On the Minimum and Maximum Selective Graph Coloring Problems in some Graph Classes

Marc Demange[*] Tınaz Ekim[†] Bernard Ries[‡]

October 21, 2015

Abstract

Given a graph together with a partition of its vertex set, the *minimum selective coloring* problem consists of selecting one vertex per partition set such that the chromatic number of the subgraph induced by the selected vertices is minimum. The contribution of this paper is twofold. First, we investigate the complexity status of the minimum selective coloring problem in some specific graph classes motivated by some models described in [9]. Second, we introduce a new problem that corresponds to the worst situation in the minimum selective coloring; the *maximum selective coloring* problem aims to select one vertex per partition set such that the chromatic number of the subgraph induced by the selected vertices is maximum. We motivate this problem by different models and give some first results concerning its complexity.

Keywords: complexity; approximation; graph classes.

[*]RMIT University, Melbourne, Victoria, Australia, e-mail: marc.demange@rmit.edu.au

[†]Bogazici University, Istanbul, Turkey, e-mail: tinaz.ekim@boun.edu.tr

[‡]Université de Fribourg, DIUF, Fribourg, Switzerland, e-mail: bernard.ries@unifr.ch

This work was supported in part by TUBITAK PIA BOSPHORUS Grant No. 111M303 and PHC BOSPHORE 2012 PROJECT N 26284RB. M. Demange was also partially supported by the French Agency for Research under the DEFI program TODO, ANR-09-EMER-010. Both supports are greatly acknowledged.

1 Introduction

All graphs that we will consider in this paper are undirected and without loops and multiple edges. Let $G = (V, E)$ be such a graph. A *stable set* (resp. *clique*) is a subset $S \subseteq V$ of pairwise nonadjacent (resp. adjacent) vertices. An induced path on k vertices is denoted by P_k. The graph obtained by taking k disjoint copies of G (with no edges between any two copies) is referred to as kG. For $V' \subseteq V$, $G[V']$ denotes the subgraph of G induced by V'. A *k-coloring* of G is a mapping $c : V \to \{1, \ldots, k\}$ ($c(u)$ is called the *color* of vertex u) such that $c(u) \neq c(v)$ for all $uv \in E$. The smallest integer k such that G is k-colorable is called the *chromatic number* of G and is denoted by $\chi(G)$. Given a graph $G = (V, E)$, the problem of deciding whether G is k-colorable is called k-Colorability and the problem of computing $\chi(G)$ together with a $\chi(G)$-coloring is called Minimum Coloring. k-Colorability is well known to be NP-complete for any $k \geq 3$ [13].

In previous works [8, 9], we have motivated and investigated a new kind of coloring problem called *minimum selective coloring*, denoted by Sel-Col. Given a graph $G = (V, E)$, consider a partition $\mathcal{V} = (V_1, V_2, \ldots, V_p)$ of the vertex set V of G. The sets $V_1, \ldots, V_p$ are called *clusters* and $\mathcal{V}$ is called a *clustering*. A *selective k-coloring* of G with respect to $\mathcal{V}$ is a mapping $c : V' \to \{1, \ldots, k\}$, where $V' \subseteq V$ with $|V' \cap V_i| = 1$ for all $i \in \{1, \ldots, p\}$, such that $c(u) \neq c(v)$ for all $uv \in E$. A *selection* is a subset of vertices $V' \subseteq V$ such that $|V' \cap V_i| = 1$ for all $i \in \{1, \ldots, p\}$. Thus, determining a selective k-coloring with respect to $\mathcal{V}$ consists in finding a selection V' such that $G[V']$ admits a k-coloring.

We may define the following two problems:

Sel-Col
Input: An undirected graph $G = (V, E)$ and a clustering $\mathcal{V} = (V_1, \ldots, V_p)$ of V.
Output: A selection V^* such that $\chi(G[V^*])$ is minimum.

Let $k \geq 1$ be a fixed integer.

k-Dsel-Col
Input: An undirected graph $G = (V, E)$ and a clustering $\mathcal{V} = (V_1, \ldots, V_p)$ of V.
Question: Does there exist a selection V' such that $\chi(G[V']) \leq k$? We call such a selection a *k-colorable selection*.

For any $k \geq 1$, k-Dsel-Col is clearly in NP in general graphs and consequently we will not mention NP-membership in the NP-completeness proofs. In [8], the complexity of the problem has been studied in some graph classes including split graphs, threshold graphs, complete bipartite graphs, nC_4's and nP_3's. In [9] several applications have been presented that motivate the problem in other graph classes (indicated in parenthesis in the next sentence). These applications concern routing and wavelength assignment (edge intersection graphs of paths in different kinds of host graphs), dichotomy-based constraint encoding (twin graphs), antenna positioning and frequency assignment ((unit) disk graphs), scheduling ((linear) interval graphs), multiple stacks TSP (permutation graphs) and berth allocation (rectangle intersection graphs). Some of these applications also motivate the particular case when all clusters are cliques; this case is called *compact clustering*. One such application is antenna positioning and frequency assignment where each set of antennas forming a cluster has pairwise intersecting impact areas because of coverage constraints. Another application comes from timetabling problems where for each event to be scheduled, the set of available time periods that form a cluster is around a given time because although some flexibility is provided, the time periods cannot deviate too much from some prescribed time (see [9] for more details). The compact clustering was already considered in [8] for mainly theoretical results, without being justified by applications.

Note that Sel-Col has previously been studied in the class of edge intersection graphs of paths under the name of *partition coloring* or *path coloring*; the main motivation in these studies was to solve the second phase (namely the wavelength assignment) of the Routing and Wavelength Assignment problem ([11, 12, 17, 18, 19], see [9] for more details on the content of these references). To the best of our knowledge, this application, and therefore the class of edge intersection graphs of paths, is the only one considered in the framework of selective coloring before [9]; as a consequence all graph classes considered in the present paper are new with respect to the study of the complexity of the selective coloring problem.

In the first part of this paper (Section 2) we study the complexity study of the problem by focusing on graph classes related to applications described in [9].

In the second part (Section 3) we introduce the opposite problem consisting in determining a selection that maximizes the chromatic number of the corresponding induced subgraph. This corresponds to the worst possible solution when considering Sel-Col. We call this new problem *maximum selective coloring* problem, and denote it by Sel-Col+. It can be formally defined as follows:

Sel-Col+
Input: An undirected graph $G = (V, E)$; a clustering $\mathcal{V} = (V_1, \ldots, V_p)$ of V.
Output: A selection V^* such that $\chi(G[V^*])$ is maximum.

Graph class	Sel-Col	Sel-Col+
Twin	2-Dsel-Col NP-c even if all clusters are two adjacent vertices, Prop 2.1	?
Planar UDG	1-Dsel-Col NP-c even if $\Delta \leq 3$, compact clustering with $\lvert V_i \rvert = 2$ or 3, $\forall i = 1, \ldots, p$, and the intersection model given, Prop 2.2 (the same shown without compact clustering in [8])	?
Linear Interval	k-Dsel-Col in $O(\lvert V\rvert + \lvert E\rvert)$ with consecutive clustering, Thm 2.8 (NP-c with general clustering [8])	Linear, Cor 3.1
Interval	k-Dsel-Col is in P if k is fixed and with compact clustering, Thm 2.4 (NP-c with compact clustering but k is not fixed, Thm 2.3 by [10])	Linear, Cor 3.1
Chordal	Already NP-hard in unit/proper/linear interval graphs [8]	Linear, Cor 3.1
Complete k-partite	NP-hard [8]	Polynomial, Prop 3.2
Permutation	1-DselCol NP-c even with sparse clustering [8]	NP-hard even with compact clustering, Rem 3.2. $\chi^+_{SEL}(G, \mathcal{V})$-approximation algorithm even in comparability graphs, Prop 3.3 and Alg 4

Table 1: Summary of the complexity situation for Sel-Col and Sel-Col+.

In the applications described in [9], the selection process was completely controlled by the user trying to minimize some scarce resource. In contrast, if one has no full control over the selection process, the selection does not necessarily minimize the use of this resource. In such a case, it becomes important to measure how bad a selection can be with respect to the use of the resources. When facing such situations, it becomes important to evaluate the impact of the worst possible selection on the resources. This motivates Sel-Col+. After introducing this new problem, we start investigating its complexity in some particular cases.

Let us summarize the complexity results on Sel-Col and Sel-Col+ in Table 1 where results with no bibliographic reference are the ones obtained in this paper and a question mark means that the complexity of the problem in the related graph class is not known.

2 Complexity of Minimum Selective Coloring

In this section, we investigate the complexity of Sel-Col in several graph classes each motivated by an application in [9]. We provide both NP-hardness and polynomial time solvability results.

2.1 Twin graphs

Let $S = \{s_1, \ldots, s_n\}$ be a set of n states. An *encoding* of length k is a mapping from S into $\{0,1\}^k$. A *dichotomy* in S is an unordered pair $\{P, Q\}$ of disjoint subsets of S, with $P \cup Q \neq \emptyset$. An encoding of S *satisfies* a dichotomy $\{P, Q\}$ if there is at least one component that takes value 0 for all states in P and value 1 for all states in Q, or vice versa. Let $C = \{D_1, \ldots, D_p\}$ be a set of dichotomies in S with $D_i = \{P_i, Q_i\}, i = 1, \ldots, p$, the Constrained Encoding problem consists in finding an encoding of S of minimum length which satisfies all the dichotomies in C.

With an instance (S, C) of the Constrained Encoding problem, we associate a graph G and a clustering $\mathcal{V}$ as follows (see [6]): with each dichotomy $D_i \in C$, we associate two vertices (P_i, Q_i) and (Q_i, P_i) corresponding to the two possible oriented pairs. Vertices (S, T) and (S', T') are linked by an edge if and only if $S \cap T' \neq \emptyset$ or $S' \cap T \neq \emptyset$; in particular (P, Q) and (Q, P) are linked by an edge. The clustering $\mathcal{V} = (V_1, \ldots, V_p)$ is defined by $V_i = \{(P_i, Q_i), (Q_i, P_i)\}$, for $i = 1, \ldots, p$. It is shown in [9] that for an integer $k \geq 1$, there exists an encoding of S of length at most k satisfying all dichotomies in C if and only if $(G, \mathcal{V})$ is selective k-colorable.

