



8-1-2008

On the Pseudo-Achromatic Number Problem

Jianer Chen

Texas A & M University - College Station

Iyad A. Kanj

DePaul University

Jie Meng

Texas A & M University - College Station

Gei Xia

Lafayette College

Fenghui Zhang

Texas A & M University - College Station

Recommended Citation

Chen, Jianer; Kanj, Iyad A.; Meng, Jie; Xia, Gei; and Zhang, Fenghui, "On the Pseudo-Achromatic Number Problem" (2008).

Technical Reports. Paper 2.

<http://via.library.depaul.edu/tr/2>

On the Pseudo-Achromatic Number Problem*

JIANER CHEN[†] IYAD A. KANJ[‡] JIE MENG[†] GE XIA[§] FENGHUI ZHANG[†]

Abstract

We study the parameterized complexity of the pseudo-achromatic number problem: Given an undirected graph and a parameter k , determine if the graph can be partitioned into k groups such that every two groups are connected by at least one edge. This problem has been extensively studied in graph theory and combinatorial optimization. We show that the problem has a kernel of at most $(k - 2)(k + 1)$ vertices that is constructable in time $O(m\sqrt{n})$, where n and m are the number of vertices and edges, respectively, in the graph, and k is the parameter. This directly implies that the problem is fixed-parameter tractable. We also study generalizations of the problem and show that they are parameterized intractable.

Keywords. pseudo-achromatic number, parameterized complexity, kernel, fixed-parameter tractability

1 Introduction

The PSEUDO-ACHROMATIC NUMBER problem is to determine whether an undirected graph G can be partitioned into k groups/classes $(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k)$ such that every two groups \mathcal{G}_i and \mathcal{G}_j , $1 \leq i < j \leq k$, are connected by at least one edge. The problem is also referred to in the literature as the GRAPH COMPLETE PARTITION problem, and is formally defined as follows:

Definition 1 Let G be an undirected graph. The *pseudo-achromatic number* of G is the largest integer p such that there exists a surjective function $f : V(G) \rightarrow \{1, \dots, p\}$ satisfying: for all i, j , where $1 \leq i, j \leq p$ and $i \neq j$, there exist $u \in f^{-1}(i)$, $v \in f^{-1}(j)$ such that $(u, v) \in E(G)$, where $f^{-1}(h)$ denotes the preimage set of h under f .

The PSEUDO-ACHROMATIC NUMBER problem is:

PSEUDO-ACHROMATIC NUMBER. Given an undirected graph G and a positive integer k , determine if the pseudo-achromatic number of G is at least k .

We will be using the informal definition more frequently than the formal one.

*A preliminary version of this manuscript will appear in the proceedings of the 34th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'08), Durham, England, June 30 - July 2, 2008.

[†]Department of Computer Science, Texas A&M University, College Station, TX 77843-3112, USA. Email: {chen, jmeng, fhzhang}@cs.tamu.edu. Supported by the NSF under grant CCF-0480683.

[‡]School of Computing, DePaul University, 243 S. Wabash Avenue, Chicago, IL 60604-2301, USA. Email: ikanj@cs.depaul.edu. Supported in part by a DePaul University Competitive Research Grant.

[§]Department of Computer Science, Lafayette College, Easton, PA 18042, USA. Email: gexia@cs.lafayette.edu.

It is easy to see that the PSEUDO-ACHROMATIC NUMBER problem is a variation of the graph coloring problem (or the achromatic number problem), the latter problem requiring the groups in the partition to be independent sets.

The PSEUDO-ACHROMATIC NUMBER problem was first introduced by Gupta in 1969 [10], and since then it has been studied extensively [1, 2, 3, 4, 8, 12, 13]. The problem is known to be NP-complete even on restricted classes of graphs [3, 8, 12].

Kortsarz et al. [12] studied the approximability of the PSEUDO-ACHROMATIC NUMBER problem. It was proved in [12] that the problem has a randomized polynomial-time approximation algorithm of ratio $O(\sqrt{\lg n})$, which can be de-randomized in polynomial time. This upper bound on the approximation ratio was shown to be asymptotically tight under the randomized model.

The PSEUDO-ACHROMATIC NUMBER problem was also considered from the extremal graph-theoretic point of view on special classes of graphs [2, 4, 13, 14, 15]. Balsubramanian et al. [1] gave a complete characterization of when the pseudo-achromatic number of the join of two graphs is the sum of the pseudo-achromatic numbers of the two graphs.

In the current paper we study the parameterized complexity of the PSEUDO-ACHROMATIC NUMBER problem. We show that the problem has a kernel of size at most $(k - 2)(k + 1)$ vertices that is computable in time $O(m\sqrt{n})$, where n and m are the number of vertices and edges, respectively, in the graph. This kernelization result directly gives an algorithm for the PSEUDO-ACHROMATIC NUMBER problem running in time $O(k^{k^2-k+2} + m\sqrt{n})$, thus showing that the problem is fixed-parameter tractable. The upper bound on the kernel size is obtained by developing elegant and highly non-trivial structural results, that are of independent interest.

We also study generalizations of the PSEUDO-ACHROMATIC NUMBER problem and prove that they are parameterized intractable. In particular, we consider the VERTEX GROUPING problem, in which an input instance has the form (G, H, k) , where G and H are two graphs, and $k = |V(H)|$. The problem asks for the existence of a surjective function $f : V(G) \rightarrow V(H)$ satisfying the property that $\forall u, v \in V(H)$, if $(u, v) \in E(H)$ then there exists $x \in f^{-1}(u), y \in f^{-1}(v)$ such that $(x, y) \in E(G)$. The PSEUDO-ACHROMATIC NUMBER problem is a special case of the VERTEX GROUPING problem in which the graph H is the complete graph on k vertices. The VERTEX GROUPING problem falls into the category of clustering problems, where a clustering of the graph G into $|V(H)|$ clusters is sought such that the inter-cluster properties are imposed by the graph H . We prove some (parameterized) intractability results for the VERTEX GROUPING problem. For example, we show that the problem is $W[1]$ -hard, even when the graph H is the h -star graph (i.e., $K_{1,h-1}$). We also show that some interesting instances of the VERTEX GROUPING problem can be solved in polynomial time.

2 Preliminaries

The reader is referred to Downey and Fellows' book [7] for more details about parameterized complexity theory.

A *parameterized problem* is a set of instances of the form (x, k) , where $x \in \Sigma^*$ for a finite alphabet set Σ , and k is a non-negative integer called the *parameter*. A parameterized problem Q is *fixed parameter tractable*, or simply FPT, if there exists an algorithm A that on input (x, k) decides if (x, k) is a yes-instance of Q in time $f(k)n^{O(1)}$, where f is a recursive function independent of $n = |x|$. In analogy to the polynomial time hierarchy, a hierarchy for parameterized complexity, called the *W-hierarchy*, has been defined. At the 0th level of this hierarchy lies the class FPT of

fixed-parameter tractable problems. The class of all problems at the i -th level of the W-hierarchy ($i > 0$) is denoted by $W[i]$. A parameterized-complexity preserving reduction (FPT-reduction) has been defined as follows. A parameterized problem Q is *FPT-reducible* to a parameterized problem Q' if there exists an algorithm of running time $f(k)|x|^c$ that on an instance (x, k) of Q produces an instance $(x', g(k))$ of Q' such that (x, k) is a yes-instance of Q if and only if $(x', g(k))$ is a yes-instance of Q' , where the functions f and g depend only on k , and c is a constant. A parameterized problem Q is *W[i]-hard* if every problem in $W[i]$ is FPT-reducible to Q . Many well-known problems have been proved to be $W[1]$ -hard including: CLIQUE, INDEPENDENT SET, SET PACKING, DOMINATING SET, HITTING SET and SET COVER. The *parameterized complexity hypothesis*, which is a working hypothesis for parameterized complexity theory, states that $W[i] \neq \text{FPT}$ for every $i > 0$.

The notion of the fixed-parameter tractability of a problem turns out to be closely related to the notion of the problem having a good data reduction (or preprocessing) algorithm. Formally speaking, a parameterized problem Q is *kernelizable* if there exists a polynomial-time reduction that maps an instance (x, k) of Q to another instance (x', k') of Q such that: (1) $|x'| \leq g(k)$ for some recursive function g , (2) $k' \leq k$, and (3) (x, k) is a yes-instance of Q if and only if (x', k') is a yes-instance of Q . The instance x' is called the *kernel* of x . It was shown that a parameterized problem is fixed-parameter tractable if and only if it is kernelizable [9].

