Near-Bipartiteness on graphs having small dominating sets: Structural characterization and algorithms

Maria Luíza López da Cruz
Advisors: Uéverton S. Souza, Raquel Bravo

[1]Universidade Federal Fluminense (UFF), Niterói, RJ, Brazil

marialopez@id.uff.br, {raquel, ueverton}@ic.uff.br

Abstract. *The Near-Bipartiteness problem asks for a partition of the vertex set of a graph $G = (V, E)$ into two subsets $\mathcal{S}$ and $\mathcal{F}$, where $\mathcal{S}$ forms an independent set and $\mathcal{F}$ induces a forest. Despite its NP-completeness, even for graphs with a diameter three, we explore this problem on graphs with a dominating edge or small dominating sets. Our work presents a polynomial-time algorithm for Near-Bipartiteness on graphs with a dominating edge, a particular case of graphs with diameter three. In addition, we prove that Connected Near-Bipartiteness, the variant where the forest must be connected, is NP-complete on the same class. Moreover, we also establish the NP-hardness of Independent Feedback Vertex Set and Acyclic Vertex Cover on this class of graphs. In addition, by extending our approach to graphs with bounded dominating sets, we achieve a huge improvement, obtaining an $O(n^2 \cdot m)$-time algorithm for Near-Bipartiteness on P_5-free graphs, improving upon the current state-of-the-art time complexity of $O(n^{16})$.*

1. Introduction

In 1972, Richard Karp presented the NP-completeness proof of 21 fundamental problems for Computer Science [Karp 1972]. FEEDBACK VERTEX SET, INDEPENDENT SET and VERTEX COVER are three of these classical problems. FEEDBACK VERTEX SET consists of finding a minimum set of vertices such that its removal eliminates all cycles of the input graph, INDEPENDENT SET consists of determining a maximum set of pairwise nonadjacent vertices (also known as a independent set), and VERTEX COVER is the problem of determining a minimum set of vertices intersecting all edges (called vertex cover) of the input graph. Note that if $\mathcal{S}$ is a independent set of $G = (V, E)$ then $\mathcal{F} = V(G) \setminus \mathcal{S}$ is a vertex cover of G.

An independent feedback vertex set (IFVS) of a graph G is a set of vertices that is independent and also a feedback vertex set of G. Defined by Yang A. and Yuan J. in [Yang and Yuan 2006], a graph $G = (V, E)$ has a *near-bipartition* $(\mathcal{S}, \mathcal{F})$ if there exist $\mathcal{S} \subseteq V$ and $\mathcal{F} = V \setminus \mathcal{S}$ such that $\mathcal{S}$ is a independent set, and $\mathcal{F}$ induces a forest. Furthermore, $\mathcal{S}$ and $\mathcal{F}$ can be empty sets. A graph that admits a near-bipartition is a *near-bipartite graph*. Note that the class of near-bipartite graphs is exactly the class of graphs having independent feedback vertex sets. Also, a graph G has an independent feedback vertex set $\mathcal{S}$ if and only if it has an acyclic vertex cover $\mathcal{F}$, i.e., a vertex cover $\mathcal{F}$ such that $G[\mathcal{F}]$ is acyclic (a vertex cover inducing a forest). The problem of recognizing near-bipartite graphs, so-called NEAR-BIPARTITENESS, is NP-complete even when restricted to graphs with maximum degree four [Yang and Yuan 2006], graphs with diameter three [Bonamy et al. 2018], line graphs [Bonamy et al. 2019], and planar graphs [Bonamy et al. 2017; Dross et al. 2017].

On the other hand, Brandstädt et al. [Brandstädt et al. 2013] proved that NEAR-BIPARTITENESS is polynomial-time solvable on cographs. Yang and Yuan [Yang and Yuan 2006] showed that NEAR-BIPARTITENESS is polynomial-time solvable for graphs of diameter at most two and that every connected graph of maximum degree at most three is near-bipartite except for the complete graph on four vertices (K_4). Besides, Bonamy et al. [Bonamy et al. 2019] proved that NEAR-BIPARTITENESS on P_5-free graphs can be solved in $O(n^{16})$ time. FPT algorithms parameterized by k for finding an independent feedback vertex set of size at most k can be found in [Agrawal et al. 2017; Li and Pilipczuk 2020; Misra et al. 2012]

A *coloring* for a graph G is an assignment of colors (labels) to all vertices of G. A *proper coloring* for G is an assignment of color $c(u)$, for each vertex $u \in V$, such that $c(u) \neq c(v)$ if $uv \in E(G)$. A graph G is *k-colorable* if there exists a proper coloring for G with at most k colors. The *chromatic number* of G, $\chi(G)$, is the smallest number k for G being k-colorable. A clear necessary condition for a graph to be near-bipartite is:

Proposition 1. *If a graph G is near-bipartite then G is 3-colorable.*

By Proposition 1, it holds that K_4 is a natural forbidden subgraph for near-bipartite graphs. A graph G is called *perfect* if for every induced subgraph H of G holds that its chromatic number equals the size of its largest clique, $\chi(H) = \omega(H)$. In particular, $\omega(G) = \chi(G)$.

