

最大出次数最小化問題の各種グラフクラスに対する計算複雑さ

朝廣雄一¹ 小野廣隆² 宮野英次³

¹九州産業大学情報科学部 〒813-8503 福岡市東区松香台 2-3-1 asahiro@is.kyusan-u.ac.jp

²九州大学大学院システム情報科学研究院 〒819-0395 福岡市西区元岡 744 ono@csce.kyushu-u.ac.jp

³九州工業大学情報工学部 〒820-8502 飯塚市川津 680-4 miyano@ces.kyutech.ac.jp

概要. 辺重みつき無向グラフが与えられ、辺の向き付けを求める問題 MMO を考える。MMO の目的は重みつき出次数の最大値を最小化することである。MMO は木に対しては \mathcal{P} 、平面二部グラフに対しては弱 \mathcal{NP} -困難、一般のグラフに対しては強 \mathcal{NP} -困難であることが、過去の研究により知られている。しかし、これらのグラフクラス間には隔たりがあり、困難さの境界が厳密でない。本稿の目的は、MMO の困難さの境界を明らかにすることであり、カクタス(木はカクタスの部分クラス)に対しては \mathcal{P} 、多重辺を持つ外平面グラフや直並列グラフに対しては弱 \mathcal{NP} -困難、平面グラフと二部グラフに対しては強 \mathcal{NP} -困難であることを示す。

Graph Classes and the Graph Orientation Problem to Minimize the Maximum Weighted Outdegree

Yuichi Asahiro¹ Hirotaka Ono² Eiji Miyano³

¹ Department of Social Information Systems, Kyushu Sangyo University, 2-3-1 Matsukadai, Higashi-ku, Fukuoka 813-8503, Japan. Email: asahiro@is.kyusan-u.ac.jp

² Department of Computer Science and Communication Engineering, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka 819-0395, Japan. Email: ono@csce.kyushu-u.ac.jp

³ Department of Systems Innovation and Informatics, Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, Fukuoka 820-8502, Japan. Email: miyano@ces.kyutech.ac.jp

Abstract. Given an undirected graph with edge weights, we are asked to find an orientation, i.e., an assignment of a direction to each edge, so as to minimize the weighted maximum outdegree in the resulted directed graph. The problem is called MMO. As previous studies, it is shown that MMO is in \mathcal{P} for trees, weak \mathcal{NP} -hard for planar bipartite graphs, and strong \mathcal{NP} -hard for general graphs. There are still gaps between those graph classes. The objective of this paper is to show tight thresholds of complexity: We show that MMO is (i) in \mathcal{P} for cactuses, (ii) weakly \mathcal{NP} -hard for multi outerplanar graphs and series-parallel graphs, and also (iii) strongly \mathcal{NP} -hard for planar graphs and bipartite graphs.

1 はじめに

グラフ $G = (V, E, w)$ を辺重みつき無向グラフとする。ここで V, E, w はそれぞれ、 G の頂点集合、辺集合、辺集合から正整数への重み関数 $w: E \rightarrow \mathbb{Z}^+$ とする。グラフ G のオリエンテーション Λ とは、任意の辺 $\{u, v\} \in E$ に対する (u, v) または (v, u) で表される向き付けの集合である。すなわち、オリエンテーション Λ により、 G に対応する有向グラフを定義できる。この有向グラフ中の頂点 u の重みつき出次数(以降、単に出次数と言う)は $\sum_{\{u, v\} \in E: (u, v) \in \Lambda} w(\{u, v\})$ で定義される。本稿では、向き付け後の有向グラフ中の

最大出次数を最小化するオリエンテーションを求めることを目的とする最大出次数最小化問題(Minimum Maximum Outdegree: MMO) について考察する。図 1 に MMO の入力とオリエンテーションの例を示す。図 1(b) においては、最大出次数は 3 であり、最適解となっている。

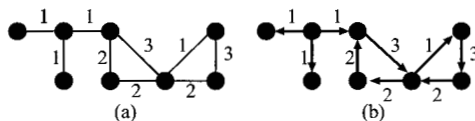


図 1: MMO の例: (a) 入力, (b) オリエンテーション.

MMO は辺の重みがすべて同一なら \mathcal{P} である [3, 9, 12] が, 一般には \mathcal{NP} -困難である [3, 1]. ただし, 入力グラフの構造を木に制限すると一般の場合でも \mathcal{P} であり [3], 入力グラフを平面グラフに制限をした場合には弱 \mathcal{NP} -困難である [1].

計算複雑さの研究としては, MMO に対して, \mathcal{P} と \mathcal{NP} -困難の境界を明らかにすることに意義がある. 本稿では, 入力グラフの構造に着目する. 図 2 は, 本稿の結果を含め MMO の計算複雑さの境界を示している. 図 2 では, 例えば, 木はカクタスの部分クラスであり, カクタスと 2 部グラフは比較不能であることを表している. \mathcal{NP} -困難性の証明に用いる帰着は, 外平面グラフに対してだけ多重グラフを用いて行われており, それ以外については単純グラフを用いて行われている. すなわち, 単純外平面グラフに対する計算複雑さは未解決である.