For a graph G we denote by $V(G)$ and $E(G)$ the set of vertices and edges of G , respectively. A *matching* M in a graph G is a set of edges such that no two edges in M share an endpoint. A matching M of G is said to be *maximum* if the cardinality of M is maximum over all matchings in G . For a vertex v and a set of vertices Γ in G , we say that v is *connected to* Γ if v is adjacent to some vertex in Γ . Similarly, for two sets of vertices Γ and Γ' in G , we say that Γ is *connected to* Γ' if there exists a vertex in Γ that is connected to Γ' . For a vertex $v \in G$ we denote by $N(v)$ the set of neighbors of v in G . For a set of vertices Γ in G we denote by $N(\Gamma)$ the set of neighbors of all the vertices of Γ in G , i.e., $N(\Gamma) = \bigcup_{v \in \Gamma} N(v)$. We denote by S_h the $(h + 1)$ -star graph (i.e., $K_{1,h}$). The vertex of degree h in S_h is referred to as the *root* of the star, and the other h vertices are referred to as the *leaves* of the star. The *size* of the star S_h is the number of vertices in it, which is $h + 1$. We say that a graph G contains S_h if S_h is a subgraph (not necessarily induced) of G .

For a background on network flows we refer the reader to [6], or to any standard book on combinatorial optimization.

3 The kernel

In this section we show how to construct a kernel of size (number of vertices) at most $(k-2)(k+1)$ for the parameterized PSEUDO-ACHROMATIC NUMBER problem. We start by presenting some structural results that are essential for the kernelization algorithm, and that are of independent interest on their own.

3.1 Structural results

The following lemma ascertains that graphs with large matchings have large pseudo-achromatic number.

Lemma 3.1 *If a graph G contains a matching of size at least $(k - 1)k/2$, then the instance (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.*

PROOF. Assuming that G contains a matching of at least $(k-1)k/2$ edges, we show how to group the vertices of G into k groups $(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k)$ so that every pair of groups is connected. For every pair of groups $(\mathcal{G}_i, \mathcal{G}_j)$ where $1 \leq i < j \leq k$, we use a distinct edge (u, v) of the matching to connect the two groups by mapping the vertex u to \mathcal{G}_i and v to \mathcal{G}_j . The remaining vertices of G are mapped arbitrarily to the groups. Since there are exactly $(k-1)k/2$ pairs of groups and at least $(k-1)k/2$ edges in the matching, every pair of groups is connected under this mapping. It follows that (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem. \square

Lemma 3.2 *If a graph G contains a set of $k-1$ (mutually) vertex-disjoint stars of sizes $2, \dots, k$, respectively, then the instance (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.*

PROOF. Let $\mathcal{S} = \{s_1, \dots, s_{k-1}\}$ be a set of vertex-disjoint stars in G , where s_i is the star graph S_i . We will map the vertices in \mathcal{S} to k groups $(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k)$ such that every pair of groups is connected.

For $i = 1, \dots, k-1$, we map the root of s_i to group \mathcal{G}_{i+1} , and we map its leaves, in a one-to-one fashion, to groups $(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_i)$. The remaining vertices in G are mapped arbitrarily to the groups. Since there is no overlap between the vertices of any two stars in \mathcal{S} , this mapping is well defined. It is easy to verify now that every two distinct groups in $(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k)$ are connected under the defined mapping. It follows that (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem. \square

Lemma 3.3 *If a graph G contains a collection of (mutually) vertex-disjoint stars each of size at least 2 and at most $k+1$, and such that the total number of vertices in all the stars is more than $(k-2)(k+1)$, then the instance (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.*

PROOF. Suppose that G contains a collection \mathcal{P} of vertex-disjoint stars, each containing at least two vertices and at most $k+1$ vertices, and such that the total number of vertices of the stars in \mathcal{P} is more than $(k-2)(k+1)$. Assume, to get a contradiction, that (G, k) is a no-instance of the PSEUDO-ACHROMATIC NUMBER problem.

Let s be the star graph S_h and s' be the star graph $S_{h'}$ such that s and s' are vertex-disjoint. By *merging* s and s' we mean creating the star graph $S_{h+h'}$ by identifying the roots of s and s' . Note that the size of the merged star is 1 less than the size of s plus the size of s' .

We construct from \mathcal{P} a sequence of vertex-disjoint stars $\mathcal{S} = \langle s_{k-1}, \dots, s_r \rangle$, for some integer $r \geq 1$, such that s_i has size at least $i+1$, for $r \leq i \leq k-1$. The procedure that constructs these stars is as follows.

For $i = k-1$ down to 1 do: if the largest star in \mathcal{P} is an S_j , where $j \geq i$, assign it to s_i , and remove it from \mathcal{P} ; Otherwise, recursively merge the two stars of largest size in \mathcal{P} and add the resulting star to \mathcal{P} until either there is only one star left in \mathcal{P} , and in which case the procedure halts, or the largest star in \mathcal{P} is an S_j , where $j \geq i$, and in which case we assign it to s_i , remove it from \mathcal{P} , and proceed to the next value of i in the for loop.

If a star s_i in \mathcal{S} was created without merging stars in \mathcal{P} , we call s_i a *single star*, otherwise, we call s_i a *merged star*.

Note the following: if s_i is a merged star created from merging a collection of stars, and if s_i is used to produce a valid grouping of G , then clearly the stars that s_i was merged from can replace s_i to produce a valid grouping of G . Therefore, assuming that (G, k) is a no-instance of the PSEUDO-ACHROMATIC NUMBER problem, the last star s_r constructed by the above procedure before halting must satisfy $r \geq 2$. Otherwise, the sequence \mathcal{S} would contain a set of $k - 1$ vertex-disjoint stars of sizes $2, \dots, k$, and by Lemma 3.2, the instance (G, k) would be a yes-instance of the problem, contradicting our assumption.

Now assume that the above procedure halts after constructing a sequence of vertex-disjoint stars $\mathcal{S} = \langle s_{k-1}, \dots, s_r \rangle$, such that s_i has size at least $i + 1$, for $2 \leq r \leq i \leq k - 1$.

We define a *monotone subsequence* of \mathcal{S} to be a consecutive subsequence $\langle s_i, s_{i-1}, \dots, s_j \rangle$ of \mathcal{S} such that either s_i, s_{i-1}, \dots, s_j are all single stars, or they are all merged stars. A monotone subsequence $\langle s_i, s_{i-1}, \dots, s_j \rangle$ of \mathcal{S} is *maximal* if it is maximal under containment.

Let $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$, $\ell \geq 1$, be a maximal monotone subsequence of \mathcal{S} , and note that $i - \ell + 1 \geq 2$ (since $r \geq 2$). We will show that the total number of vertices in the stars of \mathcal{P} that were used to form the subsequence $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$ is at most $2(i + (i - 1) + \dots + (i - \ell + 1))$. We distinguish two cases:

- **Case 1.** $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$ consists of single stars. We distinguish two subcases:
 - **Subcase 1.1.** $i = k - 1$. Since every single star contains at most $k + 1$ vertices by the statement of the lemma, the total number of vertices in the stars in the subsequence is bounded by $\ell(k + 1) \leq 2(k - 1 + k - 2 + \dots + k - \ell)$. The last inequality is true because $((k - 1) - \ell + 1) \geq 2$.
 - **Subcase 1.2.** $i < k - 1$. By the maximality of the subsequence, s_{i+1} is a merged star. Since s_i is a single star, it is easy to verify that s_i has size exactly $i + 1$. The total number of vertices in the stars in the subsequence is bounded by $\ell(i + 1) \leq 2(i + i - 1 + \dots + i - \ell + 1)$ because $i - \ell + 1 \geq 2$.
- **Case 2.** $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$ consists of merged stars. Let s_j be any star in this subsequence, and suppose that s_j was constructed by merging stars t_1, \dots, t_q in \mathcal{P} . By the construction of s_j , the total number of leaves in the stars t_1, \dots, t_{q-1} is less than j (otherwise these stars would be sufficient to produce s_j), and the size of t_q is not larger than any of the sizes of t_1, \dots, t_{q-1} . Therefore, we have:

$$|t_1| - 1 + |t_2| - 1 + \dots + |t_{q-1}| - 1 \leq j - 1, \tag{1}$$

and

$$|t_q| \leq (|t_1| + |t_2| + \dots + |t_{q-1}|)/(q - 1). \tag{2}$$

Combining Inequality (1) with Inequality (2), and noting that $q \leq j$, we obtain:

$$|t_1| + |t_2| + \dots + |t_q| \leq 2j. \tag{3}$$

Inequality (3) shows that the total number of vertices in the stars of \mathcal{P} forming s_j is at most $2j$. By applying this inequality to each star s_j in the maximal monotone subsequence $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$

of merged stars, and by the linearity of addition, we obtain that the total number of vertices of \mathcal{P} used to form the stars in $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$ is at most $2(i + (i-1) + \dots + (i-\ell+1))$.

