European Journal of Combinatorics 83 (2020) 103015

Contents lists available at ScienceDirect

European Journal of Combinatorics

journal homepage: www.elsevier.com/locate/ejc

Partitioning a graph into degenerate subgraphs

Faisal N. Abu-Khzam [a], Carl Feghali [b], Pinar Heggernes [b]

[a] Department of Computer Science and Mathematics, Lebanese American University, Beirut, Lebanon
[b] Department of Informatics, University of Bergen, Bergen, Norway

ARTICLE INFO

Article history:
Received 12 April 2018
Accepted 26 August 2019
Available online 3 September 2019

ABSTRACT

Let $G = (V, E)$ be a connected graph with maximum degree $k \geq 3$ distinct from K_{k+1}. Given integers $s \geq 2$ and $p_1, \ldots, p_s \geq 0$, G is said to be $(p_1, \ldots, p_s)$-partitionable if there exists a partition of V into sets $V_1, \ldots, V_s$ such that $G[V_i]$ is p_i-degenerate for $i \in \{1, \ldots, s\}$. In this paper, we prove that we can find a $(p_1, \ldots, p_s)$-partition of G in $O(|V| + |E|)$-time whenever $1 \geq p_1, \ldots, p_s \geq 0$ and $\sum_{i=1}^{s} p_i \geq k - s$. This generalizes a result of Bonamy et al. (2017) and can be viewed as an algorithmic extension of Brooks' Theorem and several results on vertex arboricity of graphs of bounded maximum degree. We also prove that deciding whether G is (p, q)-partitionable is $\mathbb{NP}$-complete for every $k \geq 5$ and pairs of non-negative integers (p, q) such that $(p, q) \neq (1, 1)$ and $p + q = k - 3$. This resolves an open problem of Bonamy et al. (2017). Combined with results of Borodin et al. (2000), Yang and Yuan (2006) and Wu et al. (1996), it also settles the complexity of deciding whether a graph with bounded maximum degree can be partitioned into two subgraphs of prescribed degeneracy.

1. Introduction

The concept of degenerate graphs introduced by Lick and White [12] in 1970 has since found a number of applications in graph theory, especially in graph partitioning and graph colouring problems. This is mainly because the class of degenerate graphs captures one of the earliest studied classes of graphs such as independent sets, forests and planar graphs. For example, results of

E-mail addresses: faisal.abukhzam@lau.edu.lb (F.N. Abu-Khzam), carl.feghali@uib.no (C. Feghali), pinar.heggernes@uib.no (P. Heggernes).

https://doi.org/10.1016/j.ejc.2019.103015

Thomassen [16,15] on decomposing the vertex set of a planar graph into degenerate subgraphs have lead to new proofs of the 5-colour theorem on planar graphs that do not use Euler's formula. Another example is a result of Alon, Kahn and Seymour [1] that extends the well-known Turán's theorem on the size of the largest independent set in a graph to the size of largest subgraph of any prescribed degeneracy.

In this paper, we shall investigate the complexity of partitioning the vertex set of a graph of bounded maximum degree into degenerate subgraphs. In order to make this statement more precise, we must first proceed with some definitions. Let $G = (V, E)$ be a graph, and let k be a non-negative integer. We say that G is k-degenerate if we can successively delete vertices of degree at most k in G until the empty graph is obtained. Expressed in an another way, G is k-degenerate if it admits a *k-degenerate ordering* — an ordering $x_1, \dots, x_n$ of the vertices in G such that x_i has at most k neighbours x_j in G with $j < i$. In this case, the ordering is said to *start at* x_1 and *end at* x_n.

Given integers $s \geq 2$ and $p_1, \dots, p_s \geq 0$, G is said to be *$(p_1, \dots, p_s)$-partitionable* if there exists a partition of V into sets $V_1, \dots, V_s$ such that $G[V_i]$ is p_i-degenerate for $i \in \{1, \dots, s\}$.

We shall consider the following computational problem.

Problem 1. Given a graph G and integers $s \geq 2$, $p_1, \dots, p_s \geq 0$, determine the complexity of deciding whether G is $(p_1, \dots, p_s)$-partitionable.

We briefly review some existing results related to Problem 1. Let $G = (V, E)$ be a connected graph with maximum degree $k \geq 3$ distinct from K_{k+1}, and let d be a non-negative integer. A d-colouring of G is a function $f : V \to \{1, \dots, d\}$ such that $f(u) \neq f(v)$ whenever $(u, v) \in E$. Equivalently, f is a d-colouring of G if $f^{-1}(1), \dots, f^{-1}(d)$ each forms an independent set. The earliest result on Problem 1 is most likely the celebrated theorem of Brooks [6], which states that G has a d-colouring for each $d \geq k$. Thus, given that an independent set is a 0-degenerate graph, Brooks' Theorem can be reformulated in the language of Problem 1 to state that G is $(p_1, p_2, \dots, p_s)$-partitionable for every $s \geq k$ and $p_1 = \cdots = p_s = 0$. Later on, Borodin, Kostochka and Toft [5] obtained a generalization of Brooks' theorem by showing that G remains $(p_1, \dots, p_s)$-partitionable for every $s \geq 2$ and $\sum_{i=1}^{s} p_i \geq k - s$. Observe that this result is algorithmic: Given a graph G of maximum degree $k \geq 3$ and integers $s \geq 2$ and $p_1, \dots, p_s \geq 0$ such that $\sum_{i=1}^{s} p_i \geq k - s$, one can check in polynomial time if G is $(p_1, \dots, p_s)$-partitionable, because the only computation needed is to verify whether G is isomorphic to K_{k+1}. The question remains, however, whether one can *find* such a partition efficiently whenever it exists. In this direction, Bonamy et al. [3] have already considered the case $s = 2$ with $p_1 = 0$ and $p_1 + p_2 \geq k - 2$ by showing that one can find the partition in $O(n + m)$-time if $k = 3$ and in $O(kn^2)$-time if $k \geq 4$. In the first part of this paper, we generalize the first of these two results in the following theorem.

