Selective line-graphs and partition colorings

Denis Cornaz[1], Fabio Furini[1], Enrico Malaguti[2], and Alberto Santini[3]

[1]{denis.cornaz,fabio.furini}@dauphine.fr — Université Paris-Dauphine, Paris (France)
[2]enrico.malaguti@unibo.it — University of Bologna (Italy)
[3]alberto.santini@upf.edu — Pompeu Fabra University, Barcelona (Spain)

January 19, 2019

Abstract

We extend the definition of sandwich line-graphs, a class of auxiliary graphs the stable sets of which are in 1-to-1 correspondence with the colorings of the original graph, from graphs to partitioned graphs, this way, we obtain a one-to-one correspondence between stable sets and partition colorings.

Keywords: Selective/partition coloring. Line graph. Integer Linear programming.

1 Introduction

Throughout, G is a simple undirected graph with vertex set $V(G)$ and edge set $E(G)$, with no loops nor parallel edges. A subset of $V(G)$ is *stable* if it contains no edges and it is a *clique* if it has all possible edges. A *coloring of* G is a partition of $V(G)$ into stable sets. The *chromatic number* $\chi(G)$ is the minimum number of stable sets in a coloring of G. The *coloring problem*, that is determining $\chi(G)$, is one of the most studied in graph theory and optimization, and one of the most important class is probably that of perfect graphs, those the subgraphs of which have a coloring with no more colors than the biggest clique, see *e.g.* [18].

A generalization of the coloring problem is the *selective coloring problem* (SCP, also known as *the partition coloring problem*). In this version, the vertices of G are partitioned into disjoint clusters, which can be assumed stable sets, and only one vertex needs to be colored in each cluster. Throughout the paper, we let $\mathscr{P} = \{V_1, \ldots, V_{|\mathscr{P}|}\}$ denotes a partition of $V(G)$ into non-empty stable sets, and we will refer to the pair $(G, \mathscr{P})$ as a *partitioned graph*. The members V_j of $\mathscr{P}$ are called *clusters*.

A *selective coloring of* $(G, \mathscr{P})$ is a collection of pairwise disjoint stable sets $S_1, \ldots, S_k$ of G so that a S_i hits a cluster at most once and every cluster is touched by exactly one S_i. Formally, $|S \cap V_j| = 1$ for all $V_j \in \mathscr{P}$, where $S = \cup_i S_i$ is the (disjoint) union of the S_i's. The *selective chromatic number* $\chi(G, \mathscr{P})$ *of* $(G, \mathscr{P})$ is the minimum k over all selective colorings $S_1, \ldots, S_k$. In other words, select one vertex per cluster to induce a subgraph H of G with $\chi(H)$ minimum. Observe that if every cluster V_j is a singleton, then a selective coloring is a coloring and $\chi(G, \mathscr{P})$ is the usual chromatic number.

The SCP is important for its various applications [8], in particular in telecommunications [14, 15], and as a challenging problem to solve in practice [13, 11, 19]. Its algorithmical complexity has been investigated in several classes of graphs [9] including a worse-case version of the problem [7]. Conform and perfect partitioned graphs have been studied in [2], in particular, the class of i-threshold graphs is introduced to characterize primal-dual integrality of associated polyhedra. An auxiliary graph is introduced in [4, 17], called the *sandwich line-graph* in [6] where it is shown to provide a chromatic version of Galai identities and to strengthen Lovász's sandwich theorem, that is, the semidefinite programming (SDP) relaxation for the classical coloring problem. The sandwich line-graph is used in [3, 16] to strengthen the linear programming (LP) approach. In [1] this auxiliary graph is used to find classes of graphs in which the coloring problem can be solved in polynomial time. An extension of the coloring problem, with weights, namely Max-Coloring, has been studied for its applications in batch scheduling [10]. In [5], the approach of [4] to solve Max-Coloring with the auxiliary graph, is shown to be efficient in practice and is extended to other coloring problems.