It follows from the above that, for any maximal monotone subsequence $\langle s_i, s_{i-1}, \dots, s_{i-\ell+1} \rangle$ of \mathcal{S} , the total number of vertices of \mathcal{P} used to form the stars in this subsequence is at most $2(i + (i-1) + \dots + (i-\ell+1))$. Applying the above bound to every maximal monotone subsequence of \mathcal{S} , and by the linearity of addition, we conclude that the total number of vertices in \mathcal{P} forming all the stars in \mathcal{S} is at most $(k-r)(k+r-1)$.

Noting that the number of remaining non-empty stars in \mathcal{P} cannot form an s_{r-1} , the total number of leaves in the remaining stars is at most $r-2$, and consequently, the total number of vertices in the remaining stars is at most $2(r-2)$. Therefore, the total number of vertices in \mathcal{P} is at most $(k-r)(k+r-1) + 2(r-2) = k^2 - k - (r^2 - 3r + 4)$. Since $r \geq 2$, \mathcal{P} has the maximum number of vertices when $r = 2$. It follows that the total number of vertices in \mathcal{P} is at most $(k-2)(k+1)$, contradicting the hypothesis of the lemma.

This completes the proof. \square

3.2 The auxiliary flow network and the graph pseudo-achromatic number

Let G be a graph with pseudo-achromatic number at least k , and let \mathcal{H} be a vertex grouping that partitions the vertices of G into k groups such that every pair of groups is connected.

For each pair of groups in \mathcal{H} , pick, arbitrarily, an edge connecting the groups, and designate that edge as a *critical edge*. Therefore, the set E_c of critical edges consists of exactly $\binom{k}{2} = k(k-1)/2$ edges, each connecting a different pair of groups in \mathcal{H} . The tuple (\mathcal{H}, E_c, k) will be called a *valid triple* for the graph G . All the edges in G that are not in E_c are called *noncritical edges*. A vertex in G is *critical* if it is incident to at least one critical edge; otherwise, the vertex is *noncritical*. Note that the existence of the valid triple (\mathcal{H}, E_c, k) for the graph G implies that the pseudo-achromatic number of G is at least k .

Lemma 3.4 *Let v be a noncritical vertex in G (with respect to a valid triple (\mathcal{H}, E_c, k)). Then either deleting v from G or moving v from its current group to any other group will result in a vertex grouping \mathcal{H}' such that (\mathcal{H}', E_c, k) is a valid triple for the resulting graph.*

PROOF. Since the vertex v is noncritical, v is not incident to any critical edges. Consequently, deleting v from G or moving v from one group to another group will not affect the critical edges. Therefore, in the new vertex grouping \mathcal{H}' in the resulting graph, there are still exactly k groups such that each pair of the groups is connected. \square

We will show a nice relationship between the pseudo-achromatic number of a graph and graph matchings.

Let M be a maximum matching in G . Let $I = V(G) \setminus V(M)$, and note that I is an independent set. For a vertex $u \in V(M)$ we denote by $N_I(u)$ the set $N(u) \cap I$. Let M_2 be the set of edges in M whose both ends are connected to I .

Lemma 3.5 *Let (u, v) be an edge in M_2 . Then $N_I(u) = N_I(v)$ and $|N_I(u)| = 1$.*

PROOF. By definition, both $N_I(u)$ and $N_I(v)$ are nonempty. Therefore, either $N_I(u) \neq N_I(v)$ or $|N_I(u)| > 1$ would imply the existence of two different vertices $w_1 \in N_I(u)$ and $w_2 \in N_I(v)$.

However, this would give an augmenting path (w_1, u, v, w_2) with respect to M , contradicting the maximality of the matching M . \square

Let $N_I(M_2)$ be the set $N(V(M_2)) \cap I$, and let $D = I \setminus N_I(M_2)$. We partition the edges of $M \setminus M_2$ into two sets M_1 and M_0 , where M_1 consists of all the edges in $M \setminus M_2$ that have exactly one end connected to D , and $M_0 = M \setminus (M_2 \cup M_1)$. Note that the edges in $M_0 \cup M_2$ have no end connected to D (however, an edge in M_0 or in M_1 may have an end connected to $N_I(M_2)$).

The vertices in $V(M_1)$ are further partitioned into R and L , such that R is the set of vertices in $V(M_1)$ that are connected to D , and L is the set of remaining vertices in $V(M_1)$. By definition, each edge in M_1 has exactly one end in R and one end in L . Moreover, by the definition of the set M_0 and by Lemma 3.5, the vertices in the set D can only be connected to vertices in R (note that D is an independent set). We refer the reader to Figure 1 for an illustration of the decomposition of G .

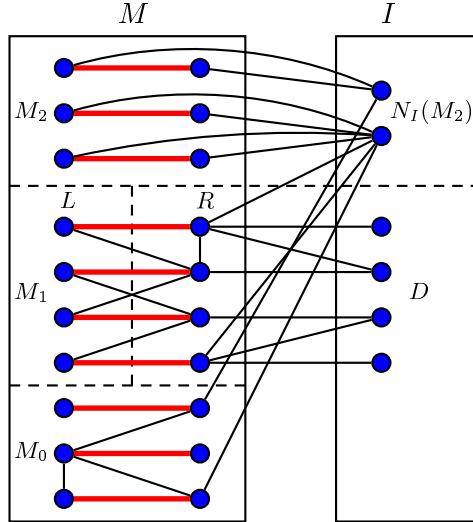


Figure 1: The decomposition of G .

Let J be the subgraph of G with vertex set $R \cup D$ and edge set $\{(u, v) \mid u \in R \text{ and } v \in D\}$. We construct a flow network J_k from J as follows. Convert each undirected edge (u, v) in J , where $u \in R$ and $v \in D$, into a directed edge $\langle u, v \rangle$ of capacity 1. Add a source s and a sink t . For each vertex $u \in R$, add a directed edge $\langle s, u \rangle$ of capacity $k - 1$; and for each vertex $v \in D$, add a directed edge $\langle v, t \rangle$ of capacity 1. We refer the reader to Figure 2 for an illustration of the flow network J_k .

In the following, we fix a valid triple (\mathcal{H}, E_c, k) for the graph G , a maximum matching M in G , and the corresponding flow network J_k . Let f^* be an integer-valued maximum flow in J_k . In case of no confusion, we will identify the vertices and edges in $J_k - \{s, t\}$ with their counterparts in G . Therefore, an edge is critical and saturated if it is critical with respect to the valid triple (\mathcal{H}, E_c, k) for G and saturated in the flow network J_k under the flow f^* .

For a vertex u , denote by f_u^* the flow through u , i.e., the total outgoing flow from u . We say that a vertex $u \in R$ is *saturated* if $f_u^* = k - 1$, and that a vertex $v \in D$ is *saturated* if $f_v^* = 1$.

Let $T_k = \{u \mid u \in D \text{ and } f_u^* = 0\}$. The main result of this subsection is to show that the instance (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem if and only if $(G - T_k, k)$ is.

We further partition the vertices in the set R into two sets R_1 and R_2 , where R_1 consists of all saturated vertices (in the flow network J_k under f^*), and $R_2 = R \setminus R_1$.

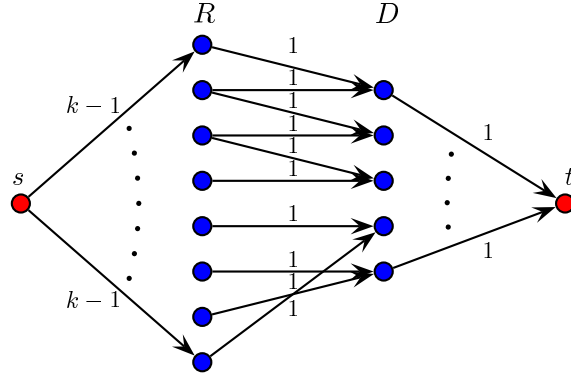


Figure 2: The flow network J_k .