Theorem 1.1. *Let G be a connected graph with maximum degree $k \geq 3$ distinct from K_{k+1}. For every $s \geq 2$ and $1 \geq p_1, \dots, p_s \geq 0$ such that $\sum_{i=1}^{s} p_i \geq k - s$, a $(p_1, \dots, p_s)$-partition of G can be found in $O(n + m)$-time.*

The proof of Theorem 1.1 appears in Section 2. Let us note by our earlier discussion that Theorem 1.1 can be viewed as an algorithmic extension of Brooks' theorem. (In fact, our approach differs from [3] but is instead a refinement of Lovász' proof [13] of Brooks' Theorem — see [2] for an algorithmic analysis of [13].) We also remark that since the definition of forests coincides with the definition of 1-degenerate graphs, Theorem 1.1 can be viewed as an algorithmic counterpart to several results on vertex arboricity of graphs; see [7,11] for some examples.

On a different track, one might also ask what happens if the maximum degree of the graph exceeds $s + \sum_{i=1}^{s} p_i$. In this direction, Yang and Yuan [18] have shown that the case $s = 2$ with $p_1 = 0$ and $p_2 = 1$ is $\mathbb{NP}$-complete for every $k \geq 4$. Wu, Yuan and Zhao [17] have also shown that the case $s = 2$ with $p_1 = p_2 = 1$ is $\mathbb{NP}$-complete for every $k \geq 5$. Extending Yang and Yuan's result, Bonamy et al. [4] have shown that the case $s = 2$ with $p_1 = 0$ and $p_2 = t - 2$ remains $\mathbb{NP}$-complete for every $t \geq 3$ and $k \geq 2t - 2$. They then posed as an open problem the case $s = 2$ with $p_1 = 0$ and $p_2 = k - 3$ for every $k \geq 5$. In the second part of this paper, we resolve this problem by proving, more generally, the following theorem.

Theorem 1.2. *For every integer $k \geq 5$ and pairs of non-negative integers (p, q) such that $(p, q) \neq (1, 1)$ and $p + q = k - 3$, deciding whether a graph with maximum degree k is (p, q)-partitionable is $\mathbb{NP}$-complete.*

The proof of Theorem 1.2 appears in Section 3. Let us note that finding the least integer d such that a graph with maximum degree 3 is d-colourable can be done in polynomial time. Indeed, one can check in polynomial time if $d = 1$ or $d = 2$ (for any arbitrary graph). If this is not the case, we check in polynomial time if the graph is isomorphic to K_4; if not, then we know by Brooks' theorem that $d = 3$. Thus, combined with the aforementioned results in [5,18,17], Theorem 1.2 settles the complexity of deciding whether a graph with bounded maximum degree can be partitioned into two subgraphs of prescribed degeneracy. More formally, we now have the following solution to the case $s = 2$ of Problem 1.

Corollary 1.3. *Given integers $p, q \geq 0$, deciding whether a graph with maximum degree $k \geq 3$ is (p, q)-partitionable is*

(i) polynomial time solvable if $k = 3$ or $p + q \geq k - 2$ or $p = q = 0$;
(ii) $\mathbb{NP}$-complete otherwise.

2. A linear time algorithm

In this section, we prove Theorem 1.1. First, we need some standard definitions.

Let k be a non-negative integer, and let G be a graph with maximum degree k. Then G is said to be k-regular if every vertex of G has degree exactly k. A vertex v of G is called a *cut vertex* of G if $G - \{v\}$ has more components than G. A *block* of G is either K_2 or a maximal 2-connected subgraph of G, and an *end block* of G is a block of G that contains exactly one cut vertex of G. A *forest partition* of G is a partition of V into $k/2$ forests if k is even and $\lfloor k/2 \rfloor$ forests and one independent set if k is odd.

Lemma 2.1. *Let G be a graph with maximum degree $k \geq 3$. If G has a forest partition that can be found in $O(n + m)$ time, then G has a $(p_1, \dots, p_s)$-partition for every $1 \geq p_1, \dots, p_s \geq 0$ such that $\sum_{i=1}^{s} p_i \geq k - s$ that can be found in $O(n + m)$ time.*

Proof. Suppose k is even (the case k is odd is entirely similar) and let $p_1, \dots, p_s$ be integers such that $1 \geq p_1, \dots, p_s \geq 0$.

Case 1 $\sum_{i=1}^{s} p_i = k - s$. Let $\sigma = \sum_{i=1}^{s}(1 - p_i)$, then σ is even. Let $\mathcal{F}$ be a forest partition of G, and partition $\sigma/2$ of the forests in $\mathcal{F}$ into two independent sets, which can be done in $O(n + m)$ time. The resulting decomposition is a $(q_1, \dots, q_t)$-partition of G, where $1 \geq q_1, \dots, q_t \geq 0$. Since $t = |\mathcal{F}| + \sigma/2 = (k + \sigma)/2 = s$ and $\sum_{i=1}^{s=t} q_i = k/2 - \sigma/2 = k - s$, this $(q_1, \dots, q_t)$-partition of G is also a $(p_1, \dots, p_s)$-partition of G. Case 1 is complete.

Case 2 $\sum_{i=1}^{s} p_i > k - s$. Let $q_1, \dots, q_s$ be integers such that $0 \leq q_i \leq p_i$ for $1 \leq i \leq s$ and such that $\sum_{i=1}^{s} q_i = k - s$. By Case 1, a $(q_1, \dots, q_s)$-partition of G can be found in $O(n + m)$ time. This partition is also trivially a $(p_1, \dots, p_s)$-partition of G. This completes Case 2 and hence the proof of the lemma. □

To prove Theorem 1.1, it suffices to show by Lemma 2.1 that for every connected graph G with maximum degree $k \geq 3$ distinct from K_{k+1}, a forest partition of G can be found in $O(n+m)$ time. We will need the next two lemmas. The proof of the first lemma is essentially the same as the proof of [8, Lemma 8] but with some minor adjustments.

Lemma 2.2. *Let G be a connected graph with maximum degree $k \geq 3$ distinct from K_{k+1}. If G is not k-regular, then a forest partition of G can be found in $O(n + m)$ time.*

Proof. Since G is connected and not k-regular, it is $(k-1)$-degenerate. Let us first compute a $(k-1)$-degenerate ordering of the vertices of G in $O(n+m)$ time as follows. We find a vertex v of degree at most $k-1$. Note that every neighbour of v has degree at most $k-1$ in $G-\{v\}$. So the recursive algorithm that consists of first deleting v and then, for each vertex deleted, deleting all of its neighbours until the empty graph is obtained gives a $(k-1)$-degenerate ordering $v_1, v_2, \ldots, v_n$ of G in $O(n+m)$ time.

Let us now proceed to find a forest partition of G in $O(n+m)$ time.

