Strength of the Upper Bounds for the Edge-Weighted Maximum Clique Problem

Fabio Ciccarelli[a], Valerio Dose[a], Fabio Furini[a], Marta Monaci[b]

[a]*Department of Computer, Control and Management Engineering "Antonio Ruberti" - Sapienza University of Rome. Via Ariosto 25, 00185 Roma, Italy.*

[b]*Dipartimento di Ingegneria e Scienze - Universitas Mercatorum. Piazza Mattei 10, 00186 Roma, Italy.*

Abstract

We theoretically and computationally compare the strength of the two main upper bounds from the literature on the optimal value of the Edge-Weighted Maximum Clique Problem (EWMCP). We provide a set of instances for which the ratio between either of the two upper bounds and the optimal value of the EWMCP is unbounded. This result shows that neither of the two upper bounds can give a performance guarantee. In addition, we provide two sets of instances for which the ratio between the two upper bounds, and its reciprocal, are unbounded. This result shows that there are EWMCP instances in which one of the upper bounds can be arbitrarily better than the other one and vice versa. Our theoretical analysis is complemented by extensive computational experiments on two benchmark datasets: the standard DIMACS instances and randomly generated instances, providing practical insights into the empirical strength of the upper bounds.

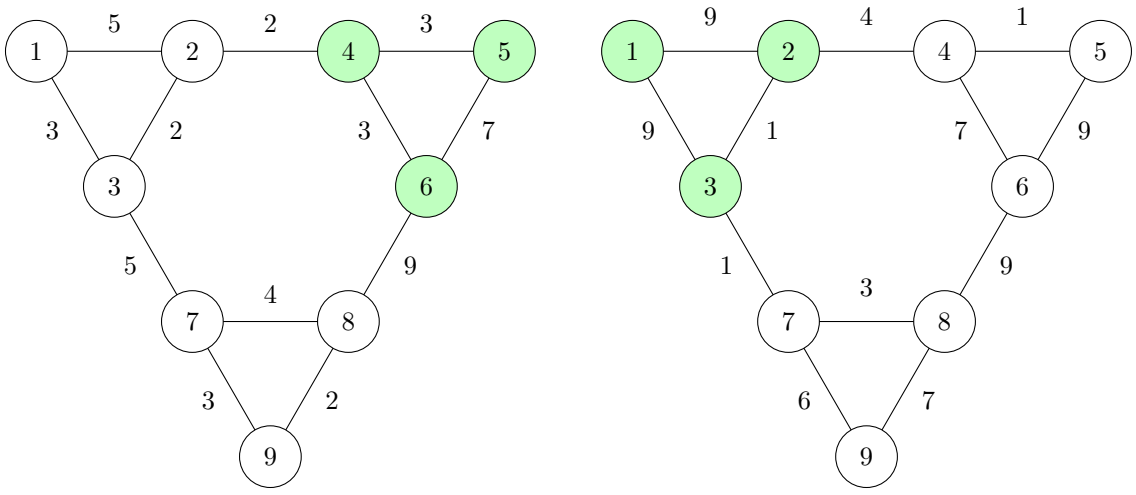
Keywords: Graph Theory, Combinatorial Optimization, Edge-Weighted Maximum Clique Problem

1. Introduction

Let $G = (V, E)$ be a *simple undirected graph*, where V is the set of vertices, each labeled with a distinct natural number, and E is the set of edges connecting pairs of vertices. We denote an edge $\{u, v\} \in E$, with $u, v \in V$, by e. We assume that each edge $e \in E$ is associated with a nonnegative integer *edge-weight* $c_e \in \mathbb{Z}_{\geq 0}$. Given a clique $C \subseteq V$ of G,

Email addresses: f.ciccarelli@uniroma1.it (Fabio Ciccarelli), valerio.dose@uniroma1.it (Valerio Dose), fabio.furini@uniroma1.it (Fabio Furini), marta.monaci@unimercatorum.it (Marta Monaci)

Figure 1: Two EWMCP instances with the same vertex set and edge set but different edge-weight vectors $\boldsymbol{c}_a$ and $\boldsymbol{c}_b$. The natural number labeling a vertex is represented inside the vertex, whereas the edge-weight of an edge is reported next to the edge. In Part (a), the clique $\{4, 5, 6\}$, highlighted in green, is a maximum total edge-weight clique for the first instance. The edge-weighted clique number of the graph $\omega(G, \boldsymbol{c}_a)$ is 13. In Part (b), the clique $\{1, 2, 3\}$, highlighted in green, is a maximum total edge-weight clique for the second instance. The edge-weighted clique number of the graph $\omega(G, \boldsymbol{c}_b)$ is 19.



(a) $\omega(G, \boldsymbol{c}_a) = 13$ (b) $\omega(G, \boldsymbol{c}_b) = 19$

namely a subset of pairwise connected vertices, its *total edge-weight* is:

$$\sum_{e \in E[C]} c_e, \tag{1}$$

where $E[C] \subseteq E$ is the set of edges connecting the vertices in C. The *Edge-Weighted Maximum Clique Problem* (EWMCP) aims to determine a clique of the graph of maximum total edge-weight. The vector $\boldsymbol{c} \in \mathbb{Z}_{\geq 0}^{|E|}$, containing all edge-weights, is called the *edge-weight vector*. We denote an instance of the EWMCP by the pair $(G, \boldsymbol{c})$. We denote by $\omega(G, \boldsymbol{c})$ the optimal value of the EWMCP and call it the *edge-weighted clique number* of the graph. It is worth noticing that, since all edge-weights are nonnegative, there exists a maximum total edge-weight clique which is inclusion-wise maximal. In Figure 1, we introduce two EWMCP instances that are used in the remainder of the article.

The EWMCP is an important generalization of the classical *Maximum Clique Problem* (MCP), with real-world applications across many domains, such as robotics, computer vision and materials science [1, 14, 17, 18]. Since the MCP is strongly $\mathcal{NP}$-hard, the same holds for the EWMCP [9, 12]. The EWMCP is an important weighted generalization of the MCP. We refer the interested reader to other relevant weighted clique-related problems; see, for instance, [3, 4, 5, 7, 8, 15, 20].

Over the past decades, numerous exact approaches have been proposed in the literature

to solve the EWMCP to proven optimality [13, 16, 19, 21, 22]. The main state-of-the-art techniques are combinatorial branch-and-bound algorithms that efficiently explore the search space through an implicit enumeration [19, 21, 22]. These algorithms rely on effective bounding techniques to limit the search space.

There are two main upper bounds for the optimal value of the EWMCP, each forming the core of the bounding procedures in the exact algorithms proposed by San Segundo et al. [19] and Shimizu et al. [22]. The first is based on a *Linear Programming* (LP) relaxation of an *Integer Linear Programming* (ILP) formulation for the EWMCP, while the second is a purely combinatorial upper bound that directly exploits the structure of feasible solutions to the EWMCP. Both upper bounds are formally defined in Section 2. To the best of our knowledge, these two upper bounds were developed independently and simultaneously, and no comparison has ever been made in the literature.

This work aims to study, both theoretically and computationally, the strength of these two upper bounds. Our first theoretical result, presented in Section 3, shows that there exists a set of instances for which the ratio between either of the two upper bounds and the optimal value of the EWMCP is unbounded. Our second theoretical result, in Section 4, provides a comparison between the two upper bounds by defining two families of instances in which either upper bound can be arbitrarily worse than the other. These results, while ruling out the possibility of obtaining any relative performance guarantee with respect to the optimal value of the EWMCP, offer insight into the structural properties of the graphs that influence the strength of the two upper bounds.