MMO はジョブスケジューリング問題の一つである $R||C_{\max}$ を制限した問題と考えられる. $R||C_{\max}$ では, ジョブの集合 J と計算機の集合 M , ならびにジョブ j を計算機 i の上で実行するのに必要な処理時間 p_{ij} が入力として与えられる. すべてのジョブをいずれかの計算機に割り当てるが, その際, 最も処理時間が長い計算機の処理時間を最小化する問題が $R||C_{\max}$ である. グラフにおいて, 頂点を計算機, 辺をジョブと考え, 辺の向き付けをその辺に対応するジョブを頂点对応する計算機に割り当てることとみなすと, MMO は, $R||C_{\max}$ において, 各ジョブは 2 つの計算機のいずれかにしか割り当てられず, その処理時間は両計算機において同じという制限をつけた問題と考えることができる.

一般の $R||C_{\max}$ については, 2 近似アルゴリズムと, $3/2$ 近似不可能性が示されている [10]. また, 処理時間がすべて整数という仮定のもとで, $2 - 1/k$ 近似アルゴリズムも知られている [5]. ここで k は各ジョブの処理時間の最大値である. また, [1] では, MMO に対して辺の重みが 1 と k の 2 種類しか存在しなくても (つまり, $R||C_{\max}$ においてジョブの処理時間が 1 と k の 2 種類しか存在しなくても), $3/2$ 近似不可能性が保持されることが示されている.

2 定義

グラフ $G = (V, E, w)$ を辺重み付き無向グラフとする. ここで V, E, w はそれぞれ, G の頂点集合, 辺集合, 辺集合から正整数への重み関数 $w: E \rightarrow \mathbb{Z}^+$

とする. $V(G)$ と $E(G)$ も G の頂点集合と辺集合を表す. 端点が u と v である無向辺は $\{u, v\}$ で表し, u から v への有向辺は (u, v) で表す. G のオリエンテーション (G に対するオリエンテーション) Λ とは, 任意の辺 $\{u, v\} \in E$ に対する (u, v) または (v, u) で表される向きの集合である. すなわち, G とオリエンテーション Λ により有向グラフを定義できる. $\Lambda(\{u, v\})$ により, $\{u, v\}$ の Λ による向き, (u, v) または (v, u) を表す. $d_G(v)$ は G における頂点 v の次数, すなわち $d_G(v) = |\{(u, v) \mid \{u, v\} \in E\}|$ である. G のオリエンテーション Λ による, v の出次数 $d_G^+(\Lambda, v)$ は, v から出ている辺の重みの総和:

$$d_G^+(\Lambda, v) = \sum_{\{u, v\} \in E: (v, u) \in \Lambda} w(\{u, v\})$$

で定義される. グラフ G が明らかならば, 簡単のために, $d(v)$ と $d^+(\Lambda, v)$ を $d_G(v)$ と $d_G^+(\Lambda, v)$ の代わりに用いることがある. G のオリエンテーション Λ のコストは, $\Delta_\Lambda(G) = \max_{v \in V} d_G^+(\Lambda, v)$ とする.

長さ l の道 P とは, $P = \langle v_0, v_1, v_2, \dots, v_l \rangle$ で表される頂点列である. また, 長さ l の閉路 C は, $C = \langle v_1, v_2, \dots, v_l, v_1 \rangle$ のように, 最初と最後の頂点と同じである道のことである. 本稿では閉路と言う時, 常に単純閉路を意味する. 閉路中のある頂点がその閉路以外の頂点に接続している時, それを門と呼ぶ. カクタスとは, 各辺が高々 1 個の閉路にだけ属するグラフである. 直並列グラフなどのグラフ構造の定義については, [4, 8] に従う.

問題 MMO を定義する. $S = \{1, 2, \dots, k\}$ とする.

問題: S -MINIMUM MAXIMUM OUTDEGREE (S -MMO)

入力: 無向グラフ $G = (V, E, w)$. ただし, w は辺重み関数 $w: E \rightarrow S$.

出力: $\Delta_\Lambda(G)$ を最小化するオリエンテーション Λ .

$\Delta^*(G)$ を, G の最適オリエンテーション OPT_G のコストとする. すなわち, $\Delta^*(G) = \Delta_{OPT_G}(G)$ である. 任意のオリエンテーション Λ について, $\Delta_\Lambda(G) \geq \Delta^*(G)$, つまり, $\Delta^*(G) \geq k$ が成立する [3]. 以下の性質は単純であるが, カクタスに対する多項式時間アルゴリズムの設計において重要な役割を持つ.

