Vol.2019-AL-175 No.9
2019/11/28

グラフの2等分割問題に対するアルゴリズムと計算複雑性

小林 靖明[1]　曽根 大雅[1,a)]　土中 哲秀[2]

概要：グラフの 2 等分割問題は，辺重み付きグラフが与えられたときに，その頂点集合を (ほぼ) 同じサイズの 2 つの部分集合に分割したときに，その間の辺の重みの総和を最大化/最小化する問題である．この問題は一般的に NP 困難であることが知られているが，グラフの木幅が定数であるときに効率的な動的計画法アルゴリズムが存在する．幅が w の木分解が与えられたとき，Jansen ら (SIAM J. Comput. 2005) は $O(2^w n^3)$ 時間で動作する動的計画法を与えた．ここで n はグラフの頂点数である．最近になって Eiben ら (ESA 2019) が 3 乗の依存度を木分解の幅の依存度を犠牲にすることで改善した．具体的には，$O(8^w w^5 n^2 \log n)$ 時間のアルゴリズムを与えた．本稿では，この問題に対する $O(2^w w^4 n^2)$ 時間アルゴリズムを与える．この結果は Eiben らが示した n に関する依存度の条件付き下界と木分解の幅 w に関する依存度の強指数時間仮説を仮定した下界に漸近的に一致する．また，いくつかのグラフクラスにおける困難性や容易性を議論する．

1. はじめに

グラフ $G=(V,E)$ の 2 **等分割** (bisection) とは頂点集合 V の 2 分割 (A,B) で要素数の差が高々 1 であるものである．G の辺重み関数 $w: E \to \mathbb{R}_+$ を考えたとき，A と B にまたがる辺の重みの総和を最大化または最小化する問題は，**最大 2 等分割問題**または**最小 2 等分割問題**と呼ばれ，様々な観点からよく研究されている問題である．よく知られている最大カット問題と最小カット問題とは状況が異なり，最大/最小 2 等分割問題のいずれも NP 困難であることが知られている．しかしながら，その状況は最大化と最小化のどちらかかによって若干異なる．注目すべきは，最大化問題が平面グラフに限っても NP 困難である [12] のに対し，最小化問題の平面グラフでの複雑性は長年未解決問題である．また，重みなしのグラフにおいて，またがる辺の数を最大化および最小化する問題も NP 困難であるが，その数がパラメータ k 以上，または k 以下になるかどうかでも状況についても異なり，最大化問題が固定パラメータ容易 (Fixed-Parameter Tractable，FPT) であることは簡単に確認できるが [*1]，最小化問題については非自明であり，固定パラメータ容易性は [5], [6] によって解決された．

2 等分割問題は木幅限定グラフ [*2] においても効率よく解くことができる．Jansen ら [12] は，木幅が w であるようなグラフに対して，$O(2^w n^3)$ 時間アルゴリズムを与えた．Eiben ら [7] は，実行時間における頂点数 n に関する依存度を減らすことに成功した．具体的には，最小重み 2 等分割問題に対し，$O(8^w w^5 n^2 \log n)$ 時間アルゴリズムを与えた．また，実行時間の条件付き下界に関しても議論を行い，要素数 n の $(\min, +)$-半環上での畳込み演算に関する問題が $O(n^{2-\varepsilon})$ 時間で解くことができない限り，木の最小 2 等分割問題が $O(n^{2-\delta})$ 時間で解けないことを示した．この畳込み演算の問題は n 要素の 2 つの数列 $(a_i)_{0 \le i < n}$，$(b_i)_{0 \le i < n}$ が与えられたとき，$c_i = \min_{0 \le j \le i}(a_j + b_{i-j})$ によって定義される数列 $(c_i)_{0 \le i < n}$ を計算する問題である．このため，木幅が定数であるようなグラフに対しては，計算時間の上界と (条件付き) 下界の間に $\log n$ のギャップが存在する．

本稿では，木幅限定グラフに対して最大/最小 2 等分割問題に対する "最適な" アルゴリズムを与える．具体的には，木幅が w であるようなグラフとその木分解が与えられたとき，$O(2^w w^4 n^2)$ 時間のアルゴリズムを与える．実行時間の n に対する依存度は n^2 であるため，Eiben ら [7] の $\log n$ のギャップを解消している．このアルゴリズムは Jansen ら [12] の動的計画法と本質的に同じであるが，より正確な実行時間の解析を与えることで改善を達成している．また，最大重み 2 等分割問題に対して，強指数時間仮説 (Strong Exponential Time Hypothesis，SETH) のもとで $2^{w-\varepsilon} n^{O(1)}$ 時間アルゴリズムが存在しないことを示す．これらの結果により，(正とは限らない任意の重み関数に