Lemma 3.6 *For each vertex $u \in R_1$, let $\text{cri-unsat}(u)$ be the set of critical unsaturated edges going out from u , and let $\text{noncri-sat}(u)$ be the set of noncritical saturated edges going out from u . Then there is an injective mapping Φ_u from $\text{cri-unsat}(u)$ to $\text{noncri-sat}(u)$ (i.e., a mapping Φ_u such that for every $e_1, e_2 \in \text{cri-unsat}(u)$, if $e_1 \neq e_2$ then $\Phi_u(e_1) \neq \Phi_u(e_2)$).*

PROOF. It suffices to show that $|\text{cri-unsat}(u)| \leq |\text{noncri-sat}(u)|$. Let $\text{cri}(u)$ be the set of critical edges going out from u , and let $\text{sat}(u)$ be the set of saturated edges going out from u . It is easy to see that the vertex u can be incident to at most $k - 1$ critical edges. Thus, $|\text{cri}(u)| \leq k - 1$. Moreover, since $u \in R_1$, u is saturated, which gives $|\text{sat}(u)| = k - 1$. Now let $\text{cri-sat}(u)$ be the set of critical saturated edges going out from u . Then $\text{cri}(u) \setminus \text{cri-sat}(u)$ is the set $\text{cri-unsat}(u)$ of critical unsaturated edges going out from u , and $\text{sat}(u) \setminus \text{cri-sat}(u)$ is the set $\text{noncri-sat}(u)$ of noncritical saturated edges going out from u . By the above analysis, we have $|\text{cri-unsat}(u)| = |\text{cri}(u) \setminus \text{cri-sat}(u)| \leq |\text{sat}(u) \setminus \text{cri-sat}(u)| = |\text{noncri-sat}(u)|$. \square

By Lemma 3.6, for each vertex $u \in R_1$ we can correspond an injective mapping Φ_u from the set $\text{cri-unsat}(u)$ of critical unsaturated edges going out from u to the set $\text{noncri-sat}(u)$ of noncritical saturated edges going out from u .

For the given valid triple (\mathcal{H}, E_c, k) , the maximum matching M in G , the flow network J_k , the maximum flow f^* on J_k , and the set of injective mappings $\{\Phi_u \mid u \in R_1\}$, we define a layered structure L that is a subgraph of the flow network J_k , as follows.

Definition 2 The 0-th level of L consists of all vertices in the set T_k . For an integer $i \geq 0$,

(1) the $(2i + 1)$ -st level of L consists of all vertices $u \in R$ such that $\langle u, v \rangle$ is a critical edge and $v \in D$ is a vertex in the $(2i)$ -th level. Every critical edge that is from a vertex in the $(2i + 1)$ -st level to a vertex in the $(2i)$ -th level is also included in L .

(2) the vertices in the $(2i + 2)$ -nd level are given as follows: for each critical unsaturated edge $e = \langle u, v \rangle$, where $u \in R_1$ is in the $(2i + 1)$ -st level and $v \in D$ is in the $(2i)$ -th level, if $\Phi_u(e) = \langle u, w \rangle$, then the vertex w is in the $(2i + 2)$ -nd level, and the edge $\langle u, w \rangle$ is also included in L .

By definition, all vertices in even levels in the layered structure L belong to the set D , and all vertices in odd levels in L belong to the set R . For any integer $i \geq 0$, all edges between the $(2i)$ -th

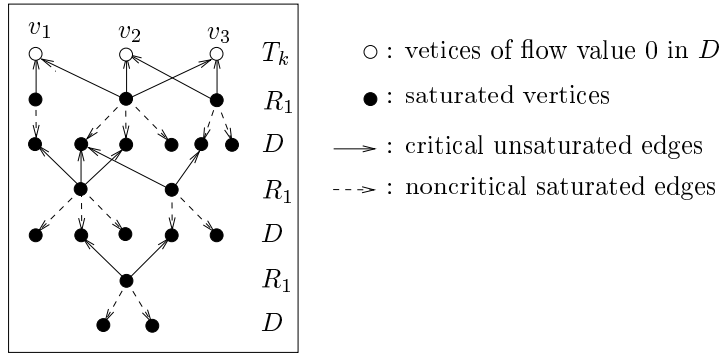


Figure 3: The layered structure L .

level and the $(2i + 1)$ -st level are critical edges whose direction is from the $(2i + 1)$ -st level to the $(2i)$ -th level; while all edges between the $(2i + 1)$ -st level and the $(2i + 2)$ -nd level are noncritical saturated with directions from the $(2i + 1)$ -st level to the $(2i + 2)$ -nd level.

Lemma 3.7 *The layered structure L has the following properties: (1) all critical edges in L are unsaturated; (2) all vertices in odd levels in L are in the set R_1 ; and (3) for each vertex v in an even level $2i$, where $i > 0$, there is exactly one edge coming into v from the $(2i - 1)$ -st level.*

PROOF. (1) Let e be a critical edge in L . If e is a directed edge from the 1-st level to the 0-th level, then the edge e must be unsaturated because all vertices in the 0-th level are in T_k , and hence are unsaturated. If $e = \langle u, v \rangle$ is from the $(2i + 1)$ -st level to the $(2i)$ -th level, for some $i > 0$, then since there is a noncritical saturated edge from the $(2i - 1)$ -st level to the vertex v in the $(2i)$ -th level, and since $v \in D$ has only one out-going edge that has a capacity 1, the critical edge e coming into the vertex v must be unsaturated.

(2) Let v be a vertex in the $(2i + 1)$ -st level in L , for some $i \geq 0$. By the definition of the layered structure L , $v \in R$, and there is a vertex sequence $(w_0, w_1, \dots, w_{2i+1})$ in the layered structure L , where $w_0 \in T_k$, $v = w_{2i+1}$, w_j is in the j -th level for all j , and for all h , the edge $\langle w_{2h+1}, w_{2h} \rangle$ is critical (which, by (1) of the current lemma, is also unsaturated), and the edge $\langle w_{2h-1}, w_{2h} \rangle$ is noncritical saturated. If the vertex v is not saturated, then the edge $\langle s, w_{2i+1} \rangle$ is unsaturated. Since $w_0 \in T_k$, the edge $\langle w_0, t \rangle$ is also unsaturated. Therefore, the path $(s, w_{2i+1}, w_{2i}, w_{2i-1}, \dots, w_1, w_0, t)$ would make a flow augmenting path in the residual network of J_k with respect to f^* , contradicting the maximality of the flow f^* in the flow network J_k . This proves that the vertex v must be saturated, i.e., $v \in R_1$.

(3) Let v be a vertex in the $(2i)$ -th level in L , for some $i > 0$. By the definition of the layered structure L , $v \in D$, and there is at least one noncritical saturated edge coming into v from the $(2i - 1)$ -st level. Moreover, since v has only one out-going edge to t that has capacity 1, v cannot have more than one incoming edge from the $(2i - 1)$ -st level that is saturated. \square

By Lemma 3.7, all vertices in odd levels in L belong to the set R_1 , and for any integer $i \geq 0$, the edges between the $(2i)$ -th level and the $(2i + 1)$ -st level are all critical unsaturated edges. We refer the reader to Figure 3 for the properties of the layered structure L .

We prove next that the layered structure L is finite.

Lemma 3.8 *Let v be a vertex in the set D that is at an even level $i > 0$ in the layered structure L . Then v is saturated and does not appear anywhere else in L .*

PROOF. By the definition of the layered structure L , there is a noncritical saturated edge $\langle u, v \rangle$ coming into v from the $(i - 1)$ -st level. Therefore, the vertex $v \in D$ is saturated.

To prove the second part of the lemma, suppose that the vertex v has two copies v_1 and v_2 in the layered structure L , which appear at the i_1 -th level and the i_2 -th level, respectively, where $i_1 \leq i_2$ are even integers. Without loss of generality, assume that the index i_1 is the smallest among all vertices in D that have multiple copies in L . We must have $i_1 < i_2$ since each vertex has at most one copy at each level in L . Moreover, $i_1 \neq 0$, since a vertex at the 0-th level is unsaturated (because it is in the set T_k) while a vertex at any other even level is saturated (by the first part of the current lemma). Therefore, we must have $0 < i_1 < i_2$.

By Lemma 3.7(3), for $j = 1, 2$, there is a unique noncritical saturated edge $\langle u_j, v_j \rangle$ from the $(i_j - 1)$ -st level to the vertex v_j . Since there is at most one saturated edge coming into a vertex in the set D , and $v_1 = v_2$, we must have $\langle u_1, v_1 \rangle = \langle u_2, v_2 \rangle$, so $u_1 = u_2$. Note that for each $j = 1, 2$, the edge $\langle u_j, v_j \rangle$ is the image of a unique edge $\langle u_j, w_j \rangle$ under the injective mapping Φ_{u_j} , where w_j is in the $(i_j - 2)$ -nd level in L . Since $u_1 = u_2$ and $\langle u_1, v_1 \rangle = \langle u_2, v_2 \rangle$, we must have $\langle u_1, w_1 \rangle = \langle u_2, w_2 \rangle$. Thus, $w_1 = w_2$ and the vertex w_1 is at the $(i_1 - 2)$ -nd level. However, this contradicts the minimality of the index i_1 . This completes the proof of the lemma. \square

Corollary 3.9 *Each edge in the flow network J_k can appear at most once in L .*

PROOF. Two edges between the same pair of adjacent levels in L cannot correspond to the same edge in J_k because no two vertices in the same level of L correspond to the same vertex. Two edges between two different pairs of adjacent levels in L cannot correspond to the same edge in J_k because either they have different flow saturations, or, they either come into or go out from, respectively, two vertices of D at different levels, which by Lemma 3.8, must be different. \square

By Lemma 3.8 and Corollary 3.9, we can conclude that the layered structure L is finite. We note that a vertex in the set R_1 may have multiple copies in the layered structure L , which, however, will not affect our discussion.