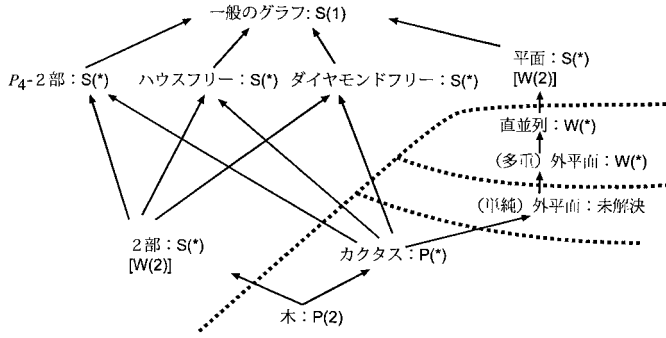


図 2: グラフクラスごとの MMO に関する計算複雑さ (W: 弱 \mathcal{NP} -困難, S: 強 \mathcal{NP} -困難, (1): [1] による結果, (2): [3] による結果, (*): 本稿に示す結果.)

主張 1 任意の頂点 v に対して, $d_G(v) \geq 2$ ならば, カクタス G には門をただか 1 個しか持たない閉路が存在する. (証明略) \square

3 カクタスに対する多項式時間アルゴリズム

本節では, カクタスに対する多項式時間アルゴリズムについて述べる. まず S -MMO を一般化した問題である, (S, T) -MINIMUM MAXIMUM OUTDEGREE ((S, T) -MMO) を 3.1 節で導入する. 3.2 節では, (S, T) -MMO の決定問題版である, (S, T) -MMO(K) を解くアルゴリズムを示す. そして, 3.3 節で (S, T) -MMO に対する多項式時間アルゴリズムを示す.

3.1 一般化した問題 (S, T) -MMO

S -MMO においては入力グラフは辺重みしか持っていないが, 頂点重みも持つグラフを入力とする問題を考えたい. まず前節で導入した記号の定義を拡張する. $G = (V, E, f, w)$ を頂点重みと辺重みを持つ無向グラフとする. ここで, V と E はそれぞれ頂点集合と辺集合を表し, f と w はそれぞれ頂点集合と辺集合から正整数への関数 $f: V \rightarrow \mathbb{Z}^+$ と $w: E \rightarrow \mathbb{Z}^+$ とする. グラフ G に対するオリエンテーション Λ による, v の出次数とは,

$$d_G^+(\Lambda, v) = f(v) + \sum_{\{u, v\} \in E: (v, u) \in \Lambda} w(\{u, v\})$$

で定義される. これら以外の各種定義は, 前の節と同様である. 問題 (S, T) -MMO を定義する:

問題: (S, T) -MINIMUM MAXIMUM OUTDEGREE ((S, T) -MMO)

入力: 無向グラフ $G = (V, E, f, w)$. ただし, f は頂点重み関数 $f: V \rightarrow T$ であり, w は辺重み関数 $w: E \rightarrow S$ である.

出力: $\Delta_\Lambda(G)$ を最小化するオリエンテーション Λ .

定理 2 $G = (V, E, f, w)$ に対して $G^0 = (V, E, w)$ と定数 c を考える. もし全ての頂点 $v \in V$ について $f(v) = c$ なら, 任意のオリエンテーション Λ に対して $\Delta_\Lambda(G) = \Delta_\Lambda(G^0) + c$ が成り立つ. (証明略) \square

この定理から次の系が得られる. よって, S -MMO に対する \mathcal{NP} -困難性に関する結果は, そのまま (S, T) -MMO に対しても適用できることが分かる.

系 3 $G = (V, E, f, w)$ を考える. もし任意の頂点 v について $f(v) = 0$ が成立しているなら, (S, T) -MMO は S -MMO と同じ問題である. (証明略) \square

全頂点に対して $f(v) = c$ が成立しているとき, $(S, \{c\})$ -MMO の代わりに (S, c) -MMO と書くことにする. 2つのグラフの組 $G = (V, E, f, w)$ と $G' = (V', E', f', w')$ に対して, $V' \subseteq V$, $E' \subseteq E$ であり, かつ 任意の辺 e について $w'(e) = w(e)$ が成立しているなら, G' を G の部分グラフと呼ぶことにする. G の部分グラフ G' がさらに, 任意の頂点 v に対して $f'(v) = f(v)$ を満たしている時に, G' を真部分グラフと呼ぶ. これ以降は, グラフ G のオリエンテー

シオン Λ は, 辺 $\{u, v\} \notin E(G)$ に対して (u, v) と (v, u) のいずれも含んでいなくてもよいものとする.

以下の4つの主張は, カクタスに対する多項式時間アルゴリズムを構築するために必要なものである.