Case 1: k is even For $i = 1, \ldots, n$, define $X_i = \{v_1, \ldots, v_i\}$. By definition, v_i has at most $k-1$ neighbours in X_{i-1}. Let $r = \frac{k}{2}$. It suffices to show that, for $2 \le i \le n$, we can compute in $O(1)$ time a partition $\{Y_1, \ldots, Y_r\}$ of X_i, where $G[Y_s]$ is 1-degenerate for $s = 1, \ldots, r$, if we have as input such a partition of X_{i-1}. We note first that finding a partition of X_1 is trivial. Suppose $i > 1$ and let $\{Z_1, \ldots, Z_r\}$ be a partition of X_{i-1} where $G[Z_s]$ is 1-degenerate for $s = 1, \ldots, r$. If v_i has more than one neighbour in every $G[Z_s]$, then v_i has at least $\sum_{i=1}^{r} 2 = k$ neighbours in X_{i-1}, a contradiction. Hence, v_i has at most one neighbour in at least one set Z_q, which we can find in $O(1)$ time since we only need to check the neighbours of v_i in X_{i-1}. We put v_i into Z_q to get the desired partition for X_i in $O(1)$ time.

Case 2: k is odd For $i = 1, \ldots, n$, define $X_i = \{v_1, \ldots, v_i\}$. By definition, v_i has at most $k-1$ neighbours in X_{i-1}. Let $r = \lceil \frac{k}{2} \rceil$. It suffices to show that, for $2 \le i \le n$, we can compute in $O(1)$ time a partition $\{Y_1, \ldots, Y_r\}$ of X_i, where $G[Y_1]$ is an independent set and $G[Y_s]$ is 1-degenerate for $s = 2, \ldots, r$, if we have as input such a partition of X_{i-1}. We note first that finding a partition of X_1 is trivial. Suppose $i > 1$ and let $\{Z_1, \ldots, Z_r\}$ be a partition of X_{i-1} where $G[Z_1]$ is an independent set and $G[Z_s]$ is 1-degenerate for $s = 2, \ldots, r$. If v_i has at least one neighbour in $G[Z_1]$ and more than one neighbour in every other $G[Z_s]$, then v_i has at least $1 + \sum_{i=2}^{r} 2 = k$ neighbours in X_{i-1}, a contradiction. Hence, v_i has either no neighbour in Z_1 or at most one neighbour in at least one other set Z_q, which we can find in $O(1)$ time since we only need to check the neighbours of v_i in X_{i-1}. We put v_i into this set to get the desired partition for X_i in $O(1)$ time. □

A pair of vertices x, y in a connected graph G is called an *eligible pair* if x and y are at distance exactly two in G and $G-\{x, y\}$ is connected. The proof of the next lemma makes use of the following result of Lovász.

Lemma 2.3 ([13]). *Let G be a 2-connected graph that is not complete or a cycle. Then an eligible pair of G can be found in $O(n+m)$ time.*

Lemma 2.4. *Let $k \ge 3$, and let $G \ne K_{k+1}$ be a 2-connected k-regular graph. Then a forest partition of G can be found in $O(n+m)$ time.*

Proof. By Lemma 2.3, we can find in $O(n+m)$ time an eligible pair of vertices x, y in G. So there is a common neighbour of x and y in G that we denote v. Let G' be the graph obtained from G by identifying x and y into a new vertex z, and let $z_1, \ldots, z_t$ denote the neighbours of z distinct from v that are common neighbours of x and y in G.

Claim 1. *There is a $(k-1)$-degenerate ordering of G' that starts at z that can be found in $O(n+m)$ time such that each z_i has at most $k-2$ neighbours earlier in the ordering.*

The proof of the claim is almost entirely contained in the proof [9, Lemma 9], but we repeat it for completeness. We shall prove Claim 1 by successively deleting vertices of G' such that the earlier a vertex is deleted, the later it occurs in the ordering.

The ordering starts at z and ends at v (note that v has degree $k-1$ in G'). The order of deletion of the remaining vertices is determined as follows. Since, by definition of an eligible pair, the graph $G^* = G' - \{z\}$ is connected, each neighbour of z distinct from v is joined to v via a path in G'. We consider each such path (in an arbitrary order) and delete each (remaining) vertex of the path distinct from v in the order in which it is encountered, if one traverses the path from the neighbour of v towards the neighbour of z on the path. Each vertex has degree at most $k-1$ at the time it is deleted and each z_i degree at most $k-2$. At this stage, we are left with a graph whose components are $(k-1)$-degenerate. Then simply successively delete the remaining vertices distinct from z of degree at most $k-1$ in this graph. The claim is proved.

Let us now find a forest partition $\mathcal{F}'$ of G' in $O(n+m)$ time with the property that z and z_i belong to different forests for each $i = 1, \dots, t$. Define the sets X_i, Y_i and Z_i as in the proof of Lemma 2.2. We put $z \in Z_1$. Note that each z_i has at most one neighbour in at least one Z_q for some $q \geq 2$ since otherwise z_i has at least $k-1$ neighbours in X_{i-1}, which contradicts Claim 1. Thus, if we put $z_i \in Z_q$, we get $\mathcal{F}'$.

To complete the proof, since every common neighbour of x and y in G is not a member of Z_1, the graph $Z_1' = Z_1 \cup \{x, y\} \setminus \{z\}$ is also a forest and, if k is odd, can be insured to be an independent set. Therefore, $\mathcal{F} = (\mathcal{F}' \setminus Z_1) \cup Z_1'$ is a forest partition of G. □

We are now able to finish the proof of Theorem 1.1.

Proof of Theorem 1.1. By Lemma 2.1, it suffices to show that we can find a forest partition of G in $O(n+m)$ time. We first check in $O(n+m)$ time whether G is k-regular. If G is not k-regular, we apply Lemma 2.2. If G is k-regular, we compute in $O(n+m)$ time a block decomposition of G (by using, for example, a depth-first search algorithm). If G is 2-connected (that is, G contains exactly one block), we can apply Lemma 2.4.

So we can assume that G is a connected k-regular graph and not 2-connected. We consider an end block B of G, and let v be the cut vertex of G that is contained in B. Let $G' = G - B$, and let $B' = B - \{v\}$. Note that G' and B' are not k-regular. Applying Lemma 2.2, we find a forest partition $\mathcal{F}'$ of G' and a forest partition $\mathcal{F}''$ of B'.

Two cases arise.