We call a vertex v at the i -th level of L a *leaf* if there is no edge in L between v and the $(i + 1)$ -st level in L . In particular, all vertices in the last level of L are leaves.

Lemma 3.10 *All leaves in the layered structure L belong to the set D .*

PROOF. Let u be a vertex in the set R_1 that is at the i -th level in L for some i . The vertex u is in L because of a critical unsaturated edge $e = \langle u, v \rangle$, where $v \in D$ is a vertex in the $(i - 1)$ -st level. By the definition of L , the edge $\Phi_u(e)$ will become an edge from u to a vertex in the $(i + 1)$ -st level, which implies that the vertex u cannot be a leaf. \square

Now we are ready for our main theorem in this subsection.

Theorem 3.11 *The instance (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem if and only if $(G - T_k, k)$ is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.*

PROOF. Since $G - T_k$ is a subgraph of G , the pseudo-achromatic number of $G - T_k$ cannot be

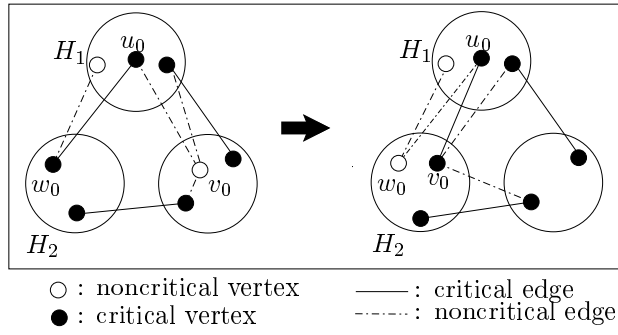


Figure 4: Moving a noncritical vertex v_0 .

larger than that of G . Therefore, if (G, k) is a no-instance of the PSEUDO-ACHROMATIC NUMBER problem, then $(G - T_k, k)$ is a no-instance of the PSEUDO-ACHROMATIC NUMBER problem.

Now suppose that (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem. Then there is a valid triple (\mathcal{H}, E_c, k) for the graph G . We fix the maximum matching M of G , the flow network J_k , the maximum flow f^* on J_k , the set R_1 of saturated vertices in R , and the set T_k of unsaturated vertices in D , as we have defined in the above discussion. Since for each valid triple (\mathcal{H}, E_c, k) for the graph G , we can define the set of injective mappings Φ_u and construct the corresponding layered structure L , we can assume, without loss of generality, that (\mathcal{H}, E_c, k) is a valid triple for G together with a set of injective mappings Φ_u , for which the corresponding layered structure L has the minimum number of vertices (note that the layered structure L is finite). We first show that all the vertices in the set T_k are noncritical under this valid triple (\mathcal{H}, E_c, k) and the injective mappings.

If the set T_k contains critical vertices, then the layered structure L has $h_0 + 1 > 1$ levels. Let v_0 be any vertex in the last level (i.e., the h_0 -th level) in L . By Lemma 3.10, $v_0 \in D$ and $h_0 > 0$ is an even number. By the definition of the layered structure L and since v_0 is a leaf, the vertex v_0 is not incident to any critical edges (recall that D is an independent set, and the vertices in D can only be connected to the vertices in R). Thus, v_0 is a noncritical vertex. Let $e_1 = \langle u_0, v_0 \rangle$ be the unique noncritical edge from the $(h_0 - 1)$ -st level to v_0 , and let $e_2 = \langle u_0, w_0 \rangle$ be the critical edge in L such that $\Phi_{u_0}(e_2) = e_1$, where w_0 is at the $(h_0 - 2)$ -nd level. Suppose that the vertices u_0 and w_0 belong to the groups H_1 and H_2 , respectively, under the grouping \mathcal{H} . We perform the following operations on the valid triple (\mathcal{H}, E_c, k) : (1) move the vertex v_0 from its current group to the group H_2 and let the new grouping be \mathcal{H}' ; and (2) designate $e_1 = \langle u_0, v_0 \rangle$ the critical edge between the groups H_1 and H_2 (so the edge $e_2 = \langle u_0, w_0 \rangle$ becomes a noncritical edge), and let $E'_c = E_c - e_2 + e_1$. See Figure 4 for an illustration of these operations.

Since v_0 is noncritical, by Lemma 3.4, it is easy to see that the triple (\mathcal{H}', E'_c, k) is a valid triple for the graph G . We also modify the injective mapping Φ_{u_0} at u_0 by simply removing the edge e_2 from the domain of Φ_{u_0} (recall that Φ_{u_0} is an injective mapping from the set $\text{cri-unsat}(u_0)$ of critical unsaturated edges going out from u_0 to the set $\text{noncri-sat}(u_0)$ of noncritical saturated edges going out from u_0): we had $e_2 \in \text{cri-unsat}(u_0)$ and $e_1 = \Phi_{u_0}(e_2) \in \text{noncri-sat}(u_0)$ under the original valid triple (\mathcal{H}, E_c, k) , while under the new valid triple (\mathcal{H}', E'_c, k) , the edge e_1 becomes critical saturated and the edge e_2 becomes noncritical unsaturated, so neither of them is in the set $\text{cri-unsat}(u_0) \cup \text{noncri-sat}(u_0)$. For all other vertices $u \in R_1$, $u \neq u_0$, in L , we keep the injective mapping Φ_u unchanged.

We consider how the layered structure L is changed under the new valid triple (\mathcal{H}', E'_c, k) and the new injective mapping Φ_{u_0} corresponding to vertex u_0 . The layered structure is started from the same set T_k and expanded level by level. An even level is expanded to the next level based on edge saturations and edge criticalities, and an odd level is expanded to the next level based on the injective mapping Φ_u on each vertex u in the current level. Therefore, the layered structure L' under the new valid triple (\mathcal{H}', E'_c, k) and the new injective mapping Φ_{u_0} is exactly the same as the old layered structure L , except when we expand from the vertex w_0 in the $(h_0 - 2)$ -nd level to the $(h_0 - 1)$ -st level: the edge $e_2 = \langle u_0, w_0 \rangle$ is not included because it is no longer critical. As a consequence, the edge $e_1 = \langle u_0, v_0 \rangle$ will not be added between the $(h_0 - 1)$ -st level and the h_0 -th level and the vertex v_0 will not appear in the h_0 -th level. We emphasize that the above reasoning holds true also because of Corollary 3.9, which states that no edge has multiple copies in the layered structure.

Therefore, the layered structure L' under the new valid triple (\mathcal{H}', E'_c, k) and the new injective mapping Φ_{u_0} can be obtained from the layered structure L under the original valid triple (\mathcal{H}, E_c, k) and the original injective mapping Φ_{u_0} by deleting the edges e_1 and e_2 and deleting the vertex v_0 in the h_0 -th level (probably also deleting the vertex u_0 if there is no other critical edge from u_0 to a vertex in the $(h_0 - 2)$ -nd level). Thus, L' has at least one fewer vertex than L . However, this contradicts our assumption that the original valid triple (\mathcal{H}, E_c, k) , together with the original injective mappings on vertices in R_1 , gives the layered structure L of the minimum number of vertices. This contradiction shows that all vertices in the set T_k must be noncritical under the valid triple (\mathcal{H}, E_c, k) .

By Lemma 3.4, deleting a noncritical vertex in a graph under a valid triple gives a valid triple for the resulting graph. Moreover, note that deleting a noncritical vertex does not convert any noncritical vertices into critical vertices because the critical edges are not changed. Therefore, if we delete all vertices in the set T_k from the graph G under the valid triple (\mathcal{H}, E_c, k) , we will obtain a valid triple (\mathcal{H}', E_c, k) for the graph $G - T_k$, which shows that the pseudo-achromatic number of the graph $G - T_k$ is at least k , i.e., $(G - T_k, k)$ is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.

This completes the proof of the theorem. □

The above theorem shows that the vertex set T_k can be safely removed from the graph G . Moreover, the graph $G - T_k$ has the following nice property.