主張 4 任意の $v \in V$ に対して $f(v) \leq f'(v)$ を満たすグラフの組 $G = (V, E, f, w)$ と $G' = (V, E, f', w)$ について $\Delta^*(G) \leq \Delta^*(G')$ が成立する. (証明略) \square

主張 5 グラフ G , ある辺 $e \in E(G)$ について $G' = G - e$ と表される G の真部分グラフ G' と, G のオリエンテーション Λ , G' のオリエンテーション $\Lambda' = \Lambda \setminus \{\Lambda(e)\}$ について, 以下が成立する: (i) $\Delta_\Lambda(G') = \Delta_{\Lambda'}(G')$, (ii) $\Delta_\Lambda(G) \geq \Delta_{\Lambda'}(G')$, (iii) $\Delta^*(G) \geq \Delta^*(G')$. (証明略) \square

主張 6 グラフ G , ある頂点 $v \in V(G)$ について $G' = G - v$ と表される G の真部分グラフ G' と G のオリエンテーション Λ , G' のオリエンテーション $\Lambda' = \Lambda \setminus \{\Lambda(\{v, u\}) \mid \{v, u\} \in E(G)\}$ について以下が成立する: (i) $\Delta_\Lambda(G') = \Delta_{\Lambda'}(G')$, (ii) $\Delta_\Lambda(G) \geq \Delta_{\Lambda'}(G')$, (iii) $\Delta^*(G) \geq \Delta^*(G')$. (証明略) \square

主張 7 $G = (V, E, f, w)$ と, 定数 K に対して $f(u) + w(e) > K$ を満たす辺 $e = \{u, v\} \in E$ を考える. もし $\Delta^*(G) \leq K$ なら $(v, u) \in OPT_G$ であり, また $G' = (V, E', f', w')$ に対して, $\Delta^*(G) = \Delta^*(G')$ が成立する. ここで, $E' = E \setminus \{e\}$, $f'(v) = f(v) + w(e)$, 任意の $x \in V \setminus \{v\}$ に対して $f'(x) = f(x)$, 任意の $e \in E'$ に対して $w'(e) = w(e)$ である. (証明略) \square

3.2 決定問題 (S, T) -MMO(K)

本節では, (S, T) -MMO に対応する以下の決定問題 (S, T) -MMO(K) に対して, 多項式時間アルゴリズムを示す. 任意のオリエンテーションのコストは, 辺の最大重み k 以上になるので, $K \geq k$ と考える.

問題: (S, T) -MMO(K)

入力: 無向グラフ $G = (V, E, f, w)$. ただし, f は頂点重み関数 $f: V \rightarrow T$ で, w は辺重み関数 $w: E \rightarrow S$ である.

問い: $\Delta_\Lambda(G) \leq K$ となるオリエンテーション Λ は存在するか?

処理 OutAll(G, Λ, v)

入力: グラフ $G = (V, E, f, w)$, (部分的)オリエンテーション Λ , 頂点 $v \in V$.

ステップ 1: 全ての $\{u, v\} \in E$ に対して (v, u) を Λ に加える.

ステップ 2: v と v に接続する辺を G から削除.

図 3: 処理 OutAll

処理 FixEdge(G, K, Λ, e)

入力: グラフ $G = (V, E, f, w)$, 定数 K , (部分的)オリエンテーション Λ , 辺 $e = \{u, v\} \in E$.

ステップ 1: もし $f(u) + w(e) > K$ なら (v, u) を Λ に追加し $f(v) = f(v) + w(e)$ とする. そうでないなら (u, v) を Λ に追加し $f(u) = f(u) + w(e)$ とする.

ステップ 2: e を削除.

図 4: 処理 FixEdge

提案するアルゴリズム AlgCactus 中で用いる3つの処理 OutAll, FixEdge と OrientCycle をまず示す. OutAll(G, Λ, v) (図 3) は, ある頂点 v に接続する全ての辺の向きを決定し, v とそれらの辺を削除する. The second procedure FixEdge(G, K, Λ, e) (図 4) は, 主張 7 に基づき, ある辺 e の向きを決定し, e を削除する. OrientCycle(G, Λ, C) (図 5) は, 門を高々1個だけ持つ閉路 C の辺の向きを決定する. 図 6 に AlgCactus の全体を示す.

図 7 に AlgCactus の動作例を示す. 図 7(a) に示すグラフと $K = 3$ が入力とし, 点線で描かれた頂点や辺はアルゴリズムにより削除されたものを表し, 箱に入った数は頂点の重みのうち0より大きいものを表している. 最初に OutAll が頂点 s に対して適用され (図 7(b)), AlgCactus の記述中の条件 (1) または (2) を満たす頂点や辺は存在しなくなる. そして OrientCycle が, 門 t を持つ閉路 C に対して適用される (図 7(c)). 次は, 辺 $\{t, u\}$ に対して FixEdge が適用される (図 7(d)). 最後に OutAll が頂点 t に適用されて, 残った辺に FixEdge が適用されて, 最

処理 OrientCycle(G, Λ, C)

入力: グラフ $G = (V, E, f, w)$, (部分的) オリエンテーション Λ , 門を高々1つしか持たない閉路 $C = \langle v_1, v_2, \dots, v_l, v_1 \rangle$.

