$ ならば，新たなインスタンス $(X \setminus \{v\}, Y, E)$ とする．

規則 2 半二部グラフ (X, Y, E) とする．$u \in Y$ について，u の隣接頂点集合 $N_u \subseteq X$ とするとき，全ての異なる 2 頂点 $v, w \in N_u$ について，$\{v, w\} \in E(G)$ であるならば，新たなインスタンス $(X, Y \setminus \{u\}, E)$ とする．

規則 3 半二部グラフ (X, Y, E) とする．$v \in Y$ について，v の次数が 1 ならば，新たなインスタンス $(X, Y \setminus \{v\}, E)$ とする．

さらに，不在辺について，次の還元規則を導入する．

規則 4 半 2 部グラフ (X, Y, E) で不在辺の集合を M とする．ある不在辺 $m_i \in M$ について，m_i の 2 端点に隣接する，Y 上の頂点集合を V_{m_i} とする．$|V_{m_i}| \geq |M|$ ならば，不在辺集合 M から m_i を削除し，新たなインスタンス

$(X, Y, E \cup \{m_i\})$ とする．

この還元規則 4 の妥当性を示す．

証明． 還元規則の適応前のインスタンスと還元規則適応後のインスタンスの正負が一致すること，すなわち，「$G = (X, Y, E)$ において，頂点被覆 X を根とするクリークマイナーが存在する．」$\Leftrightarrow$「$G' = (V, E \cup \{m_i\})$ において，頂点被覆 X を根とするクリークマイナーが存在する．」を示す．

まず，必要性，「$G = (X, Y, E)$ において，X を根とするクリークマイナーが存在する．」$\Rightarrow$「$G' = (X, Y, E \cup \{m_i\})$ において，X を根とするクリークマイナーが存在する．」を示す．これは，自明であり，辺集合が小さいインスタンスでクリークマイナーが存在するならば，辺集合が大きいインスタンスでもクリークマイナーは存在する．

次に，十分性「$G' = (X, Y, E \cup \{m_i\})$ において，頂点被覆 X を根とするクリークマイナーが存在する．」$\Rightarrow$「$G = (X, Y, E)$ において，X を根とするクリークマイナーが存在する．」を示す．仮定より，任意の不在辺 $m_j \in M \setminus \{m_i\}$ は，縮約操作において完全グラフを得る上で，縮約元の Y の頂点が存在する．これらの合計は，高々 $|M|-1$ 頂点であり，すべての頂点が，V_{m_i} 上の頂点であっても，$|V_{m_i}| \geq |M|$ より，少なくとも 1 頂点以上の V_{m_i} 上の頂点は，縮約元の頂点とはなっていないので，この頂点を m_i の 2 端点のいずれかに縮約することで，完全グラフを得ることは可能である．よって，命題は証明された．

□

上述の縮約規則を適用できなくなるまで適用することにより，以下の定理を得る．

定理 4.1 指定された頂点被覆を根とするクリークマイナー問題は，不在辺の本数を $k = |M|$ としたとき，$O(k^2)$ の頂点と $O(k^3)$ の辺のカーネルを持つ．

証明． 還元規則 1，還元規則 2 の適応後，X 上の任意の頂点は，不在辺の端点の 1 つとなる．よって自明に $|X| \leq k+1$ となる．

また，還元規則 3，還元規則 4 の適応後，Y 上の任意の頂点は，不在辺の端点の 1 つとなり，X 上の任意の不在辺 $m_i \in M$ において，$|V_{m_i}| < k$ が成立する．よって，$|Y| \leq k(k-1)$ となる．

同様に，X 上の頂点間の辺の本数は，$|X| \leq k+1$ より，高々 $\frac{1}{2}(k+1)k - k$ となり，X 上の頂点と Y 上の頂点間の辺の本数は，$|X| \leq k+1$，$|Y| \leq k(k-1)$ より，高々 $(k+1)k(k-1)$ となる．□