A first contribution of this paper is a procedure to construct an auxiliary graph of a partitioned graph, to extend some properties of the sandwich line-graph to the SCP. A further generalization of the problem is the *Selective Max-Coloring Problem* (SMCP). In this version each cluster V_j has an associated weight $c(V_j)$. Each color gets the weight of the heaviest cluster it covers and the objective is to find a selective coloring of minimum weight, denoted $\chi(G, \mathscr{P}, c)$. A second contribution of this paper is to show that the line-graph method for the SCP can be easily extended to the SMCP and that it outperforms the natural Mixed-Integer model for the SMCP in computational tests.

2 A line-graph solution method for the Selective Coloring Problem

The *clique-partition number* $\overline{\chi}(G)$ is equal to the chromatic number $\chi(\overline{G})$ of the complementary graph $\overline{G}$ of G. An equivalent definition for a perfect graph is that is satisfies $\alpha(G) = \overline{\chi}(G)$ for all subgraphs, where the *stability number* $\alpha(G)$ *of* G is the maximum cardinality of a stable set of G. The *complementary partitioned graph* $(\overline{G}, \mathscr{P})$ *of* $(G, \mathscr{P})$ is the partitioned graph with an edge uv if and only if u, v belongs to different clusters and uv is not an edge of G. So the clusters remain stable sets in $(\overline{G}, \mathscr{P})$. We will also consider *partitioned digraph* $(\vec{G}, \mathscr{P})$ that is a partitioned graph where the edges of G have been orientated. A *selective clique-partition of* $(G, \mathscr{P})$ is a selective coloring of $(\overline{G}, \mathscr{P})$. The problem of determining the *selective clique-partition number* $\overline{\chi}(G, \mathscr{P}) = \chi(\overline{G}, \mathscr{P})$ is that of determining the selective chromatic number in the complementary partitioned graph. Trivially, any algorithm computing $\overline{\chi}(G,\mathscr{P})$ solves SCP.

A graph parameter $\beta(G)$ is said to be *sandwich* if it satisfies $\alpha(G) \leq \beta(G) \leq \overline{\chi}(G)$ for all graphs G. For instance, the fractional clique-partition number $\overline{\chi}_f(G)$ is sandwich, and the so-called sandwich theorem by Lovász states that the theta function $\vartheta(G)$ is a sandwich graph parameter [18].

The main definition in this section is the following:

Definition 1. *A pair of arcs $\{e,f\}$ of a digraph is* simplicial *if*

1. *e and f have the same tail,*
2. *the heads of e,f are adjacent.*

Let $L(G)$ be the *line-graph of* G, that is, it has vertex-set $V(L(G)) = E(G)$ the edge-set of G, with an edge $ef \in E(L(G))$ if e and f are adjacent in G. A *sandwich line-graph of* G is any graph built from $L(G)$ by removing edges corresponding to simplicial pairs of arcs in some orientation $\vec{G}$ of G without 3-dicycles. Given an orientation $\vec{G}$, the corresponding sandwich line-graph is denoted by $S(\vec{G})$. Before extending sandwich line-graphs to partitioned graphs, let us summarize their properties (see [6]).

For every sandwich graph parameter $\beta(\cdot)$, for any graph G, and for any orientation $\vec{G}$ of G without 3-dicycle, then the following *chromatic Gallai identities* holds:

$$\alpha(G) + \overline{\chi}(S(\vec{G})) = |V(G)| = \alpha(S(\vec{G})) + \overline{\chi}(G) \tag{1}$$

which implies that the modified parameter $|V(G)| - \beta(S(\vec{G}))$ is still sandwich. Restricted to triangle-free graphs, $S(\vec{G}) = L(G)$ and (1) is equivalent to the classical Gallai identities ($\alpha(G) + \tau(G) = |V(G)| = \nu(G) + \rho(G)$, see *e.g.* [18]). Sandwich line-graphs improve LP bound for stability ($\overline{\chi}_f$) and SDP bound for coloring (ϑ), since the following inequalities holds:

$$\alpha(G) \leq |V(G)| - \overline{\chi}_f(S(\vec{G})) \leq \overline{\chi}_f(G) \qquad \vartheta(G) \leq |V(G)| - \vartheta(S(\vec{G})) \leq \overline{\chi}(G) \tag{2}$$

Not all properties of sandwich graphs extends to partitioned graphs but in the following we extend the second equality in (1). As for sandwich line-graphs, there is a 1-to-1 correspondence between selective colorings and stable sets of the auxiliary graph.