ステップ 1: C が門を持たない場合,

(a): $(v_1, v_2), (v_2, v_3), \dots, (v_l, v_1)$ を Λ に加える.

C が 1 個の門 v_1 を持つ場合,

(b): もし $w(\{v_1, v_2\}) < w(\{v_1, v_l\})$ なら $(v_1, v_2), (v_2, v_3), \dots, (v_l, v_1)$ を Λ に追加し, $f(v_1) = f(v_1) + w(\{v_1, v_2\})$ とする.

(c): そうでないなら, $(v_1, v_l), (v_l, v_{l-1}), \dots, (v_2, v_1)$ を Λ に追加し, $f(v_1) = f(v_1) + w(\{v_1, v_l\})$ とする.

ステップ 2: C の辺と v_1 以外の頂点を削除.

図 5: 処理 OrientCycle

最終的なオリエンテーションが完成する.

補題 8 AlgCactus は (S, T) -MMO(K) に対して正しい答を出力し, $O(|V| \log |V|)$ 時間の前処理と $O(|E|)$ 時間で実行できる. (証明略) □

3.3 多項式時間アルゴリズム

最適解のコストの上限を示し, それにより $\{1, \dots, k\}$ -MMO が多項式時間で解けることを示す.

補題 9 カクタス G に対して, (S, T) -MMO の最適解 $\Delta^*(G)$ は $\Delta^*(G) \leq f_{\max} + 2w_{\max}$ を満たす. ここで f_{\max} と w_{\max} はそれぞれ頂点と辺の最大重みを表す. (証明略) □

頂点と辺の重みは整数と仮定しているので, 上の補題から (S, T) -MMO(K) を二部探索の要領で, $O(\log(f_{\max} + w_{\max}))$ 回解くことにより, (S, T) -MMO の最適解を得ることができるとわかる. また AlgCactus を異なる K に対して複数回実行するとしても, 前処理は一度だけ実行すればよい. さらに, $\{1, \dots, k\}$ -MMO は $(\{1, \dots, k\}, 0)$ -MMO と同一の問題であるから, 次の定理を得る.

アルゴリズム AlgCactus(G, K)

入力: カクタス $G = (V, E, f, w)$, 定数 K .

出力: Yes (とオリエンテーション Λ) または No.

ステップ 0: $\Lambda := \emptyset$, $G' := G$ とする.

ステップ 1: 次の条件を満たす頂点 $u \in V(G')$ があれば, OutAll(G', Λ, u) を実行.

$$f(u) + \sum_{\{u, v\} \in E(G')} w(\{u, v\}) \leq K, \quad (1)$$

ステップ 2: 条件 (2): $f(u) + w(e) > K$ を満たす辺 $e = \{u, v\} \in E(G')$ があれば, FixEdge(G', K, Λ, e) を実行.

ステップ 3: 条件 (1) または (2) を満たす頂点や辺がなくなるまでステップ 1 と 2 を繰り返す.

ステップ 4: $f(v) > K$ を満たす頂点 $v \in V(G')$ があるなら, 'No' を出力して停止.

ステップ 5: 孤立点を削除. それにより G' に頂点が無くなれば, 'Yes' (と Λ) を出力し停止.

ステップ 6: 門を高々1個しか持たない閉路 C を見付け, OrientCycle(G', Λ, C) を実行し, ステップ 1 に戻る.

図 6: アルゴリズム AlgCactus

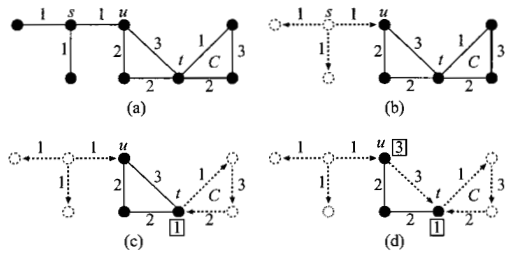


図 7: AlgCactus の動作例: (a) 入力グラフ, (b) OutAll の頂点 s への適用, (c) OrientCycle の閉路 C への適用, (d) FixEdge の辺 $\{t, u\}$ への適用.

定理 10 $\{1, \dots, k\}$ -MMO はカクタスに対して $O(|V| \log |V| + |E| \log k)$ 時間で解ける. (証明略) □

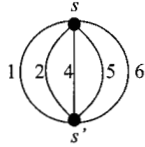


図 8: 定理 11 の証明.

4 \mathcal{NP} -困難性

$\{1, \dots, k\}$ -MMO の \mathcal{NP} -困難性を、各種グラフクラス: 外平面, 直並列, 平面, 二部, P_4 -二部, ダイヤモンドフリー, ハウスフリーについて示す. 紙面の都合で, 帰着の部分だけ述べ証明の詳細は省略する.

定理 11 $\{1, \dots, k\}$ -MMO は, (多重)外平面グラフに対して, 弱 \mathcal{NP} -困難である.