[1] 京都大学
[2] 中央大学
a) taiga_sone@iip.ist.i.kyoto-u.ac.jp

[*1] 大きさ $\lceil |E|/2 \rceil$ の 2 等分割が常に存在することが証明できる [10]．さらに，[10], [16] は大きさ $\lceil |E|/2 \rceil + k$ 以上の 2 等分割があるかどうかを判定する問題の固定パラメータ容易性を示している．

[*2] 木幅限定グラフよりも広いクラスであるクリーク幅限定グラフにおいても多項式時間可解である [8]．

おいて) 本稿で与えたアルゴリズムは木幅に対する指数部分の依存度と頂点数に対する多項式部分の依存度がともに (SETH および $(\min, +)$ 畳み込み問題の下界を仮定すると) 最適であると言える．

また，2 等分割問題に対するグラフクラスに関する困難性や多項式時間可解性についても議論する．特に，いくつかのグラフクラスにおける最大/最小 2 等分割の困難性を示す．既存の困難性の帰着を用いることで，スプリットグラフ，比較可能グラフ，AT-free グラフ，claw-free グラフにおいての最大 2 等分割問題が NP 困難であることを示し，これらによってスプリットグラフ，比較不能グラフにおける最小 2 等分割問題の NP 困難性を示す．さらに，2 部グラフと補 2 部グラフにおいては，どちらの問題も NP 困難であることを示す．この結果については新しいアイデアに基づく帰着によって証明する．その一方で，ライングラフにおいては最大 2 等分割問題は多項式時間可解であることを示す．

2. 準備

$G = (V, E)$ を無向グラフとする．以降では，n は与えられたグラフの頂点数を表すとする．頂点 $v \in V$ について，$N(v)$ は v の隣接点集合とする．頂点集合 $X \subseteq V$ について，$G[X]$ で X によって誘導される G の部分グラフを表す．G の 2 **等分割**とは V の 2 分割 (A, B) で $||A| - |B|| \leq 1$ であるようなもののことである．辺に重みのないグラフにおいては，2 等分割の**大きさ**または**サイズ**を A と B にまたがる辺の数，つまり，$|\{\{a, b\} \in E : a \in A, b \in B\}|$ と定義する．また，重み付きグラフの場合においては，A と B にまたがる辺の重みの総和でサイズを定義する．

本稿の次の節では，木分解を用いた動的計画法について取り組む．グラフ G の**木分解**とは，頂点集合が I である根付き木 T と V の部分集合族 $\{X_i : i \in I\}$ の対 $(T, \{X_i : i \in I\})$ で，以下の性質を満たすものである．

- $\bigcup_{i \in I} X_i = V$.
- 任意の $e \in E$ について，$e \subseteq X_i$ を満たす $i \in I$ が存在．
- 任意の $v \in V$ について，$v \in X_i$ を満たす添字の集合 $\{i \in I : v \in X_i\}$ は T において部分木を誘導する．

G の頂点と木分解の頂点を区別するために，木分解についてはノードと呼ぶことにする．また，文脈から集合族 $\{X_i : i \in I\}$ が明らかなときは，単に T を木分解と呼ぶことにする．木分解 T の**幅**とは，$|X_i|$ の最大値から 1 減じた値であり，G の**木幅**とは，G が幅 w の木分解を持つような最小の非負整数 w のことである．

集合 X と要素 v について，$X \cup \{v\}$ や $X \setminus \{v\}$ の代わりに，それぞれ $X + v$ や $X - v$ と書くことにする．木分解 T が**好適**であるとは，T の各ノード $i \in I$ が次のいずれかの性質を満たすことである．

- **葉ノード**: i が葉であるならば，$X_i = \emptyset$.
- **導入ノード**: i はちょうどひとつの子 $j \in I$ を持ち，ある $v \in V \setminus X_j$ について $X_i = X_j + v$ を満たす．
- **忘却ノード**: i はちょうどひとつの子 $j \in I$ を持ち，ある $v \in V \setminus X_i$ について $X_j = X_i + v$ を満たす．
- **結合ノード**: i はちょうどふたつの子 $j, k \in I$ を持ち，$X_i = X_j = X_k$ を満たす．

与えられた木分解から好適な木分解への変換は効率良く行うことができる．

補題 1 ([13])**.** G の幅 w の木分解が与えられたとき，G の幅 w の好適な木分解を求める $O(w^2 n)$ 時間アルゴリズムが存在する．さらに，その好適な木分解は $O(wn)$ 個のノードを持つ．