However, the complexity of 3-COLORING and NEAR-BIPARTITENESS are not necessarily the same, depending on the graph class being explored. Grötschel, Lovász and Schrijver [Grötschel et al. 1984] proved that COLORING is solved in polynomial time for perfect graphs, while Brandstädt et al. [Brandstädt et al. 2013] proved that NEAR-BIPARTITENESS is NP-complete in the same graph class. NEAR-BIPARTITENESS can also be seen as a variant of 2-COLORING. For an input graph G, the question is whether its vertex set can be colored with two colors (not necessarily properly coloring) such that one color class is K_2-free (a independent set), and the other is cycle-free (i.e., induces a forest). Other 2-COLORING variants have already received attention in the literature. In [Achlioptas 1997], Achlioptas studied the problem of determining if there exists a bipartition of $V(G)$ where each part (color class) is H-free for some fixed graph H. He showed that for any graph H on more than two vertices, the problem is NP-complete. Another variant was considered by Schaefer [Schaefer 1978], who asked whether a given graph G admits a 2-coloring of the vertices such that each vertex has exactly one neighbor with the same color as itself. Schaefer proved that such a problem is NP-complete even for planar cubic graphs. The problem studied by Schaefer [Schaefer 1978] is a particular case of a defective coloring called $(2,1)$-coloring. A (k,d)-coloring of a graph G is a k-coloring of $V(G)$ such that each vertex has at most d neighbors with the same color. Some studies on $(2,1)$-coloring include [Borodin et al. 2013; Cowen et al. 1997; Lima et al. 2021]. In addition, the problem of finding a bipartition where each part induces a subgraph of minimum degree at least k (for a given integer k) was studied in [Bang-Jensen and Bessy 2019]. Also, the problem of partitioning the edge set of a graph into a independent set of edges (matching) and a forest has been studied in [Lima et al. 2017; Protti and Souza 2018].

2. Near-Bipartiteness Problem and Related Variations

In the master's thesis, driven by studies on variants of 2-coloring and the natural relevance of independent feedback vertex sets, we focused on the NEAR-BIPARTITENESS problem and its variants, which we define as follows:

NEAR-BIPARTITENESS

Instance: A simple undirected graph $G = (V, E)$.
Question: Does G have a near-bipartition $(\mathcal{S}, \mathcal{F})$?

INDEPENDENT FEEDBACK VERTEX SET

Instance: A simple undirected graph $G = (V, E)$.
Goal: Find (if any) a minimum independent feedback vertex set of G, i.e., a near-bipartition $(\mathcal{S}, \mathcal{F})$ that minimizes $|\mathcal{S}|$.

ACYCLIC VERTEX COVER

Instance: A simple undirected graph $G = (V, E)$.
Goal: Find (if any) a minimum acyclic vertex cover of G, i.e., a near-bipartition $(\mathcal{S}, \mathcal{F})$ of G that minimizes the size of $\mathcal{F}$.

Recall that the complement of an acyclic vertex cover is an independent feedback vertex set. So, the reader can assume that we are also dealing with the maximization version of both problems. Besides, we consider the problem of determining whether a graph G can have its set of vertices partitioned into a independent set and a *tree*, called CONNECTED NEAR-BIPARTITENESS, which was shown to be NP-complete even on bipartite graphs of maximum degree four [Brandstädt et al. 1998].

CONNECTED NEAR-BIPARTITENESS

Instance: A simple undirected graph $G = (V, E)$.
Question: Does G have a near-bipartition $(\mathcal{S}, \mathcal{F})$ such that $G[\mathcal{F}]$ is connected?

Motivated by the fact that NEAR-BIPARTITENESS remains NP-complete on graphs with diameter three [Bonamy et al. 2018], we first analyse the problem on graphs having a dominating edge, a natural subclass of graphs with diameter 3. In such a case, we show that NEAR-BIPARTITENESS can be solved in polynomial time, but CONNECTED NEAR-BIPARTITENESS is NP-complete. We also prove the NP-hardness of finding a *minimum* independent feedback vertex set or a *minimum* acyclic vertex cover on graphs having a dominating edge. Finally, we present a $O(n^2 \cdot m)$-time algorithm to solved NEAR-BIPARTITENESS on P_5-free graphs, improving the current $O(n^{16})$-time state of the art [Bonamy et al. 2019].

The mathematical proofs were crafted using reduction techniques to prove their complexity. Problems like 2-Satisfiability (2SAT), a polynomial problem, and 1-in-3-SAT, an NP-complete variation, were crucial for these reductions. Additionally, the Positive-Min-Ones-2SAT problem, also NP-complete, played a significant role. Due to space limitations, only proofs for Theorem 1 and 2 are presented in detail, while others are briefly mentioned with proof strategies outlined. All proofs can be found in [da Cruz 2023].