Case 1 There exist a forest $F' \in \mathcal{F}'$ and a forest $F'' \in \mathcal{F}''$ such that both F' and F'' contain at least one neighbour of v. In this case, we pair off

- F' with F'',
- the forests in $\mathcal{F}' \setminus F'$ with the forests in $\mathcal{F}'' \setminus F''$ arbitrarily, and
- the independent set in $\mathcal{F}'$ with the independent set in $\mathcal{F}''$ (if k is odd).

This yields a forest partition of $G - \{v\}$ that we denote $\mathcal{F}^*$. If F' and F'' each contains exactly one neighbour of v, then $\mathcal{F} = (\mathcal{F}^* \setminus (F' \cup F'')) \cup (F' \cup F'' \cup \{v\})$ is a forest partition of G (that can be found in $O(n+m)$ time). So we can assume that $F' \cup F''$ contains at least three neighbours of v. Suppose that k is even. Since $|\mathcal{F}^*| = k/2$ and v has degree exactly k, by the pigeonhole principle there exists a forest $F^* \in \mathcal{F}^* \setminus (F' \cup F'')$ that contains at most one neighbour of v. Hence $\mathcal{F} = (\mathcal{F}^* \setminus F^*) \cup (F^* \cup \{v\})$ is a forest partition of G. Similarly, if k is odd, then $\mathcal{F}^*$ contains either a forest that contains at most one neighbour of v or an independent set that does not contain a neighbour of v. In either case, a forest partition of G can be found. This completes Case 1.

Case 2 k is odd, the independent set $I' \in \mathcal{F}'$ and the independent set $I'' \in \mathcal{F}''$ together contain at least two neighbours of v. In this case, we pair off

- I' with I'' and
- the forests in $\mathcal{F}'$ with the forests in $\mathcal{F}''$ arbitrarily.

This yields a forest partition of $G - \{v\}$ that we denote $\mathcal{F}^*$. Since $|\mathcal{F}^*| = \lceil \frac{k}{2} \rceil$ and v has degree exactly k, there must be a forest $F^* \in \mathcal{F}^*$ that contains at most one neighbour of v. Thus $\mathcal{F} = (\mathcal{F}^* \setminus F^*) \cup (F^* \cup \{v\}$ is a forest partition of G. This completes Case 2.

From Cases 1 and 2, the one outstanding case to complete the proof of the theorem is when k is odd and v has:

- precisely one neighbour in G' that also belongs to the independent set in $\mathcal{F}'$, and
- no neighbour in B' that belongs to the independent set in $\mathcal{F}''$.

In the remainder of the proof, we shall circumvent the second bullet point by constructing a new forest partition of B' in $O(n+m)$ time whose independent set contains at least one neighbour of v. The following claim will be essential.

Claim 2. *There is a $(k-1)$-degenerate ordering of the vertices of B' that starts at some neighbour w of v that can be found in $O(n+m)$ time.*

Let u be a vertex in $B \setminus (N(v) \cup \{v\})$. Since B is an end block of G, it is 2-connected. Thus, by Menger's Theorem, there are at least two internally disjoint paths in B linking u and v. Clearly, at least one of these paths contains some neighbour of v distinct from w. We successively delete vertices of degree at most $k-1$ in B' starting with every neighbour of v in B' distinct from w towards every other vertex distinct from w. At the end of this procedure, we delete w. This proves the claim.

Using the ordering given by Claim 2, we can now proceed (as in the proof of Lemma 2.2) to obtain a forest partition $\mathcal{F}''$ of B' such that w belongs to the independent set of $\mathcal{F}''$ (by simply placing $w \in Z_1$ at the start of the algorithm) in $O(n+m)$ time.

Given that we have guaranteed that at least two vertices in the neighbourhood of v belong to the independent set, the theorem follows. □

3. Hardness for large maximum degree

In this section, we prove Theorem 1.2. This will be done by exhibiting polynomial time reductions from new variants of SAT, where each reduction "corresponds" to some combination of values of p and q in a (p,q)-partition of the graph. Let us first introduce these new variants of SAT.

Recall that an instance $(X, \mathcal{C})$ of SAT consists of a set of Boolean variables $X = \{x_1, \ldots, x_n\}$ and a collection of clauses $\mathcal{C} = \{C_1, \ldots, C_m\}$, such that each clause is a disjunction of literals, where a *literal* is either x_i or its negation $\neg x_i$ for some $x_i \in X$. A function $g : X \to \{\text{true}, \text{false}\}$ is called a *satisfying truth assignment* if $\theta = C_1 \wedge \ldots \wedge C_m$ is evaluated to true under g. A SAT instance $(X, \mathcal{C})$ is called an RSAT *instance* if each clause is a disjunction of either exactly two literals or exactly four literals, and each literal appears at most twice in $\mathcal{C}$. A clause in $\mathcal{C}$ is called a *k-clause* for some positive integer k if it contains exactly k literals. We will reduce from the following variants of RSAT.

NAE-RSAT	
Instance:	An instance $(X, \mathcal{C})$ of RSAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment with at least one true literal and at least one false literal per clause?

EXACT-RSAT	
Instance:	An instance $(X, \mathcal{C})$ of RSAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment with exactly one true literal per clause?

ALL-RSAT	
Instance:	An instance $(X, \mathcal{C})$ of RSAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment with at least one true literal per 4-clause and exactly one true literal per 2-clause?

Lemma 3.1. *Each of the above variants of* RSAT *is* $\mathbb{NP}$*-complete.*

We require the following well-known $\mathbb{NP}$-complete decision problems; cf. Garey and Johnson [10].

An instance $(X, \mathcal{C})$ of SAT is a 4-SAT instance if every clause in $\mathcal{C}$ is a 4-clause.

4-SAT	
Instance:	An instance $(X, \mathcal{C})$ of 4-SAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment?

NAE 4-SAT	
Instance:	An instance $(X, \mathcal{C})$ of 4-SAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment with at least one true literal and at least one false literal per clause?

1-IN-4 SAT	
Instance:	An instance $(X, \mathcal{C})$ of 4-SAT.
Question:	Does $(X, \mathcal{C})$ have a satisfying truth assignment with exactly one true literal per clause?

Proof of Lemma 3.1. Clearly, each of the above variants of RSAT is in $\mathbb{NP}$. We simultaneously show that they are $\mathbb{NP}$-hard by exhibiting a generic reduction from an instance $(X, \mathcal{C})$ of SAT in which every clause contains exactly four literals. (Our proof is identical to the proof that the variant of 3-SAT in which every literal appears in at most two clauses is $\mathbb{NP}$-hard. It will merely suffice to make a few additional observations.)