On the computational side, in Section 5 we report extensive results on two sets of instances to empirically assess the relative strength of the two upper bounds. When the optimal value of the EWMCP can be computed, we also report the corresponding optimality gap. Finally, in Section 6, we draw the main conclusions, highlighting the structural properties of graphs that favor one upper bound over the other, and outlining directions for future research.

2. The two upper bounds on $\omega(G, c)$

In this section, we define the two upper bounds on $\omega(G, \boldsymbol{c})$ that are analyzed and compared in this article. For completeness, we also include a proof of validity for each upper bound, following the derivations from the original works in which they were defined.

To this end, we first introduce the necessary notation. Let $I \subseteq V$ be a subset of pairwise disconnected vertices, called an *independent set*. Let $\mathscr{I}$ denote the collection of all independent sets of the graph. In addition, let

$$\mathscr{C} := \{I_1, I_2, \ldots, I_k\}$$

be a partition of the vertex set V into k disjoint, non-empty independent sets. We call such a partition a *coloring* of the vertices of the graph. We interpret the data of a

coloring as an assignment of colors (labeled by natural numbers) to the vertices such that any pair of connected vertices receives a different color.

2.1. The first upper bound

Following San Segundo et al. [19], we describe in this section the derivation of the first upper bound on $\omega(G, \boldsymbol{c})$, which is based on a LP relaxation of an ILP formulation of the EWMCP, with the constraints restricted to a given coloring.

For every $u \in V$, let x_u be a binary variable that takes value 1 if and only if vertex u is selected in the clique. For every edge $e \in E$, let y_e be a binary variable that takes value 1 if and only if both its endpoints u and v are selected in the clique. Using these binary variables, an ILP formulation for the EWMCP reads as follows:

$$\max_{\substack{\boldsymbol{x}\in\{0,1\}^{|V|}\\ \boldsymbol{y}\in\{0,1\}^{|E|}}} \left\{ \sum_{e\in E} c_e\, y_e : \quad y_e \le x_u,\ y_e \le x_v,\ e \in E, \quad \sum_{u\in I} x_u \le 1,\ I \in \mathscr{I} \right\}. \tag{2}$$

The objective function of Model (2) maximizes the total edge-weight of the selected clique. The first two sets of constraints linearize the following logical implication: if an edge $e \in E$ is part of the selected clique, then both its endpoints $u, v \in V$ must also be part of the selected clique. The last set of constraints imposes that, for every independent set of the graph, at most one vertex can belong to the selected clique. This ensures that the feasible solutions of Model (2) correspond to cliques in the graph. The optimal value of Model (2) is the edge-weighted clique number $\omega(G, \boldsymbol{c})$. It is worth noting that, without loss of generality, the binary constraints on the $\boldsymbol{y}$ variables can be dropped, since they are implicitly enforced by the binary nature of the $\boldsymbol{x}$ variables and the structure of the objective function. Therefore, the $\boldsymbol{y}$ variables can be considered as continuous free variables without affecting the correctness of the model.

By imposing the last set of constraints of Model (2) only on a given coloring $\mathscr{C} \subseteq \mathscr{I}$, and by relaxing the integrality constraints on the variables $\boldsymbol{x}$ with nonnegativity constraints, we obtain a LP relaxation of the ILP Model (2). To derive the dual of this LP relaxation, we introduce, for each edge $e \in E$, two dual variables ϱ_{eu} and ϱ_{ev} associated with the constraints $y_e \le x_u$ and $y_e \le x_v$, respectively. In addition, for each $I \in \mathscr{C}$, we define a dual variable π_I corresponding to the constraint $\sum_{u\in I} x_u \le 1$. The dual of the LP relaxation described above reads as follows:

$$UB_1(G, \boldsymbol{c}, \mathscr{C}) \ := \ \min_{\substack{\boldsymbol{\pi}\ge\boldsymbol{0}\\ \boldsymbol{\varrho}\ge\boldsymbol{0}}} \quad \sum_{I\in\mathscr{C}} \pi_I \tag{3a}$$

$$\varrho_{eu} + \varrho_{ev} \ = \ c_e, \qquad e \in E, \tag{3b}$$

$$\pi_I \ge \sum_{e\in\delta(u)} \varrho_{eu}, \qquad I \in \mathscr{C}, \quad u \in I, \tag{3c}$$

where $\delta(u) \subseteq E$ is the set of all edges incident to vertex $u \in V$.

By LP duality, the optimal value of model (3) provides an upper bound on $\omega(G, \boldsymbol{c})$. This value, denoted by $UB_1(G, \boldsymbol{c}, \mathscr{C})$, constitutes the first upper bound studied in this article.

It is worth noting that the values of the variables $\boldsymbol{\varrho}$, in any optimal solution of Model (3), can be interpreted as distributing the weight of each edge among its two endpoints. This interpretation is then used in the reminder of the article to illustrate the examples, and to make a comparison with the second upper bound. In more detail, for each edge $e \in E$, the optimal value of the variables ϱ_{eu} and ϱ_{ev} represent the portions of the edge-weight c_e assigned to the endpoints u and v, respectively. Moreover, for every $u \in V$, the quantity

$$\gamma_u := \sum_{e \in \delta(u)} \varrho_{eu} \tag{4}$$

can be interpreted as the total weight assigned to vertex u. Constraints (3c) then imply that, for each $I \in \mathscr{C}$, the variable π_I must be at least as large as the weight γ_u for every $u \in I$. In this interpretation, each variable π_I represents then the weight of the independent set I, and the objective function (3a) computes the total weight across all independent sets in the coloring $\mathscr{C}$. Then we have:

$$UB_1(G, \boldsymbol{c}, \mathscr{C}) = \sum_{I \in \mathscr{C}} \max\{\, \gamma_u : u \in I \,\}. \tag{5}$$

Note that the upper bound employed in the branch-and-bound algorithm of San Segundo et al. [19] corresponds to a heuristic solution of Model (3), computed on subgraphs generated during the branching process. The proposed heuristic procedure is designed to reduce the computational effort while computing good-quality upper bounds. In contrast, in this paper we focus on evaluating the strength of the upper bound itself. To this end, we consider optimal solutions of Model (3), corresponding to the best possible upper bounds within this family.

2.2. The second upper bound

Following Shimizu et al. [22], we describe in this section the derivation of the second upper bound on $\omega(G, \boldsymbol{c})$, obtained by exploiting the structure of feasible EWMCP solutions and a coloring of the vertices of the graph.

Given a coloring $\mathscr{C}$ of k independent sets, for each vertex $u \in V$, let $\tau(u)$ denote the index of the (unique) independent set that contains u. For every $u \in V$, we define the quantity

$$\sigma_u := \sum_{h \in \{1,2,\ldots,\tau(u)-1\}} \max\{\, c_e : e \in \delta(u),\ v \in I_h \,\}, \tag{6}$$

which can be interpreted as the total weight assigned to vertex u. This weight corresponds to the sum, over all independent sets that precede the one containing u in the ordering, of the maximum edge-weight connecting u to a vertex in each of those independent sets.