3. On graphs having a dominating edge

In this section, we consider the problem of partitioning a graph having a dominant edge into a independent set and a *tree* (CONNECTED NEAR-BIPARTITENESS), as well as the problem of partitioning it into a independent set and a forest (NEAR-BIPARTITENESS).

Theorem 1. CONNECTED NEAR-BIPARTITENESS *is* NP*-complete even when restricted to graphs having a dominating edge.*

Proof. The proof is based on a reduction from 1-IN-3SAT, a well-known NP-complete problem [Garey and Johnson 1979]. In such a problem we are given a formula φ in conjunctive normal form where each clause is limited to at most three literals, and asked whether there exists a satisfying assignment so that exactly one literal in each clause is set to true. Given an instance φ of 1-IN-3SAT, we construct a graph G such that φ has a truth assignment such that each clause has exactly one literal set to true if and only if G is partitionable into a independent set and a tree. From φ we construct G as follows:

1. first consider $G = (\{u, v\}, \{uv\})$;
2. add a chordless cycle C of size 4 in G induced by $\{k_1, k_2, k_3, k_4\}$, and add edges from u for all vertices in C;
3. add a chordless cycle $C' = l_1, m, l_2, n_1, n_2$;
4. add the edges ul_1, ul_2, vm, vn_1 and vn_2;
   At this point, notice that every $(\mathcal{S}, \mathcal{T})$-partition of G has $v \in \mathcal{S}$ and $u \in \mathcal{T}$.
5. for each variable x_i of φ create vertices v_{x_i} and $v_{\overline{x}_i}$ and add edges $v_{x_i}v_{\overline{x}_i}$, uv_{x_i} and $uv_{\overline{x}_i}$;
6. for each clause C_j of φ create a vertex c_j in G and add the edge vc_j;
7. Finally, add an edge $c_jv_{x_i}$ if the clause C_j contains the literal x_i, and add an edge $c_jv_{\overline{x}_i}$ if the clause C_j contains the literal $\overline{x}_i$.

Figure 1 shows a graph G constructed from a 3-CNF formula as previously described. Furthermore, it presents an $(\mathcal{S}, \mathcal{T})$-partition of G derived from a 1-in-3 truth assignment of φ. In this representation, the white vertices form the independent set, while the black vertices form the tree.

If φ is a 3-CNF formula having a truth assignment A such that each clause has exactly one literal set as true, then we can construct an $(\mathcal{S}, \mathcal{T})$-partition of G by setting $\mathcal{S} = \{k_1, k_3, l_1, v\} \cup \{v_{x_i} : x_i = \mathit{false} \in A\} \cup \{v_{\overline{x}_i} : x_i = \mathit{true} \in A\}$ (clearly $\mathcal{S}$ is a independent set). Since A defines a 1-in-3 truth assignment then each vertex c_j has exactly one neighbor in $G[V \setminus \mathcal{S}]$ then $\mathcal{T} = V \setminus \mathcal{S}$ induces a tree. Conversely, if G admits an $(\mathcal{S}, \mathcal{T})$-partition then, by construction, it holds that $v \in \mathcal{S}$ and $u \in \mathcal{T}$. This implies that every vertex c_j belongs to $\mathcal{T}$, and that for each pair $v_{x_i}, v_{\overline{x}_i}$ exactly one of these vertices belongs to $\mathcal{T}$. Also, since $\mathcal{T}$ is connected each c_j has at least one neighbor in $\mathcal{T}$, thus as $\mathcal{T}$ is acyclic each vertex c_j has exactly one neighbor in $\mathcal{T}$ (each c_j must be a leaf in $\mathcal{T}$). Therefore, we can construct a 1-in-3 truth assignment by setting $x_i = \mathit{true}$ iff $v_{x_i} \in \mathcal{T}$. □

Contrasting with Theorem 1, we show that when we remove the connectivity constraint, i.e., we look for a forest instead of a tree, the problem becomes polynomial-time solvable.