4.2 木幅による判定

上述の還元規則だけでは，不在辺の本数が多い場合，判定に時間がかかる．よって，次の補題を用いて，負のインスタンスを判定する．

補題 4.2 ([7]) 半 2 部グラフ G の木幅を，$\mathrm{tw}(G)$ としたとき，$\mathrm{tw}(G) < |X| - 1$ ならば，負のインスタンスである．

4.3 バックトラック法

上述の還元規則と判定規則を用いたバックトラック法による厳密アルゴリズムを次に示す．

アルゴリズムでは，入力のインスタンスに対して，規則 1，規則 2，規則 3，規則 4 の適応を繰り返し，不在辺集合が空となった場合，正のインスタンスと判定する．不在辺集合が空でない場合，補題 4.2 を用いて，負のインスタンスを判定する．補題 4.2 を用いて負と判定されなかった場合，最終的に縮約元の頂点に対するバックトラック法を用いて，インスタンスの正負を判定する．

5. 実験的評価

既に示したアルゴリズムについて，その性能を実験に基いて評価する．但し，実際の実験ではアルゴリズムの実行時間に影響しないので，規則 1，2，3 は行っていない．

5.1 インスタンス

PACE（Parameterized Algorithm and Computation Experiments challenge）2017 コンペティション [2] における厳密木幅部門に対する玉木らの提出 [3] では，木幅計算における前処理として，安全なセパレータ [5] の検出が発見的解法により行なわれている．彼らは，与えられたグラフに対して貪欲木分解アルゴリズムを用いてセパレータを列挙し，各セパレータについて，切り離される連結成分とセパレータによって誘導される部分グラフから半 2 部グラフを生成し，半 2 部グラフ上で指定された根付きクリークマイナー問題を解いている [1]．

本実験では，PACE2017 における厳密木幅部門の公開インスタンスの 100 つ [4] に対して，玉木らの提出上で生成される半二部グラフをベンチマークインスタンスとして用いた．

5.2 実験環境

実験環境は次の通りである．

- CPU: Intel Core Xeon 2.20GHz
- RAM: 1024GB,
- OS: Ubuntu 14.04 LTS, Trusty Tahr,
- プログラミング言語: Java 1.8,
- JVM: 1.8.0_131,
- 最大ヒープサイズ: 24GB.

5.3 実験結果

全インスタンス 160219 個の正負の判定理由を表 1 に

アルゴリズム 1 半 2 部グラフ G 上にクリークマイナーが存在するか判定するアルゴリズム

```
function JudgeCliqueMinorExist(G)
    G の不在辺をリストアップし，集合 M とする．
    while G が還元可能 do
        for i = 0 to |X| do
            if ∀v_j ∈ X\{v_i}, {v_i, v_j} ∈ E(G) then
                X ← X\{v_i}
            end if
        end for
        for i = 0 to |Y| do
            v_i ∈ Y の隣接頂点集合を N_{v_i} とする．
            if ∀x, y ∈ N_{v_i}, {x, y} ∈ E(G) then
                Y ← Y\{v_i}
            end if
        end for
        for i = 0 to |Y| do
            if v_i ∈ Y の次数=1 then
                Y ← Y\{v_i}
            end if
        end for
        for i = 0 to |M| do
            if m_i ∈ M に隣接する Y 上の頂点数 |V_{m_i}| ≥ |M| then
                M ← M\{m_i}
            end if
        end for
    end while
    if |M| = 0 then
        return TRUE
    else if 木幅 tw(G) < |X| − 1 then
        return FALSE
    end if
    return FindCliqueMinor(0)
end function
function FindeCliqueMinor(i)
    if i > |M| then
        return TRUE
    end if
    if m_i ∈ M の端点の 1 つが，ある頂点 v ∈ V_{m_i} の縮約先となっている then
        return FindCliqueMinor(i+1)
    end if
    for j = 0 to |V_{m_i}| do
        m_i ∈ M の端点の 1 つを v_j ∈ V_{m_i} の縮約先とする．
        if FindCliqueMinor(i + 1) = TRUE then
            return TRUE
        end if
        m_i ∈ M の端点のもう一方を v_j ∈ V_{m_i} の縮約先とする．
        if FindCliqueMinor(i + 1) = TRUE then
            return TRUE
        end if
    end for
    return FALSE
end function
```