The second upper bound studied in this article, denoted by $UB_2(G, \boldsymbol{c}, \mathscr{C})$, is then defined as:

$$UB_2(G, \boldsymbol{c}, \mathscr{C}) := \sum_{I \in \mathscr{C}} \max\{\, \sigma_u : u \in I \,\}. \tag{7}$$

To verify that this is indeed an upper bound for $\omega(G, \boldsymbol{c})$, consider any clique C in G. By construction, each vertex in the clique C belongs to a different independent set in the coloring $\mathscr{C}$. For every independent set $I_h \in \mathscr{C}$, let u_h be the unique vertex in C that belongs to I_h, if such a vertex exists; otherwise, let u_h be any vertex in I_h. We then obtain:

$$\sum_{e \in E[C]} c_e = \sum_{\substack{r \in \{1,2,\ldots,k\}: \\ C \cap I_r \neq \emptyset}} \sum_{\substack{h \in \{1,2,\ldots,r-1\}: \\ C \cap I_h \neq \emptyset}} c_{\{u_r, u_h\}} \tag{8a}$$

$$\leq \sum_{\substack{r \in \{1,2,\ldots,k\}: \\ C \cap I_r \neq \emptyset}} \max \left\{ \sum_{h \in \{1,2,\ldots,\tau(u)-1\}} c_{\{u,u_h\}} : u \in I_r \right\} \tag{8b}$$

$$\leq \sum_{\substack{r \in \{1,2,\ldots,k\}: \\ C \cap I_r \neq \emptyset}} \max \left\{ \sum_{h \in \{1,2,\ldots,\tau(u)-1\}} \max\{c_e : e \in \delta(u),\ v \in I_h\} : u \in I_r \right\} \tag{8c}$$

$$\leq \underbrace{\sum_{I \in \mathscr{C}} \max\{\sigma_u : u \in I\}}_{= UB_2(G, \boldsymbol{c}, \mathscr{C})}. \tag{8d}$$

The equality in (8a) rewrites the sum of the weights of the edges in the clique C by indexing over pairs of independent sets in $\mathscr{C}$ that contain vertices of C. The inequality in (8b) is obtained by maximizing the inner sum over all possible choices of a vertex $u \in I_r$ and by extending the sum to all indices $h < r$. The inequality in (8c) follows by maximizing each term of the inner sum over all possible choices of a vertex $v \in I_h$. Finally, the inequality in (8d) is obtained by extending the sum to all independent sets in $\mathscr{C}$. This derivation follows the proof of [22, Lemma 1].

Note that the upper bound $UB_2(G, \boldsymbol{c}, \mathscr{C})$ depends not only on a coloring, but also on the specific ordering of the independent sets within the coloring, given by the indices of the independent sets. Using the same partition in independent sets, but varying the labeling could result in different upper bounds. In Section 5, we empirically measure the impact of different orderings on the quality of this upper bound.

It is worth mentioning that Shimizu et al. [22] also proposed an alternative upper bound, which can be interpreted as the value of a specific feasible solution to the dual formulation (3). The main idea is to assign the full weight of each edge $e \in E$ to the endpoint with the smaller index in the coloring. Assuming without loss of generality that for each edge $e \in E$, we have $\tau(u) < \tau(v)$, this feasible solution is defined as follows:

$$\tilde{\varrho}_{eu} := c_e, \qquad \tilde{\varrho}_{ev} := 0, \quad e \in E, \qquad \tilde{\pi}_I := \max\left\{ \sum_{e \in \delta(u)} \tilde{\varrho}_{eu} \colon u \in I \right\}, \quad I \in \mathscr{C}.$$

The upper bound obtained from this dual solution is, by construction, greater than or equal to the optimal value of the dual Model (3). Accordingly, the second upper bound proposed by Shimizu et al. [22] is dominated by $UB_1(G, \boldsymbol{c}, \mathscr{C})$, and is therefore not considered in our study.

2.3. Examples of the upper bounds computation

In Figure 2, we report the values of the two upper bounds for the graph shown in Figure 1 and its two different edge-weight vectors. The figure also includes all the values needed to compute the upper bounds, as described in Sections 2.1 and 2.2. These examples illustrate that there exist graphs, edge-weight vectors, and colorings for which $UB_1(G, \boldsymbol{c}, \mathscr{C})$ is smaller than $UB_2(G, \boldsymbol{c}, \mathscr{C})$ (see Parts 2a and 2c of Figure 2), and vice versa (see Parts 2b and 2d of Figure 2).

3. Theoretical comparison of the two upper bounds with $\omega(G, c)$

In this section, we provide a sequence of EWMCP instances in which the ratio between either of the two upper bounds and the edge-weighted clique number becomes arbitrarily large as the number of vertices in the graphs tends to infinity.

For every integer $n \geq 1$, let $G_n^1 = (V_n^1, E_n^1)$ be a graph with a vertex set containing $2n$ vertices, partitioned into two cliques $C_1, C_2 \subseteq V_n^1$, where

$$C_1 = \{1, 2, \ldots, n\}, \qquad C_2 = \{n+1, n+2, \ldots, 2n\}.$$

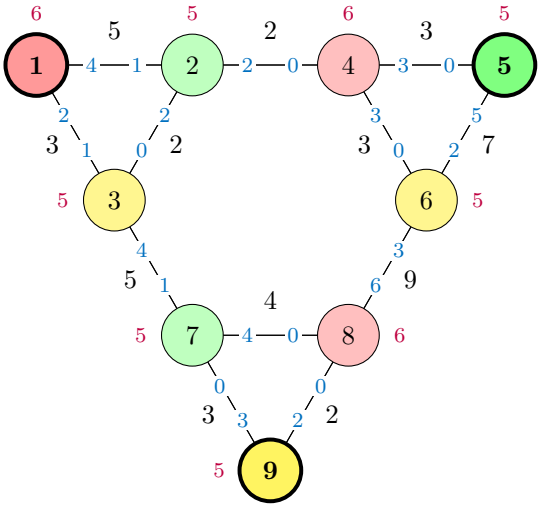
The edge set of the graph includes all edges within the two cliques, each assigned an edge-weight of 1. Moreover, for each vertex $u \in C_1$, there is an edge $\{u, n+u\}$ with edge-weight

$$\bar{c} = \frac{n(n-1)}{2} - 1,$$

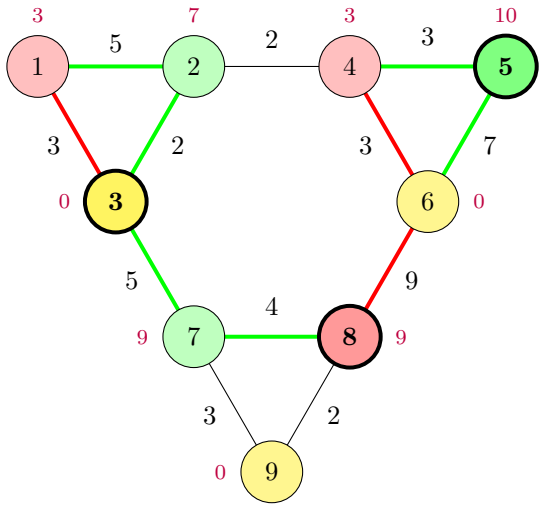
and, by construction, the vertex $n+u$ belongs to C_2. These edge-weights define the edge-weight vector $\boldsymbol{c}_n^1$ for the graph G_n^1. An illustration of G_n^1 is provided in Figure 3.