A graph $G = (V, E)$ constructed from an instance (S, C) as described above is called a *twin graph*. See Figure 1 for an example of a twin graph for states $S = \{1, 2, 3, 4\}$ and dichotomies $C = \{\{\{1,3\},\{2\}\}; \{\{1\},\{2,4\}\}; \{\{1,2\},\{3,4\}\}; \{\{1,4\},\{3\}\}\}$. The selection shown with circled vertices is 3-colorable and corresponds to an encoding φ of S of length 3: the first color includes $\{\{1,3\},\{2\}\}$ and $\{\{1\},\{2,4\}\}$, the second one includes $\{\{3,4\},\{1,2\}\}$ and the third one includes $\{\{1,4\},\{3\}\}$; the state 1 will be coded $\varphi(1) = (010)$ since for the first and the third colors it appears in the first part of the dichotomy while in the second color it appears in the second part of the dichotomy; for state 2, two equivalent choices are possible: $\varphi(2) = (110)$ or $\varphi(2) = (111)$ (see [9] for more information about this model).

In what follows we suppose that the twin graphs are given with a related list of dichotomies, which avoids to consider the problem of deciding whether a given graph is a twin graph. As

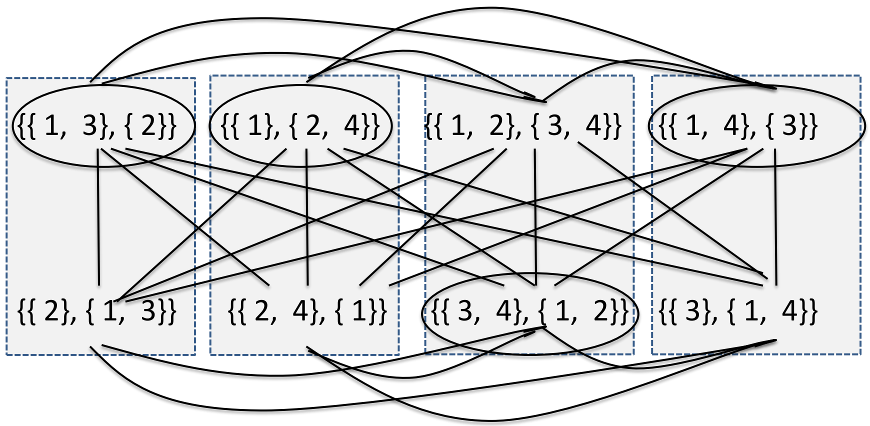

Figure 1: An example of a twin graph for $S = \{1,2,3,4\}$ and dichotomies $C = \{\{\{1,3\},\{2\}\};\{\{1\},\{2,4\}\};\{\{1,2\},\{3,4\}\};\{\{1,4\},\{3\}\}\}$.

mentioned in [9], a twin graph is 1-selective colorable if and only if it is bipartite. Consequently 1-Dsel-Col is polynomial in twin graphs. The following result shows that deciding whether there exists an encoding of length at most 2 satisfying all dichotomies of a given set is NP-complete.

Proposition 2.1 2-Dsel-Col *is NP-complete in twin graphs even if all clusters consist of two adjacent vertices.*

Proof: We use a reduction from 4-Colorability which is NP-complete (see [13]). Consider a graph $G = (V,E)$ on n vertices $v_1,\ldots,v_n$. Without loss of generality, we may assume that G has no isolated vertex.

Consider now an arbitrary orientation of the edges of G and associate with every vertex $v_i \in V$ the oriented pair (P_i,Q_i) where P_i is the set of edges oriented as arcs starting at v_i and Q_i is the set of edges oriented as arcs ending at v_i. Each non-oriented pair $\{P_i,Q_i\}$, $v_i \in V$, is a dichotomy in E since G has no isolated vertex.

Let us now construct the corresponding twin graph. With each dichotomy $\{P_i,Q_i\}$, $v_i \in V$, we associate two vertices (P_i,Q_i) and (Q_i,P_i). By definition of a twin graph, we link two vertices (P_i,Q_i), (P_j,Q_j), $i \neq j$, if and only if $P_i \cap Q_j \neq \emptyset$ or $P_j \cap Q_i \neq \emptyset$. This holds if and only if $v_iv_j \in E$: in this case $\{v_iv_j\} = P_i \cap Q_j$ if v_iv_j is oriented from i to j and $\{v_iv_j\} = P_j \cap Q_i$ else. Similarly (Q_i,P_i), (Q_j,P_j), $i \neq j$ are linked if and only if $v_iv_j \in E$.

Finally, two vertices (P_i, Q_i), (Q_j, P_j) are linked if and only if $i = j$. As a consequence, the resulting twin graph $G' = (V', E')$ corresponds to two disjoint copies of G linked by a perfect matching consisting of edges $(P_i, Q_i)(Q_i, P_i)$, $v_i \in V$. We then consider the clustering $\mathcal{V} = \{V_0, V_1, \ldots, V_n\}$ defined by $V_i = \{(P_i, Q_i), (Q_i, P_i)\}$, $i = 0, 1, \ldots, n$. By construction, $(G', \mathcal{V})$ is selective 2-colorable if and only if V can be partitioned into to sets that both induce a bipartite graph in G. This is equivalent to say that G is 4-colorable, which concludes the proof. $\square$

As mentioned in [9], the structure of Sel-Col allows to distinguish two subproblems: the *selection problem* which aims to select one vertex per cluster (formally finding a feasible solution of Sel-Col) and the problem of deciding whether a given selection induces a k-colorable graph. This second problem, called *evaluation problem*, corresponds to evaluating the objective function of a given feasible solution. In [9], some examples are presented which illustrate that the complexity status of these two problems (the selection problem and the evaluation problem) are independent and that the hardness of Sel-Col can result either from the selection problem or from the evaluation problem or from both.

To conclude this section, we show that 3-Dsel-Col remains hard even for a subclass of twin graphs for which an optimal selection is known. This implies that the evaluation problem is hard for twin graphs.

We consider a slight modification of the reduction given in Proposition 2.1. We start from a graph $G = (V, E)$ with $V = \{v_1, \ldots, v_n\}$ corresponding to an instance of 3-Colorability which is NP-complete. From any orientation of G, we define the sets P_i, Q_i as previously and add a state s_0 to each set P_i to obtain $P'_i = \{s_0\} \cup P_i$. The twin graph associated with the dichotomies $\{P'_i, Q_i\}$ is composed of two copies $G_1 = (V^1, E^1), G_2 = (V^2, E^2)$ of G that are completely linked, denoted by ${}^2G$: V^1 corresponds to all pairs (P'_i, Q_i) and V^2 to all symmetric pairs (Q_i, P'_i). Indeed, $\forall i, s_0 \notin Q_i$ and consequently the introduction of s_0 does not change the edges between two vertices $(P'_i, Q_i), (P'_j, Q_j)$ or between two vertices $(Q_i, P'_i), (Q_j, P'_j)$. However every two vertices (P'_i, Q_i), (Q_j, P'_j), are adjacent since $s_0 \in P'_i \cap P'_j$.

Taking the clustering $\mathcal{V} = \{V_0, V_1, \ldots, V_n\}$ defined by $V_i = \{(P'_i, Q_i), (Q_i, P'_i)\}, i = 0, 1, \ldots, n$, we have $\chi_{SEL}({}^2G, \mathcal{V}) = \chi(G)$. Indeed, V_1 (or V_2) being a selection, we have $\chi_{SEL}({}^2G, \mathcal{V}) \leq \chi(G)$ and any coloring of a selection gives a coloring of the original graph since no two vertices in different copies of G can receive the same color; so, $\chi_{SEL}({}^2G, \mathcal{V}) \geq \chi(G)$. Consequently both V^1 and V^2 correspond to an optimal selection, making the selection process trivial in this case. In terms of encoding schemes, this represents a situation where the selection process is easy (select for instance all oriented dichotomies (P,Q) with $s_0 \in P$) but deciding whether a 3-dimensional encoding can represent all dichotomies is still hard since it corresponds to 3-Colorability in G. Obviously, deciding whether 2 dimensions are enough is polynomial in this case.

2.2 Unit Disk Graphs

Next we consider the minimum selective coloring problem in unit disk graphs. This is motivated by a frequency assignment problem [9] which will be reconsidered in Section 3.1. The model also motivates the compact clustering case. A graph $G = (V,E)$ is a *(unit) disk graph*, denoted by (U)DG, if one can associate with every vertex v a disk (of radius 1) in the plane such that two vertices are adjacent if and only if the corresponding disks intersect. Deciding whether a given graph is a UDG is known to be NP-complete [3]. Hence, we will suppose here that an intersection model is given with the graph.

Note also that k-Dsel-Col is NP-complete in UDGs for any fixed $k \geq 3$ even if all clusters contain a single vertex since under this assumption, the problem is equivalent to k-Colorability which is known to be NP-complete in planar unit disk graphs for $k = 3$ [5, 20] and in unit disk graphs for $k > 3$ [16] even if an intersection model is known. In [8], it is shown that 1-Dsel-Col is NP-complete in graphs isomorphic to nP_3, even if clusters are of size either 2 or 3. This class is trivially included in planar UDGs and consequently, 1-Dsel-Col is NP-complete in planar UDGs with clusters of size 2 or 3, even if an intersection model is known. Here we go further by investigating the complexity of the compact clustering case.

Proposition 2.2 *1-Dsel-Col is NP-complete in planar UDGs of maximum degree 3 with compact clustering and clusters containing either 2 or 3 vertices, even if an intersection model is known.*

Proof: Our reduction combines ideas from the proof of NP-completeness of 3-Colorability in planar UDGs [5, 20] and from the proof of NP-completeness of 1-Dsel-Col in planar graphs of maximum degree 3 with compact clustering and clusters of size 2 or 3 given in [8].