Lemma 3.12 *The vertices in the graph $G' = G - T_k$ can be decomposed into a collection \mathcal{P} of vertex-disjoint stars, each star of size at least 2 and at most $k + 1$.*

PROOF. We will exhibit the collection of vertex-disjoint stars \mathcal{P} in G' . We will denote by $V_{\mathcal{P}}$ the set of vertices of the stars in the collection \mathcal{P} , and by $E_{\mathcal{P}}$ the set of edges of the stars in \mathcal{P} .

The set of vertices of G' consists of the vertices in the matching M , the vertices in $N_I(M_2)$, and the vertices in D with a non-zero flow value. For a vertex u in R , let $S(u)$ be the star graph formed by the incident edge to u in M_1 , together with the set of saturated edges in G' incident on u . Clearly, each such star $S(u)$ has size at least 2 and at most $k + 1$ since the capacity of u in J_k is $k - 1$. Moreover, for any two vertices u and v in R , the two star graphs $S(u)$ and $S(v)$ share no vertices; otherwise, there would be a shared vertex $w \in S(u) \cap S(v)$ of capacity 1 in J_k with two saturated edges incident on it, contradicting the flow properties. We add all such stars $S(u)$ to the collection \mathcal{P} .

We also include in \mathcal{P} a maximal set of disjoint S_2 stars such that the root of each S_2 star is a vertex in $N_I(M_2)$ and its leaves are the end points of the same edge in M_2 . Moreover, for every edge in M_2 whose endpoints are not yet in $V_{\mathcal{P}}$, we include it in \mathcal{P} as an S_1 stars. Finally we include in \mathcal{P} the matching edges in M_0 as S_1 stars.

It is clear that all the stars included in \mathcal{P} are vertex-disjoint, and that each star has size at least 2 and at most $k + 1$.

We claim that $V_{\mathcal{P}}$ contains all the vertices of G' . First observe that $V_{\mathcal{P}}$ contains the endpoints of all the edges in M . Second, since every vertex v in $D - T_k$ is incident on a saturated edge in G' , v is included in \mathcal{P} . Moreover, since by definition every vertex $u \in N_I(M_2)$ forms an S_2 star with two vertices w and v , where (w, v) is an edge in M_2 , and since by Lemma 3.5 no other vertex in $N_I(M_2)$ can form a star with the vertices w and v , it follows from the construction of \mathcal{P} that $u \in V_{\mathcal{P}}$. Therefore, every vertex u in $N_I(M_2)$ is in \mathcal{P} , and $V_{\mathcal{P}}$ contains all the vertices of G' as desired. \square

3.3 Putting it all together: the kernelization algorithm

Consider the decomposition of G defined in Subsection 3.2, and let M and T_k be as defined in Subsection 3.2. The kernelization algorithm is given in Figure 5.

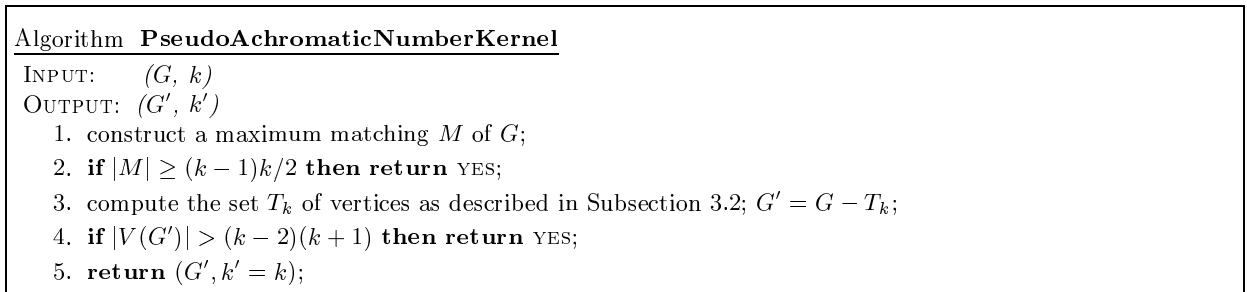


Figure 5: The kernelization algorithm.

Theorem 3.13 *Given an instance (G, k) of the PSEUDO-ACHROMATIC NUMBER problem, the algorithm PseudoAchromaticNumberKernel either decides the instance (G, k) correctly, or returns an instance (G', k') of the problem such that G' is a subgraph of G , $k' \leq k$, and (G, k) is a yes-instance if and only if (G', k') is. Moreover, the algorithm runs in time $O(m\sqrt{n})$, where n and m are the number of vertices and edges, respectively, in G .*

PROOF. If the size of the maximum matching M in G is at least $(k - 1)k/2$, then by Lemma 3.1, G is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem. Therefore, the algorithm **PseudoAchromaticNumberKernel** makes the right decision in step 2.¹

¹We note that step 2 is not essential to the algorithm and can be omitted. However, since the computation of the maximum matching M is essential to the computation of the set of vertices T_k in step 3, there is no harm in checking the size of the matching M and accepting the instance in case the size is large enough. Moreover, this step makes sense, especially from a practical point of view, as there is no need to carry on further with the computation of a maximum flow, and subsequently of the set of vertices T_k , if the graph contains a large matching and the instance can be accepted.

By Theorem 3.11, (G, k) is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem if and only if (G', k') is.

It suffices to argue that if $|V(G')| > (k-2)(k+1)$ (note $k' = k$), then (G', k') , and hence (G, k) , is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem, and the algorithm makes the right decision in step 4.

By Lemma 3.12, the set $V(G')$ can be decomposed into a collection of vertex-disjoint stars \mathcal{P} , each star of size at least 2 and at most $k+1$. Since $|V(G')| > (k-2)(k+1)$, it follows that the number of vertices in \mathcal{P} is more than $(k-2)(k+1)$. Consequently, \mathcal{P} satisfies the statement of Lemma 3.3, and (G', k') is a yes-instance of the PSEUDO-ACHROMATIC NUMBER problem.

Finally, to see that the algorithm **PseudoAchromaticNumberKernel** runs in time $O(m\sqrt{n})$, note first that the maximum matching M can be computed in $O(m\sqrt{n})$ time by a standard maximum matching algorithm [6]. Noting that the flow network J_k is a bipartite graph with at most $O(n)$ vertices and $O(m)$ edges, the maximum flow f^* in J_k can be computed in time $O(m\sqrt{n})$ [6]. All other steps can be performed in time $O(m)$, and the theorem follows. \square

Corollary 3.14 *The PSEUDO-ACHROMATIC NUMBER problem has a kernel of at most $(k-2)(k+1)$ vertices that is computable in time $O(m\sqrt{n})$, where n and m are the number of vertices and edges, respectively, in the graph, and k is the parameter.*

Remark. Note that our upper-bound analysis of the size of the kernel returned by the algorithm **PseudoAchromaticNumberKernel** is tight. This can be seen by considering a graph G that consists of $(k-1)k-2 = (k-2)(k+1)$ vertices which are the endpoints of $(k-1)k/2-1$ edges in a matching. The algorithm **PseudoAchromaticNumberKernel** on input (G, k) will return (G, k) as is, and without any modifications. Clearly, (G, k) is a no-instance of the PSEUDO-ACHROMATIC NUMBER problem.

Using the $(k-2)(k+1)$ upper bound on the kernel size, we can solve the PSEUDO-ACHROMATIC NUMBER problem by enumerating all possible assignments of the vertices in the graph to the k groups, then checking whether any such assignment yields a valid grouping. We have the following corollary:

Corollary 3.15 *The PSEUDO-ACHROMATIC NUMBER problem can be solved in time $O(k^{k^2-k+2} + m\sqrt{n})$, and hence is fixed-parameter tractable, where n and m are the number of vertices and edges, respectively, in the graph.*

PROOF. Given an instance (G, k) of the PSEUDO-ACHROMATIC NUMBER problem, where G has n vertices and m edges, we apply the algorithm **PseudoAchromaticNumberKernel** to (G, k) . The algorithm runs in $O(m\sqrt{n})$ time and either accepts the instance (G, k) correctly, or returns a kernel (G', k) where G' has at most $(k-2)(k+1)$ vertices. Now if G' can be partitioned into k groups that are mutually connected, then every vertex in G' must belong to one of the k groups. Therefore, there are at most $k^{(k-2)(k+1)}$ ways to partition G' into k groups. For each such partitioning, we can check whether the corresponding groups are mutually connected; this can be done in time $O(k^4)$. If we do not succeed in finding a valid partitioning then clearly the algorithm can reject the instance; otherwise, the algorithm returns a valid partitioning. The total running time of the algorithm is $O(k^4 \cdot k^{(k-2)(k+1)} + m\sqrt{n})$, which is $O(k^{k^2-k+2} + m\sqrt{n})$. \square

4 Hardness results for the VERTEX GROUPING problem

Recall from Section 1 that in the VERTEX GROUPING problem we are given an instance (G, H, k) , where G and H are two graphs, and $k = |V(H)|$, and the problem asks for the existence of a surjective function $f : V(G) \rightarrow V(H)$ satisfying the property that for all $u, v \in V(H)$, if $(u, v) \in E(H)$ then there exist $x \in f^{-1}(u)$ and $y \in f^{-1}(v)$ such that $(x, y) \in E(G)$. The VERTEX GROUPING problem can be defined more intuitively as follows.