証明の概略: 弱 \mathcal{NP} -困難問題である PARTITION[6] から多項式時間帰着する. PARTITION とは, 正整数の集合 $S = \{s_1, s_2, \dots, s_n\}$ が与えられ, $\sum_{s_i \in S'} s_i = \sum_{s_i \in S \setminus S'} s_i$ となる部分集合 $S' \subseteq S$ が存在するかどうかを問う問題である.

PARTITION のインスタンス $S = \{s_1, s_2, \dots, s_n\}$ から辺重みつきグラフ $G = (V, E, w)$ を作る. 頂点集合を $V = \{s, s'\}$ とする. 辺集合は, 頂点 s, s' 間の n 本の多重辺 e_1, e_2, \dots, e_n とする. ここで, 各辺 e_i の重み $w(e_i) = s_i$ とする. このようにして作られるグラフは外平面グラフであり, この帰着は多項式時間で実行できる. 図 8 は例として $S = \{1, 2, 4, 5, 6\}$ の場合に作られるグラフを示している.

$W = \sum_{s_i \in S} s_i / 2$ とすると, $\sum_{s_i \in S'} s_i = W$ となる部分集合 $S' \subseteq S$ が存在する時に限り, 最大次数が W 以下になるオリエンテーションが存在することを証明できる. (詳細は省略) \square

図 8 のグラフは直並列グラフでもあるので, 多重辺を持つ直並列グラフに対しての \mathcal{NP} -困難性は, 上の定理同様に成り立つ. 次の定理は, 多重辺を持たない直並列グラフに対する \mathcal{NP} -困難性を示すもので, 再度 PARTITION からの多項式時間帰着による.

定理 12 $\{1, \dots, k\}$ -MMO は, 単純直並列グラフに対して 弱 \mathcal{NP} -困難である.

証明の概略: PARTITION のインスタンス $S = \{s_1, s_2, \dots, s_n\}$ から辺重みつきグラフ $G = (V, E, w)$

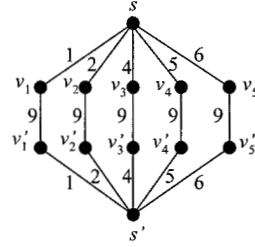


図 9: 定理 12 の証明.

を作る. 頂点集合 V は, (i) 頂点 s と s' , ならびに (ii) 各 s_i に対応して v_i と v'_i を含む. 頂点数は $2n+2$ となる. ここで, $W = \sum_{s_i \in S} s_i / 2$ と定義する. 辺集合 E は 各 $1 \leq i \leq n$ について, $\{s, v_i\}$, $\{v_i, v'_i\}$, $\{v'_i, s'\}$ からなる $3n$ 本の辺を含む. 各辺の重みは, $w(\{s, v_i\}) = w(\{v'_i, s'\}) = s_i$, $w(\{v_i, v'_i\}) = W$ とする. このようにして作られるグラフは直並列グラフであり, この帰着は多項式時間で実行できる. 図 9 は例として $S = \{1, 2, 4, 5, 6\}$ の場合に作られるグラフを示している. この帰着により, $\sum_{s_i \in S'} s_i = W$ となる部分集合 $S' \subseteq S$ が存在する時に限り, 最大次数が W 以下になるオリエンテーションが存在することを証明できる. (詳細は省略) \square

次に, 二部グラフと平面グラフに対する, $\{1, k\}$ -MMO の強 \mathcal{NP} -困難性について述べる. いずれの証明も, SAT の一種からの多項式時間帰着による. SAT とは, 論理変数の集合 $U = \{x_1, \dots, x_n\}$ と CNF 式 $\phi = \bigwedge_{c_i \in C} c_i$ が与えられ, ϕ を true にする変数割り当てが存在するかどうかを問う問題である. ここで C は U 上の項の集合である.

実際に証明に用いる SAT を変形した問題について述べる. At-Most-3SAT(2L) は SAT に対して, 各項は高々 3 個のリテラルしか含まず, 各リテラルは高々 2 回しか ϕ に出現しないと制限をつけた問題である. At-Most-3SAT(2L) の強 \mathcal{NP} -困難性は, [6] の p.259 に記載の問題 [LO1] からの帰着で示される. また, 各項が正リテラルか負リテラルのいずれかしか含まない時, CNF 式は単調であると言う. 入力を単調 CNF 式に限定した SAT も強 \mathcal{NP} -困難であり [7], さらに At-Most-3SAT(2L) の入力を単調 CNF 式に限定しても, 強 \mathcal{NP} -困難である [2]. 次に平面 3SAT について述べる. CNF 式 ϕ に対して, $V = U \cup C$ を頂点集合とするグラフ $G(\phi) = (V, E)$ を考える. ここで, 辺集合 E は, 項 c がリテラル x

または \bar{x} を含む場合に, $\{x, c\}$ を含むとする. このグラフ $G(\phi)$ が平面グラフの場合に, ϕ が平面であるという. 平面 3SAT とは, 入力される CNF 式が平面で, かつ各項はリテラルを高々 3 個ずつ含むという制限をつけた SAT で, 強 \mathcal{NP} -困難である [11].