Let $\theta = C_1 \wedge \ldots \wedge C_m$. If a variable $y \in X$ appears (as y or $\neg y$) in $k > 1$ clauses, then we replace y with a set of new variables $y^1, \ldots, y^k$ in the following way: we replace the first occurrence of y with y^1, the second occurrence of y with y^2, etc. If some of these occurrences are negated then we replace those occurrences with the negated versions of the new variables. We repeat this for each variable that appears in more than one clause. Next we link the new variables for y to each other with a set of clauses $(y^1 \vee \neg y^2), (y^2 \vee \neg y^3), \ldots, (y^k \vee \neg y^1)$. We denote by $(X', \mathcal{C}' = \{C'_1, \ldots, C'_{m'}\})$ the resulting instance, and let $\theta' = C'_1 \wedge \ldots \wedge C'_{m'}$. Notice that every literal appears in at most two clauses of $\mathcal{C}'$. Moreover, every 2-clause in $\mathcal{C}'$ has exactly one true literal since in every satisfying truth assignment g' of the variables in X', we have $g'(y^1) = \cdots = g'(y^k)$ for every $y \in X$. Thus, for every satisfying truth assignment to the variables in X and X',

- θ has at least one true literal and at least one false literal per clause if and only if θ' has at least one true literal and at least one false literal per clause;
- θ has exactly one true literal per clause if and only if θ' has exactly one true literal per clause;
- θ has at least one true literal per clause if and only if θ' has least one true literal per 4-clause and exactly one true literal per 2-clause.

Given that 4-SAT, NAE 4-SAT and 1-IN-4 SAT are $\mathbb{NP}$-complete, it follows that ALL-RSAT, NAE-RSAT and EXACT-RSAT are $\mathbb{NP}$-hard. This completes the proof. $\square$

Proof of Theorem 1.2. The problem is clearly in $\mathbb{NP}$. To show that it is $\mathbb{NP}$-hard, we simultaneously exhibit two reductions from a generic instance of RSAT.

Given an instance $(X, \mathcal{C})$ of RSAT we construct two graphs G and H of maximum degree $k \geq 5$ in the following way. We let $f : \bigcup_{C \in \mathcal{C}} C \to \{1, 2, 3, 4\}$ be a function that associates an integer between 1 and 4 to every literal such that

- if literal ℓ is the jth occurrence of y for some $y \in X$, then $f(\ell) = j$, and
- if ℓ is the jth occurrence of $\neg y$ for some $y \in X$, then $f(\ell) = 2 + j$.

Next, to each variable $x \in X$, we associate a variable gadget $S(x, k)$ as illustrated in Fig. 1 for $S(x, 5)$. It is a graph with 12 vertices $a_{x,i}, a'_{x,i}, a^*_{x,i}$ for $i \in \{1, \ldots, 4\}$, six vertices $\widehat{x}, \widehat{x}', \widehat{x}^*, \widetilde{x}, \widetilde{x}', \widetilde{x}^*$ and six copies $K^{(\widehat{x})}, K^{(\widetilde{x})}, K^{(1)}, K^{(2)}, K^{(3)}, K^{(4)}$ of a complete graph on $k-2$ vertices. We add edges between vertices $a_{x,i}, a'_{x,i}, a^*_{x,i}$ for $i = 1, \ldots, 4$ and $\widehat{x}, \widehat{x}', \widehat{x}^*, \widetilde{x}, \widetilde{x}', \widetilde{x}^*$ in the way depicted in Fig. 1. Moreover, we add edges from

- each of $a_{x,i}, a'_{x,i}, a^*_{x,i}$ to every vertex of $K^{(i)}$,
- each of $\widehat{x}, \widehat{x}', \widehat{x}^*$ to every vertex of $K^{(\widehat{x})}$, and
- each of $\widetilde{x}, \widetilde{x}', \widetilde{x}^*$ to every vertex of $K^{(\widetilde{x})}$.

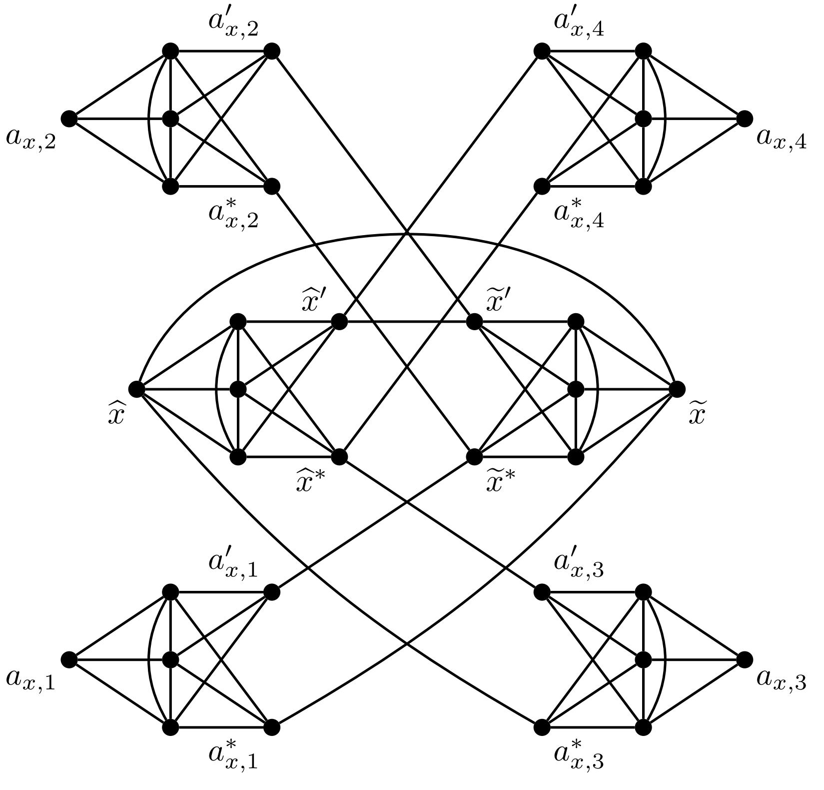

Fig. 1. The graph $S(x, 5)$.