示す．

ここでバックトラック法による判定の内，制限時間内に判定が終了したものと，タイムアウトによって負と判定されたものは別で集計している．今回の実験ではタイムアウトの制限時間を 1000ms とした．

表 1 インスタンスの正負判定理由

判定理由	個数
還元規則	58407
木幅による判定	58119
バックトラック法 (判定終了)	43357
バックトラック法 (タイムアウト)	336

また，バックトラック法で判定されたインスタンスのうち，153 個については，玉木らが PACE の提出 [3] で用いた発見的解法では負のインスタンスと判定されており，今回の実験で，本稿のアルゴリズムにより，新たに正と判定されたものである．

次に，還元規則の性能について，実験結果を示す．図 4 は，還元規則によって削除された不在辺の本数について，元々の不在変数集合全体に対する割合を示したものである．図 4 の縦軸が削除された割合を示し，横軸はインスタンスが名前順に並んでいる．

全インスタンスの内，100%縮小されたインスタンスと 0%縮小されたインスタンスが非常に多く，100%のインスタンスは 58407 個，0%のインスタンスは 100441 個であった．

そして，アルゴリズムの実行時間についての実験結果を示す．インスタンスには玉木らの PACE の提出で用いられた発見的解法の実行時間の情報があり，これを用いて本アルゴリズムの性能を評価する．

図 5 から図 8 は，玉木らの発見的解法と本稿のアルゴリズムの実行時間の比較である．縦軸はミリ秒の実行時間差で横軸はインスタンスが名前順に並んでいる．

図 5 は還元規則によって正と判定されたインスタンスについて，その判定時間と玉木らの発見的解法の判定時間との差分を示している．但し，差分が正のインスタンスにおいて，還元規則の方が判定時間が短く，差分が負のインスタンスにおいて，還元規則の方が判定時間が長いことを示す．これは以降の図についても同様である．還元規則によって正と判定されたインスタンスのうち，発見的解法に比べて判定時間が 0ms 以上早いインスタンスは 35999 個であった．

図 6・図 7 は木幅を用いた判定によって負と判定されたインスタンスについて，その判定時間と玉木らの発見的解法の判定時間との差分を示している．図 7 は図 6 から，木幅計算において，非常に時間のかかったインスタンスである，ex115con_022 を取り除いたものである．木幅を用いた判定によって，負と判定されたインスタンスのうち，発見的解法に比べて判定時間が 0ms 以上早いインスタンスは 5257 個であった．

図 8 は，バックトラック法によって正負が判定されたインスタンスについて，その判定時間と玉木らの発見的解法の判定時間との差分を示している．但し，図 8 は厳密に判

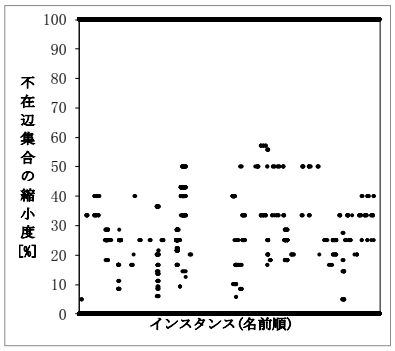

図 4　還元規則による不在辺集合の縮小度

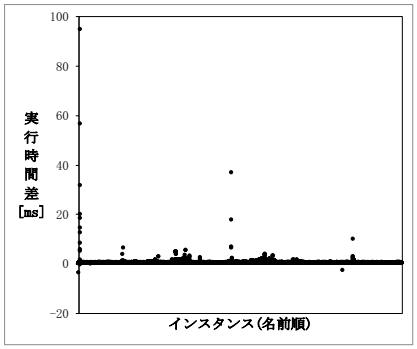

図 5　発見的解法との時間差 (還元規則)