3. 木幅限定グラフ

本節では，辺重み付きグラフ $G = (V, E)$ と G の木分解 T が与えられたときに，2 等分割問題に対する木分解に基づく動的計画法を与える．この節においては任意の辺重み関数 $w : E \to \mathbb{R}$ を許すこととする．この設定においては最大 2 等分割問題と最小 2 等分割問題は等価であるため，本節では最大化問題として取り扱う．

3.1 FPT アルゴリズム

本節では $(T, \{X_i : i \in I\})$ を幅が w の G の好適な木分解とする．ノード $i \in I$ について，V_i を X_i または i の子孫 $j \in I$ において X_j に出現するような G の頂点からなる集合とする．

以下で示すアルゴリズムは Jansen ら [12] が与えたものと本質的に等価である．そのため，アルゴリズムの正しさについては [12] に従う．

T の各ノード $i \in I$ と $S \subseteq X_i$，$0 \leq d \leq |V_i|$ について，$\mathrm{bs}(i, S, d)$ を $G[V_i]$ の 2 等分割 (A_i, B_i) で $A_i \cap X_i = S$，$|A_i| = d$ であるようなものの中の最大サイズと定義する．ここで，A_i と B_i それぞれは空であることを許す．以下で示す動的計画法は $\mathrm{bs}(i, S, d)$ の値を木分解に沿ってボトムアップに計算する．

葉ノード

$i \in I$ を葉ノードとする．$V_i = X_i = \emptyset$ であるため，$\mathrm{bs}(i, \emptyset, 0) = 0$ である．

導入ノード

$i \in I$ を導入ノードとする．$v \in X_i \setminus X_j$ を X_i で導入される頂点とする．ここで $j \in I$ はノード i の唯一の子である．$G[V_i]$ における v の隣接点はすべて X_i に含まれるため，各 $S \subseteq V$ と $0 \leq d \leq |V_i|$ について

$$\mathrm{bs}(i, S, d) = \begin{cases} \mathrm{bs}(j, S - v, d - 1) + w(\{v\}, X_i \setminus S) & (v \in S) \\ \mathrm{bs}(j, S, d) + w(\{v\}, S) & (\text{それ以外}) \end{cases}$$

という再帰式で $\mathrm{bs}(i,S,d)$ が計算可能である．ここで，互いに素な頂点集合 $X,Y \subseteq V$ において $w(X,Y)$ は X と Y にまたがる辺の重みの総和であるとする．

忘却ノード

$i \in I$ を忘却ノードとする．$v \in X_j \setminus X_i$ を X_i で忘却される頂点とする．ここで $j \in I$ はノード i の唯一の子である．$G[V_i] = G[V_j]$ であるため，$\mathrm{bs}(i,S,d)$ は以下の再帰式によって計算できる．

$$\mathrm{bs}(i,S,d) = \max(\mathrm{bs}(j,S+v,d), \mathrm{bs}(j,S,d)).$$

結合ノード

$i \in I$ を結合ノードとし，$j,k \in I$ をその子ノードとする．T の定義より $X_i = X_j = X_k$ である．導入，忘却ノードとは異なり，$\mathrm{bs}(i,S,d)$ をわずかながらに違う考え方に基づき計算を行う．この違いはアルゴリズムの正しさについては影響を与えないが，計算時間の解析にとっては重要である．各 $S \subseteq X_i$ と $0 \le d \le |V_i|$ について，

$$\mathrm{bs}(i,S,d) = \max_{0 \le d' \le d}(\mathrm{bs}(j,S,d') + \mathrm{bs}(k,S,d-d')) - w(S, X_i \setminus S)$$

が成り立つことは Jansen らによって示されている．ここでは，各 d について上の再帰式を計算するのではなく，$0 \le d_j \le |V_j|$ と $0 \le d_k \le |V_k|$ を満たすすべての対 (d_j, d_k) について，$(i, S, d_j + d_k)$ をインデックスに持つ表の値を更新する動的計画法を考える．このとき，各 $S \subseteq X_i$ について $\mathrm{bs}(i,S,*)$ の形をした値が $O(|V_j||V_k|)$ で計算できることが鍵である．

実行時間の解析

各導入ノードまたは忘却ノード $i \in I$ については $\mathrm{bs}(i,*,*)$ のすべての値を $O(2^w wn)$ 時間で計算可能である．また，各結合ノード $i \in I$ については $\mathrm{bs}(i,*,*)$ のすべての値を

$$O(2^w w^2) + \sum_{S \subseteq X_i} O(|V_j||V_k|) = O(2^w |V_j||V_k|)$$

で計算可能である．ここで，左辺の 1 項目はすべての $S \subseteq X_i$ について $w(S, X_i \setminus S)$ を求める計算時間である．再び，$j,k \in I$ はノード i のふたつの子であるとする．$|V_j|$ と $|V_k|$ に対する自明な上界は n であるため，動的計画法全体の実行時間は $O(2^w wn^3)$ である．これは補題 1 より得ることができる．