The inclusion-wise maximal cliques of G_n^1 are C_1, C_2, with total edge-weight $\frac{n(n-1)}{2}$, and the n cliques of cardinality 2 given by the edges connecting C_1 and C_2, with total

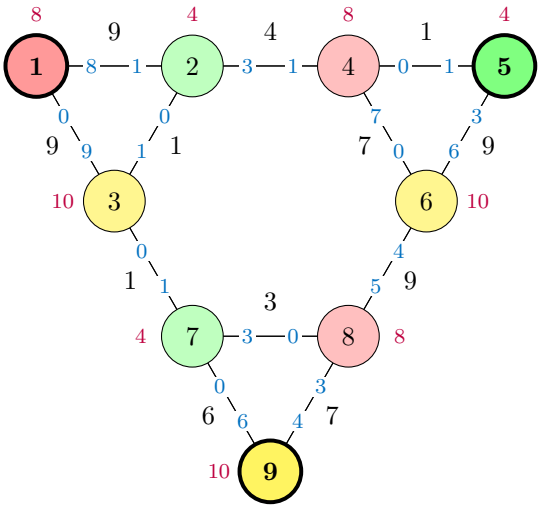
Figure 2: The values of the two upper bounds $UB_1(G, \boldsymbol{c}, \mathscr{C})$ and $UB_2(G, \boldsymbol{c}, \mathscr{C})$ for the instances in Figure 1, with respect to the following coloring: $I_1 = \{3, 6, 9\}$ (yellow), $I_2 = \{1, 4, 8\}$ (red), $I_3 = \{2, 5, 7\}$ (green). For each edge $e \in E$, the black label next to the edge shows its weight — in Parts 2a and 2b, this corresponds to the edge-weight vector $\boldsymbol{c}_a$ of Part 1a of Figure 1; in Parts 2c and 2d, to the edge-weight vector $\boldsymbol{c}_b$ of Part 1b of Figure 1. In Parts 2a and 2c, for each edge $e \in E$, the blue labels on the edge report the values of the variables ϱ_{eu} and ϱ_{ev} in an optimal solution of Model (3). For each vertex $u \in V$, the value γ_u (see the definition (4)) is shown in purple next to the vertex. For each color $I \in \mathscr{C}$, the vertex $u \in I$ with the maximum weight γ_u is highlighted in bold; this value corresponds to the value of the variable π_I in the same optimal solution of Model (3). In Parts 2b and 2d, for each vertex $u \in V$, the value σ_u (see the definition (6)) is shown in purple next to the vertex. Additionally, the edges contributing to σ_u are drawn in the same color as vertex u. For each color $I \in \mathscr{C}$, the vertex $u \in I$ with the maximum weight σ_u is highlighted in bold.



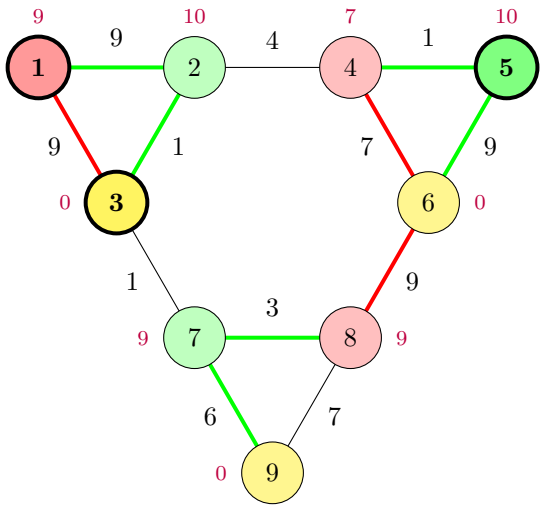
(a) $UB_1(G, \boldsymbol{c}_a, \mathscr{C}) = 16$



(b) $UB_2(G, \boldsymbol{c}_a, \mathscr{C}) = 19$

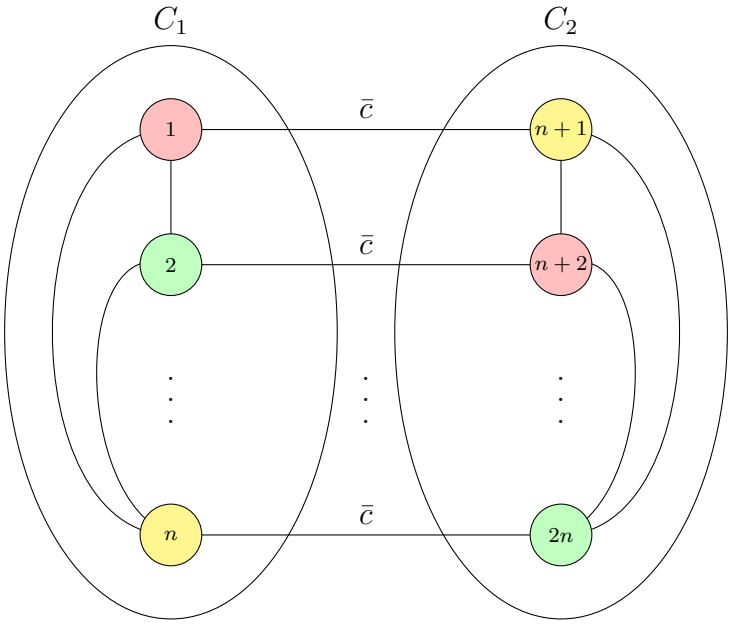


(c) $UB_1(G, \boldsymbol{c}_b, \mathscr{C}) = 22$



(d) $UB_2(G, \boldsymbol{c}_b, \mathscr{C}) = 19$

Figure 3: Representation of the graph G_n^1, where the vertices are colored using the minimum number of colors. Each vertex is labeled with a natural number displayed inside the node. The two ellipses represent the cliques C_1 and C_2. The edge-weight $\bar{c}$ of each edge connecting a vertex in C_1 to a vertex in C_2 is shown next to the edge. All edges within each clique have weight one; these values are omitted from the figure.



edge-weight equal to $\bar{c}$. Hence, the edge-weighted clique number of the EWMCP instance $(G_n^1, \boldsymbol{c}_n^1)$ is

$$\omega(G_n^1, \boldsymbol{c}_n^1) \;=\; \frac{n(n-1)}{2}. \tag{9}$$

Since G_n^1 contains a clique of cardinality n, any coloring of its vertices requires at least n colors. It follows that for every coloring $\mathscr{C}$ of the vertices of G_n^1, we have $|\mathscr{C}| \geq n$. Moreover, any independent set I in any coloring $\mathscr{C}$ contains at most one vertex from C_1 and at most one from C_2, and therefore $|I| \leq 2$.

The following proposition states that the ratio between the upper bound $UB_1(G, \boldsymbol{c}, \mathscr{C})$ and edge-weighted clique number $\omega(G, \boldsymbol{c})$ can be arbitrarily large.

Proposition 1. *The sequence* $\{(G_n^1, \boldsymbol{c}_n^1)\}$ *of EWMCP instances is such that*

$$\frac{UB_1(G_n^1, \boldsymbol{c}_n^1, \mathscr{C}_n^1)}{\omega(G_n^1, \boldsymbol{c}_n^1)} \to +\infty \qquad \textit{as} \qquad n \to +\infty,$$

where $\mathscr{C}_n^1$ *is any coloring of the vertices of* G_n^1.