We will use in particular a reduction from Restricted Planar 3-Sat which was shown to be NP-complete in [7] and which is defined as follows: we are given a set X of variables as well as a set C of clauses over X such that each clause contains either 2 or 3 literals;

furthermore each variable occurs exactly 3 times, once as a negative literal and twice as a positive literal; finally the bipartite graph $H = (X \cup C, E)$, where $xc \in E$ if the variable corresponding to x appears (as positive or negative literal) in the clause corresponding to c, is planar; we want to decide whether there exists a truth assignment such that each clause contains at least one true literal.

Let $\mathcal{I}$ be an instance of Restricted Planar 3-Sat with variables $x_1, \ldots, x_n$ and clauses $c_1, \ldots, c_m$. We first revisit the construction proposed in [8]. Consider the planar bipartite graph $H = (X \cup C, E)$ associated with $\mathcal{I}$ and a vertex $x_i \in X$, corresponding to variable x_i, as well as its neighbors $c_j, c_k, c_\ell \in C$ corresponding to the clauses in which x_i appears. Suppose without loss of generality that x_i appears as a negative literal in c_k (and hence it appears as a positive literal in c_j and in c_ℓ). We delete x_i from H and replace it by the graph H_i with vertex set $\{x_i^1, x_i', x_i'', \overline{x_i}, x_i^2\}$ and edge set $\{x_i^1 x_i', x_i' x_i^2, x_i' x_i'', x_i'' \overline{x_i}\}$ (these edges are called *variable edges*). Then we make c_j adjacent to x_i^1, c_k adjacent to $\overline{x_i}$ and c_ℓ adjacent to x_i^2. We do this for every vertex $x_i \in X$ and call H' the resulting graph which is still planar and has maximum degree 3. Let $Z = \{x_i^1, x_i^2, \overline{x_i}, i = 1, \ldots, n\}$ be the set of vertices representing the occurrences of the variables in the clauses. For every clause $c_j \in C$, we will denote by $z_j^h \in Z, j \in \{1, \ldots, m\}, h = 1, 2, 3$ the vertices representing the occurrences of the literals appearing in clause c_j (if a clause contains only 2 literals, we simply set $h = 1, 2$).

We now complete the construction of [8] as follows. Consider a vertex $c_j \in C$, associated with clause c_j, and its neighbors in H' denoted by z_j^1, z_j^2 and eventually z_j^3 representing the 2 or 3 literals appearing in clause c_j. We remove vertex c_j from H' and replace it by an edge with vertices c_j^1, c_j^2 if c_j contains two literals and by a triangle with vertices c_j^1, c_j^2, c_j^3 if c_j contains 3 literals (these edges are called *clause edges*). We then add edges $z_j^h c_j^h$ for $h = 1, 2$ if c_j contains 2 literals and for $h = 1, 2, 3$ if c_j contains 3 literals. We will call these edges *H-edges*. We do so for every clause $c_j \in C$ and denote by H'' the resulting graph. Clearly, in H'', H-edges are in one-to-one correspondence with the edges of the graph H. The graph H'' is clearly still planar and has maximum degree 3. The edge set of H'' is made of variable edges, clause edges and H-edges. We finally replace every H-edge $e = z_j^h c_j^h$, $j \in \{1, \ldots, m\}, h \in \{1, 2, 3\}$ by an even length path $z_j^h, y_j^{h,1}, \ldots, y_j^{h,2k_{jh}+1}, c_j^h$ and denote by $H^\star$ the resulting graph. As shown in [5] it is possible to choose integers $k_{jh}, j \in \{1, \ldots, m\}, h \in \{1, 2, 3\}$ such that $H^\star$ is a UDG and moreover an intersection model can be built in polynomial time. Note that $H^\star$ is still planar and has maximum degree 3.

The reader is referred to Figure 2 for an example of the graph $H^\star$ associated with the instance $(x_1 \vee \overline{x_3}) \wedge (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_3)$ of Restrictive Planar 3-Sat. The graph is

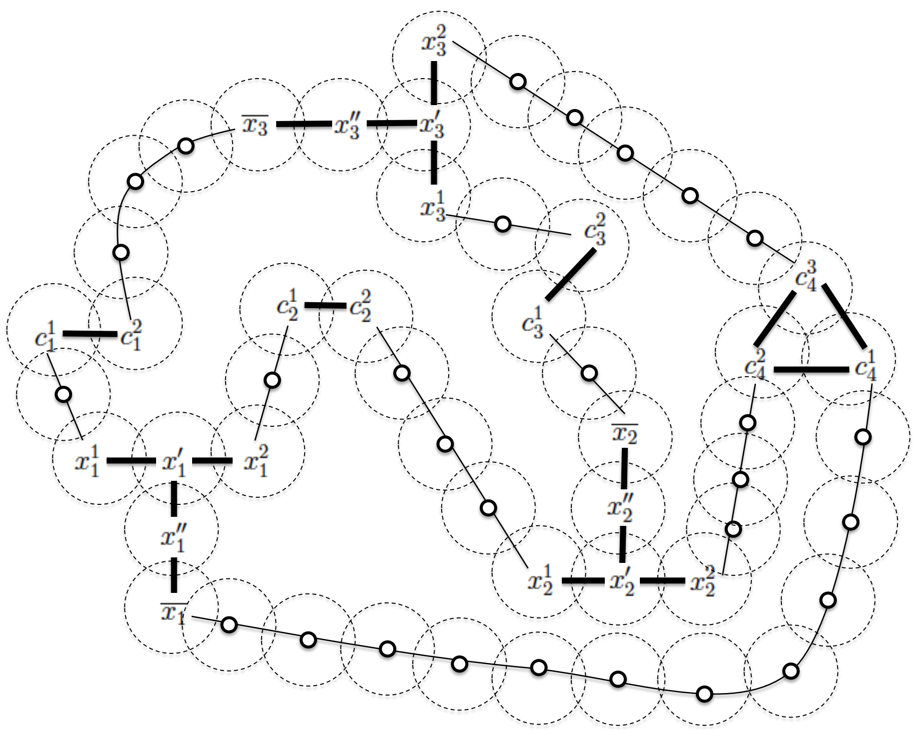

Figure 2: An example of graph $H^\star$ with an intersection model for the instance $(x_1 \vee \overline{x_3}) \wedge (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee x_2 \vee x_3)$.

represented together with an intersection model. Thick edges are clause and variable edges while thin edges correspond to the edges of the even length paths.

To complete the reduction we define a clustering $\mathcal{V} = (V_1, \ldots, V_p)$ for $H^\star$ as follows: for every $i \in \{1, \ldots, n\}$, x'_i and x''_i form a cluster; for every $j \in \{1, \ldots, m\}$, $c_j^h, h = 1, 2$ or $h = 1, 2, 3$ form a cluster of size 2 or 3; for every H-edge we define clusters $\{z_j^h, y_j^{h,1}\}, \{y_j^{h,2}, y_j^{h,3}\}, \ldots,$ $\{y_j^{h,2k_{jh}}, y_j^{h,2k_{jh}+1}\}$, all of size 2.

The reduction can be done in polynomial time and the instance $(H^\star, \mathcal{V})$ of 1-Dsel Col satisfies all conditions of the proposition. To complete the proof we need to show that the instance $\mathcal{I}$ of Restricted Planar 3-SAT is a yes-instance if and only if $(H^\star, \mathcal{V})$ is 1-selective colorable. Suppose first there is a truth assignment satisfying all clauses in $\mathcal{I}$. For every variable x_i, $i \in \{1, \ldots, n\}$, select x'_i and $\overline{x_i}$ in the solution if x_i is false and x''_i together with x_i^1 and x_i^2 if x_i is true. For every clause c_j, we select one vertex c_j^h such that z_j^h is a true literal and include it in the solution. Then, for every $j \in \{1, \ldots, m\}$ and $h \in \{1, 2\}$ (resp. $h \in \{1, 2, 3\}$), we consider the path $z_j^h, y_j^{h,1}, \ldots, y_j^{h,2k_{jh}+1}, c_j^h$ and complete the solution with $y_j^{2\ell}, \ell = 1, \ldots, k_{jh}$ if the literal associated with z_j^h is true and with $y_j^{2\ell+1}, \ell = 0, \ldots, k_{jh}$ if the literal associated with z_j^h is false. It can be easily verified that the resulting solution is a stable set intersecting each cluster exactly once, which shows that $(H^\star, \mathcal{V})$ is 1-selective colorable.

Conversely, suppose that $(H^\star, \mathcal{V})$ is 1-selective colorable and let S be the corresponding stable set. For every variable x_i, $i \in \{1, \ldots, n\}$, if $x'_i \in S$ then variable x_i is set to false and if $x''_i \in S$ then variable x_i is set to true. This defines a truth assignment for every variable. For every clause $c_j, j \in \{1, \ldots, m\}$, we consider the vertex in S belonging to the cluster associated with c_j. We need to show that this vertex, say c_j^h, corresponds to a true literal z_j^h. In the path $z_j^h, y_j^{h,1}, \ldots, y_j^{h,2k_{jh}+1}, c_j^h$, since $c_j^h \in S$ necessarily $\forall \ell \in \{1, \ldots, k_{jh}\}, y_j^{2\ell} \in S$ and thus $z_j^h \in S$, meaning that z_j^h is true in the truth assignment we have computed. This concludes the proof. $\square$

In Sections 2.3 and 2.4 we will present some polynomial cases of the minimum selective coloring problem in interval graphs (IG) with compact clustering when k is fixed and unit interval graphs (UIG) with consecutive clustering (definitions will be given in these sections). These particular cases can also be seen as special cases of the present problem in DGs and UDGs which contain respectively IGs and UIGs.

2.3 Interval graphs

In [8, 9], some scheduling problems are introduced which motivate the study of the minimum selective coloring problem in interval graphs, in particular with compact clustering. A graph $G = (V, E)$ is called an *interval graph*, if one can associate an interval on the real line with every vertex such that two intervals intersect if and only if the corresponding vertices are adjacent. The following was shown in [10].