この素朴な計算時間の見積もりを改善するために，結合ノードについては別の解析を行う．すべての導入ノードとすべての忘却ノードの計算時間の総和は $O(2^w w^2 n^2)$ である．以下では，全結合ノードについて総計算時間が $O(2^w w^4 n^2)$ で上から抑えられることを示す．各結合ノード $i \in I$ について $n_i = \sum_{j \preceq i} |X_j|$ とする．ここで，ノード $j \preceq i$ はノード i または i の子孫全体での $|X_j|$ の総和とする．つまり，n_i は $|V_i|$ と一致するとは限らず，一般に $|V_i| \le n_i$ である．よって，以下では，

$$\sum_{X_i:\ \text{結合ノード}\ X_i} n_j n_k = O(w^4 n^2). \qquad (1)$$

が成り立つことを示す．このことを確かめるために，各ノード $i \in I$ の X_i の頂点についてラベル $v_{1,i}, v_{2,i}, \ldots, v_{|X_i|,i}$ を割り当てる．ここで重要なことは，G のひとつの頂点にふたつ以上の異なるラベルが割り当てられることがあり，ここでは，それらを異なる頂点とみなすこととする．以下では，これらを区別するために，ラベル付き頂点と呼ぶこととする．ここで，T が $O(wn)$ ノードもつことと，各ノード $i \in I$ について $|X_i| \le w+1$ であることに着目すると，異なるラベルを持つ頂点の個数は $O(w^2 n)$ である．これらに従って式 1 を評価すると，各結合ノード $i \in I$ について，$n_j n_k$ の項は，n_j 頂点と n_k 頂点の異なるラベルを持つ 2 部グラフ上における辺の個数であると解釈できる．また，各ラベル付き頂点対は (それらを含むノードの最近祖先ノードにおいて) 高々 1 度しかカウントされないため，式 1 は全体として，$O((w^2 n)^2)$ で上から抑えられる．よって，式 1 が得られる．

定理 1. 幅が w の G の木分解が与えられたとき，$O(2^w w^4 n^2)$ 時間で最大 2 等分割を計算することができる．

3.2 アルゴリズムの最適性

Eiben ら [7] は最初の節で説明した $(\min, +)$ 畳み込み計算が $O(n^{2-\varepsilon})$ 時間で計算できない限り，木の最小 2 等分割問題に対して $O(n^{2-\delta})$ 時間アルゴリズムが存在しないことを証明した．この結果は，前節のアルゴリズムの n に関する依存度が条件付きで最適であることを示している．

この節では，前節のアルゴリズムが木幅に関する依存度についても，強指数時間仮説 (SETH)[11] のもとで最適であることを示す．SETH は精細な複雑性 (fine-grained complexity) の研究においてよく用いられる計算量理論の仮定であり，その帰結のひとつとして，SETH が正しいと仮定すると n 変数の CNFSAT に対する $(2-\varepsilon)^n$ 時間アルゴリズムの非存在性を導くことができる．このことを用いることで様々な条件付き計算時間の下界を得ることができる．この節の主張を示すために以下の事実を利用する．

定理 2 ([15])**.** SETH が正しいと仮定すると，幅が w の木分解が入力として与えられたとき，任意の $\varepsilon > 0$ について，最大カット問題を解く $(2-\varepsilon)^w n^{O(1)}$ 時間アルゴリズムは存在しない．

最大カット問題から最大 2 等分割問題へは標準的な帰着によって下界を示すことができる．最大カット問題のインスタンスが与えられたとき，n 個の孤立点を加える．このとき，最大カット問題の最適値の最大 2 等分割問題の最適値は一致する．また，最大 2 等分割問題の最適解から最大

カット問題の最適解を追加して孤立点を除くことで容易に得られる．この帰着は木幅を保存するため，以下の定理が得られる．

定理 3. SETH が正しいと仮定すると，幅が w の木分解が入力として与えられたとき，任意の $\varepsilon > 0$ について，最大 2 等分割問題を解く $(2-\varepsilon)^w n^{O(1)}$ 時間アルゴリズムは存在しない．