定理 13 任意の $k \geq 2$ に対して, 入力を二部グラフに限定した $\{1, k\}$ -MMO は 強 \mathcal{NP} -困難である.

証明の概略: Monotone At-Most-3SAT(2L) からの多項式時間帰着により証明する. Monotone At-Most-3SAT(2L) のインスタンスとして, n 変数 $\{x_1, \dots, x_n\}$ と m 項 $\{c_1, \dots, c_m\}$ からなる CNF 式 ϕ が与えられたとする. ここで, 正 (負) リテラルしか含まない項を正項 (負項) と呼ぶことにする.

ϕ からグラフ G_ϕ を構成する (図 10). ガジェットとして, (a) 変数に対応するものと (b) 項に対応するもの, さらに (c) 特別ガジェットを用意する. (a) 各変数ガジェットはそれぞれリテラル x_i と \bar{x}_i に対応する 2 頂点 x_i と \bar{x}_i と, 重さ k の辺 $\{x_i, \bar{x}_i\}$ からなる. (b) 項ガジェットは, 項 c_j に対応する頂点 c_j であり, 項 c_j に含まれるリテラルに対応する頂点と重さ 1 の辺で接続する. (c) 特別ガジェットは $2k$ 個の頂点 s_1, s_2, \dots, s_{2k} からなる閉路であり, 各辺の重さは k である. 正項 (負項) が i 変数を含んでいる場合には, その項に対応する頂点と特別ガジェット中の $s_1, s_3, \dots, s_{2(k-i)-1}$ ($s_2, s_4, \dots, s_{2(k-i+1)}$) を接続する. 以上の構成により, 各頂点の次数はすべて $k+1$ であり, またグラフ G_ϕ は二部グラフになっている. また以上の帰着は, 多項式時間でできる.

G_ϕ に対して, もし ϕ が充足可能なら $\Delta^*(G_\phi) \leq k$ であり, もし ϕ が充足不能なら $\Delta^*(G_\phi) \geq k+1$ であることを証明できる. (詳細は省略) \square

定理 13 からすぐに, 次の系を得る.

系 14 入力を二部グラフに限定しても, $\mathcal{P} = \mathcal{NP}$ でない限り, $\{1, k\}$ -MMO には近似度が $1+1/k$ より小さい擬多項式時間アルゴリズムは存在しない. \square

また, 二部グラフは, ハウスフリー, ダイヤモンドフリー, P_4 -二部でもあるので, 次の系も言える.

系 15 $\{1, k\}$ -MMO は, ハウスフリー, ダイヤモンドフリー, P_4 -二部の各グラフクラスに対しても, 強 \mathcal{NP} -困難である. さらに, $\mathcal{P} = \mathcal{NP}$ でない限り, そ

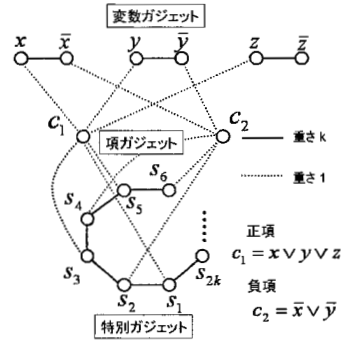


図 10: 定理 13 の証明.

これらのグラフクラスに対して近似度が $1+1/k$ より小さい擬多項式時間アルゴリズムは存在しない. \square

定理 16 任意の $k \geq 2$ に対して, 入力を平面グラフに限定した $\{1, k\}$ -MMO は 強 \mathcal{NP} -困難である.

証明の概略: 基本的な方針は定理 13 の証明と同じであるが, 平面 3SAT から帰着する.

平面 3SAT のインスタンス ϕ とその平面描画が与えられたとする. ϕ に対して, (a) 変数ガジェット, (b) 項ガジェット, (c) 特別ガジェットからなるグラフ G'_ϕ を構成する. (a) 変数 x に対応する変数ガジェットは, $3l$ 個の頂点と $3l$ 個の辺を持つ. ここで l は ϕ 中の x の出現回数である. 簡単のために, 変数 x は, 項 c_1, c_2, \dots, c_l に出現し, 与えられた ϕ の平面描画中でもこれらの項はこの順に描かれているとする. (図 11 の上図). $x^{(i)}$ と $\bar{x}^{(i)}$ のラベルのついた頂点を $2l$ 個と, $d^{(l)}$ とラベルのついた頂点を l 個用意する. これらのラベルは, 項 c_i の順番に対応する. これらの頂点間に, 重さ k の辺 $\{x^{(i)}, \bar{x}^{(i)}\}$ と, 重さ 1 の辺 $\{\bar{x}^{(i)}, d^{(i)}\}$ と, 重さ 1 の辺 $\{d^{(i)}, x^{(i+1)}\}$ を $i = 1, 2, \dots, l$ ($l+1 \equiv 1$) に対して用意する. この変数ガジェット自体は平面グラフである. (b) ϕ の項 c_j に対応して頂点 c_j を項ガジェットとして用意する. 変数 x が c_1, c_2, \dots, c_l に出現するとする. もし, リテラル $x(x)$ が項 c_j に出現していたら, 重さ 1 の辺 $\{c_j, x^{(j)}\}$ ($\{c_j, \bar{x}^{(j)}\}$) を作る (図 11 の下図). (c) 特別ガジェットは, $k+1$ 頂点と, 重さ k の $k+1$ 辺からなる閉路である. 各項と, 変数ガジェット中の各頂点 $d^{(i)}$ に対して 1 つずつ特別ガジェットを用意する. ある項が 1 (2 または 3) 個の変数を含んで