Let G be an undirected graph. We define an operation on G , called *vertex grouping*, applied to a subset of vertices S as follows: remove all the vertices in S from G , add a new vertex w , and connect w to all the neighbors of S in $G - S$. The VERTEX GROUPING problem is:

VERTEX GROUPING: Given two graphs G and H , where H is a graph of k vertices, and k is the parameter, decide if H can be obtained from G by a sequence of vertex grouping operations.

If H in the above definition is the complete graph on k vertices, then the VERTEX GROUPING problem becomes the PSEUDO-ACHROMATIC NUMBER problem, and hence is fixed parameter tractable. The following theorem shows that the VERTEX GROUPING problem is parameterized intractable in general.

Theorem 4.1 *The VERTEX GROUPING problem is $W[1]$ -hard.*

PROOF. We reduce the $W[1]$ -hard problem INDEPENDENT SET to the VERTEX GROUPING problem.

Let (G, k) be an instance of the INDEPENDENT SET problem. Construct a graph G' by adding a new vertex w to G and connecting w to every vertex in G . Let H be a $(k + 1)$ -star with root r_H . Define the mapping π that, on an instance (G, k) of INDEPENDENT SET, produces the instance $(G', H, k + 1)$ of VERTEX GROUPING. Clearly, the mapping π is computable in polynomial time, and hence π is an FPT-reduction. We show that (G, k) is a yes-instance of INDEPENDENT SET if and only if $(G', H, k + 1)$ is a yes-instance of VERTEX GROUPING.

In effect, suppose that (G, k) is a yes-instance of INDEPENDENT SET, and let I be an independent set in G of size k . Consider the function $f : V(G') \rightarrow V(H)$ that maps the k vertices of I in G' to the k leaves of the star H , in a one-to-one fashion, and maps all other vertices of G' to the root r_H of H . Then it is easy to verify that H is a vertex grouping of G' under the function f .

Conversely, suppose that H is a vertex grouping of G' under a function f . Consider any set of vertices I in G of cardinality k satisfying $f(I) = V(H) \setminus \{r_H\}$. Clearly, such a set I exists by the definition of the vertex grouping. Note that f is a bijection from I to $V(H) \setminus \{r_H\}$. Now for any two distinct vertices u and v of I , u and v are not adjacent in G , otherwise, by the definition of vertex grouping, $f(u)$ and $f(v)$ would be adjacent in H . It follows that I is an independent set of size k in G . This completes the proof. \square

The Exponential Time Hypothesis (ETH) states that many NP-hard problems including 3-SAT, INDEPENDENT SET, and VERTEX COVER, cannot be solved in time $2^{o(n)}$. ETH has become a working hypothesis for many researchers in the area of exact and parameterized algorithms. It was shown in [5] that, unless ETH fails, INDEPENDENT SET cannot be solved in time $n^{o(k)}$. It was also shown in [5] that if a parameterized problem Q is reducible to a parameterized problem Q' by an FPT reduction, called *linear fpt-reduction*, that preserves the order of the parameter and

does not increase the size of the instance by more than a polynomial factor, and if Q cannot be solved in time $n^{o(k)}$ then it follows that Q' cannot be solved in time $n^{o(k)}$. Clearly, the reduction from INDEPENDENT SET to VERTEX GROUPING, given in the proof of Theorem 4.1, is a linear fpt-reduction. Therefore, we have the following theorem:

Theorem 4.2 *Unless ETH fails, the VERTEX GROUPING problem cannot be solved in time $n^{o(k)}$, where n and k are the number of vertices in G and H , respectively.*

Determining the complexity of the GRAPH ISOMORPHISM problem is an outstanding open problem that has been attracting the attention of researchers in theoretical computer science for decades. Although no polynomial time algorithm was developed for the problem, it seems unlikely that the problem is NP-hard [11].

We illustrate a relationship between the GRAPH ISOMORPHISM problem and the VERTEX GROUPING problem. Let G_1 and G_2 be two graphs on n vertices. We are interested in knowing how “similar” G_1 and G_2 are, under the notion of vertex grouping defined above. For this purpose, we introduce the following parameterized problem:

GRAPH STRUCTURAL SIMILARITY: given two graphs G_1 and G_2 on n vertices, and a parameter k , decide if there exists a graph H of k vertices such that both (G_1, H, k) and (G_2, H, k) are yes-instances of the VERTEX GROUPING problem.

Intuitively, the graph structural similarity measures the degree of similarity (i.e., k) between two graphs under the notion of vertex grouping. In particular, if $k = n$, then the GRAPH STRUCTURAL SIMILARITY problem is equivalent to the GRAPH ISOMORPHISM problem. We have the following parameterized intractability result for the GRAPH STRUCTURAL SIMILARITY problem:

Theorem 4.3 *The GRAPH STRUCTURAL SIMILARITY problem is $W[1]$ -hard.*

PROOF. As was shown in Theorem 4.1, the VERTEX GROUPING problem is $W[1]$ -hard when the graph H is a star. An FPT-reduction can be constructed that takes an instance (G, H, k) , where G has n vertices and H is a k -star, of the VERTEX GROUPING problem to an instance (G_1, G_2, k) of the GRAPH STRUCTURAL SIMILARITY problem, where $G_1 = G$ and G_2 is the n -star. Observing that any sequence of vertex grouping operations that are applied to G_2 can only result in a star graph, the $W[1]$ -hardness of the GRAPH STRUCTURAL SIMILARITY problem follows. \square

The reduction described in the proof of the above theorem is clearly a linear fpt-reduction. Therefore, it follows from Theorem 4.2 that:

Theorem 4.4 *Unless ETH fails, the GRAPH STRUCTURAL SIMILARITY problem cannot be solved in time $n^{o(k)}$, where n is the number of vertices in G_1 and G_2 , and k is the parameter.*

5 An easy instance of the VERTEX GROUPING problem

In this section we will show that some instances of the VERTEX GROUPING problem can be solved in polynomial time. We will consider the interesting case when the graph H , in the instances (G, H, k) of the VERTEX GROUPING problem, is the simple path P_k on k vertices.

For two vertices u and v in G , denote by the *distance* between u and v , $d_G(u, v)$, the length of a shortest path between u and v in G . Let G be an undirected graph, and k a positive integer. We start by providing a characterization of when P_k can be obtained from G by a sequence of vertex grouping operations. Equivalently, we provide a characterization of when G can be partitioned in k groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_i , $i = 2, \dots, k-1$, is connected and only connected to groups \mathcal{G}_{i-1} and \mathcal{G}_{i+1} . We consider first the case when G is connected.

Lemma 5.1 *Let G be a connected graph. Then G can be partitioned into k groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_i , $i = 2, \dots, k-1$, is connected and only connected to groups \mathcal{G}_{i-1} and \mathcal{G}_{i+1} , if and only if there exist two vertices u and v in G satisfying $d_G(u, v) \geq k-1$.*

PROOF. Suppose that there exist two vertices u and v in G satisfying $d_G(u, v) \geq k-1$. Let $(u = u_1, u_1, \dots, u_h = v)$ be a shortest path between u and v in G , where $h \geq k$. For $i = 1, \dots, k-1$, let $\mathcal{G}_i = \{w \in G \mid d_G(u, w) = i-1\}$, and with an abuse of the notation, let $\mathcal{G}_k = \{w \in G \mid d_G(u, w) \geq k-1\}$, and note that \mathcal{G}_i is nonempty, for $i = 1, \dots, k$ because $u_i \in \mathcal{G}_i$. Since G is connected, every vertex in G must appear in one of the k groups $\mathcal{G}_1, \dots, \mathcal{G}_k$. Moreover, by the definition of the groups and the connectedness of G , each group \mathcal{G}_i , $i = 2, \dots, k$, is connected and only connected to groups \mathcal{G}_{i-1} and \mathcal{G}_{i+1} .

Conversely, suppose that the vertices in G can be grouped into k groups, $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_i , $i = 2, \dots, k-1$, is connected and only connected to groups \mathcal{G}_{i-1} and \mathcal{G}_{i+1} . Let u be a vertex in \mathcal{G}_1 and v a vertex in \mathcal{G}_k . Since G is connected, there exists a shortest path between u and v in G . Clearly, any path between u and v must pass through at least one vertex in each of the groups \mathcal{G}_i , $i = 2, \dots, k-1$, and hence must have length at least $k-1$. It follows that $d_G(u, v) \geq k-1$. \square

Now we address the case when G is not connected.