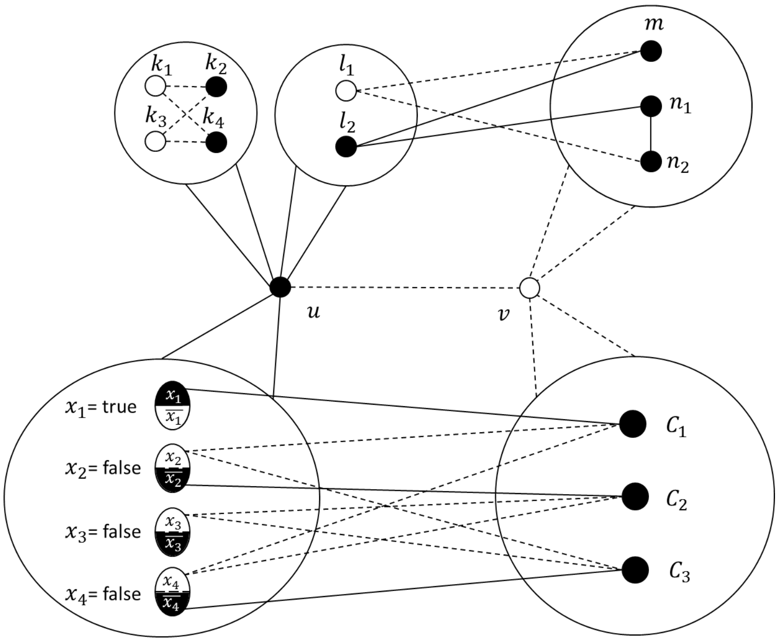

Figure 1. Example of graph G and an near-bipartition representation from the formula $\varphi = (x_1 + x_2 + x_4) \cdot (\overline{x}_2 + x_3 + x_4) \cdot (x_2 + x_3 + \overline{x}_4)$,

.

Theorem 2. *Given a graph G and a dominating edge of G, one can determine in $O(n^2)$ time whether G is a near-bipartite graph.*

Proof. Let $u, v \in V(G)$ be two vertices of G such that uv is a dominant edge of G. Suppose that G has a near-bipartition $(\mathcal{S}, \mathcal{F})$. Without loss of generality, we may assume that G does not have vertices with degree one. At this point, we may consider just two cases:

Case 1. Suppose that $u, v \in \mathcal{F}$. As $uv \in E(F)$, then $N(u) \cap N(v) \subseteq \mathcal{S}$, otherwise $\mathcal{F}$ has cycles. Thus, $N(u) \cap N(v)$ must be a independent set. For a remaining vertex w belonging to either $N(u) \setminus N(v)$ or $N(v) \setminus N(u)$: if it has a neighbor in $\mathcal{S}$ then it must belong to $\mathcal{F}$; if it has a neighbor z ($z \neq u$ and $z \neq v$) that must be in $\mathcal{F}$, then w must belong to $\mathcal{S}$, otherwise, the edge wz together with uv induces a cycle in $\mathcal{F}$. Thus, by checking if $N(u) \cap N(v)$ is independent and then successively applying the operations previously described according to a Breadth-First Search from $N(u) \cap N(v)$, in linear time, we can either conclude that such a near-bipartition with $u, v \in \mathcal{F}$ does not exist, or build a partition (S', F', U) of $V(G)$ such that S' is independent, $F' \supseteq \{u, v\}$ induces a forest, and U is the set of unclassified vertices. Note that, by construction, no vertex in U has neighbors in $S' \cup F' \setminus \{u, v\}$. Since any pair of adjacent vertices together with u and v induces a cycle, G has a near-bipartition $(\mathcal{S}, \mathcal{F})$ with $\{u, v\} \subseteq \mathcal{F}$ if and only if $G[U]$ has an independent vertex cover, which is equivalent to U inducing a bipartite graph.

Case 2. Suppose that $u \in \mathcal{S}$ and $v \in \mathcal{F}$. If $u \in \mathcal{S}$ and $v \in \mathcal{F}$ then $N(u) \subseteq \mathcal{F}$. Thus, $N(u)$ must induce a forest and $N(u) \cap N(v)$ must be a independent set. At this point, only the vertices belonging to $N(v) \setminus N[u]$ are unclassified. Let $B = N(v) \setminus \{u\}$. If G has a near-bipartition $(\mathcal{S}, \mathcal{F})$ then $G[B]$ must be bipartite, so that its vertices can be partitioned into two sets (B_1, B_2) such that $B_1 \subseteq \mathcal{S}$ and $B_2 \subseteq \mathcal{F}$. Thus, we must find a bipartition of B that satisfies the following conditions: $N(u) \cap N(v) \subseteq B_2$; and, for each component T of $G[N(u) \setminus N[v]]$ (which is a tree) it holds that: For each $w \in B_2$, $|N_T(w)| \leq 1$ (otherwise $\{w\} \cup V(T)$ induces a cycle); and, T has at most one neighbor in B_2 (otherwise $\mathcal{F}$ has cycles). Note that any bipartition (B_1, B_2) satisfying the above restrictions is sufficient to form a near-bipartition such that $B_1 \cup \{u\} = \mathcal{S}$. Now, we can reduce the problem of finding such a bipartition of $G[B]$ to the 2SAT problem by building a 2-CNF formula φ as follows:

1. for each vertex $w \in B$ create a variable x_w;
2. for each vertex $w \in N(u) \cap N(v)$ create a clause (x_w);
3. for each edge $w_1 w_2 \in E(G[B])$ create the clauses $(x_{w_1} + x_{w_2})$ and $(\overline{x}_{w_1} + \overline{x}_{w_2})$;
4. for each vertex $w \in B$ with at least two neighbors in the same component T of $G[N(u) \setminus N[v]]$, create a clause $(\overline{x}_w)$;
5. For each component T of $G[N(u) \setminus N[v]]$, and for each pair of vertices w_1, w_2 in the neighborhood of T, create a clause $(\overline{x}_{w_1} + \overline{x}_{w_2})$.