Proof. Given any independent set $I \in \mathscr{C}_n^1$, by adding up the Constraints (3c) related to

I, we obtain:

$$|I|\,\pi_I \geq \sum_{u\in I}\sum_{e\in\delta(u)} \varrho_{eu} \quad \Longrightarrow \quad 2\,\pi_I \geq \sum_{u\in I}\sum_{e\in\delta(u)} \varrho_{eu},$$

since $|I| \leq 2$. Moreover, adding up all these last inequalities for every independent set $I \in \mathscr{C}_n^1$, we then have:

$$2\sum_{I\in\mathscr{C}_n^1} \pi_I \geq \sum_{e\in E} (\varrho_{eu} + \varrho_{ev}),$$

since any coloring is a partition of the vertex set. Then, according to Constraints (3b), the following holds:

$$\underbrace{\sum_{I\in\mathscr{C}_n^1} \pi_I}_{=UB_1(G_n^1,\boldsymbol{c}_n^1,\mathscr{C}_n^1)} \geq \frac{1}{2}\sum_{e\in E_n^1} c_e = \frac{1}{2}\sum_{e\in E_n^1[C_1]} c_e + \frac{1}{2}\sum_{e\in E_n^1[C_2]} c_e + \frac{1}{2}n\,\bar{c} = \underbrace{\frac{n(n-1)}{2}}_{=\omega(G_n^1,\boldsymbol{c}_n^1)} + \frac{1}{2}n\,\bar{c}.$$

Finally, it follows that:

$$\lim_{n\to+\infty} \frac{UB_1(G_n^1,\boldsymbol{c}_n^1,\mathscr{C}_n^1)}{\omega(G_n^1,\boldsymbol{c}_n^1)} \geq \lim_{n\to+\infty} 1 + \frac{n}{2} - \frac{1}{n-1} = +\infty.$$

□

The following proposition states that the ratio between the upper bound $UB_2(G,\boldsymbol{c},\mathscr{C})$ and edge-weighted clique number $\omega(G,\boldsymbol{c})$ can be arbitrarily large.

Proposition 2. *The sequence $\{(G_n^1,\boldsymbol{c}_n^1)\}$ of EWMCP instances is such that*

$$\frac{UB_2(G_n^1,\boldsymbol{c}_n^1,\mathscr{C}_n^1)}{\omega(G_n^1,\boldsymbol{c}_n^1)} \to +\infty \qquad as \qquad n\to+\infty,$$

where $\mathscr{C}_n^1$ is any coloring of the vertices of G_n^1.

Proof. For any edge $\{w, n+w\} \in E_n^1$ with $w \in \{1,2,\ldots,n\}$, connecting a vertex in C_1 with a vertex in C_2, its endpoints belong to distinct independent sets of $\mathscr{C}_n^1$. Let $V' \subseteq V_n^1$ be the subset of vertices $w \in C_1$ such that $\tau(w) < \tau(n+w)$ and vertices $n+w \in C_2$ such that $\tau(w) > \tau(n+w)$. Let $V'' := V_n^1 \setminus V'$. This implies that each vertex v in V' is connected to a vertex u in V'' with an edge of weight $\bar{c}$ and we have $\tau(v) < \tau(u)$. Then, we have:

$$\sum_{v\in V_n^1}\sigma_v = \sum_{v\in V'}\sigma_v + \sum_{u\in V''}\sigma_u = \sum_{v\in V'}\sigma_v + \sum_{u\in V''}\sum_{h\in\{1,2,\ldots,\tau(u)-1\}} \max\{c_e\colon e\in\delta(u), v\in I_h\}.$$

Since, for every $u \in V''$ there exists an edge of weight $\bar{c}$ belonging to the set $\{e \in \delta(u)\colon v \in I_h, h < \tau(u)\}$, and since $|V''| = n$, we have:

$$\sum_{v\in V_n^1}\sigma_v \;\geq\; n\,\bar{c}.$$

For every $h \in \{1, 2, \ldots, k\}$ we have:

$$\max\{\sigma_u \colon u \in I_h\} \;\geq\; \frac{1}{2} \sum_{u \in I_h} \sigma_u,$$

since the right-hand side is at most large as the average of the values in $\{\sigma_u \colon u \in I_h\}$. The latter is true because $|I_h| \leq 2$ for every $h \in \{1, 2, \ldots, k\}$. Adding up these inequalities for all $h \in \{1, 2, \ldots, k\}$ yields

$$\underbrace{\sum_{I \in \mathscr{C}_n^1} \max\{\sigma_u \colon u \in I\}}_{=UB_2(G_n^1, \boldsymbol{c}_n^1, \mathscr{C}_n^1)} \;\geq\; \frac{1}{2} \sum_{v \in V_n^1} \sigma_v \;\geq\; \frac{n\,\bar{c}}{2} = \frac{n}{2} \left(\underbrace{\frac{n(n-1)}{2}}_{=\omega(G_n^1, \boldsymbol{c}_n^1)} - 1 \right).$$

Finally, it follows that

$$\lim_{n \to +\infty} \frac{UB_2(G_n^1, \boldsymbol{c}_n^1, \mathscr{C}_n^1)}{\omega(G_n^1, \boldsymbol{c}_n^1)} \geq \lim_{n \to +\infty} \frac{n}{2} - \frac{1}{n-1} = +\infty.$$

□

4. Theoretical comparison between the two upper bounds

In this section, we provide two sequences of EWMCP instances in which the ratio between the two upper bounds, or its reciprocal, becomes arbitrarily large as the number of vertices in the graphs tends to infinity.

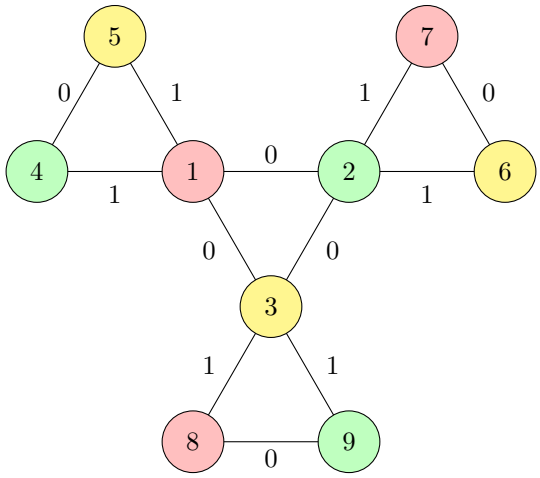
For every integer $n \geq 1$, let $G_n^2 = (V_n^2, E_n^2)$ be a graph with vertex set containing a set C_0 of n vertices, and, for each vertex $u \in C_0$, a set C_u of additional $n-1$ vertices, with $C_u \cap C_0 = \emptyset$. Moreover, the sets C_u for every $u \in C_0$ are pairwise disjoint. The edge set contains only the edges that make C_0 and $C_u \cup \{u\}$, for every $u \in C_0$, cliques. Hence, the graph is composed of n *peripheral cliques* $C_u \cup \{u\}$ of size n, each sharing exactly one vertex u with the *central clique* C_0. For every edge $e \in E$, its edge-weight is defined as follows:

$$c_e = \begin{cases} 1 & \text{if } u \in C_0 \text{ and } v \in C_u, \\ 0 & \text{otherwise.} \end{cases}$$

That is, the edges connecting the central clique to the peripheral cliques have weight 1, and all the remaining edges (within each clique) have weight 0. These edge-weights define the edge-weight vector $\boldsymbol{c}_n^2$ for graph G_n^2. An illustration of a graph of this family, with $n = 3$, is provided in Figure 4.