Theorem 2.3 *[10] The decision version of* SEL-COL *is NP-complete in the strong sense in interval graphs with compact clustering, even if the vertices of the graph can be partitioned into three cliques.*

However, for a fixed k the problem to decide whether a k-colorable selection exists becomes polynomial.

Theorem 2.4 *For any fixed k, Algorithm 2 solves k-*DSEL-COL *in polynomial time for interval graphs G with compact clustering $\mathcal{V}$. Moreover, if $\chi_{SEL}(G, \mathcal{V}) \leq k$, then it computes a selective k-coloring.*

Proof: <u>Notations</u>

We consider an instance of k-DSEL-COL defined by an interval graph $G = (V, E)$, whose vertices are associated with a set of n intervals $\mathcal{I} = \{I_1, \ldots, I_n\}$, and a compact clustering $\mathcal{V} = (V_1, \ldots, V_p)$ of V. Since vertices represent intervals, we will use the terms vertices and intervals indifferently. We suppose that G is given by its interval representation. For every $r \in \{1, \ldots, p\}$, we denote by G_r the graph $G_r = G[V_1 \cup \ldots \cup V_r]$. Every solution of the problem restricted to G_r corresponds to a selection of r intervals $\mathcal{J}_r = \{J_1, \ldots, J_r\}$ with $J_i \in V_i, i = 1, \ldots, r$ and will be called a *partial selection* on G_r or just of order r if the related graph is defined without ambiguity; a partial selection of order p is just a solution of the original problem.

<u>Main idea</u>

Algorithm 2 solves the problem by dynamic programming. It is based on a notion of *states* defined below. States are associated with partial solutions (i.e. selections on graphs G_r, $1 \leq r \leq p$). Many different solutions may be associated with a given state, however the definition of states ensures that all partial selections with the same state are either all feasible or all unfeasible; more precisely feasibility of a partial selection can be immediately and easily stated by reading its state. A state associated with feasible partial selections is called *feasible*; the total number of feasible states is $O(n^k)$. As shown by the analysis, one only needs to compute one particular feasible solution for any feasible state.

The dynamic programming process consists in filling in a table T line by line. T has one column per feasible state and one line per cluster, each entry $T[r,\sigma]$ being either a k-colorable partial selection of order r and of state σ or $\emptyset$ if such partial selection does not exist. Line r ($2 \leq r \leq p$) is filled in from line $r-1$; to this aim, given a state σ of a partial selection of order $r-1$, Algorithm 1 computes the state $\tilde{\sigma}$ of a partial selection of order r obtained by adding one particular interval $I \in V_r$ to a partial selection of order $r-1$. $\tilde{\sigma}$ only depends on σ and I.

<u>Clusters' ordering</u>

The hypothesis that the clustering is compact allows us to define a specific order of clusters. For every cluster $V_i, i \in \{1,\ldots,p\}$, we consider $H_i = \cap_{I \in V_i} I$; since the clustering is compact, $H_i \neq \emptyset, i = 1,\ldots,p$. We assume that the clusters are ordered so that the intervals $H_i, i = 1,\ldots,p$ are in non-decreasing order of their left endpoint.

<u>Definition of states</u>

Consider a partial selection of order r, i.e. r intervals $\mathcal{J}_r = \{J_1,\ldots,J_r\}$ with $J_i \in V_i, i = 1,\ldots,r$. We denote $J_i = [a_i,b_i]$ and associate with $\mathcal{J}_r$ its state σ defined by an array of $(k+1)$ values $\sigma = [S_0,\ldots,S_k]$, where $S_\ell = \max\{x, |\{J \in \mathcal{J}_r, x \in J\}| \geq k+1-\ell\}$ with $\max(\emptyset) = -\infty$. $\sigma = [S_0,\ldots,S_k]$ is a non-decreasing sequence. It is feasible if and only if $S_0 = -\infty$, which means that the interval graph associated with $\mathcal{J}_r$ is k-colorable.

Note that for all $\ell \in \{0,\ldots,k\}$, we have $S_\ell \in \{-\infty\} \cup \{b_1,\ldots,b_r\}$ (so S_ℓ can take at most $n+1$ different values) and consequently, the number of different states for any partial solution is at most $(n+1)^{(k+1)}$ and the number of feasible states is at most $(n+1)^k$. We denote by Σ the set of all possible feasible states, in one-to-one correspondance with the set of all non-decreasing sequences of k numbers, each belonging to $\{-\infty\} \cup \{b, \exists a, [a,b] \in \mathcal{I}\}$.

<u>Analysis</u>

The **for**-loop from line 2 to line 4 of Algorithm 2 fills in the first line of T by computing all possible states corresponding to selecting one interval in V_1. Suppose that the $r-1$ first lines are all filled in, with $2 \leq r \leq p$; then given a solution $\mathcal{J}_{r-1}$ on G_{r-1} and an interval $I = [a,b] \in V_r$, the state $\widetilde{\sigma}$ of the solution $\mathcal{J}_{r-1} \cup \{I\}$ on G_r only depends on the state σ associated with $\mathcal{J}_{r-1}$ and the interval I. This state is computed in $O(k)$ time by Algorithm 1. Line 3 of Algorithm 1 deals with values of $\widetilde{S}_\ell$'s not affected by I. To justify the correctness

Algorithm 1 Updating states

Require: A feasible state $\sigma = [S_0, \dots, S_k]$ of a partial selection $\mathcal{J}_{r-1}$ on G_{r-1}, $2 \leq r \leq p$, and an interval $I = [a, b] \in V_r$.

Ensure: The state $\widetilde{\sigma} = [\tilde{S}_0, \dots, \tilde{S}_k]$ of the solution $\mathcal{J}_{r-1} \cup \{I\}$ on G_r.

$\ell_0 \leftarrow \min(\{\ell, S_\ell > b\} \cup \{k+1\})$
for $\ell \geq \ell_0$ AND $\ell \leq k$ **do**
    $\widetilde{S}_\ell = S_\ell$
$\widetilde{S}_{\ell_0 - 1} = b$
$\ell_1 \leftarrow \min(\{\ell, S_\ell \geq a\} \cup \{\ell_0 - 1\})$
for $\ell_1 \leq \ell \leq \ell_0 - 1$ **do**
    $\widetilde{S}_{\ell-1} \leftarrow S_\ell$
for $\ell \leq \ell_1 - 2$ **do**
    $\widetilde{S}_\ell \leftarrow S_\ell$

Algorithm 2 k-DSEL-COL in interval graphs with compact clustering

Require: An interval graph G defined by a set of intervals $\mathcal{I}$ and a compact clustering $\mathcal{V} = (V_1, \dots, V_p)$.

Ensure: Either a selection $\mathcal{J} \subset \mathcal{I}$ inducing a k-colorable graph or the information that it does not exist.

Order clusters in non-decreasing order of left endpoint of $\cap_{I \in V_i} I$
for $I = [a, b] \in V_1$ **do**
    $\sigma \leftarrow (-\infty, \dots, -\infty, b)$
    $T[1, \sigma] \leftarrow \{I\}$
for $r = 2$ **to** p **do**
    for $I = [a, b] \in V_r$ **do**
        for σ such that $T[r-1, \sigma] \neq \emptyset$ **do**
            Compute the state σ' of solution $T[r-1, \sigma] \cup I$ by using Algorithm 1
            if σ' is feasible **then**
                $T[r, \sigma'] \leftarrow T[r-1, \sigma] \cup \{I\}$
if $\forall \sigma \in \Sigma, T[p, \sigma] = \emptyset$ **then**
    G is not k-selective colorable
else
    Select a state σ such that $T[p, \sigma] \neq \emptyset$
    return $T[p, \sigma]$

of Line 4 of Algorithm 1, note that the ordering chosen for clusters guarantees that, for every $I' = [a', b'] \in V_1 \cup \ldots \cup V_{r-1}$ and $I = [a, b] \in V_r$, we have $a' < b$. If $\ell_0 = k+1$, then no interval in $\mathcal{J}_{r-1}$ ends after b and we have $b = \widetilde{S}_k$. If $\ell_0 \leq k$ and $S_{\ell_0 - 1} < b$, then, using the above property, exactly $k + 1 - \ell_0 + 1$ intervals contain b: I and the $k + 1 - \ell_0$ intervals of $\mathcal{J}_{r-1}$ containing S_{ℓ_0}. If $S_{\ell_0 - 1} = b$, then the same holds but we have $\widetilde{S}_{\ell_0 - 1} = \widetilde{S}_{\ell_0 - 2} = b$. Finally, Line 7 and Line 9 (of Algorithm 1) allow to compute values of $\widetilde{S}_\ell$ for $\ell < \ell_0$. Algorithm 2 computes all possible feasible states of feasible solutions in lines 1 to p of table T. A k-colorable selection corresponds then to a feasible state on the pth line of T. The complexity of Algorithm 2 is $O(n(n+1)^k) = O(n^{k+1})$: Line 2 (of Algorithm 2) takes $O(|\Sigma|)$, while Line 8 is executed $O(n|\Sigma|)$ times and requires Algorithm 1 of complexity $O(k)$. This concludes the proof. $\square$

Given an interval graph G defined by the set I of intervals and a compact clustering $\mathcal{V} = (V_1, \ldots, V_p)$, we denote by $G_\mathcal{V}$ the intersection graph of the sets $C_i = \cup_{I \in V_i} I$. Since $\mathcal{V}$ is compact, $G_\mathcal{V}$ is an interval graph and its clique number $\omega(G_\mathcal{V})$ can be computed in polynomial time [15]. It is straightforward to verify that $\chi_{SEL}(G, \mathcal{V}) \leq \omega(G_\mathcal{V}) \leq \omega(G)$. So, if $G_\mathcal{V}$ has bounded clique number, then the previous result applies and Algorithm 2 computes $\chi_{SEL}(G, \mathcal{V})$.