At this point, it is easy to see that φ is satisfied if and only if $G[B]$ has a partition (B_1, B_2) as requested (variables equal to true correspond to the vertices of B_2). Since φ can be built in $O(n^2)$ time with respect to the size of $G[B]$ and 2SAT can be solved in linear time [Aspvall et al. 1982], a near-bipartition $(\mathcal{S}, \mathcal{F})$ of G can be found in $O(n^2)$ time (if any). Now, let $\{u, v, x\} \subseteq V(G)$ be a dominating set of G which induces a triangle. If G has a near-bipartition $(\mathcal{S}, \mathcal{F})$ then $|\{u, v, x\} \cap \mathcal{S}| = 1$. Suppose that G has a near-bipartition $(\mathcal{S}, \mathcal{F})$ with $x \in \mathcal{S}$ and $u, v \in \mathcal{F}$. Clearly $N(u) \cap N(v) \cap N(x) = \emptyset$, otherwise G has a K_4. Also, $N(u) \cap N(v) \subseteq \mathcal{S}$. Therefore, as described in case 1, we can either conclude that such a near-bipartition does not exist, or build a partition (S', F', U) of $V(G)$ where $S' \subseteq \mathcal{S}$ and $F' \subseteq \mathcal{F}$. Without loss of generality, we can assume that $N(x) \subseteq F'$, and $N(x) \cap N(u)$ as well as $N(x) \cap N(v)$ are independent sets. Notice that vertices in U have only neighbors in $U \cup N(x)$. To classify the vertices in U, let $B = N(u) \cup N(v) \setminus \{u, v\}$. Similar to case 2, we must find a bipartition of B that satisfies the following conditions: $B \cap S' \subseteq B_1$ and $B \cap F' \subseteq B_2$; and, for each component T of $G[N(x) \setminus (B \cup \{u, v\})]$ (which is a tree) holds that: For each $w \in B_2$, $|N_T(w)| \leq 1$ (otherwise $\{w\} \cup V(T)$ induces a cycle); and, T has at most one neighbor in B_2 (otherwise $\mathcal{F}$ has cycles). Finally, it is easy to see that, as in case 2, we can find such a bipartition of $G[B]$ (if any), in linear time, using 2SAT. $\square$

Theorem 3. INDEPENDENT FEEDBACK VERTEX SET *is* NP-*hard when restricted to graphs having a dominating edge.*

First, it is noted that the Positive Min-Ones-2SAT problem [Misra et al. 2013] is equivalent to the Minimum Vertex Cover problem [Karp 1972], a well-known NP-complete problem. Given an instance φ of Positive Min-Ones-2SAT, we can construct a graph G using the same construction as Theorem 1 (disregarding negative literals). At this point, the variables x_i defined as true are equivalent to the vertices v_{x_i} assigned to S. Therefore, φ has a value assignment to its variables with at most k variables set as true such that φ is satisfied if and only if G is partitionable into an independent set $\mathcal{S}$ and a forest $\mathcal{F}$ such that $|\mathrm{S}| \leq k + 4$.

Theorem 4. ACYCLIC VERTEX COVER *is* NP-*hard when restricted to graphs having a dominating edge.*

The proof constructs a reduction from the NP-complete POSITIVE MIN-ONES-2SAT problem to ACYCLIC VERTEX COVER problem. Initially, is defined a construction algorithm transforms a given CNF formula into a graph with a dominating edge. (similar to the one used in Theorem 1) . This algorithm incorporates modifications to adapt to ACYCLIC VERTEX COVER's requirements, adding edges between clause vertices and variable vertices based on the literals present in the clauses of the CNF formula. The proof establishes the equivalence between satisfying assignments of the CNF formula and near-bipartitions of the graph, where true literals form one set ($\mathcal{F}$) and false literals the other ($\mathcal{S}$). Conversely, given a near-bipartition of the graph, the proof shows how to construct a satisfying assignment for the CNF formula. This involves selecting variables to be true or false based on their presence in the $\mathcal{F}$ or $\mathcal{S}$ sets, respectively. Lastly, it ensures the constructed graph's acyclic nature and validates the transformation between satisfying assignments and near-bipartitions, affirming the NP-hardness of ACYCLIC VERTEX COVER in graphs with a dominant edge.

4. On P_5-free graphs

In 2019, Bonamy, Dabrowski, Feghali, Johnson, and Paulusma showed that NEAR-BIPARTITENESS and INDEPENDENT FEEDBACK VERTEX SET can be solved in $O(n^{16})$ time. In 1990, Bacsó and Tuza [Bacsó and Tuza 1990] showed that any connected P_5-free graph has a dominating clique or a dominating P_3. In 2016, Camby and Schaudt generalized this result and showed that such a dominating set can be computed in polynomial time.