Definition 2. *The* partition line-graph $L(G,\mathscr{P})$ of a partitioned graph $(G,\mathscr{P})$ *is the graph with vertex-set $V(L(G,\mathscr{P})) = E(G)$ the edge-set of G, with an edge $ef \in E(L(G,\mathscr{P}))$ if there is a cluster V_j so that both e and f have a vertex in V_j.*

Clearly, if every cluster is a singleton then $L(G,\mathscr{P}) = L(G)$. In fact, $L(G,\mathscr{P}) = L(G/\mathscr{P})$ where $G/\mathscr{P}$ is the graph obtained by contracting each cluster. Observe also that stable sets of $L(G,\mathscr{P})$ are matchings of G so that no cluster is covered more than once by an edge of the matching. In the following, we let $\vec{G}$ denotes some orientation of G.

Definition 3. *The* selective line-graph $S(\vec{G},\mathscr{P})$ of $(G,\mathscr{P})$ *is the graph obtained from the partition line-graph $L(G,\mathscr{P})$ by removing edges ef corresponding to simplicial pairs of $\vec{G}$.*

Theorem 1. *The equality $\overline{\chi}(G,\mathscr{P}) + \alpha(S(\vec{G},\mathscr{P})) = |\mathscr{P}|$ holds for any partitioned graph $(G,\mathscr{P})$ and any orientation $\vec{G}$ whithout 3-dicyle.*

Proof. First observe that if a selective clique-partition of $(G,\mathscr{P})$ has a clique of size one, then any vertex in the cluster containing this clique can be replaced by any other vertex of the cluster to form another selective clique-partition with as many cliques. In the following we assume that each cluster has a particular vertex which is automatically chosen to be the clique of size one if there is one in the cluster. Let us prove the equality now.

It suffices to show that there is a 1-to-1 correspondence (up to choosing an arbitrary vertex per cluster) between the set of all stable sets of the selective line-graph $S(\vec{G},\mathscr{P})$ and the set of all selective clique-partitions of $(G,\mathscr{P})$, and that this correspondence is so that $|\mathscr{K}|+|S|=|\mathscr{P}|$, for any pair $(\mathscr{K},S)$ in correspondence, where $\mathscr{K}$ is the selective clique-partition of $(G,\mathscr{P})$ and S the stable set of $S(\vec{G},\mathscr{P})$. Indeed, in particular, it would imply the equality for the two optima. The rest of the proof consists in describing the 1-to-1 correspondence.

Let S be a stable set of $S(\vec{G},\mathscr{P})$, so S is a set of arcs in the digraph $\vec{G}$. Let U be the set of the vertices of G covered by S and let $(U_1,S_1),\dots,(U_p,S_p)$ be the (non-trivial) components of the partial subdigraph (U,S) of $\vec{G}$. Since two adjacent arcs in S must form a simplicial pair of $\vec{G}$, it follows that (U_i,S_i) is a simplicial out-star of $\vec{G}$, that is, a partial subdigraph whose vertex-set $U_i=\{u_i,u_i^1,\dots,u_i^{|S_i|}\}$ is a clique of G and whose arc-set has the form $S_i=\{u_iu_i^1,\dots,u_iu_i^{|S_i|}\}$, where u_i is the center of S_i and $u_i^1,\dots,u_i^{|S_i|}$ are the leaves of S_i. Since every cluster is a stable set, clearly, no cluster of $(G,\mathscr{P})$ contains two vertices of the clique U_i. Since only two simplicial arcs of S may have a vertex in a same cluster, no cluster contains two centers u_i,u_j. Moreover, no cluster contains a center u_i and a leaf u_j^k. Hence every cluster contains at most one vertex of U. We can build a selective clique-partition $\mathscr{K}$ of $(G,\mathscr{P})$ by choosing the p non-trivial cliques (with at least two vertices) corresponding to the components of (U,S) and complete it with 1-cliques by choosing the particular vertex for each cluster not covered by S. The graph obtained from (V,S) by contracting each cluster is a forest with $|S|$ arcs, with $|\mathscr{K}|$ components, and with $|\mathscr{P}|$ vertices, hence $|\mathscr{K}|+|S|=|\mathscr{P}|$.