辺の重みを正に限定した場合，最小 2 等分割問題に対しては上の帰着を利用することはできない．最小 2 等分割問題と最大 2 等分割問題とはいずれも NP 困難であるものの，グラフクラスにおける困難性に関しては必ずしも困難性が一致するとは知られていない．特に平面グラフにおいては，最大 2 等分割問題が NP 困難であるのに対し，最小 2 等分割問題の困難性が未解決である，このように疎なグラフにおいての最小 2 等分割問題の困難性を証明することがそれほど容易ではない．正の重みの最小 2 等分割問題に対する $(2-\varepsilon)^w n^{O(1)}$ 時間アルゴリズムの条件付き非存在性に関しては依然として未解決である．

4. 様々なグラフクラスにおける困難性

本節では，重みなしの無向グラフにおいて，いくつかのグラフクラスにおいても最大および最小 2 等分割問題が NP 困難であることを示す．第 3.2 節においては，最大カット問題から最大 2 等分割問題への帰着を与えた．この帰着から導かれることとしては，グラフクラス $\mathcal{C}$ が，

- $\mathcal{C}$ において最大カット問題が NP 困難
- 任意の $G \in \mathcal{C}$ について，G に任意の個数の孤立点を追加したグラフ G' が $G' \in \mathcal{C}$ である

を満たせば最大 2 等分割問題がグラフクラス $\mathcal{C}$ において NP 困難である．よって以下のことがわかる．

定理 4. 最大 2 等分割問題はグラフをスプリットグラフ，比較可能グラフ，AT-free グラフ，claw-free グラフに限っても NP 困難である．

最大カット問題はスプリットグラフ [1]，比較可能グラフ [17] でも NP 困難であることが知られている．AT-free グラフや claw-free グラフの部分クラスである補 2 部グラフ (co-bipartite graph) でも最大カット問題が NP 困難であることが知られている [1] が，このグラフクラスは上の 2 番目の条件を満たさない．2 部グラフおよび補 2 部グラフにおける 2 等分割問題の困難性は次節で議論する．

与えられたグラフ G の頂点数が偶数 $2n$ であると仮定する．G の補グラフを $\overline{G}$ とすると，G が大きさ k 以上 2 等分割を持つことと，$\overline{G}$ が大きさ $n^2 - k$ 以下の 2 等分割を持つことが等価である．よって以下の定理が成立する．

定理 5. 最小 2 等分割問題はグラフをスプリットグラフ，比較不能グラフに限っても NP 困難である．

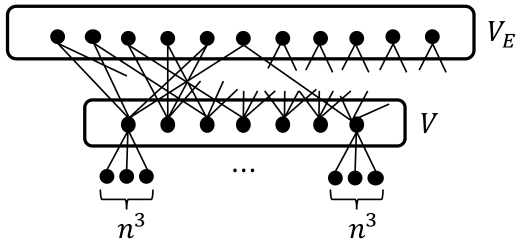

図 1 帰着グラフ G'

4.1 2 部グラフ

定理 6. 最小 2 等分割問題はグラフを 2 部グラフに限っても NP 困難である．

証明. 頂点数が偶数である 4-正則グラフにおける最小 2 等分割問題は NP 困難であることが知られている [3]．よって，4-正則グラフにおける最小 2 等分割問題から帰着する．4-正則グラフ $G = (V, E)$ が与えられたとする．ただし，$|V| = 2n$ とする．まず，各辺 $e = \{u, w\}$ を細分する，すなわち，e に対応する頂点 v_e を用いて，辺 $\{u, v\}$ を $\{u, v_e\}$, $\{v_e, w\}$ に置き換える．その後，各 $v \in V$ に対して，n^3 個のペンダント点をつける．ペンダント点の集合を V^p とする．このようにして得られたグラフを $G' = (V \cup V_E \cup V^p, E')$，ただし $V_E = \{v_e : e \in E\}$ とする（図 1）．今，G は 4-正則グラフなので，$|V_E| = |E| = (4 \cdot 2n)/2 = 4n$ であり，$|V \cup V_E \cup V^p| = 2n + 4n + 2n^4 = 6n + 2n^4$ である．また，図 1 からわかるように，G' は 2 部グラフである．以下では，G が大きさ k 以下の 2 等分割を持つ必要十分条件は，G' が大きさ k 以下の 2 等分割を持つことであることを示す．