Corollary 2.5 *If G is an interval graph with compact clustering $\mathcal{V}$ and if $\omega(G_\mathcal{V}) \leq k$ for a constant k, then $\chi_{SEL}(G, \mathcal{V})$ can be computed in polynomial time.*

A *circular arc graph* $G = (A, E)$ defined by a set A of arcs on a circle has a vertex for each arc in A and two vertices are adjacent if they correspond to intersecting arcs; of course interval graphs constitute a subclass of circular arc graphs. Since there is a one-to-one correspondence between vertices and arcs of a circular arc graph, we will use both terms interchangeably. Let the *load* $\lambda(G)$ be the maximum number of arcs a point on the circle may belong to. Of course, in the particular case of an interval graph, the load is equal to the clique number and to the chromatic number. In a circular arc graph however, we have $\lambda(G) \leq \omega(G) \leq \chi(G)$ since circular arc graphs are not perfect (moreover, k-Colorability is NP-complete in circular arc graphs, but becomes polynomial if k is fixed [14]). Given a clustering $\mathcal{V}$ of G we can define the Minimum Selective Load problem in the same way as the Minimum Selective Clique problem, denoting by $\lambda_{SEL}(G, \mathcal{V})$ its optimal value.

Consider then a point x on the circle and denote by A_x the set of arcs in A containing x; $G[A \setminus A_x]$ is an interval graph. Consider then a set of arcs $P \subset A_x$ such that $|P \cap V_i| \leq$

$1, i = 1, \ldots, p$, and for every arc $p \in P$ we break it at x so as to define two disjoint arcs p^- and p^+ respectively before and after x in a clockwise orientation of the circle. Then $\widetilde{G} = G[(A \setminus (A_x \cup \bigcup_{i, V_i \cap P \neq \emptyset} V_i)) \cup \{p^-, p^+, p \in P\}]$ is an interval graph and each arc $a \in (A \setminus (A_x \cup \bigcup_{i, V_i \cap P \neq \emptyset} V_i)) \cup \{p^-, p^+, p \in P\}$ can be immediately associated to an interval $I(a)$ such that $\{I(a), a \in (A \setminus (A_x \cup \bigcup_{i, V_i \cap P \neq \emptyset} V_i)) \cup \{p^-, p^+, p \in P\}\}$ is an interval representation of $\widetilde{G}$. Moreover, defining $\widetilde{\mathcal{V}} = (V_i \setminus (V_i \cap A_x), i$ such that $V_i \cap P = \emptyset, \{p^-\}, \{p^+\}, p \in P)$, $\omega_S(\widetilde{G}, \widetilde{\mathcal{V}})$ is exactly the value of a minimum selective load in G for which P is the related selection among A_x.

If $\lambda(G) \leq k$ for a constant k, one can use this for every set $P \subset A_x$ satisfying $|P \cap V_i| \leq 1, i = 1, \ldots, p$, so we have:

Corollary 2.6 *$\lambda_{SEL}(G, \mathcal{V})$ can be computed in polynomial time in circular arc graphs of bounded load and for a clustering $\mathcal{V}$ satisfying $\cap_{a \in V_i} a \neq \emptyset, i = 1, \ldots, p$.*

2.4 Linear Interval Graphs

In [9], a quality test scheduling problem was introduced which motivated the study of the minimum selective coloring problem in *linear interval graphs* (LIG), introduced in [4] and defined below. Let L be a line and V a finite set of points of L; given a set of intervals from L (an interval means a proper subset of L homeomorphic to $[0, 1]$), the related linear interval graph $G = (V, E)$ has vertex set V and $u, v \in V$ are adjacent in G if u, v belong both to a same interval. It follows from this definition that if v_i is adjacent to v_j, with v_i being to the left of v_j in L, then v_i, v_j and all vertices lying between these two vertices in L form a clique. We say that $v_i < v_j$ if v_i lies on the left of v_j in L. When dealing with selective coloring in this graph class, a clustering $\mathcal{V} = \{V_1, \ldots, V_p\}$ of the vertex set is called *consecutive* if for every set $V_\ell \in \mathcal{V}$ we have the following property: if $v_i, v_j \in V_\ell$ with $v_i < v_j$, then $v_r \in V_\ell$ for all v_r such that $v_i \leq v_r \leq v_j$. Linear interval graphs occurred in the quality test scheduling problem described in [9] and consecutive clustering was naturally appearing in this application. As illustrated in Theorem 2.8, this restriction allows to get a polynomial result while the problem is hard in LIG under general clustering (see Table 1).

Before explaining how we can solve the problem in polynomial time in this graph class with consecutive clustering, let us mention a few properties concerning linear interval graphs. First we will show that linear interval graphs are equivalent to *proper interval graphs* (PIG) (i.e., interval graphs admitting an interval representation in which no interval properly contains

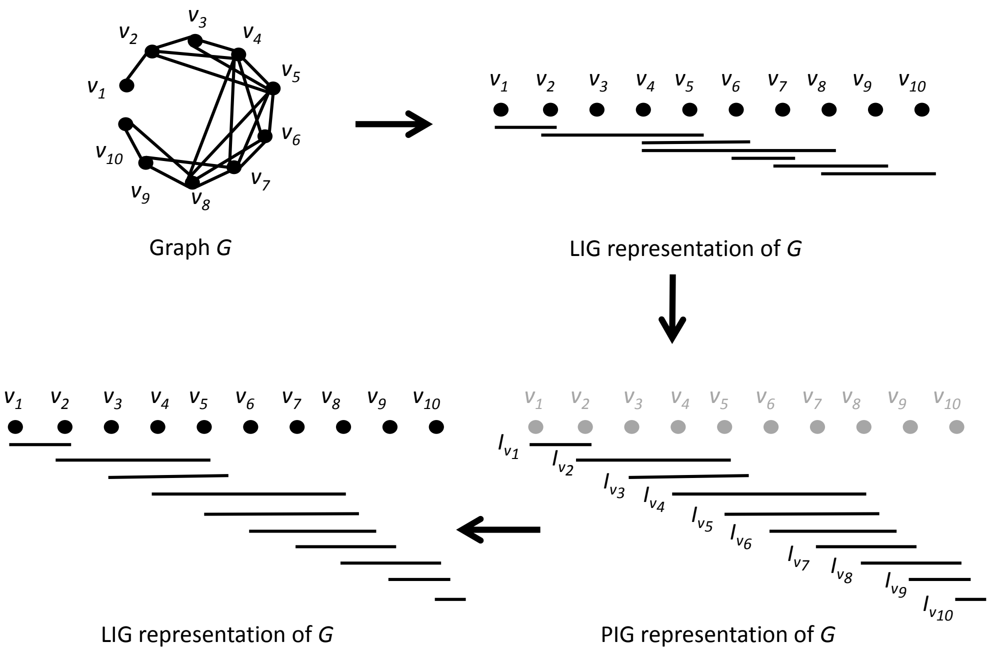

Figure 3: Transformations between LIG and PIG representations.

another), which are in turn equivalent to *unit interval graphs* (UIG) (i.e., interval graphs admitting an interval representation in which each interval has unit length) as shown in [21].

Proposition 2.7 *A graph $G = (V, E)$ is a LIG if and only if it is a PIG. Moreover, given any LIG representation R_{LIG} of G with points $v_1, \ldots, v_n$, one can construct in linear time (in the size of R_{LIG}) a PIG representation of G such that the order of left endpoints of the intervals $I_{v_1}, \ldots, I_{v_n}$ is the same as the order of points $v_1, \ldots, v_n$ in R_{LIG} and vice versa.*

Proof: The reader is referred to Figure 3 for the following constructions.

Assume G is a LIG and consider a LIG representation of G with points $v_1, \ldots, v_n$ on the real line L and intervals representing cliques. Now, one can obtain a proper interval representation of G by assigning an interval I_{v_i} to each vertex v_i in the following manner: I_{v_i} starts at the point v_i on L and ends at the rightmost right endpoint of all the intervals containing the point v_i. Note that if $v_j > v_i$ then the right endpoint of I_{v_j} is not smaller than the right endpoint of I_{v_i}. Now, if in the resulting representation there are more than one interval with the same right endpoint, order them according to their left endpoints (which are all different since all points v_i are distinct). Let $I_{v_i}, \ldots, I_{v_{i+j}}$ be these ordered intervals (they necessarily correspond to consecutive points) and v_r their common right end point in L. Then replace $I_{v_{i+t}} = [v_i, v_r]$ with $[v_i, v_r + t \times \epsilon]$ for $t = 1, \ldots, j$ and $\epsilon > 0$. It is easy to see that, by definition, such intervals never properly contain another. Moreover, by choosing ϵ sufficiently small ($\epsilon < \min |v_j - v_i|/n$) this does not change the related interval graph. Besides, this is a proper interval representation of G. Indeed, if two vertices u, v are adjacent in G, then in its LIG representation, there is an interval I containing their corresponding points. Consequently, in the above described PIG representation, I_u and I_v contain the right endpoint of I and therefore u and v are adjacent. Now, if two vertices u and v are non-adjacent in G, then it means that none of the intervals in its LIG representation contains both points representing u and v. Assume without loss of generality that the point u lies on the left of the point v on the real line L. Then we have in particular that the interval containing u and having the rightmost right endpoint does not contain v and therefore I_u does not contain the point v which is the starting point of I_v; hence u and v are non-adjacent and G is a PIG.

Conversely, assume $G = (V, E)$ is a PIG and consider a PIG representation of G. We need to show that we can define points on the real line L and a set of intervals of L such that $u, v \in V$ are adjacent if and only if they are both contained in some interval. For each interval I in the proper interval representation, we add a point on the real line L corresponding to the left endpoint of I and consider the same set of intervals as in the PIG representation. We claim that this is a LIG representation of G. Consider two adjacent vertices $u, v \in V$ and the corresponding intervals I_u, I_v. Without loss of generality, assume that the left endpoint of I_u comes before the left endpoint of I_v. Then the corresponding points on L belong both to the interval I_u. Now let $u, v \in V$ be two non-adjacent vertices in G. Then their intervals I_u and I_v do not intersect. Without loss of generality, we may assume that the left endpoint of I_u comes before the left endpoint of I_v. If there was an interval I containing the corresponding points on L then this interval would necessarily contain I_u, a contradiction since we started with a PIG representation. Thus G is a LIG.