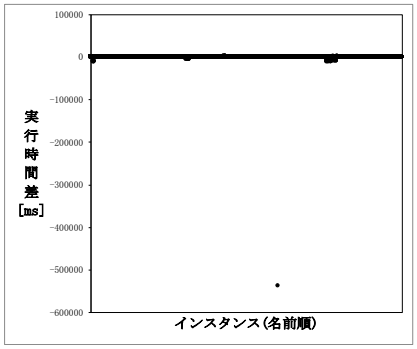

図 6　発見的解法との時間差 (木幅による判定規則)

定されたインスタンスとタイムアウトによって判定されたインスタンスの両方の差分を集計したものであり，木幅を用いた判定規則の判定時間を除いたものである．バックトラック法によって正負が判定されたインスタンスのうち，発見的解法に比べて判定時間が 0ms 以上早いインスタンスは 629 個であった．

6. 終わりに

指定された頂点被覆を根とするクリークマイナー問題の NP 完全性の証明によって，頂点集合を根とするクリークマイナー問題は，より簡略化された問題においても計算が困難であることが示された．

PACE のインスタンスを用いた実験において，カーネル化後のバックトラック法では，計算時間が非常に掛かったが，一方で，還元規則や木幅による判定規則のみで正負を

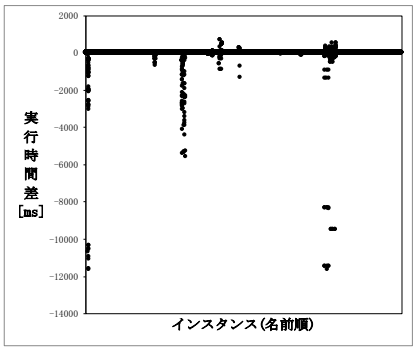

図 7　発見的解法との時間差 (木幅による判定規則)

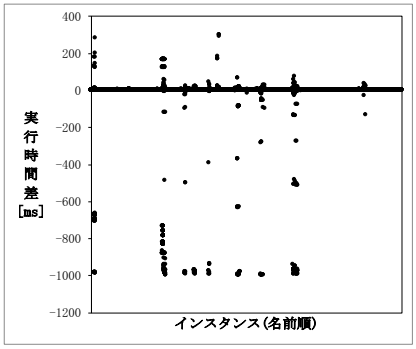

図 8　発見的解法との時間差 (バックトラック法)

判定できるインスタンスも多いことが分かった．

今後の研究課題として，他の還元規則や，還元規則の効果を最大化できるような，指定された頂点集合を根とするクリークマイナー問題からの指定された頂点被覆を根とするクリークマイナー問題のインスタンスの生成方法などが考えられる．

参考文献

[1] Tamaki, Hisao. "Positive-instance driven dynamic programming for treewidth." arXiv preprint arXiv:1704.05286 (2017).

[2] The Parameterized Algorithm and Computation Experiments challenge. https://pacechallenge.wordpress.com. Accessed: 2018-01-05.

[3] GitHub repository PACE2017-TrackA for TCS-Meiji on PACE2017 Track A submission. https://github.com/TCS-Meiji/PACE2017-TrackA. Accessed: 2018-01-05.

[4] GitHub repository PACE2017-2017-TrackA-instances for PACE-challenge. https://github.com/PACE-challenge/PACE-2017-TrackA-instances. Accessed: 2018-01-05.

[5] Bodlaender, Hans L., and Arie MCA Koster. "Safe separators for treewidth." Discrete Mathematics 306.3 (2006): 337-350.

[6] Richard M. Karp. "Reducibility Among Combinatorial Problems." Complexity of Computer Computations. New York: Plenum. (1972): 85 - 103.

[7] 大塚 広夢，玉木 久夫，牧井 慶太朗．「根付きクリークマイナーアルゴリズムの設計と実験による評価」．情報処理学会研究報告，第 166 回アルゴリズム研究会, 2018.

© 2018 Information Processing Society of Japan