In this section, using the same approach presented in Theorem 2, we show how to handle NEAR-BIPARTITENESS on graphs having a dominating clique or a dominating P_3. Our results imply a faster algorithm to solve NEAR-BIPARTITENESS on P_5-free graphs in time $O(n^4)$. Interestingly, we can observe that the same technique combined with Bacsó and Tuza's result is not very useful to get a more efficient algorithm for INDEPENDENT FEEDBACK VERTEX SET on P_5-free graphs, due our Theorem 3 showing that this problem remains NP-complete on graphs having a dominating edge.

Theorem 5. *Given a graph G and a dominating triangle of G, one can determine in $O(n^2)$ time whether G is a near-bipartite graph.*

For this proof, initially, a dominating set $u, v, z \subseteq V(G)$ inducing a triangle is considered. The proof assumes z belongs to set $\mathcal{S}$ while u and v belong to set $\mathcal{F}$. Next, it's noted that $N(z)$, the neighbors of z, must form a forest within $\mathcal{F}$, as z belongs to $\mathcal{S}$. Moreover, the common neighbors of u and v cannot be in $N(z)$ to avoid the formation of a K_4. Further analysis focuses on the unclassified vertices, denoted as $B = N(v) \cup N(u) \setminus u, v, z$. As $G[B]$ must be bipartite, the goal is to partition its vertices into sets (B_1, B_2), with $B_1 \subseteq \mathcal{S}$ and $B_2 \subseteq \mathcal{F}$, satisfying specific conditions. These conditions ensure that G maintains its near-bipartite structure. Any bipartition (B_1, B_2) meeting the specified criteria is deemed sufficient to form a near-bipartition, where $B_1 \cup z = \mathcal{S}$. The problem of finding such a bipartition is then reduced to the 2SAT problem, concluding the proof.

Theorem 6. *Given a graph G and a dominating induced P_3 of G, one can determine in $O(n^4)$ time whether G is a near-bipartite graph.*

This proof involves the analysis of four cases. Each of them considers different configurations of the dominating set $\{u, v, z\}$ and the relationships between the vertices.
– **Case 1**: The vertices u and v belong to $\mathcal{F}$ while z is in $\mathcal{S}$. The proof begins by observing that $N(z)$ forms a forest, and $N(u) \cap N(v) \cap N(z) = \emptyset$ to prevent cycles. The unclassified vertices (B) are partitioned into sets B_1 and B_2, respecting certain conditions, and the problem is reduced to a 2SAT problem to find a bipartition of B. In this case the proof is similar to that of Theorem 5. The same holds if $\{v, z\} \in \mathcal{F}$ and $\{u\} \in \mathcal{S}$.
– **Case 2**: Only vertex $v \in \mathcal{F}$, while $\{u, z\} \in \mathcal{S}$. To maintain the independence of $\mathcal{S}$, the neighborhood of u and z, $N(u) \cup N(z)$, must belong entirely to $\mathcal{F}$, ensuring that $\mathcal{S}$ remains edge-free. Consequently, $N(u) \cup N(z)$ must form a forest, and $G[N(u) \cup N(z)]$ must not contain any path between vertices of $N(v) \cap (N(u) \cup N(z))$ to prevent cycles in $\mathcal{F}$. Additionally, the vertices in $N(v) \setminus (N(u) \cup N(z))$ must induce a bipartite graph. Also, for a vertex w in this set, if it has a neighbor $p \neq v$ that reaches v in $G[N(u) \cup N(z)]$, then p must be in $\mathcal{S}$. At this point, similarly to Theorem 2, a 2SAT formula can be used to decide which unclassified vertices of $N(v) \setminus (N(u) \cup N(z))$ must be in $\mathcal{F}$ and $\mathcal{S}$, with a time complexity of $O(n^2)$.
– **Case 3**: All vertices of the induced P_3 are in $\mathcal{F}$. In this scenario, any vertex with at least two neighbors in u, v, z must belong to the independent set $\mathcal{S}$. After that, we need observe that the set $(N(v) \setminus (N(u) \cup N(z))) \cup (N(z) \setminus (N(u) \cup N(v))) \cup (N(u) \setminus (N(v) \cup N(z)))$ forms a bipartite graph B. By iteratively applying a 2-coloring process from $(N(v) \cap N(z)) \cup (N(u) \cap N(v)) \cup (N(u) \cap N(z))$, for any remaining vertex w in this bipartite graph, (determining whether it belongs to $\mathcal{F}$ or $\mathcal{S}$ based on its neighbors) in linear time, it determines whether a near-bipartition of G with $v, u, z \in \mathcal{F}$ exists. If so, a partition (S', F', U) of $V(G)$ is constructed, where S' is independent, $F' \supseteq v, u, z$ induces a forest, and U represents unclassified vertices. Thus, we conclue that G has a near-bipartition $(\mathcal{S}, \mathcal{F})$ with $v, u, z \subseteq \mathcal{F}$ if and only if $G[U]$ has an independent vertex cover, which can be checked in linear time by verifying if U induces a bipartite graph.
– **Case 4**: u and z belong to the dominating set $\mathcal{F}$, while vertex v belongs to the independent set $\mathcal{S}$. This case is divided into two subcases. In case A, it is investigated whether u and z are in the same tree of $\mathcal{F}$, while in case B, the scenario where u and z are disconnected in $\mathcal{F}$ is considered.