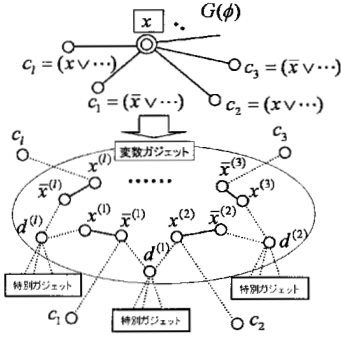


図 11: 定理 16 の証明.

いるならば、その項用の特別ガジェット中の $k(k-1)$ または $k-2$ 個の頂点に、頂点 c_j を重さ 1 の辺で接続する。各頂点 $d^{(i)}$ については、その頂点用の特別ガジェット中の $k-1$ 頂点に重さ 1 の辺で接続する。よって各 $d^{(i)}$ の次数は正確に $k+1$ となる。

以上の構成により、 G'_ϕ は平面グラフとなる。なぜならば、 ϕ が平面であるので $G(\phi)$ は平面グラフであり、かつ G'_ϕ は $G(\phi)$ の変数ガジェットを、平面グラフに置き換えたものだからである。グラフ G'_ϕ に対して、もし ϕ が充足可能なら $\Delta^*(G'_\phi) \leq k$ であり、もし ϕ が充足不能ならば $\Delta^*(G'_\phi) \geq k+1$ であることを証明できる。(詳細は省略) □

以上の定理 16 の証明より、下記の系も得られる。

系 17 入力平面グラフに限定しても、 $\mathcal{P} = \mathcal{NP}$ でない限り、 $\{1, k\}$ -MMO には近似度が $1+1/k$ より小さい擬多項式時間アルゴリズムは存在しない。□

5 おわりに

本稿では各種グラフクラスに対する MMO の計算複雑さを明らかにした。紙面の都合で証明の大部分を省略したので、詳細は [2] を参照してほしい。

謝辞

本研究の一部は科研費 (16092222 16092223, 17700022, 18700014, 18700015) の助成を受けたものである。

参考文献

[1] Asahiro, Y., Jansson, J., Miyano, E., Ono, H., and Zenmyo, K. Approximation algorithms

for the graph orientation minimizing the maximum Weighted outdegree, *Lecture Notes in Comp. Sci.*, Vol. 4508, pp. 167–177, 2007.

[2] Asahiro, Y., Miyano, E., Ono, H. Graph classes and the complexity of the graph orientation minimizing the maximum outdegree. *Conferences in Research and Practice in Information Technology*, Vol. 77, pp. 97–106, 2008.

[3] Asahiro, Y., Miyano, E., Ono, H., and Zenmyo, K. Graph orientation algorithms to minimize the maximum outdegree, *Int. J. Foundations of Comp. Sci.*, **18**(2), pp. 197–215, 2007.

[4] Brandstädt, A., BangLe, V., and Spinrad, J. P. *Graph Classes: A Survey*, SIAM, 1987.

[5] Gairing, M., Lüicking, T., Mavronicolas, M., and Monien, B. Computing Nash equilibria for scheduling on restricted parallel links, *Proc. 36th ACM STOC*, pp. 613–622, 2004.

[6] Garey, M., and Johnson, D. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Co., 1979.

[7] Gold, E. M. Complexity of automaton identification from given data, *Information and Control*, **37**(3), pp. 302–320, 1978.

[8] Gross, J. L., and Yellen, J.(eds) *Handbook of Graph Theory*, CRC Press, 2004.

[9] Kowalik, L. Approximation scheme for lowest outdegree orientation and graph density measures, *Lecture Notes in Comp. Sci.*, Vol.4288, pp. 557–566, 2006.

[10] Lenstra, J. K., Shmoys, D. B., and Tardos, É. Approximation algorithms for scheduling unrelated parallel machines, *Mathematical Programming*, **46**(3), pp. 259–271, 1990.

[11] Lichtenstein, D. Planar formulae and their uses, *SIAM Journal on Computing*, **11**(2), pp. 329–343, 1982.

[12] Venkateswaran, V. Minimizing maximum in-degree, *Discrete Applied Mathematics*, **143**(1–3), pp. 374–378, 2004.