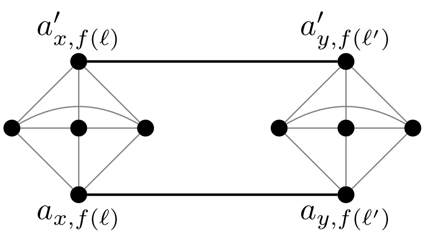

Fig. 2. The clause gadget that connects $S(x, 5)$ and $S(y, 5)$ via black edges, where literal ℓ corresponds to variable x and literal ℓ' to variable y. Grey edges are edges of $S(x, 5)$ or $S(y, 5)$ and black edges are edges of G and H.

Next, for each 2-clause in $\mathcal{C}$ with literals ℓ and ℓ' corresponding, respectively, to variables x and y for some $x, y \in X$, we construct a 2-clause gadget that connects $S(x, k)$ and $S(y, k)$ in the way depicted in Fig. 2. We refer to $a_{x,f(\ell)}$ and $a_{y,f(\ell')}$ as *special vertices* of the gadget. Finally, for each 4-clause in $\mathcal{C}$ with literals $\ell_1, \dots, \ell_4$ corresponding, respectively, to variables x_1, x_2, x_3, x_4 for some $x_1, x_2, x_3, x_4 \in X$, we construct a 4-clause gadget that connects $S(x_1, k), \dots, S(x_4, k)$ in the way depicted in Fig. 3. We also refer to each $a_{x_i,f(\ell_i)}$ as a *special vertex* of the gadget. This completes the construction of G and H. Note that G and H both have maximum degree $k \geq 5$.

Claim 3. *Let $p, q \geq 0$ such that $p + q = k - 3$. Then in every partition of $S(x, k)$ into a p-degenerate subgraph P and a q-degenerate subgraph Q either*

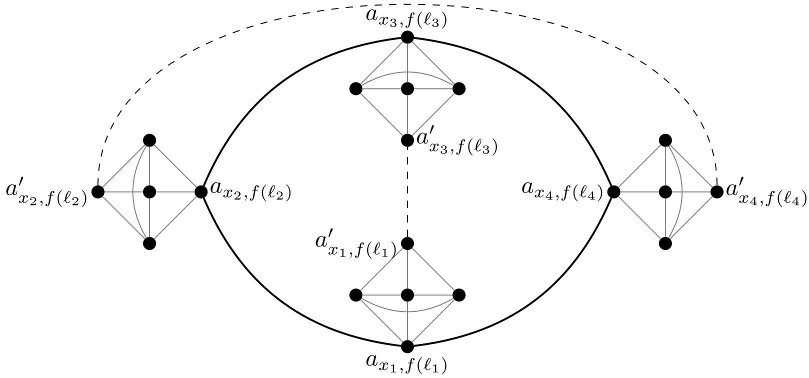

Fig. 3. The clause gadget that connects $S(x_1, 5), \ldots, S(x_4, 5)$ via black and dashed edges, where literal ℓ_i corresponds to variable x_i for $1 \leq i \leq 4$. Grey edges are edges of $S(x_i, 5)$ for $1 \leq i \leq 4$. Black edges are edges of both G and H while dashed edges are edges of H only.

- *$a_{x,1}, a_{x,2} \in V(P)$ and $a_{x,3}, a_{x,4} \in V(Q)$, or*
- *$a_{x,1}, a_{x,2} \in V(Q)$ and $a_{x,3}, a_{x,4} \in V(P)$.*

Proof sketch of claim. It suffices to show that if $a_{x,1} \in V(P)$, then $a_{x,2} \in V(P)$ and $a_{x,3}, a_{x,4} \in V(Q)$. Let us then assume that $a_{x,1} \in V(P)$. Recall that the set of neighbours of $a_{x,1}$ in $S(x, k)$ induces a complete graph $K^{(1)}$ on $k-2$ vertices. Given that $p+q = k-3$, it follows that exactly p vertices of $K^{(1)}$ are members of $V(P)$ while the other $q+1$ vertices of $K^{(1)}$ are members of $V(Q)$. This implies that both $a'_{x,1}$ and $a^*_{x,1}$ belong to $V(P)$.

Let us now show that $\widetilde{x} \in V(Q)$. If $\widetilde{x} \in V(P)$, then again we find that $\widetilde{x'}$, $\widetilde{x^*}$ and exactly p vertices of $K^{\widetilde{x}}$ are members of $V(P)$ while the other $q+1$ vertices of $K^{\widetilde{x}}$ are members of $V(Q)$. But the set $\{a_{x,1}, \widetilde{x}, \widetilde{x'}, \widetilde{x^*}\} \cup ((K^{(1)} \cup K^{\widetilde{x}}) \cap V(P))$ induces a graph of minimum degree $p+1$, which contradicts that P is p-degenerate. Hence $\widetilde{x} \in V(Q)$.

Let us next show that $\widehat{x}, a_{x,2} \in V(P)$. If $\widehat{x} \in V(Q)$ then, by the same reasoning, $\widehat{x^*}$, $\widehat{x'}$ and q vertices of $K^{\widehat{x}}$ are members of $V(Q)$ while the other $p+1$ vertices of $K^{\widehat{x}}$ are members of $V(P)$. Since $\widetilde{x} \in V(Q)$, we similarly find that $\widetilde{x'}, \widetilde{x^*} \in V(Q)$. But the set $V(Q) \cap (\{\widehat{x}, \widehat{x^*}, \widehat{x'}, \widetilde{x}, \widetilde{x'}, \widetilde{x^*}\} \cup K^{\widetilde{x}} \cup K^{\widehat{x}})$ induces a graph of minimum degree $q+1$, which contradicts that Q is q-degenerate. Similarly, if $a_{x,2} \in V(Q)$ then the set $V(Q) \cap (\{\widehat{x}, \widehat{x^*}, \widehat{x'}, a_{x,2}, a'_{x,2}, a^*_{x,2}\} \cup K^{(2)} \cup K^{\widehat{x}})$ induces a graph of minimum degree $q+1$. Hence $\widehat{x}, a_{x,2} \in V(P)$ as needed.

It remains to show that $a_{x,3}, a_{x,4} \in V(Q)$. Using the fact that $\widehat{x} \in V(P)$, one can argue as before that $a_{x,3}$ and $a_{x,4}$ are indeed both members of $V(Q)$. $\square$

Claim 4. *Let $p, q \geq 0$ such that $p+q = k-3$. Consider any partition of $S(x, k)$ into a p-degenerate subgraph P and a q-degenerate subgraph Q. Then each vertex in P (respectively, Q) that is not a special vertex or a neighbour of a special vertex has degree p in P (respectively, degree q in Q).*

Proof. This follows from the proof of Claim 3. $\square$

We distinguish three cases depending on the values of p and q.