A coloring $\mathscr{C}_n^2$ of the vertices of G_n^2 with the minimum number n of colors can be constructed as follows. For all $u \in C_0$, an independent set I_u is created containing the

Figure 4: Representation of the graph G_3^2, where the vertices are colored using the minimum number of colors. Each vertex is labeled with a natural number displayed inside the node. The edge-weights are shown next to the corresponding edges.



vertex u and one vertex for every set C_v with $v \neq u$, which has not been included in an independent set yet. By construction, every edge of weight 1 joins vertices in different independent sets in $\mathscr{C}_n^2$. Additionally, each I_u is an independent set of cardinality n and $\mathscr{C}_n^2 := \{I_u \colon u \in C_0\}$ is indeed a coloring of G_n^2.

The following proposition states that the ratio between the upper bound $UB_2(G, \boldsymbol{c}, \mathscr{C})$ and $UB_1(G, \boldsymbol{c}, \mathscr{C})$ can be arbitrarily large.

Proposition 3. *The sequence* $\{(G_n^2, \boldsymbol{c}_n^2)\}$ *of EWMCP instances is such that*

$$\frac{UB_2(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)}{UB_1(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)} \to +\infty \qquad as \qquad n \to +\infty,$$

where $\mathscr{C}_n^2$ *is the coloring of* G_n^2 *defined above.*

Proof. We now define a feasible solution to Model (3) for the instance $(G_n^2, \boldsymbol{c}_n^2)$, leading to an upper bound to $UB_1(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)$. We assign the weight of every edge of weight 1, entirely on its endpoint in the peripheral clique namely

$$\tilde{\varrho}_{eu} = 0, \qquad \tilde{\varrho}_{ev} = 1, \qquad u \in C_0,\ v \in C_u.$$

We have that each vertex in the peripheral cliques receives a total weight equal to 1, while each vertex in the central clique receives a total weight equal to 0. Hence, the feasible solution is completed by setting

$$\tilde{\pi}_I = 1, \quad I \in \mathscr{C}_n^2.$$

Therefore

$$UB_1(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2) \;\leq\; \sum_{I \in \mathscr{C}_n^2} \tilde{\pi}_I \;=\; n.$$

Let us now compute the value of $UB_2(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)$. We label the vertices $u \in C_0$ with the integers $\{1, 2, \ldots, n\}$, and we index accordingly the independent sets of $\mathscr{C}_n^2$ as $\{I_1, I_2, \ldots, I_n\}$, as required for the definition of $UB_2(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)$ (note that the value of the upper bound does not depend on the chosen labeling, in this special case). We know that, for each vertex in the central clique $u \in \{1, 2, \ldots, n\}$, belonging to the independent set $I_u \in \mathscr{C}_n^2$, there is exactly one neighbor of weight 1 in each of the independent sets I_v with $v < u$. Hence, for every $u \in \{1, 2, \ldots, n\}$, we have:

$$\sigma_u = u - 1.$$

On the other hand, each vertex in the peripheral cliques $v \notin C_0$ has at most one incident edge of weight 1, hence $\sigma_v \leq 1$.

Therefore, for all independent sets I_u with $u \in \{1, 2, \ldots, n\}$, the maximum $\max\{\sigma_v \colon v \in I_u\}$ is attained by the central vertex u. Then, for every $u \in \{1, 2, \ldots, n\}$, we have:

$$\max\{\sigma_v \colon v \in I_u\} = u - 1,$$

and thus

$$UB_2(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2) \;=\; \sum_{u=1}^{n} (u-1) \;=\; \frac{n(n-1)}{2}.$$

Finally, it follows that

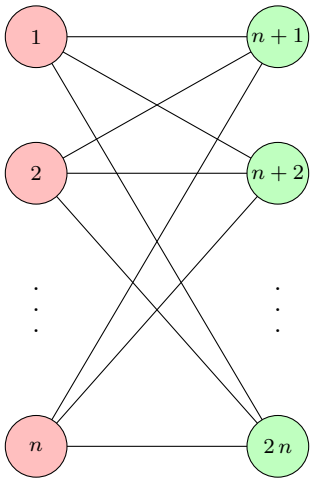
$$\lim_{n \to +\infty} \frac{UB_2(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)}{UB_1(G_n^2, \boldsymbol{c}_n^2, \mathscr{C}_n^2)} \geq \lim_{n \to +\infty} \frac{\frac{n(n-1)}{2}}{n} = \lim_{n \to +\infty} \frac{n-1}{2} = +\infty.$$

□

For every integer $n \geq 1$ let $G_n^3 = (V_n^3, E_n^3)$ be a graph where the vertex set vertex set V_n^3 has cardinality $2n$ and it is partitioned into two independent sets $I_1 = \{1, 2, \ldots, n\}$ and $I_2 = \{n+1, n+2, \ldots, 2n\}$, each of cardinality n. By construction, no edges exist between vertices within the same independent set, while each vertex in I_1 is connected to every vertex in I_2, i.e., for each $u \in I_1$, $v \in I_2$ there exists an edge $e \in E$ connecting u and v. To each edge $e \in E_n^3$ we assign the edge-weight $c_e = 1$. These edge-weights define the edge-weight vector $\boldsymbol{c}_n^3$ for the graph G_n^3. An illustration of the graph G_n^3 is provided in Figure 5.

A coloring of the vertices of G_n^3 with the minimum number of colors is naturally given as $\mathscr{C}_n^3 = \{I_1, I_2\}$.

Figure 5: Representation of the graph G_n^3, where the vertices are colored using the minimum number of colors. Each vertex is labeled with a natural number displayed inside the node. All edge-weights are equal to 1; these values are omitted from the figure.



The following proposition states that the ratio between the upper bound $UB_1(G, \boldsymbol{c}, \mathscr{C})$ and $UB_2(G, \boldsymbol{c}, \mathscr{C})$ can be arbitrarily large.

Proposition 4. *The sequence* $\{(G_n^3, \boldsymbol{c}_n^3)\}$ *of EWMCP instances is such that*

$$\frac{UB_1(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)}{UB_2(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)} \to +\infty \qquad as \qquad n \to +\infty,$$

where $\mathscr{C}_n^3$ *is the coloring of* G_n^3 *defined above.*

Proof. We first show that the optimal value of Model (3), namely $UB_1(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)$, is n. We notice that, in any feasible solution of Model (3), for every $u \in I_1$ and $v \in I_2$, Constraints (3c) relative to the edge $e = \{u, v\}$ write as

$$\pi_{I_1} \geq \sum_{e \in \delta(u)} \varrho_{eu}, \qquad \pi_{I_2} \geq \sum_{e \in \delta(v)} \varrho_{ev}.$$

Adding up these constraints over all vertices therefore yields

$$n\big(\pi_{I_1} + \pi_{I_2}\big) \;\geq\; \sum_{e \in E} c_e,$$

where we used Constraint (3b).

Moreover, all edge-weights are equal to 1, so the sum on the right hand side equals $|E_n^3| = n^2$. Consequently, $\pi_{I_1} + \pi_{I_2} \geq n$, and every feasible solution of Model(3) has value at least n.

This lower bound of n on $UB_1(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)$ is tight: if we assign each edge-weight wholly to its endpoint in I_2 and make the variable assignment $\pi_{I_1} = 0,\ \pi_{I_2} = n$, all constraints of Model (3) are satisfied and the objective function equals n.