G が大きさ k 以下の 2 等分割 (V_1, V_2) を持つとする．ここで，$|V| = 2n$ より，$|V_1| = |V_2|$ である．各 $i \in \{1, 2\}$ に対して，$V'_i = V_i$ とする．さらに，$v \in V_i$ のペンダント点も V'_i に全て加える．また，$u, v \in V_i$ となる辺 $e = \{u, v\}$ に対して，v_e を V'_i に加える．ここで，G' において，V'_1, V'_2 に含まれていない点は，G におけるカット辺 e に対応する点 v_e である．これらの点は，$V'_1 \cap V$ に含まれる点と $V'_2 \cap V$ に含まれる点を 1 点ずつ持つので，v_e を V'_1 と V'_2 のどちらに入れても G' においてカット辺はちょうど 1 しか増えない．ここで，$|V'_1| = |V'_2|$ となるように適当にカット辺に対応する v_e を V'_1, V'_2 にそれぞれ加える．

主張 1. $|V'_1| = |V'_2|$ を満たすように残りの頂点を振り分けることができる．

証明. もし，そのように振り分けることができないとすると，すでに V'_1 または V'_2 に属する V_E の頂点が $2n+1$ 以上であることが観察できる．これは $|V_E| = 4n$ より得ることができる．一方で，$G[V_1]$ および $G[V_2]$ は 4-正則であるため，高々 $2n$ 本の辺しか持てない．これは上で述べた事実と反するため，主張が成り立つ． □

よって，(V_1', V_2') は G' における 2 等分割であり，大きさは k である．

逆に，G' における大きさ k の 2 等分割 (V_1', V_2') が与えられたとする．

主張 2. 2 等分割 (V_1', V_2') は，$|V_1' \cap V| = |V_2' \cap V|$ を満たす．

証明. 背理法で示す．$|V_1' \cap V| > |V_2' \cap V|$ であると仮定する．このとき，$|V_1' \cap V| \geq n+1$ である．今，(V_1', V_2') は 2 等分割であるから，$|V_1'| = 3n + n^4$ である．$|V_1' \cap V| \geq n+1$ と $|V_1'| = 3n + n^4$ から，$v \in V_1 \cap V$ のペンダント点の中で，V_2 に含まれるものが少なくとも $n^3 - 2n + 1$ 個存在する．任意の $n \geq 5$ に対して，$n^3 - 2n + 1 > 4n^2 > k$ であるから，(V_1', V_2') の大きさが k 以下であることに矛盾する．□

主張 2 より $|V_1' \cap V| = |V_2' \cap V|$ であるから，$V_1 = V_1' \cap V$，$V_2 = V_2' \cap V$ とすると，(V_1, V_2) は G における 2 等分割である．また，その大きさは G' におけるカット辺かつ V と V_E の間にある辺の総和より小さいので k 以下である．□

定理 7. 最大 2 等分割問題はグラフを 2 部グラフに限っても NP 困難である．

証明. 4-正則グラフにおける最小 2 等分割問題から帰着する．定理 6 の G' を用いて，G が k 以下の 2 等分割を持つ必要十分条件は，大きさ G' が $2n^4 + 8n - k$ 以上の 2 等分割を持つことであることを示す．G における大きさ k の 2 等分割 (V_1, V_2) が与えられて，$V_1' = V_1$，$V_2' = V_2$ とする．さらに，各 $v \in V_1$ のペンダント点を V_2'，各 $v \in V_2$ のペンダント点を V_1 に加える．各 $v \in V_E$ に関して，もし $N(v) \subseteq V_1$ なら $v \in V_2$, $N(v) \subseteq V_2$ なら $v \in V_1$ とする．それ以外の $v_e \in V_E$ に関して，v_e は V_1 と V_2 の点をそれぞれ 1 点ずつ隣接点として持つので，e は (V_1, V_2) におけるカット辺である．G は 4-正則グラフなので，辺数は偶数であり，$|V_1'| = |V_2'|$ となるように V_E の点を V_1'，V_2' に適当に加える．今，(V_1, V_2) の大きさは k なので，(V_1', V_2') の大きさは，$2n^4 + 2 \cdot 4n - k$ となる．ここで，$|V_E| = |E| = 4n$ である．

逆に，G' における大きさ $2n^4 + 8n - k$ 以上の 2 等分割 (V_1', V_2') が与えられたとする．定理 6 と同様に以下を示す．

主張 3. 2 等分割 (V_1', V_2') は，$|V_1' \cap V| = |V_2' \cap V|$ を満たす．

証明. 背理法で示す．$|V_1' \cap V| > |V_2' \cap V|$ であると仮定する．このとき，$|V_1' \cap V| \geq n+1$ である．今，(V_1', V_2') は 2 等分割であるから，$|V_1'| = 3n + n^4$ である．$|V_1' \cap V| \geq n+1$ と $|V_1'| = 3n + n^4$ から，$v \in V_1 \cap V$ のペンダント点の中で，V_1 に含まれるものが少なくとも $n^3 - 4n - 1$ 個存在する．このとき，カット辺の数は高々 $2 \cdot 4n + 2n^4 - (n^3 - 4n - 1) = 2n^4 - n^3 + 12n + 1$ となる．$k \leq 4n^2$ と $2n^4 + 2 \cdot 4n - n^2 - (2n^4 - n^3 + 12n + 1) = n^3 - n^2 - 4n - 1$ から，$n \geq 5$ のとき $2n^4 + 2 \cdot 4n - k > 2n^4 - n^3 + 12n + 1$ となる．これは，(V_1', V_2') の大きさが $2n^4 + 8n - k$ 以上であることに矛盾する．□