Case 1 $p = 1$ and $q \geq 2$.

In this case, we reduce from ALL-RSAT. More precisely, we will show that $(X, \mathcal{C})$ has a satisfying truth assignment with exactly one true literal per 2-clause if and only if G admits a partition into a p-degenerate graph P and a q-degenerate graph Q. By Lemma 3.1, deciding whether G has a (p, q)-partition with $p = 1$ and $q \geq 2$ is $\mathbb{NP}$-hard.

Suppose that $(X, \mathcal{C})$ has a satisfying truth assignment with exactly one true literal per 2-clause. For each $x \in X$ and each literal ℓ corresponding to x, if ℓ is set to true, then we put $a_{x,f(\ell)}$ in $V(Q)$, and if ℓ is set to false, then we put $a_{x,f(\ell)}$ in $V(P)$.

By Claim 3, this partial (p, q)-partition of G extends to a (p, q)-partition of each variable gadget of G. To see that this gives a partition of G into a p-degenerate graph P and a q-degenerate Q, notice that

- the degrees of vertices in each of P and Q are not affected by the 2-clause gadgets, and
- no cycle in P is formed by the 4-clause gadgets given that at least one special vertex of each 4-clause gadget belongs to Q.

Using Claim 4, one may then easily check that a p-degenerate ordering of the vertices in P and a q-degenerate ordering of the vertices in Q can be obtained.

Conversely, suppose that G admits a partition into a p-degenerate graph P and a q-degenerate graph Q. For each $x \in X$ and each literal ℓ corresponding to x, if $a_{x,f(\ell)} \in V(Q)$, then we set ℓ to true, and if $a_{x,f(\ell)} \in V(P)$, then we set ℓ to false.

By Claim 3, this is a valid truth assignment to the variable in X. Notice that at least one special vertex of each 4-clause gadget is a member of $V(Q)$ given that $p = 1$. Notice also that exactly one special vertex of each 2-clause gadget is a member of $V(Q)$, for if two special vertices, say $a_{x,i}$ and $a_{x,j}$, of a 2-clause gadget are in $V(Q)$, then $a_{x,i}$, $a_{x,j}$, their neighbours in Q and $a'_{x,i}$ and $a'_{x,j}$ would induce a graph of minimum degree $q + 1$ in Q. Similarly, if both $a_{x,i}$ and $a_{x,j}$ are in $V(P)$, then these vertices together with their neighbours in P and $a'_{x,i}$ and $a'_{x,j}$ would induce a graph of minimum degree 2 in P. Hence we have a satisfying truth assignment of $(X, \mathcal{C})$ such that each 4-clause has at least one true literal and each 2-clause exactly one true literal. This completes Case 1.

Case 2 $p = 0$ and $q \geq 2$.

In this case, we reduce from EXACT-RSAT. More precisely, we will show that $(X, \mathcal{C})$ has a satisfying truth assignment with exactly one true literal per clause if and only if H admits a partition into a p-degenerate graph P and a q-degenerate graph Q. By Lemma 3.1, deciding whether H has a (p, q)-partition with $p = 0$ and $q \geq 2$ is $\mathbb{NP}$-hard.

Suppose that $(X, \mathcal{C})$ has a satisfying truth assignment with exactly one true literal per clause. For each $x \in X$ and each literal ℓ corresponding to x, if ℓ is set to true, then we put $a_{x,f(\ell)}$ in $V(P)$, and if ℓ is set to false, then we put $a_{x,f(\ell)}$ in $V(Q)$. By Claim 3, this partial partition of H can be extended to a (p, q)-partition of each variable gadget. Clearly, this forms a (p, q)-partition of every 2-clause gadget. To see that it also forms a (p, q)-partition of every 4-clause gadget (and therefore a (p, q)-partition of H), consider a 4-clause gadget with special vertices $a_{x_1,f(\ell_1)}, \ldots, a_{x_4,f(\ell_4)}$. Suppose without loss of generality that $a_{x_1,f(\ell_1)}$ are in $V(P)$ and $a_{x_2,f(\ell_2)}, a_{x_3,f(\ell_3)}, a_{x_4,f(\ell_4)}$ are in $V(Q)$. We extend this partition to the rest of the 4-clause gadget so that $a'_{x_1,f(\ell_1)} \in V(P)$ and $a'_{x_2,f(\ell_2)}, a'_{x_3,f(\ell_3)}, a'_{x_4,f(\ell_4)} \in V(Q)$. It is clear that no two vertices in $V(P)$ are adjacent so it remains to show that vertices in $V(Q)$ induce a q-degenerate subgraph. Notice that vertex $a'_{x_4,f(\ell_4)}$ has q neighbours in $V(Q)$ since its neighbours that are not in $K^{(4)}$ are in $V(P)$. The procedure of first deleting $a'_{x_4,f(\ell_4)}$, followed by the neighbours of $a'_{x_4,f(\ell_4)}$ in Q, followed by $a_{x_4,f(\ell_4)}, a_{x_2,f(\ell_2)}$ and $a_{x_3,f(\ell_3)}$ in this order etc. (the rest of details are left to the reader) a q-degenerate ordering of vertices in Q can be obtained.

Conversely, suppose that H has a (p, q)-partition. For each $x \in X$ and each literal ℓ corresponding to x, if $a_{x,f(\ell)}$ is in $V(P)$, we set ℓ to true, and if $a_{x,f(\ell)}$ is in $V(Q)$, we set ℓ to false. By Claim 3, this is a valid truth assignment to the variable in X. As in Case 1, exactly one special vertex of each 2-clause is in $V(P)$. Consider a 4-clause with special vertices $a_{x_1,f(\ell_1)}, \ldots, a_{x_4,f(\ell_4)}$. Exactly one of these special vertices is in $V(P)$:

- If at least two of them are in $V(P)$, then they are not adjacent (since $p = 0$). Thus $a_{x_i,f(\ell_i)}, a_{x_{i+2},f(\ell_{i+2})} \in V(P)$ (for some $i = 1, 2$), which implies $a'_{x_i,f(\ell_i)}, a'_{x_{i+2},f(\ell_{i+2})} \in V(P)$. This is impossible since $a'_{x_i,f(\ell_i)}$ and $a'_{x_{i+2},f(\ell_{i+2})}$ are adjacent.
- If all of them are members of $V(Q)$, then, considering edges of the gadget that are in H but not in G, one can find a subgraph of Q with minimum degree $q + 1$.