Lemma 5.2 *Let G be an undirected graph, and assume that G is not connected. Let C_1, \dots, C_ℓ , where $\ell > 1$, be the connected components of G . Then G can be partitioned into k groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_i , $i = 2, \dots, k-1$, is connected and only connected to groups \mathcal{G}_{i-1} and \mathcal{G}_{i+1} , if and only if there exist vertices u_i and v_i in C_i , for $i = 1, \dots, \ell$, such that $d_G(u_1, v_1) + \dots + d_G(u_\ell, v_\ell) \geq k-1$.*

PROOF. We prove the statement for the case $\ell = 2$, and the proof for the general case follows by an inductive argument. Let C_1 and C_2 be the connected components of G .

Let u_1, v_1 be two vertices in C_1 , and u_2, v_2 be two vertices in C_2 such that $d_G(u_1, v_1) + d_G(u_2, v_2) \geq k-1$. By Lemma 5.1, we can group the vertices in C_1 into groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_r$, where $r = d_G(u_1, v_1) + 1$, such that each group \mathcal{G}_j , $j = 2, \dots, r-1$, is connected and only connected to groups \mathcal{G}_{j-1} and \mathcal{G}_{j+1} . Similarly, we can group the vertices of C_2 into groups $\mathcal{G}_{r+1}, \mathcal{G}_{r+2}, \dots, \mathcal{G}_{r+s}$, where $s = d_G(u_2, v_2) + 1$, such that each group \mathcal{G}_j , $j = r+2, \dots, r+s-1$, is connected and only connected to groups \mathcal{G}_{j-1} and \mathcal{G}_{j+1} . Now by grouping the vertices in \mathcal{G}_r and \mathcal{G}_{r+1} together, we obtain a grouping for G into groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_{r+s-1}$, where $r+s-1 = d_G(u_1, v_1) + d_G(u_2, v_2) + 1 \geq k$, such that each group \mathcal{G}_j , $j = 2, \dots, r+s-2$ is connected and only connected to groups \mathcal{G}_{j-1} and \mathcal{G}_{j+1} . Finally, by grouping the vertices in all the groups \mathcal{G}_j , where $j \geq k$, together, and calling the resulting group, without loss of generality, \mathcal{G}_k , we obtain a grouping of G into groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_j , $j = 2, \dots, k-1$ is connected and only connected to groups \mathcal{G}_{j-1} and \mathcal{G}_{j+1} .

To prove the converse, suppose that the vertices in G can be grouped into groups $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$, such that each group \mathcal{G}_j , $j = 2, \dots, k-1$, is connected and only connected to groups \mathcal{G}_{j-1} and \mathcal{G}_{j+1} . Since each of C_1 and C_2 is connected, the vertices of C_1 must appear in a consecutive subsequence $\mathcal{G}_p, \mathcal{G}_{p+1}, \dots, \mathcal{G}_{p+x}$ of the groups in $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$. Similarly, the vertices of C_2 must appear in a consecutive subsequence $\mathcal{G}_q, \mathcal{G}_{q+1}, \dots, \mathcal{G}_{q+y}$ of the groups in $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$. Since every vertex in G must appear in C_1 or in C_2 , and since any two adjacent groups in the sequence $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_k$ are connected, we have $(x+1) + (y+1) - 1 \geq k$, which implies that $x+y \geq k-1$. Let $u_1 \in \mathcal{G}_p$ and $v_1 \in \mathcal{G}_{p+x}$. Since C_1 is connected, there is a shortest path between u_1 and v_1 of length at least x . Similarly, there exists a shortest path from a vertex $u_2 \in \mathcal{G}_q$ to a vertex $v_2 \in \mathcal{G}_{q+y}$ of length at least y . It follows that $d_G(u_1, v_1) + d_G(u_2, v_2) \geq x+y \geq k-1$.

This completes the proof. \square

Theorem 5.3 *let G be a graph on n vertices and m edges. Then in time $O(nm)$ it can be decided whether P_k can be obtained from G by a sequence of vertex grouping operations.*

PROOF. The proof follows from Lemma 5.1 and Lemma 5.2, and the fact that the shortest distance between all pairs of vertices in a graph can be computed in $O(nm)$ time by running a breadth first search algorithm at every vertex in the graph. \square

6 Concluding remarks

In this paper we studied the PSEUDO-ACHROMATIC NUMBER problem from the parameterized complexity point of view. Using interesting and non-trivial techniques from matching theory and network flows, we were able to show that the problem admits a kernel of quadratic size that is computable in time $O(m\sqrt{n})$, where n and m are the number of vertices and edges, respectively, in the graph, and k is the parameter. The result directly implies that the PSEUDO-ACHROMATIC NUMBER problem is fixed-parameter tractable, and gives a straight-forward brute-force algorithm that runs in $O(k^{k^2-k+2} + m\sqrt{n})$ time for the problem. Improving on this trivial upper bound for solving the problem remains an interesting open problem.

We also considered a generalization of the PSEUDO-ACHROMATIC NUMBER problem: the VERTEX GROUPING problem. Although the PSEUDO-ACHROMATIC NUMBER problem, which is a special case of the VERTEX GROUPING problem, is fixed-parameter tractable, we showed that the VERTEX GROUPING problem is in general $W[1]$ -hard. We also showed that an interesting special case of the VERTEX GROUPING problem is solvable in polynomial time.

References

- [1] R. BALASUBRAMANIAN, V. RAMAN, AND V. YEGNANARAYANAN, On the pseudoachromatic number of join of graphs, *International Journal of Computer Mathematics*, 80-9, 2003 , pp. 1131- 1137.
- [2] V. BHAVE, On the pseudoachromatic number of a graph, *Fundamenta Mathematicae*, 102(3), 1979, pp. 159-164.
- [3] H. BODLAENDER, Achromatic number is NP-complete for cographs and interval graphs, *Information Processing Letters*, 32-3, 1989, pp.135-138.

- [4] B. BOLLOBÁS, B. REED, AND A. THOMASON, An extremal function for the achromatic number, *Graph Structure Theory*, 1991, pp.161-166. 2005, pp.18-37.
- [5] J. CHEN, X. HUANG, I. KANJ, AND G. XIA, Strong computational lower bounds via parameterized complexity, *Journal of Computer and System Sciences*, 72-8, 2006, pp.1346-1367.
- [6] T. CORMEN, C. LEISERSON, R. RIVEST, AND C. STEIN, *Introduction to Algorithms*, 2nd ed., McGraw-Hill Book Company, Boston, MA, 2001.
- [7] R. DOWNEY AND M. FELLOWS, *Parameterized Complexity*, Springer-Verlag, 1999.
- [8] K. EDWARDS AND C. MCDIARMID, The complexity of harmonious coloring for trees, *Discrete Applied Mathematics*, 57, 1995, pp.133-144.
- [9] R. DOWNEY, M. FELLOWS, AND U. STEGE, Parameterized complexity: a framework for systematically confronting computational intractability, in *Contemporary Trends in Discrete Mathematics*, (R. Graham, J. Kratochvíl, J. Nešetřil, and F. Roberts eds.), *Proc. DIMACS-DIMATIA Workshop, Prague 1997, AMS-DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 49, (1999), pp. 49–99.
- [10] R. P. GUPTA, Bounds on the chromatic and achromatic numbers of complementary graphs. in *Recent Progress in Combinatorics, Proc. 3rd Waterloo Conference on Combinatorics, Waterloo* (ed. W.T. Tutte), Academic Press, New York, 1969, pp. 229-235.
- [11] J. KÖBLER, U. SCHÖNING, AND J. TORÁN, *The Graph Isomorphism Problem: Its Structural Complexity*, Birkhäuser, 1993.
- [12] G. KORTSARZ, J. RADHAKRISHNAN, S. SIVASUBRAMANIAN. Complete partitions of graphs, *Proc. 16th Annual ACM-SIAM symposium on Discrete algorithms*, 2005, pp. 860-869.
- [13] E. SAMPATHKUMAR AND V. BHAVE, Partition graphs and coloring numbers of graphs, *Discrete Mathematics*, 16, 1976, pp.57-60.
- [14] V. YEGNANARAYANAN, On pseudocoloring of graphs, *Utilitas Mathematica*, 62, 2002, pp. 199-216.
- [15] V. YEGNANARAYANAN, The pseudoachromatic number of a graph, *Southern Aian Bulletin of Mathematics*, 24, 2002, pp. 129-136.