Let $\mathscr{K}=\{K_1,\dots,K_{|\mathscr{K}|}\}$ be a selective clique-partition of $(G,\mathscr{P})$. Since $\vec{G}$ has no 3-dicycle, every clique K_i induces an acyclic subdigraph of $\vec{G}$, which thus is spanned by exactly one out-star S_i. So the union $S=\bigcup_i S_i$ is a stable set in the selective line-graph $S(\vec{G},\mathscr{P})$. Moreover, $|S|=\sum_{i=1}^{|\mathscr{K}|}(|K_i|-1)=|\mathscr{P}|-|\mathscr{K}|$. □

Selective coloring and clique-partition problems are equivalent, Theorem 1 shows how to solve SCP in $(G,\mathscr{P})$ by finding a maximum stable set in the selective line-graph of $(\overline{G},\mathscr{P})$.

2.1 Example of the transformation

Figure 1 shows a partitioned graph $(G,\mathscr{P})$ and its partition line-graph $L(G,\mathscr{P})$.

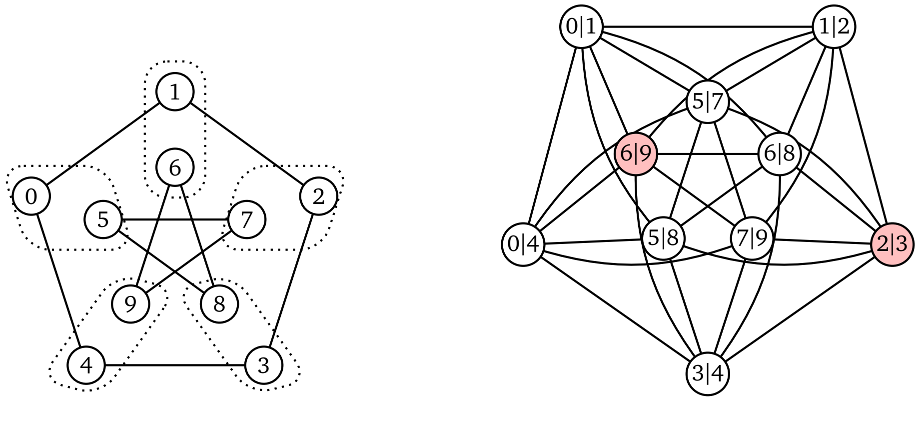

Figure 1: Partitioned graph and its partition line-graph.

Since the graph G in Figure 1 has no triangle, then any orientation $\vec{G}$ has no simplicial pair, and then $L(G,\mathscr{P})=S(\vec{G},\mathscr{P})$. The stable set $S=\{6|9,2|3\}$ of $L(G,\mathscr{P})$ corresponds to the selective clique-partition $\mathscr{K}=\{\{0\},\{2,3\},\{6,9\}\}$ of $(G,\mathscr{P})$, which illustrate Theorem 1.

For another example, suppose that we add the edge $\{6,7\}$ to G and that we choose the orientation $\vec{G}$ from vertex i to vertex j if $i>j$, then $\{(9,6),(9,7)\}$ becomes a simplicial pair. In $S(\vec{G},\mathscr{P})$, the vertex $6|7$ is added, adjacent to vertices $0|1,1|2,2|3,5|7,7|9,6|8,6|9$, and the edge between $6|9$

and 7|9 is deleted. Thus $S = \{6|9, 7|9, 5|8\}$ is a stable set of $S(\vec{G}, \mathscr{P})$. It corresponds to the selective clique-partition $\mathscr{K} = \{\{6,7,9\},\{5,8\}\}$ of $(G, \mathscr{P})$.

2.2 Extension to the Selective Max-Coloring Problem

Max-Coloring is a weighted extension of the classical coloring problem. Given (G, c) where G is a graph and c is a cost vector in $\mathbb{Z}^{V(G)}$, the weight of a coloring $\mathscr{K} = \{K_1, \ldots, K_{|\mathscr{K}|}\}$ of G is $w(\mathscr{K}) = \sum_{i=1}^{|\mathscr{K}|} \max_{v \in K_i} c(v)$, and it has to be minimized.