To conclude, it is enough to observe that the described representations have the desired property. □

In [8], it is shown that 1-DSEL-COL is NP-complete in linear/proper/unit interval graphs. The next theorem shows that with consecutive clustering it can be solved in linear time.

Theorem 2.8 *k-DSEL-COL in LIG with consecutive clustering can be solved in time* $O(|V|+$

Algorithm 3 k-DSEL-COL in LIG with consecutive clustering

Require: A LIG $G = (V, E)$ with consecutive clustering $\mathcal{V} = \{V_1, \ldots, V_p\}$.
Ensure: Yes, if a selective k-coloring exists, No if it does not exist.

```
Set V* = {v*_1, ..., v*_k} where v*_i is the first (i.e., leftmost) vertex of V_i for i = 1, ..., k.
for i = 1 to p − k do
    if V* ∩ V_i is not complete to V_{i+k} then
        select the first (i.e., leftmost) non-neighbor of V* ∩ V_i in V_{i+k}, denoted by v*_{i+k}, and
        add it to V*;
    else
        return No, (G, 𝒱) is not selective k-colorable;
return Yes, G[V*] is k-colorable and hence (G, 𝒱) is selective k-colorable.
```

$|E|$).

Proof: Let $G = (V, E)$ be a LIG and let $\mathcal{V} = \{V_1, V_2, \ldots, V_p\}$ be a consecutive clustering of V. We apply Algorithm 3 to the graph G; using the LIG representation of G it first selects the leftmost possible vertex in the k first clusters and then greedily selects (when it is possible) one vertex in clusters $k+1, \ldots, p$ in such a way it does not create a clique of size $k+1$. As shown in the analysis, if the algorithm fails at some step in selecting one vertex in one cluster, then the graph is not k-selective colorable and in the opposite case it builds a k-colorable selection.

Clearly, Algorithm 3 runs in time $O(|V| + |E|)$ since each vertex and each edge is considered at most once. Thus we are left with the proof of correctness of Algorithm 3. Note that for any selection V', $G[V']$ is k-colorable if and only if $G[V']$ does not contain a clique of size greater than or equal to $k+1$, since LIG are perfect graphs (this follows from Proposition 2.7 and the fact that interval graphs are perfect).

First assume that Algorithm 3 finds a vertex in each cluster and let V^* be the set of chosen vertices. Suppose by contradiction that $G[V^*]$ is not k-colorable. It follows from the above that it necessarily contains a clique of size greater than or equal to $k+1$. The definition of LIG and the fact that $|V^* \cap V_i| = 1, i = 1 \ \ldots p$, imply that in such a case, there necessarily exists a clique K of size $k+1$ in $G[V^*]$ induced by vertices $v_l^*, \ldots, v_{l+k}^*$ for some $l \in \{1, \ldots, p-k\}$. But this contradicts the fact that we choose in V_{l+k} a non-neighbor of $V^* \cap V_l = \{v_l^*\}$. Thus, if Algorithm 3 computes V^* such that $|V^* \cap V_i| = 1, i = 1, \ldots, p$, then $G[V^*]$ is k-colorable and hence G admits a selective k-coloring.

Now assume that Algorithm 3 does not find such a set V^*. Hence there exists $V_i \in \mathcal{V}$ such that $V^* \cap V_i = \{v_i^*\}$ is adjacent to all the vertices in V_{i+k}. It follows from the definition of

LIG, that the vertices in the sets $V_{i+1}, \dots, V_{i+k}$ form a clique K and in addition that v_i^* as well as all the vertices $v_j \in V_i$ such that $v_i^* < v_j$ are pairwise adjacent and complete to K. Thus, if a solution exists, then it must necessarily contain a vertex $v_r \in V_i$ such that $v_r < v_i^*$ since otherwise we will always get a clique of size greater than or equal to $k+1$. But now, $\{v_{i-k}^*, v_{i-k+1}^*, \dots, v_{i-1}^*, v_r\}$ form a clique of size $k+1$ since v_i^* was the first non-neighbor of v_{i-k}^*. Repeating the same argument, we conclude that we must choose a vertex $v_s \in V_{i-k}$ such that $v_s < v_{i-k}^*$. Continuing in the same manner, we finally conclude that we must choose a vertex $v \in V_j$, $j \in \{1, \dots, k\}$ such that $v < v_j^*$, which is clearly impossible. Thus no solution exists. □

3 Maximum Selective Graph Coloring Problem

In this section, we first consider some applications of Sel-Col in order to emphasize the use of Sel-Col+ in each one of these contexts. Readers may refer to [9] for more details on various models which are briefly described here. Second, we consider the complexity and the approximability of Sel-Col+ in graph classes encountered in the aforementioned applications, namely, perfect graphs, chordal graphs and comparability graphs.

3.1 Motivation

Let us consider the *Antenna Positioning and Frequency Assignment Problem* (AP-FAP) where a GSM operator has to decide for each base station a position among a predefined set such that the number of frequencies assigned to the base stations while avoiding all interferences is minimized. If each position is represented by a vertex, two vertices corresponding to positions that are close enough for possible interference (in case the same frequency is assigned to the antennae) are adjacent, and the vertex set corresponding to possible positions for a base station forms a cluster, then AP-FAP boils down to Sel-Col in this clustered (unit) disk graph. Now, assume that a central institution or an external stakeholder decides where to install the base stations in each region instead of the GSM operator. This may be preferable for instance in order to ensure that the electromagnetic waves are restricted to a certain level and/or to optimize some other criteria in terms of the overall GSM network including all operators. In this case, although the operator has a set of predefined locations for each base station, the selection of the location is not made in a way to minimize the total number of frequencies to be used. However, it is important for the operator to assess the maximum number of frequencies needed in the worst case. In other words, the operator is interested in knowing the cost of the worst selection.

In the framework of *Scheduling Problems*, Sel-Col models the problem of minimizing the use of some resource while all jobs are scheduled, given that for each job, one can choose a period among a set of available time periods. Indeed, this problem corresponds to Sel-Col in the graph having a vertex per available time period, edges between vertices whose corresponding time periods intersect and clusters consisting of vertices corresponding to the set of available time periods for a same job. In [9], timetabling for speakers in a conference, quality test scheduling and berth allocation problems are the scheduling problems considered within this framework, illustrating Sel-Col in respectively interval graphs, linear interval graphs and rectangle intersection graphs. Let us focus on the specific example of the construction of a timetabling and imagine that we are given the available time periods of each speaker but the speakers are free to choose the period they will use (or the scheduling of the speakers will be made only a few days before the conference starts). However, for organizational reasons, one has to book the seminar rooms in advance. Since each room has a cost, we want to book a minimum number of rooms, but of course, there should be enough rooms for all speakers whichever period they choose (or they are scheduled to). Consequently, we have to book as many rooms as the value of the chromatic number corresponding to the worst possible selection. We will see in Section 3.2 that unlike Sel-Col, one can solve Sel-Col+ in polynomial time in interval graphs (even in chordal graphs containing interval graphs).

Another application that motivates the study of Sel-Col+ in permutation graphs is the so-called *Multiple Stacks TSP* [9] where items should be collected from some pick-up network and distributed in some delivery network. This time, we will see that Sel-Col+ is also NP-hard in permutation graphs just like Sel-Col and consequently, in Section 3.2 we provide an approximation algorithm with performance guarantee in this case.

Motivated by the above applications, the *maximum selective coloring* problem Sel-Col+ is the problem of finding the worst selection, i.e. the selection which needs a maximum number of colors. More formally, given a graph $G = (V, E)$ and a clustering $\mathcal{V}$ of V, Sel-Col+ is the problem of finding the largest integer k for which G admits a selection V^* such that $\chi(G[V^*]) = k$. This optimal value is called the *worst selective chromatic number*, denoted by $\chi^+_{SEL}(G, \mathcal{V})$ and a selection V^* realizing $\chi^+_{SEL}(G, \mathcal{V})$ is called a *worst selection*.

Given a graph $G = (V, E)$ and a partition $\mathcal{V}$ of V, we also define the Maximum Selective

Clique problem as the problem of finding a selection V^* such that $\omega(G[V^*])$ is maximized. The size of such a clique, called *maximum selective clique*, is denoted by $\omega^+_{SEL}(G, \mathcal{V})$. Clearly, for any $(G, \mathcal{V})$, we have $\omega^+_{SEL}(G, \mathcal{V}) \leq \chi^+_{SEL}(G, \mathcal{V})$. Note that $\omega^+_{SEL}(G, \mathcal{V})$ is equal to the maximum number of clusters a single clique can intersect. Given such a clique K intersecting ℓ clusters, an optimal selection for the Maximum Selective Clique of value ℓ can be obtained by selecting one vertex in K for clusters intersecting K and by completing this set to a selection by arbitrarily choosing a vertex from the remaining clusters. Similarly, to approximate ω^+_{SEL} it is enough to compute in polynomial time a clique intersecting a large number of clusters (see Proposition 3.3).

By definition, for any $(G, \mathcal{V})$ we have $\chi_{SEL}(G, \mathcal{V}) \leq \chi^+_{SEL}(G, \mathcal{V})$. However, it can be noted that $\chi_{SEL}(G, \mathcal{V}) \leq \omega^+_{SEL}(G, \mathcal{V})$ does not necessarily hold. Indeed, consider a 5-cycle C_5 where each vertex forms a cluster by itself. Clearly, we have $\chi_{SEL}(G, \mathcal{V}) = 3$ but $\omega^+_{SEL}(G, \mathcal{V}) = 2$.

Note that, similarly to $\omega^+_{SEL}(G, \mathcal{V})$, we can define $\omega_{SEL}(G, \mathcal{V})$ as the minimum value of $\omega(G[V'])$ among all possible selections V' in $(G, \mathcal{V})$.