主張 3 より，$|V_1' \cap V| = |V_2' \cap V|$ であり，V とペンダント点の間の辺は $2n^4$ しかないので，少なくとも $8n - k$ 本のカット辺が V と V_E の間に存在する．もし，V_E において，$V_1 \cap V$ と $V_2 \cap V$ の隣接点を 1 つずつ持つ点が k より多いならば，カット数は $8n - k$ よりも少なくなる．したがって，そのような点は高々 k 個しか存在しない．今，$V_1 = V_1' \cap V$，$V_2 = V_2' \cap V$ とすると，$V_1 \cap V$ と $V_2 \cap V$ の隣接点を 1 つずつ持つ点 $v_e \in V_E$ に対して，e は G における (V_1, V_2) のカット辺である．主張 3 より $|V_1' \cap V| = |V_2' \cap V|$ なので，(V_1, V_2) は大きさ k 以下の 2 等分割である．□

定理 6，7 より，以下の系も得られる．

系 1. 最小 2 等分割問題と最大 2 等分割問題はグラフを補 2 部グラフに限っても NP 困難である．

5. ライングラフ

Guruswami [9] は，辺に重みの無いライングラフにおいて最大カット問題が多項式時間で解けることを示した．本節では，その方法が最大 2 等分割問題にも適用できることを示す．

$G = (V, E)$ をグラフとする．G の**ライングラフ**とは無向グラフ $L(G) = (V_L, E_L)$ で，$V_L = E$，$e, f \in E$ が端点を共有するとき，かつそのときに限り e, f が隣接するようなグラフのことである．Guruswami は頂点の 2 分割が最大カットとなるための十分条件を示し，任意のライングラフはその条件を満たす分割をもつことを示した．

補題 2 ([9]). $G = (V, E)$ を (ライングラフとは限らない) 無向グラフとする．$C_1, C_2, \ldots, C_k$ を辺素なクリークの集合とし，その和集合は E 全体を被覆すると仮定する．このとき，任意の $1 \leq i \leq k$ について $||A \cap C_i| - |B \cap C_i|| \leq 1$ を満たすような V の 2 分割 (A, B) が存在するとき，その (A, B) は最大カットである．

最大 2 等分割の大きさは最大カットの大きさ以下であるため，この補題の 2 分割が 2 等分割であればその最大性を保証できることがわかる．Guruswami は以下のように 2 分割を構成した．与えられたライングラフ $L(G)$ の基礎グラフを G とする．G にひとつの頂点 r を追加し，その頂点と G のすべての奇数次数の頂点と隣接させたグラフを G' とする．まず，G' が連結である場合を考えると，G' は閉じたオイラー路を持つ．G' のオイラー路をひとつ固定して，r からそのオイラー路に沿って辺を a と b で交互にラベル付けする．a のラベルがついた辺集合を A とし b のラベルがついた辺集合を B とする．このように定義した E の 2 分割 (A, B) はライングラフ $L(G)$ において補題

2 の条件を満たす．これは，G の各頂点に接続する辺集合は $L(G)$ においてクリークを成し，各頂点においてラベルが a である辺と b である辺の数の差が高々 1 であることからわかる．次に，(A, B) が 2 等分割であることを示す．G' のオイラー路において r に接続される辺を考えると，最初に通る辺と最後に通る辺を除けば，a と b のラベルを持つ辺数が同数である．また，最初の辺は必ず a である．最後の辺のラベルが a であるとき，全体の辺数は奇数であるため最終的に得られる分割は $|A| + 1 = |B|$ である．最後の辺のラベルが b であるときは，辺数が偶数かつ r に接続する辺は同数のラベルを持つため，$|A| = |B|$ である．いずれの場合も G における a と b のラベルを持つ辺数の差は高々 1 のため，(A, B) は 2 等分割である．

G' が連結でない場合にも，各連結成分で閉じたオイラーを考え，それらの辺集合を適切に分割することで，全体の辺集合を補題 2 を満たすような 2 等分割を得ることができる．