We consider a more general problem, extending Max-Coloring to partition graphs. Given $(G, \mathscr{P}, c)$ where $(G, \mathscr{P})$ is a partitioned graph and c is a cost vector in $\mathbb{Z}^{\mathscr{P}}$, defined the weight of a selective coloring $\mathscr{K} = \{K_1, \ldots, K_{|\mathscr{K}|}\}$ of G as $w(\mathscr{K}) = \sum_{i=1}^{|\mathscr{K}|} \max_{V_j : V_j \cap K_i \neq \emptyset} c(V_j)$. The SMCP is to find a selective coloring with minimum weight. Clearly, when clusters are singleton SMCP is exactly Max-Coloring.

We can reduce SMCP to a well-studied problem, namely MWSS, the Maximum Weighted Stable Set problem. Actually, SMCP is equivalent to finding a selective clique-partition with minimum weight, where the weight is defined similarly. We add two new steps in the transformation of Theorem 1:

1. Choose the orientation $\vec{G}$ of the edges of G as follows: for each edge $\{u, v\}$ then $u \in V_i$ and $v \in V_j$ for two distinct clusters V_i, V_j, orientate $\{u, v\}$ form u to v if $c(V_i) \geq c(V_j)$, in the other direction otherwise.

2. Attribute the cost $\vec{c}(uv) = c(V_j)$ to each vertex uv of $S(\vec{G}, \mathscr{P})$, where $u \in V_i$, $v \in V_j$ and $c(V_i) \geq c(V_j)$.

Let $\{\mathscr{K}, S\}$ be a pair in correspondence as in the proof of Theorem 1. So S is the disjoint union of out-stars S_i the center u_i of which is in the cluster spanned by S_i with maximum cost. It follows that

$$\sum_{i=1}^{|\mathscr{K}|} \sum_{k=1}^{|S_i|} \vec{c}(u_i u_i^k) = \sum_{j=1}^{|\mathscr{P}|} c(V_j) - \sum_{i=1}^{|\mathscr{K}|} \max_{V_j : V_j \cap S_i \neq \emptyset} c(V_j) \tag{3}$$

Hence, by denoting $c(\mathscr{P}) = \sum_{j=1}^{|\mathscr{P}|} c(V_j)$, by denoting $\alpha(G, c)$ the maximum of $\sum_{v \in S} c(v)$ over all stable sets of G, and by denoting $\overline{\chi}(G, \mathscr{P}, c)$ the minimum weight of a selective clique-partition of $(G, \mathscr{P}, c)$, we obtain the following corollary of Theorem 1:

Corollary 1. $\overline{\chi}(G, \mathscr{P}, c) + \alpha(S(\vec{G}, \mathscr{P}), \vec{c}) = c(\mathscr{P})$.

3 Results

The purpose of the computational experiments is to validate that the line-graph method, implemented solving a Maximum Weighted Stable Set Problem on the selective line-graph, outperforms a state-of-the-art solver for on a MIP formulaion of the SMCP.

We ran the experiments on a cluster with quad-core Intel Xeon E5 processors running at 2.20GHz, reserving 128GB of RAM and using `CPLEX 12.8.1` as the MIP Solver. To solve the problem with the proposed approach, we used the Maximum Weighted Stable Set solver of `exactcolors` package, originally designed for the solving the pricing phase of a column generation algorithmfor the vertex-coloring problem [12].

3.1 Instance generation

We generated a total of 531 graphs as follows. Each graph is identified by a triplet $(|V|, d, p)$ where:

- $|V| \in \{10, 20, \ldots, 100\}$ is the number of vertices in the graph.

- $d \in \{.1, .2, \ldots, .9\}$ is the density of the graph.
- $p \in \{.05, .1, .15, .2, .25, .3\}$ is the average fraction of vertices in each cluster. For example, $p = .25$ corresponds to a graph with 4 clusters, each containing roughly 25% of the vertices. Each cluster was assigned an integer weight uniformly distributed in $[1, 100]$.