In case A, when u and z are in the same tree of $\mathcal{F}$, pairs of vertices (a_u, a_z) where a_u and a_z are neighbors of u and z, respectively, are explored. We have $O(n^2)$ pairs a_u, a_z, and in $O(m)$ time we can check if $\{a_u, a_z\} \cup N(v)$ induces a forest having a tree containing u and z. For each pair, it is verified whether $(N(u) \cup N(z)) \setminus (a_u, a_z \cup N[v])$ forms a bipartite graph B. If yes, a bipartition of B satisfying certain conditions can be found, allowing to construct a near-bipartition of G with $a_u, a_z \cup N(v) \subseteq \mathcal{F}$. Such a bipartition can be found using a 2SAT formula. The overall running time for case A is $O(n^4)$, because we consider $O(n^2)$ pairs and for each one the described procedure can be performed in $O(n^2)$ time. In case B, when u and z are disconnected in $\mathcal{F}$, it is examined whether $N(u) \cap N(z)$ is an independent set and whether $G[N(u) \cup N(z)]$ is bipartite. If yes, the 2SAT method is used to classify the vertices of $N(z) \cup N(u)$ into $\mathcal{S}$ and $\mathcal{F}$. In both cases, techniques of bipartite graphs and 2SAT are used to find a near-bipartition of G. The total time to examine all cases is $O(n^4)$, where n is the number of vertices in G.

Next, we improve the Bonamy, Dabrowski, Feghali, Johnson, and Paulusma's result [Bonamy et al. 2019] concerning NEAR-BIPARTITENESS on P_5-free graphs.

Corollary 1. NEAR-BIPARTITENESS *on P_5-free graphs can be solved in $O(n^2 \cdot m)$ time.*

Near-bipartite graphs are K_4-free and K_4's can be found in $O(m^2)$ time. Also, near-bipartite P_5-free graphs have either a dominating triangle or a dominating P_3 due to Bacsó and Tuza's result [Bacsó and Tuza 1990]. Hence, it is enough to apply Theorem 5 and Theorem 6. Since G is P_5-free then Case 4A of Theorem 6 can be performed in $O(n^2 \cdot m)$ time, since either $a_u = a_z$ or $a_u a_z$ is an edge of G.

5. Concluding remarks

The master's dissertation explores the NEAR-BIPARTITION problem and its variants in graphs with dominating edges and P_5-free graphs, offering polynomial-time algorithms for NEAR-BIPARTITENESS and investigating the NP-completeness of CONNECTED NEAR-BIPARTITENESS. Moreover, it extends the analysis to graphs with bounded dominating sets, presenting an improved $O(n^2 \cdot m)$-time algorithm for NEAR-BIPARTITENESS in P_5-free graphs. Additionally, it is noted in [Bacsó and Tuza 1990] that every connected P_5-free graph contains a dominating clique or a dominating P_3. Thus, given a connected P_5-free graph G, to determine if it admits a near-bipartition can be achieved by: verifying the existence of a dominating K_4 in $O(n^4)$ time; confirming the existence of a dominating P_3 or K_3 in $O(n^3)$ time; and executing the algorithm described in Theorem 5 or Theorem 6 in $O(m.n^2)$ time.

PROBLEM ANALYZED	TIME COMPLEXITY
Partitioning problem in $\mathcal{S}, \mathcal{T}$ in graphs with dominating edge (CONNECTED NEAR-BIPARTITENESS)	NP-Complete
Near-Bipartiteness problem in graphs with dominating edge	$O(n^2)$
Near-Bipartiteness problem for graphs with dominating edge being $\lvert\mathcal{S}\rvert$ minimum (INDEPENDENT FEEDBACK VERTEX SET)	NP-hard
Near-Bipartiteness problem for graphs with dominating edge being $\lvert\mathcal{F}\rvert$ minimum (ACYCLIC VERTEX COVER)	NP-hard
Near-Bipartiteness problem for graphs with dominating K_3	$O(n^2)$
Near-Bipartiteness problem for graphs with dominating P_3	$O(n^4)$
Near-Bipartiteness problem for P_5-free graphs	$O(n^2 \cdot m)$

References

[Achlioptas 1997] Achlioptas, D. (1997). The complexity of G-free colourability. *Discrete Mathematics*, 165-166:21–30.