Remark 3.1 *Let $G = (V, E)$ be a perfect graph with partition $\mathcal{V}$ of V. Then $\chi^+_{SEL}(G, \mathcal{V}) = \omega^+_{SEL}(G, \mathcal{V})$ and $\chi_{SEL}(G, \mathcal{V}) = \omega_{SEL}(G, \mathcal{V})$.*

Let us note at that point that unlike in perfect graphs, it is not enough to require these equalities for all induced subgraphs in order to obtain a meaningful definition of selective-perfectness. A formal notion of selective-perfect graphs is introduced and studied in [2].

3.2 Complexity and Approximation results

To the best of our knowledge, Sel-Col+ has not been considered yet in the literature. However the above models motivate its systematic study for different graph classes. Here, we do a first step in this direction by investigating first complexity questions for Sel-Col+. It is straightforward to see that, in the general case, there is no link between the complexity of Sel-Col and Sel-Col+. In particular, given a graph $G = (V, E)$ with a clustering $\mathcal{V} = \{V_1, \ldots, V_p\}$, it suffices to add to G a stable set (resp. a clique) of size p with no edge between G and the stable set (resp. the clique); then we add exactly one vertex of the stable set (resp. the clique) to each cluster. Thus we obtain a new graph $\widetilde{G}$ with a clustering $\widetilde{\mathcal{V}}$ verifying:

$$\chi_{SEL}(\widetilde{G}, \widetilde{\mathcal{V}}) = 1 \text{ (select the vertices of the stable set) and } \chi^+_{SEL}(\widetilde{G}, \widetilde{\mathcal{V}}) = \chi^+_{SEL}(G, \mathcal{V}).$$

(resp. $\chi^+_{SEL}(\widetilde{G}, \widetilde{\mathcal{V}}) = p$ (select the vertices of the clique) and $\chi_{SEL}(\widetilde{G}, \widetilde{\mathcal{V}}) = \chi_{SEL}(G, \mathcal{V})$.

On the other hand, for instances with only one vertex per cluster both problems SEL-COL and SEL-COL+ are equivalent from a complexity point of view, showing in particular that SEL-COL+ is hard in general.

In the sequel, we present first complexity and approximation results for SEL-COL+; in particular we point out some cases where $\chi^+_{SEL}(G, \mathcal{V})$ can be computed in polynomial time while the computation of $\chi_{SEL}(G, \mathcal{V})$ is NP-hard and give some theoretical links between both problems. This provides first ideas for a more systematic study of the complexity of SEL-COL+.

We have seen that interval graphs are of special interest for both SEL-COL and SEL-COL+. In [8], it is shown that computing $\chi_{SEL}(G, \mathcal{V})$ is NP-hard in unit/proper/linear interval graphs. However, $\chi^+_{SEL}(G, \mathcal{V})$ can easily be computed in an even larger class of graphs, namely chordal graphs. A graph is *chordal* if it does not contain any induced cycle of length at least four.

Proposition 3.1 *Let $\mathcal{G}$ be a class of perfect graphs for which we can enumerate all maximal cliques in polynomial time. Then,* SEL-COL+ *can be solved in polynomial time in $\mathcal{G}$.*

Proof: Let $G = (V, E)$ be a graph in $\mathcal{G}$ and let $\mathcal{V}$ be a partition of V. From Remark 3.1, it follows that it is enough to determine $\omega^+_{SEL}(G, \mathcal{V})$. To this end, we enumerate all maximal cliques of G, which can be done in polynomial time. Then we choose the clique K intersecting a maximum number of clusters. In each cluster intersecting K we select exactly one vertex from K and complete the selection by choosing one vertex in each remaining cluster arbitrarily. $\square$

Since for chordal graphs we can enumerate all maximal cliques in time $O(n)$ (see [15]), we obtain the following corollary.

Corollary 3.1 SEL-COL+ *can be solved in linear time in chordal graphs.*

Another graph class for which SEL-COL is NP-hard [8] but SEL-COL+ can be solved in polynomial time is the class of complete k-partite graphs. A graph $G = (V, E)$ is a *complete k-partite graph* if its vertex set can be partitioned into k stable sets $U_1, \ldots, U_k$ such that between any two stable sets U_i, U_j, $i \neq j$, there are all possible edges.

Proposition 3.2 Sel-Col+ *can be solved in polynomial time in complete k-partite graphs.*

Proof: Let $G = (U_1, \ldots, U_k, E)$ be a complete k-partite graph and let $\mathcal{V} = (V_1, \ldots, V_p)$ be a clustering of V. Since complete k-partite graphs are perfect, it follows from the above and Remark 3.1 that it is enough to determine $\omega^+_{SEL}(G, \mathcal{V})$. In order to do so, we will reduce our problem to a Maximum Flow problem which can be solved in polynomial time (see for instance [1]). Let us denote by $v^i_1, \ldots, v^i_{|V_i|}$ the vertices in cluster V_i, for $i = 1, \ldots, p$. Let us now construct the following network: consider the vertices $v^i_1, \ldots, v^i_{|V_i|}$, for $i = 1, \ldots, p$; for every cluster V_i, $i = 1, \ldots, p$, we add a vertex w_i and add all arcs (w_i, v^i_j) for $j = 1, \ldots, |V_i|$ and $i = 1, \ldots, p$; for every set U_ℓ, $\ell = 1, \ldots, k$, we add a vertex u_ℓ; then, for every vertex $v^i_j \in U_\ell$, we add an arc (v^i_j, u_ℓ); we add two vertices s, t (the source and the sink of our network) as well as the arcs (s, w_i) for $i = 1, \ldots, p$ and (u_ℓ, t) for $\ell = 1, \ldots, k$; finally, we set the capacity of each arc to 1. This clearly gives us a network N which can be constructed in polynomial time given the graph G.

Now we claim that $\omega^+_{SEL}(G, \mathcal{V}) = q$ if and only if the value of a maximum flow in N is q. Indeed, let K be a clique of size q in G intersecting each cluster at most once. Let $V_{i_1}, \ldots, V_{i_q}$ be the clusters in G containing exactly one vertex of K. Clearly each such vertex belongs to a different partition set among $U_1, \ldots, U_k$. Without loss of generality, we may assume that $\{v^{i_j}_1\} = K \cap V_{i_j} \cap U_{i_j}$, for $j = 1, \ldots, q$. Then we obtain a flow in N of value q as follows: for $j = 1, \ldots, q$, we set the flow value to one on each arc $(s, w_{i_j}), (w_{i_j}, v^{i_j}_1), (v^{i_j}_1, u_{i_j}), (u_{i_j}, t)$.

Conversely, assume that there exists a flow of value q in N. Since all capacities of the arcs are equal to one and each vertex, except s and t, has exactly one outgoing arc or one incoming arc, it follows that there exist q vertex disjoint paths from s to t (not considering vertices s and t). For each such path $(s, w_i), (w_i, v^i_j), (v^i_j, u_\ell), (u_\ell, t)$, we consider vertex v^i_j in cluster V_i. Then it follows from the construction of N and the fact that the paths are vertex-disjoint (not considering vertices s and t) that these q vertices induce a clique intersecting q distinct clusters. □

A class of graphs $\mathcal{C}$ is called *auto-complementary* if for all $G \in \mathcal{C}$ we have $\bar{G} \in \mathcal{C}$, where $\bar{G}$ is the complement of G.

Remark 3.2 *Let $\mathcal{C}$ be an auto-complementary class of perfect graphs. If* 1-Dsel-Col *is NP-complete in $\mathcal{C}$, then* Sel-Col+ *is NP-hard in $\mathcal{C}$.*

Proof: Sel-Col+ in a graph $G \in \mathcal{C}$ with a clustering $\mathcal{V}$ consists in finding a clique intersecting a maximum number of clusters in G. This is equivalent to finding a stable set intersecting a maximum number of clusters in $(\bar{G}, \mathcal{V})$. It is straightforward to verify that 1-Dsel-Col polynomially reduces to this last problem. $\square$

As a consequence, Sel-Col+ is NP-hard in permutation graphs (i.e., comparability graphs whose complements are also comparability graphs; see definition below) even with compact clustering since 1-Dsel-Col is NP-complete in permutation graphs even if each cluster in $\mathcal{V}$ is a stable set (sparse clustering) [8]. Note also that, if G is a perfect graph and each cluster in $\mathcal{V}$ is a stable set, then $\chi^+_{SEL}(G, \mathcal{V}) = \omega(G)$ and hence it can be computed in polynomial time.

For NP-hard cases, it is natural to ask whether the problem can be approximated in polynomial time. The next proposition gives a first approximation result for Sel-Col+ in comparability graphs which generalize permutation graphs.

Given an undirected graph G, a *transitive orientation* of G is the assignment of orientations to the edges of G in such a way that if xy and yz are respectively oriented from x to y and from y to z, then there is an edge xz oriented from x to z. A graph G is a *comparability graph* if its edges are transitively orientable. It is known that a graph G is a permutation graph if and only if both G and its complement $\bar{G}$ are comparability graphs [15].

Algorithm 4 Approximation algorithm for χ^+_{SEL} in comparability graphs.

Require: A comparability graph $G = (V, E)$ with a clustering $\mathcal{V} = \{V_1, \ldots, V_p\}$.
Ensure: A selection V' satisfying $\chi(G[V']) \geq \sqrt{\chi^+_{SEL}(G, \mathcal{V})}$; an optimal coloring of $G[V']$

```
Compute a transitive orientation O of G;
Construct partial subgraphs G_{1...p} and G_{p...1} of G (see proof of Proposition 3.3)
Compute a maximum clique in G_{1...p} and in G_{p...1} and let K' be the largest one;
Complete K' into a selection V' of (G, V) by greedily adding one vertex per cluster not
    intersecting K';
return V' and a minimum coloring of G[V'].
```

Proposition 3.3 *Let* $G = (V, E)$ *be a comparability graph with a clustering* $\mathcal{V} = \{V_1, \ldots, V_p\}$ *of* V*. Algorithm 4 is polynomial and approximates* $\chi^+_{SEL}(G, \mathcal{V})$ *within a ratio of* $\sqrt{\chi^+_{SEL}(G, \mathcal{V})}$.