There are $9 \cdot 6 = 54$ possible graphs for each value of $|V|$, except for $|V| = 10$ when $p = .05$ would not make sense and, therefore, we only have $9 \cdot 5 = 45$ graphs. This gives a total of $54 \cdot 9 + 45 = 531$ graphs. Both the graphs[1] and the source code of the solver[2] are available online.

3.2 Mixed-Integer Model for the SMCP

The Mixed-Integer model we used to solve the SMCP in $(G, \mathscr{P}, c)$ uses two sets of variables. Binary variables x_v^i are defined for each vertex $v \in V(G)$ and each color $i \in [k]$. (Here k is any upper bound on the number of colors and $[k] = \{1, \ldots, k\}$.) They take value 1 if and only if vertex v is colored with color i. Continuous variables z_i are defined for each color $i \in [k]$ and denote the cost of color i in the solution. A natural formulation for the SMCP is then:

$$
(\text{MIP}_{\text{SMCP}}) \quad
\begin{aligned}
\chi(G, \mathscr{P}, c) = \quad \min \quad & \sum_{i=1}^{k} z_i && && (4)\\
s.t. \quad & \sum_{i=1}^{k} \sum_{v \in V_j} x_v^i = 1 && j = 1, \ldots, |\mathscr{P}| && (5)\\
& x_u^i + x_v^i \leq 1 && \{u, v\} \in E(G),\ i \in [k] && (6)\\
& z_i \geq c(V_j) \cdot x_v^i && j = 1, \ldots, |\mathscr{P}|,\ v \in V_j,\ i \in [k] && (7)\\
& x_v^i \in \{0, 1\} && v \in V(G),\ i \in [k] && (8)\\
& z_i \in \mathbb{R}^+ && i \in [k] && (9)
\end{aligned}
$$

Objective function (4) minimizes the sum of the costs of the colors, which are defined by constraints (7). Constraints (5) impose that one vertex per cluster is colored, and constraints (6) impose that adjacent vertices do not receive the same color. Finally, constraints (8) and (9) define the variables of the formulation.

3.3 Computational experiments

Table 1 synthetically compares the Mixed-Integer model MIP_{SMCP} and the line-graph method. Columns under "MIP" refer to the model, while columns under "Line-Graph" refer to the line-graph method. Columns "Opt" give the number of optimal solutions found by each method, while column "Gap%" reports the average gap, computed as UB-LB/UB. (Where UB is the value of the incumbent solution and LB is the best dual bound.) Columns "Time" report the average running time in seconds. We ran both algorithms with a time limit of 1 hour and include the unsolved instances in the computation of the average time. Finally, for the line-graph algorithm, we report under columns "nV" and "nE", respectively, the average number of vertices and edges of the final graphs on which we solve the Maximum Weighted Stable Set Problem.

The line-graph algorithm is able to solve all instances very quickly, whereas using the Mixed-Integer model leads to having unsolved instances and larger average times. More in details, our approach solves all the 531 instances in less than 12 seconds in average, whereas the direct use of Cplex only 509 instances in less than 1 hour.

[1] `https://github.com/alberto-santini/selective-graph-colouring/tree/master/max-weight-sgcp/instances`

[2] `https://github.com/alberto-santini/selective-graph-colouring/tree/master/max-weight-sgcp/source-code`

	MIP			*Line-Graph*				
$\|V\|$	Opt	Gap%	Time	nV	nE	Opt	Gap%	Time
10	45	0	0.05	21	132	45	0	0.00
20	54	0	0.20	82	2 115	54	0	0.00
30	54	0	0.79	189	10 795	54	0	0.01
40	54	0	3.02	335	37 078	54	0	0.02
50	53	0.8	90.22	523	87 929	54	0	0.24
60	51	2.1	204.63	750	188 612	54	0	0.28
70	49	3.5	336.07	1021	342 170	54	0	2.45
80	50	3.8	271.67	1324	591 088	54	0	1.48
90	49	5.1	363.18	1688	942 813	54	0	12.37
100	50	3.7	281.82	2073	1 453 197	54	0	5.62

Table 1: Summary of the results obtained comparing the Mixed-Integer model MIP_{SMCP} with the algorithm based on line graphs for the Selective Max-Coloring Problem.