[Agrawal et al. 2017] Agrawal, A., Gupta, S., Saurabh, S., and Sharma, R. (2017). Improved algorithms and combinatorial bounds for independent feedback vertex set. In *11th International Symposium on Parameterized and Exact Computation (IPEC 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[Aspvall et al. 1982] Aspvall, B., Plass, M., and Tarjan, R. (1982). A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 14(4).

[Bacsó and Tuza 1990] Bacsó, G. and Tuza, Z. (1990). Dominating cliques in P_5-free graphs. *Periodica Mathematica Hungarica*, 21(4):303–308.

[Bang-Jensen and Bessy 2019] Bang-Jensen, J. and Bessy, S. (2019). Degree-constrained 2-partitions of graphs. *Theoretical Computer Science*, 776:64–74.

[Bonamy et al. 2017] Bonamy, M., Dabrowski, K. K., Feghali, C., Johnson, M., and Paulusma, D. (2017). Recognizing graphs close to bipartite graphs. In *MFCS 2017*.

[Bonamy et al. 2018] Bonamy, M., Dabrowski, K. K., Feghali, C., Johnson, M., and Paulusma, D. (2018). Independent feedback vertex sets for graphs of bounded diameter. *Information Processing Letters*, 131:26–32.

[Bonamy et al. 2019] Bonamy, M., Dabrowski, K. K., Feghali, C., Johnson, M., and Paulusma, D. (2019). Independent feedback vertex set for P_5-free graphs. *Algorithmica*, 81(4):1342–1369.

[Borodin et al. 2013] Borodin, O., Kostochka, A., and Yancey, M. (2013). On 1-improper 2-coloring of sparse graphs. *Discrete Math.*, 313(22):2638–2649.

[Brandstädt et al. 2013] Brandstädt, A., Brito, S., Klein, S., Nogueira, L. T., and Protti, F. (2013). Cycle transversals in perfect graphs and cographs. *Theoretical Computer Science*, 469:15–23.

[Brandstädt et al. 1998] Brandstädt, A., Le, V. B., and Szymczak, T. (1998). The complexity of some problems related to graph 3-colorability. *Discrete Applied Mathematics*, 89(1):59–73.

[Cowen et al. 1997] Cowen, L., Goddard, W., and Jesurum, C. E. (1997). Defective coloring revisited. *J. Graph Theory*, 24(3):205–219.

[da Cruz 2023] da Cruz, M. L. L. (2023). Near-bipartiteness on graphs having small dominating sets. Master's thesis, Universidade Federal Fluminense (UFF).

[Dross et al. 2017] Dross, F., Montassier, M., and Pinlou, A. (2017). Partitioning a triangle-free planar graph into a forest and a forest of bounded degree. *European Journal of Combinatorics*, 66:81–94.

[Garey and Johnson 1979] Garey, M. R. and Johnson, D. S. (1979). *Computers and intractability: A Guide to the Theory of NP-completeness*, volume 174.

[Grötschel et al. 1984] Grötschel, M., Lovász, L., and Schrijver, A. (1984). Polynomial algorithms for perfect graphs. *Ann. Discrete Math*, 21:325–356.

[Karp 1972] Karp, R. M. (1972). *Reducibility among combinatorial problems*. Springer.

[Li and Pilipczuk 2020] Li, S. and Pilipczuk, M. (2020). An improved FPT algorithm for independent feedback vertex set. *Theory of Computing Systems*, 64(8):1317–1330.

[Lima et al. 2017] Lima, C. V., Rautenbach, D., Souza, U. S., and Szwarcfiter, J. L. (2017). Decycling with a matching. *Infor. Proc. Letters*, 124:26 – 29.

[Lima et al. 2021] Lima, C. V., Rautenbach, D., Souza, U. S., and Szwarcfiter, J. L. (2021). On the computational complexity of the bipartizing matching problem. *Annals of Operations Research*.

[Misra et al. 2013] Misra, N., Narayanaswamy, N., Raman, V., and Shankar, B. S. (2013). Solving min ones 2-sat as fast as vertex cover. *Theoretical Computer Science*, 506.

[Misra et al. 2012] Misra, N., Philip, G., Raman, V., and Saurabh, S. (2012). On parameterized independent feedback vertex set. *Theoretical Computer Science*, 461:65–75.

[Protti and Souza 2018] Protti, F. and Souza, U. S. (2018). Decycling a graph by the removal of a matching: new algorithmic and structural aspects in some classes of graphs. *Discrete Mathematics & Theoretical Computer Science*, 20(2).

[Schaefer 1978] Schaefer, T. J. (1978). The complexity of satisfiability problems. In *Proceedings of the tenth annual ACM Symposium on Theory of Computing*.

[Yang and Yuan 2006] Yang, A. and Yuan, J. (2006). Partition the vertices of a graph into one independent set and one acyclic set. *Discrete Mathematics*, 306(12).