Hence

$$UB_1(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3) = n.$$

As far as the bound $UB_2(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)$ is concerned, we observe that every vertex $u \in I_1$ is such that $\sigma_u = 0$, since there are no independent set with an index lower than 1. Whereas every vertex $v \in I_2$ is such that $\sigma_v = 1$ since it is connected only to vertices in I_1 with edges of weight 1. Hence, we have:

$$\max\{\sigma_u \colon u \in I_1\} = 0, \qquad \max\{\sigma_u \colon v \in I_2\} = 1$$

giving the upper bound

$$UB_2(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3) = 1.$$

Finally, it follows that

$$\lim_{n \to +\infty} \frac{UB_1(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)}{UB_2(G_n^3, \boldsymbol{c}_n^3, \mathscr{C}_n^3)} = \lim_{n \to +\infty} \frac{n}{1} = +\infty.$$

□

Note that the values of the two upper bounds in the examples studied in this section, and thus the conclusion of the two propositions, depend on the choice of a particular coloring of the vertices of the graphs G_n^2 and G_n^3. Nevertheless, our choices feature a minimum number of colors and the results hold with any coloring with this natural property.

5. Computational comparison

In this section, we report the results of our extensive experimental analysis, conducted on both DIMACS instances and randomly generated instances commonly used in the literature for the Edge-Weighted Maximum Clique Problem (EWMCP). This computational campaign provides practical insights into the empirical strength of the two upper bounds on $\omega(G, \boldsymbol{c})$ analyzed in this work: $UB_1(G, \boldsymbol{c}, \mathscr{C})$, proposed by San Segundo et al. [19] and described in Section 2.1, and $UB_2(G, \boldsymbol{c}, \mathscr{C})$, proposed by Shimizu et al. [22] and described in Section 2.2.

The purpose of our tests is twofold. On the one hand, we aim to determine which of the two upper bounds is closer to $\omega(G, \boldsymbol{c})$ in practice. On the other hand, we seek to assess whether the empirical strength of the two upper bounds depends on instance characteristics (such as the number of vertices and graph density) or on algorithmic choices (such as the type of coloring and the ordering of its independent sets).

Both upper bounds, $UB_1(G, \boldsymbol{c}, \mathscr{C})$ and $UB_2(G, \boldsymbol{c}, \mathscr{C})$, depend on the coloring of the graph. In our computational experiments, each instance is tested with six different colorings. One of these is generated using the DSatur heuristic [2], while the remaining five are obtained by first generating a random ordering of the vertices and then assigning each vertex, in that order, to the first available color that does not contain any of its neighbors. For the sake of brevity and since the results reported in the following tables and figures are aggregated over the six colorings, we refer to the two upper bounds as UB_1 and UB_2 in the remainder of this section.

All experiments are conducted on a Linux machine equipped with a 2.30 GHz CPU and 512 GB of RAM. The code is implemented in C++, and we use IBM ILOG CPLEX 22.11 to compute UB_1, as it requires solving the LP Model (3). Extensive preliminary tests show that the most efficient algorithm for this purpose is the barrier method, without performing the crossover phase, since we are only interested in the value of the optimal solution of Model (3). Indeed, our computational experiments demonstrate that the barrier algorithm solves the problem to optimality in a time orders of magnitude lower than that required by alternative LP solvers, such as the primal simplex, dual simplex, or network simplex methods.

Detailed numerical results, beyond those presented in the paper, along with the source code for computing the upper bounds, are available at the following GitHub repository: `https://github.com/FabioCiccarelli/EWMCP_Bounds.git`. This material supports precise benchmarking and facilitates comparisons with alternative approaches. We also hope that our findings and the accompanying resources will stimulate further research on upper bounds for $\omega(G, \boldsymbol{c})$, with the aim of identifying tighter values that can enhance the performance of bounding procedures in exact branch-and-bound algorithms for the EWMCP.

5.1. Benchmark datasets of instances

In this work, we consider both DIMACS instances and randomly generated (RANDOM) instances.

- The DIMACS instances, see [6], consist of 78 graphs that are commonly used in the literature to evaluate the performance of exact algorithms for the EWMCP, such as those proposed by Shimizu et al. [22] and San Segundo et al. [19]. These algorithms are able to determine $\omega(G, \boldsymbol{c})$ for 46 of the 78 instances, making this benchmark a valuable reference for assessing the strength of upper bounds.
- The RANDOM instances are generated according to specific pairs of number of vertices and edge density $(|V|, \mu)$, following common practice in the literature. For each combination of $|V| \in \{10, 20, \ldots, 100\}$ and $\mu \in \{0.1, 0.2, \ldots, 0.9\}$, we generate 10 instances. Additionally, for the case $|V| = 100$, we include higher densities in the range $[0.91, 0.99]$ with a step size of 0.01, resulting in a total of 990 graphs. These instances are designed to evaluate the influence of instance characteristics on upper bound quality. Graphs with up to 100 vertices are

considered, as they can be solved to optimality by state-of-the-art exact methods, allowing for a direct comparison with $\omega(G, \boldsymbol{c})$. Smaller graphs are included to assess the impact of the number of vertices, while high-density graphs (91–99%) are included due to their well-known difficulty for exact EWMCP algorithms.

Edge-weights are assigned following a standard approach commonly adopted in the literature [10, 11, 21, 19]. Specifically, for each edge $e \in E$, the edge-weight is defined as $c_e := ((u + v) \bmod 200) + 1$.

5.2. Results for DIMACS instances

In this section, we present the results of the tests conducted on the DIMACS instances. In Table 1, the DIMACS instances are grouped into families based on the similarity of their names. For each family, the table reports its name, the number of instances it contains (column #), and how many of them are solved to optimality by the exact algorithms proposed in Shimizu et al. [22] and San Segundo et al. [19] (column # opt). The next columns provide the minimum, maximum, and average *percentage gap* from $\omega(G, \boldsymbol{c})$ (when known) obtained by UB_1 across the instances in each family, computed as the ratio between the difference $UB_1 - \omega(G, \boldsymbol{c})$ and $\omega(G, \boldsymbol{c})$, multiplied by 100. These are followed by the corresponding statistics for UB_2. Finally, the last columns of Table 1 report the minimum, maximum, and average *percentage difference* (% diff.) between the two upper bounds, computed for each instance as the ratio between the difference $UB_1 - UB_2$ and the greater of the two values, multiplied by 100. As mentioned above, both UB_1 and UB_2 have been computed on each instance using six different colorings (five random colorings and one obtained with the DSatur heuristic). All results reported in the table are based on the best upper bound value obtained among these six colorings, for both UB_1 and UB_2.

The results reported in Table 1 clearly demonstrate the superior strength of UB_2 for all the families of instances. Its value is, on average, consistently lower than that of UB_1 across the entire testbed. As shown in the final columns of the table, the average percentage difference between the two bounds is nearly 43% in favor of UB_2, confirming that, on average, the value of UB_2 is approximately half that of UB_1. In extreme cases, the percentage difference exceeds 90%. Despite this, both upper bounds appear to be rather loose with respect to $\omega(G, \boldsymbol{c})$ when it is known. The average gap for UB_1 is approximately 80%, while that for UB_2 is around 71%, with only rare instances where the latter falls below 10%, a threshold that is never reached by UB_1.