Proof: Let $k = \chi^+_{SEL}(G, \mathcal{V})$. Using Remark 3.1 and the fact that comparability graphs are perfect, we have $k = \omega^+_{SEL}(G, \mathcal{V})$. We will show how to compute in polynomial time a clique intersecting at least $\sqrt{k}$ clusters. Completing it into a selection by adding one vertex per cluster not intersecting this clique we get a selection V' such that $\omega(G[V']) \geq \sqrt{k}$.

We consider a transitive orientation $\mathcal{O}$ of $G = (V, E)$ and then define two comparability graphs, $G_{1\dots p}$ and $G_{p\dots 1}$ with respect to this orientation. $G_{1\dots p} = (V, E_{1\dots p})$ (resp. $G_{p\dots 1} = (V, E_{p\dots 1})$) is a partial subgraph of G obtained by keeping only edges xy such that $x \in V_i$, $y \in V_j$ for $1 \leq i < j \leq p$ and xy is oriented from x to y (resp. from y to x) in $\mathcal{O}$. It is easy to verify that $\mathcal{O}$ induces a transitive orientation of both $G_{1\dots p}$ and $G_{p\dots 1}$. Indeed assume that there are edges xy and yz in $G_{1\dots p}$ oriented from x to y and from y to z. By definition of $G_{1\dots p}$, x, y and z belong respectively to clusters V_i, V_j and V_ℓ such that $i < j < \ell$ and since $\mathcal{O}$ is a transitive orientation, there is an edge xz (in G) oriented from x to z which also belongs to $G_{1\dots p}$. The same result holds for $G_{p\dots 1}$.

Let us consider a clique K^* in G of size $|K^*| = k = \chi^+_{SEL}(G, \mathcal{V})$ and such that $\forall i \in \{1, \dots, p\}, |K^* \cap V_i| \leq 1$. The orientation $\mathcal{O}$ induces a transitive orientation of K^* which gives an order on its vertices. So we may assume that $K^* = \{v_1, \dots, v_k\}$ such that $v_i v_j$ is oriented from v_i to v_j for all i and j such that $1 \leq i < j \leq k$. We then define a permutation $\sigma = (\sigma_1, \dots, \sigma_k)$ such that for all $i \in \{1, \dots, k\}, v_i \in V_{\sigma_i}$. We denote by G_σ the permutation graph, of order k, associated with σ. A stable set in G_σ, corresponding to an increasing sub-sequence of σ, is associated with a clique in $G_{1,\dots,p}$ of the same size. Similarly, a clique in G_σ, corresponding to a decreasing sub-sequence of σ, is associated with a clique in $G_{p,\dots,1}$ of the same size. So we have

$$\omega(G_{1,\dots,p}) \geq \alpha(G_\sigma), \tag{1}$$

$$\omega(G_{p,\dots,1}) \geq \omega(G_\sigma). \tag{2}$$

On the other hand, since G_σ is perfect and of order k we have $\alpha(G_\sigma)\omega(G_\sigma) \geq k$. Indeed, in a perfect graph G, we have $\chi(G) = \omega(G)$ and since every color class is a stable set (hence of size less than or equal to $\alpha(G)$), the number of vertices in G is at most $\alpha(G)\omega(G)$. Consequently, we have

$$\max(\omega(G_{1,\dots,p}), \omega(G_{p,\dots,1})) \geq \sqrt{k}. \tag{3}$$

Note that any clique of $G_{1,\dots,p}$ or $G_{p,\dots,1}$ defines a clique of G intersecting at most once each cluster since in $G_{1,\dots,p}$ and $G_{p,\dots,1}$, there is no edge between vertices of a same cluster. Consequently, by computing a maximum clique of $G_{1,\dots,p}$ and $G_{p,\dots,1}$ and taking the largest one, we get a clique of G intersecting at least $\sqrt{k}$ clusters, which shows the result. Furthermore, Algorithm 4 runs in polynomial time since determining a transitive orientation and computing a maximum clique are both polynomial in comparability graphs [15], which concludes the proof. $\square$

Finally, let us conclude with two remarks on the equivalence of some problems related to Sel-Col and Sel-Col+.

Remark 3.3 *Let $G = (V, E)$ be a graph and let $\mathcal{V} = \{V_1, \dots, V_p\}$ be a clustering of V. Then* 1-Dsel-Col *is equivalent to deciding whether $\omega^+_{SEL}(\bar{G}, \mathcal{V}) = p$.*

Remark 3.4 *Let $G = (V, E)$ be a graph and let $\mathcal{V} = \{V_1, \dots, V_p\}$ be a clustering of V. Then* 1-Dsel-Col *is equivalent to deciding whether $\chi^+_{SEL}(\bar{G}, \mathcal{V}) = p$.*

4 Conclusion

In this paper, we investigate the complexity of Sel-Col in graph classes motivated by various applications that were presented in [9]. In particular, we provide NP-hardness results for Sel-Col in twin graphs and unit disk graphs (even with specific clustering). However, we show that restricting the clustering may change the complexity status: although it is NP-complete in interval graphs, we show that, for a fixed k, k-Dsel-Col becomes polynomial in interval graphs with compact clustering. Similarly, Sel-Col is hard in linear interval graphs but becomes polynomial time solvable when restricted to consecutive clustering.

In addition, we introduce a new problem, Sel-Col+, which corresponds to evaluating the cost of the worst selection and thus provides an upper bound on χ_{SEL}. We emphasize that solving Sel-Col+ can be helpful in many contexts: we revisit some models for Sel-Col and show that Sel-Col+ can also be motivated by these models. We start to investigate the complexity of this new problem in different graph classes. In this paper, we mainly focus on comparing the complexity of Sel-Col and Sel-Col+. We give an example where Sel-Col is easy while Sel-Col+ is hard. Symmetrically we point out classes of graphs where Sel-Col is NP-hard but solving Sel-Col+ becomes polynomial time solvable. In this case, it gives an upper bound for the optimal value of Sel-Col. We also give an example where the hardness of Sel-Col implies the hardness of Sel-Col+. This is the case in permutation graphs and consequently Sel-Col+ is hard in permutation graphs and thus in comparability graphs. Finally, we give an approximation algorithm with a square-root factor performance guarantee for Sel-Col+ in comparability graphs. We leave as an open problem to find whether it can be approximated within a constant approximation ratio.

As further work, we plan to systematically study the complexity of Sel-Col+ in different graph classes and in particular in the classes motivated by applications.

References

[1] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, Network Flows: Theory, Algorithms and Applications, *Prentice Hall*, 1998.

[2] F. Bonomo, D. Cornaz, T. Ekim, B. Ries, Perfectness of clustered graphs, *Discrete Optimization* 10 (2013) 296–303.

[3] H. Breu, D. G. Kirkpatrick, Unit disk graph recognition is NP-hard, *Computational Geometry: Theory and Applications*, 9(1-2) (1998) 3–24.

[4] M. Chudnovsky, P. Seymour, Claw-free graphs VII. Quasi-line graphs, *Journal of Combinatorial Theory Series B* 102 (2012) 1267–1294.

[5] B. N. Clark, C. J. Colbourn, D. S. Johnson Unit disk graphs, *Discrete Mathematics*, 86(1–3) (1990) 165–177.

[6] O. Couderc, A New Paradigm for Dichotomy-based Constrained Encoding, *in Proceedings of Design, Automation and Test in Europe (DATE)*, Paris, 1998.

[7] E. Dahlhaus, P. D. Seymour, C. H. Papadimitriou, M. Yannakakis, The complexity of multiterminal cuts, *SIAM Journal on Computing* 23 (1994) 864–894.

[8] M. Demange, J. Monnot, P. Petrica, B. Ries, On the complexity of the selective graph coloring problem in some special classes of graphs, *Theoretical Computer Science* 540-541 (2014) 89–102.

[9] M. Demange, T. Ekim, B. Ries, C. Tanasescu, On Some Applications of the Selective Graph Coloring Problem, *European Journal of Operational Research* 240 (2015) 307–314.

[10] M. Demange, J. Monnot, B. Ries, Selective coloring in interval graphs, manuscript (2015).

[11] T. Erlebach, K. Jansen, The complexity of path coloring and call scheduling, *Theoretical Computer Science* 255 (2001) 33–50.

[12] Y. Frota, N. Maculan, T. F. Noronha, C. C. Ribeiro, A Branch-and-Cut Algorithm for Partition Coloring, *Networks*, 55(3) (2010) 194–204.

[13] M.R. Garey, D.S. Johnson, Computers and intractability, a guide to the theory of $\mathcal{NP}$-completeness, *Freeman, New York* (1979).

[14] M.R. Garey, D.S. Johnson, G.L. Miller, C.H. Papadimitriou, The complexity of coloring circular arcs and chords, *SIAM Journal on Algebraic and Discrete Methods* 1(2) (1980) 216-227

[15] M. C. Golumbic, *Algorithmic graph theory and perfect graphs.* Computer Science and Applied Mathematics, Academic Press, 1980.

[16] A. Gräf, M. Stumpf, G. Weissenfels, On Coloring Unit Disk Graphs, *Algorithmica* 20(3) (1998) 277–293.

[17] L. Guangzhi, R. Simha, The partition coloring problem and its application to wavelength routing and assignment, *in Proceedings of the First Workshop on Optical Networks*, 2000, Dallas.

[18] E. A. Hoshino, Y. Frota, C. C. de Souza, A Branch-and-Price Approach for The Partition Coloring Problem, *Operations Research Letters* 39(2) (2011) 132–137.

[19] T. F. Noronha, C. C. Ribeiro, Routing and wavelength assignment by partition colouring, *European Journal of Operational Research* 71(3) (2006) 797–810.

[20] R. Peeters, On coloring j-unit sphere graphs, Tilburg University, Technical Report, 1991.

[21] F.S. Roberts, Indifference graphs, Proof Techniques in Graph Theory, F. Harary, ed., *Academic Press*, NY (1969), 139–146.