This shows that we have a satisfying truth assignment of $(X, \mathcal{C})$ with exactly one true literal per clause. This completes Case 2.

Case 3 $p, q \geq 2$.

In this case, we reduce from NAE-RSAT. More precisely, we must show that a $(X, \mathcal{C})$ has a satisfying truth assignment with at least one false literal and at least one true literal per clause if and only if H admits a (p, q)-partition. By Lemma 3.1, deciding whether H has a (p, q)-partition with $p, q \geq 2$ is $\mathbb{NP}$-hard. Since the arguments are entirely similar to those of Cases 1 and 2, we leave the details to the reader.

The proof of the theorem is complete. □

4. Concluding remarks

Let us first note that a straightforward adaptation of the proof of Lemma 2.2 leads to the following statement.

Proposition 4.1. *Given integers $s \geq 2$ and $p_1, \ldots, p_s \geq 0$, a $(p_1, \ldots, p_s)$-partition of a graph G with maximum degree $k \geq 3$ that is not k-regular can be found in $O(n+m)$-time as long as $\sum_{i=1}^{s} p_i \geq k - s$.*

It is therefore unlikely that the time complexity increases by more than a factor of n in the outstanding case where G is k-regular.

Conjecture 4.1. *Let G be a connected graph with maximum degree $k \geq 3$ distinct from K_{k+1}. For every $s \geq 2$ and $p_1, \ldots, p_s \geq 0$ such that $\sum_{i=1}^{s} p_i \geq k - s$, a $(p_1, \ldots, p_s)$-partition of G can be found in $O(n^2)$ time.*

We make a few remarks on the case $s \geq 3$ with $\sum_{i=1}^{s} p_i < k - s$. A simple application of Proposition 4.1 leads to the following statement.

Proposition 4.2. *Given non-negative integers $p, q, p_1, p_2, \ldots, p_t, q_1, \ldots, q_{t'}$ such that $\sum p_i = p - t$ and $\sum q_i = q - t'$, if a graph G is (p, q)-partitionable, then G is also $(p_1, \ldots, p_t, q_1, \ldots, q_{t'})$-partitionable.*

Proposition 4.2 can be understood to mean (although rather imprecisely) that the complexity of Problem 1 in the situation when $\sum_{i=1}^{s} p_i < k - s$ does not increase as s increases. Phrased differently, Proposition 4.2 states informally that if one can find a partition into two subgraphs with prescribed degeneracy, then one can also find a partition of the same graph into more than two subgraphs with prescribed degeneracy, provided some condition on the sum of the prescribed degeneracies is met.

We therefore hoped that the problem is $\mathbb{NP}$-complete whenever s is as large as possible (that is, when a partition into independent sets is sought) as this would suggest that the problem is $\mathbb{NP}$-complete for every $s \geq 2$. As it happens, however, when s is of maximum value, the problem is tractable as long as k is not too small and $(k - s) - \sum_{i=1}^{s} p_i$ is not very large [14]. This might indicate that determining the frontier between tractability and hardness for every value of s in Problem 1 will be a difficult task.

Acknowledgements

The authors are grateful to both referees for their careful reading of the paper and for their suggestions that significantly improved the presentation of the paper. This work received support from the Research Council of Norway via the project CLASSIS, grant number 249994.

References

[1] N. Alon, J. Kahn, P.D. Seymour, Large induced degenerate subgraphs, Graphs Combin. 3 (1) (1987) 203–211.
[2] B. Baetz, D.R. Wood, Brooks' vertex-colouring theorem in linear time, CoRR abs/1401.8023 (2014).
[3] M. Bonamy, K.K. Dabrowski, C. Feghali, M. Johnson, D. Paulusma, 2017.
[4] M. Bonamy, K.K. Dabrowski, C. Feghali, M. Johnson, D. Paulusma, Recognizing graphs close to bipartite graphs with an application to colouring reconfiguration, CoRR abs/1707.09817 (2017).

[5] O.V. Borodin, A.V. Kostochka, B. Toft, Variable degeneracy: extensions of Brook's and Gallai's theorems, Discrete Math. 214 (1–3) (2000) 101–112,.
[6] R.L. Brooks, On colouring the nodes of a network, Math. Proc. Camb. Phil. Soc. 37 (2) (1941) 194–197.
[7] P.A. Catlin, H.-J. Lai, Vertex arboricity and maximum degree, Discrete Math. 141 (1–3) (1995) 37–46,.
[8] C. Feghali, M. Johnson, D. Paulusma, A reconfigurations analogue of Brook's theorem and its consequences, J. Graph Theory 83 (4) (2016) 340–358,.
[9] C. Feghali, M. Johnson, D. Paulusma, Kempe equivalence of colourings of cubic graphs, European J. Combin. 59 (2017) 1–10.
[10] M.R. Garey, D.S. Johnson, Computers and Intractability, volume 29, wh freeman New York, 2002.
[11] H.V. Kronk, J. Mitchem, Critical point-arboritic graphs, J. Lond. Math. Soc. 2 (3) (1975) 459–466.
[12] D.R. Lick, A.T. White, K-degenerate graphs, Canad. J. Math. 22 (1970) 1082–1096.
[13] L. Lovász, Three short proofs in graph theory, J. Combin. Theory Ser. B 19 (1975) 269–271.
[14] M. Molloy, B. Reed, Colouring graphs whose chromatic number is almost their maximum degree, in: Latin American Symposium on Theoretical Informatics, Springer, 1998, pp. 216–225.
[15] C. Thomassen, Decomposing a planar graph into degenerate graphs, J. Combin. Theory Ser. B 65 (2) (1995) 305–314.
[16] C. Thomassen, Decomposing a planar graph into an independent set and a 3-degenerate graph, J. Combin. Theory Ser. B 83 (2) (2001) 262–271,.
[17] Y. Wu, J. Yuan, Y. Zhao, Partition a graph into two induced forests, J. Math. Study 29 (1996) 1–6,.
[18] A. Yang, J. Yuan, Partition the vertices of a graph into one independent set and one acyclic set, Discrete Math. 306 (12) (2006) 1207–1216,.