ライングラフが与えられたとき，その基礎グラフは線形時間で計算可能であり [14], [18]，その基礎グラフにおけるオイラー路も線形時間で計算可能であるため，ライングラフ上の最大 2 等分割問題は線形時間で解くことが可能である．

6. おわりに

本稿では，最大/最小 2 等分割問題に対して，グラフの幅 w の木分解が入力して与えられたとき，$O(2^w w^4 n^2)$ 時間で動作する動的計画法を与えた．これは Eiben ら [7] らの実行時間 $O(8^w w^5 n^2 \log n)$ を完全している．また，Eiben らの下界の結果と SETH から導かれる下界を用いることで，本結果の実行時間は幅 w と頂点数 n のいずれについても (条件付きで) 最適であることがわかった．また，グラフクラフを制限した場合の 2 等分割問題の容易さ/困難さは未解決であるものが多い．特に，平面グラフにおける最小 2 等分割問題の複雑性は未解決である．さらには，最大カット問題の複雑性が未解決であるグラフクラスにおいて，最大/最小 2 等分割が効率よく解けるかどうかは興味深い問題である．特に，順列グラフや区間グラフにおける最大カット問題の複雑性は長年未解決であり，2 等分割のようにさらなる制約が付与されたときに問題を簡単にし得るかどうかは未知である．

謝辞　本研究の一部は JST CREST JPMJCR1401 および JSPS 科研費 JP17H01788，JP19K21537 の助成を受けている．

参考文献

[1] H. L. Bodlaender, K. Jansen: On the complexity of the maximum cut problem. Nordic Journal of Computing 7(1): 14–31 (2000)
[2] A. Boyacı, T. Ekim, M. Shalom: A polynomial-time algorithm for the maximum cardinality cut problem in proper interval graphs. Information Processing Letters 121, 29–33 (2017)
[3] T. N. Bui, S. Chaudhuri, F. T. Leighton, M. Sipser: Graph bisection algorithms with good average case behavior. Combinatorica 7(2), 171–191 (1987)
[4] B. Courcelle, S. Olariu: Upper bounds to the clique width of graphs. Discrete Applied Mathematics 101(1–3), 77–144 (2000)
[5] M. Cygan, P. Komosa, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, S. Saurabh, M. Wahlström: Randomized contractions meet lean decompositions. arXiv:1810.06864 (2018)
[6] M. Cygan, D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, S. Saurabh: Minimum bisection if fixed-parameter tractable. SIAM Journal on Computing 48(2), 417–450 (2019)
[7] E. Eiben, D. Lokshtanov, A. E. Mouawad: Bisection of bounded treewidth graphs by convolution. In: Proceedings of ESA 2019, LIPIcs, pp. 42:1–42:11 (2019)
[8] F. V. Fomin, P. A. Golovach, D. Lokshtanov, S. Saurabh: Almost Optimal Lower Bounds for Problems Parameterized by Clique-Width. SIAM Journal on Computing 43(5), 1541–1563 (2014)
[9] V. Guruswami: Maximum cut on line and total graphs. Discrete Applied Mathematics 92(2–3), 217–221 (1999)
[10] G. Gutin, A. Yeo: Note on maximal bisection above tight lower bound. Information Processing Letters 110(21), 966-969 (2010)
[11] R. Impagliazzo, R. Paturi: On the complexity of k-sat. Journal of Computer and System Sciences 62(2), 367–375 (2001)
[12] K. Jansen, M. Karpinski, A. Lingas, E. Seidel: Polynomial time approximation Schemes for MAX-BISECTION on planar and geometric graphs. SIAM Journal on Computing 35(1), 110–119 (2005)
[13] T. Kloks: Treewidth. Computations and Approximations. LNCS, vol. 842. Springer-Verlag, Berlin (1994)
[14] P. G. H. Lehot: An optimal algorithm to detect a line graph and output its root graph. Journal of the ACM 21: 569–575 (1974)
[15] D. Lokshtanov, D. Marx, S. Saurabh: Known Algorithms on Graphs of Bounded Treewidth Are Probably Optimal. ACM Transactions on Algorithms 14(2), 13:1–13:30 (2018)
[16] M. Mnich, R. Zenklusen: Bisections above tight lower bounds: In Proceedings of WG 2012, LNCS vol. 7551, pp. 184–193 (2012)
[17] R. V. Pocai: The Complexity of SIMPLE MAX-CUT on Comparability Graphs. Electronic Notes in Discrete Mathematics 55, 161–164 (2016)
[18] N. D. Roussopoulos: A max $\{m, n\}$ algorithm for determining the graph H from its line graph G. Information Processing Letters 2(4), 108–112 (1973)