It is also worth noting that computing UB_1 is considerably more time-consuming than UB_2, as it requires solving Model (3). For instances with many edges, and consequently a large number of variables, solving the model can take several tens of seconds and, in some cases, even exceed a few hundred seconds. In contrast, the evaluation of UB_2 is extremely fast and incurs negligible computational time.

Finally, it is worth noting that both upper bounds generally exhibit weaker performance when computed using random colorings, compared to those obtained with the DSatur

Table 1: Performance of the two upper bounds on $\omega(G, \boldsymbol{c})$ for DIMACS instances, grouped by family. Each instance is tested with six different colorings.

Instances			UB_1			UB_2					
			% gap			% gap			% diff.		
family	#	# opt	min	max	avg	min	max	avg	min	max	avg
brock	12	6	93.9	96.5	95.3	89.8	93.7	92.1	35.0	53.4	45.5
C	7	1	76.3	78.7	77.6	65.9	72.0	69.8	19.6	64.1	43.8
c-fat	7	7	24.7	44.9	34.6	4.6	52.0	33.2	-16.6	27.1	1.1
gen	5	2	74.4	84.2	78.8	62.4	77.0	68.7	29.8	57.6	36.2
hamming	6	4	45.6	93.0	69.7	1.2	87.6	51.2	29.7	77.8	46.6
johnson	4	3	77.5	93.5	84.2	54.3	82.7	65.3	43.4	83.2	62.3
keller	3	1	92.7	95.3	94.0	81.4	90.8	85.8	48.8	86.0	70.8
MANN	4	1	63.6	67.1	64.9	36.0	49.5	42.3	33.8	62.6	53.9
p_hat	15	8	89.2	99.5	95.6	80.5	99.0	92.9	30.2	62.9	43.9
san	11	10	58.3	98.6	86.2	35.0	94.0	73.7	22.9	93.8	49.0
sanr	4	3	94.5	98.4	96.9	90.7	97.1	94.7	26.8	48.8	40.6
Total	78	46	24.7	99.5	79.5	1.2	99.0	71.4	-16.6	93.8	43.1

heuristic. This is likely due to DSatur's tendency to produce colorings with fewer independent sets. Indeed, our computational experiments consistently show that as the cardinality of $\mathscr{C}$ increases, the values of both UB_1 and UB_2 tend to increase accordingly.

To complement the aggregated analysis by family, Table 2 reports detailed results for each individual DIMACS instance, considering only the coloring produced by the DSatur heuristic. For each instance, the table lists its name, number of vertices $|V|$, number of edges $|E|$, edge density (in percentage, denoted as % μ), and the value of $\omega(G, \boldsymbol{c})$ (when known). The following columns report, for both UB_1 and UB_2, the value of the upper bound, its percentage gap from $\omega(G, \boldsymbol{c})$, and the time in seconds required to compute it (this last metric is omitted for UB_2, as its computation time is negligible). The last column shows the percentage difference between the two upper bounds. For each instance, the smaller of the two upper bound values is highlighted in bold.

The results shown in Table 2 confirm that UB_2 outperforms UB_1 on nearly all instances, with the only exceptions being `c-fat200-1` and `c-fat200-5`. The largest average percentage difference between the two bounds is observed on instance `san1000`, reaching 93.8%.

Despite the dominance of UB_2 over UB_1, it is important to note that both upper bounds are generally weak and often far from $\omega(G, \boldsymbol{c})$. In many instances, the gap exceeds 90% for both bounds, highlighting their limited tightness. Conversely, only in a few cases does either bound achieve a gap below 10%.

Table 2: Performance of the two upper bounds on $\omega(G, \boldsymbol{c})$ for DIMACS instances and the coloring produced by DSatur heuristic.

Instances					UB_1			UB_2		
name	$\|V\|$	$\|E\|$	% μ	$\omega(G, \boldsymbol{c})$	val	% gap	time	val	% gap	% diff.
brock200_1	200	14,834	74.5	21,230	346,396	93.9	0.1	**208,068**	89.8	39.9
brock200_2	200	9,876	49.6	6,542	139,042	95.3	0.1	**80,743**	91.9	41.9
brock200_3	200	12,048	60.5	10,303	209,409	95.1	0.1	**120,926**	91.5	42.3
brock200_4	200	13,089	65.8	13,967	271,661	94.9	0.1	**164,661**	91.5	39.4
brock400_1	400	59,723	74.8	-	1,277,900	-	0.5	**677,822**	-	47.0
brock400_2	400	59,786	74.9	-	1,276,930	-	0.5	**697,422**	-	45.4
brock400_3	400	59,681	74.8	46,785	1,269,990	96.3	0.6	**695,741**	93.3	45.2
brock400_4	400	59,765	74.9	54,304	1,286,460	95.8	0.6	**720,641**	92.5	44.0
brock800_1	800	207,505	64.9	-	3,218,480	-	3.1	**1,530,610**	-	52.4
brock800_2	800	208,166	65.1	-	3,351,520	-	2.8	**1,600,580**	-	52.2
brock800_3	800	207,333	64.9	-	3,183,140	-	3.3	**1,509,330**	-	52.6
brock800_4	800	207,643	65.0	-	3,264,490	-	3.8	**1,522,150**	-	53.4
c-fat200-1	200	1,534	7.7	7,734	**10,677**	27.6	0.0	10,740	28.0	-0.6
c-fat200-2	200	3,235	16.3	26,389	36,326	27.4	0.0	**30,111**	12.4	17.1
c-fat200-5	200	8,473	42.6	168,200	**304,982**	44.8	0.1	350,016	51.9	-12.9
c-fat500-1	500	4,459	3.6	10,738	14,422	25.5	0.1	**12,771**	15.9	11.5
c-fat500-2	500	9,139	7.3	38,350	50,942	24.7	0.1	**40,211**	4.6	21.1
c-fat500-5	500	23,191	18.6	205,864	292,208	29.5	0.2	**227,384**	9.5	22.2
c-fat500-10	500	46,627	37.4	804,000	1,174,240	31.5	0.5	**856,328**	6.1	27.1
C125.9	125	6,963	89.8	66,248	279,003	76.3	0.1	**194,171**	65.9	30.4
C250.9	250	27,984	89.9	-	928,442	-	0.3	**653,854**	-	29.6
C500.9	500	112,332	90.0	-	3,458,980	-	1.6	**2,194,440**	-	36.6
C1000.9	1,000	450,079	90.1	-	12,581,900	-	8.8	**7,134,690**	-	43.3
C2000.5	2,000	999,836	50.0	-	9,876,680	-	32.8	**3,903,260**	-	60.5
C2000.9	2,000	1,799,532	90.0	-	47,423,100	-	41.5	**25,110,300**	-	47.1
C4000.5	4,000	4,000,268	50.0	-	35,853,700	-	274.2	**12,867,500**	-	64.1
gen200_p0.9_44	200	17,910	90.0	94,362	441,099	78.6	0.1	**265,920**	64.5	39.7
gen200_p0.9_55	200	17,910	90.0	150,839	590,043	74.4	0.1	**401,289**	62.4	32.0
gen400_p0.9_55	400	71,820	90.0	-	1,107,870	-	0.8	**470,090**	-	57.6
gen400_p0.9_65	400	71,820	90.0	-	1,528,460	-	0.7	**804,721**	-	47.4
gen400_p0.9_75	400	71,820	90.0	-	1,521,530	-	0.7	**807,171**	-	47.0
hamming6-2	64	1,824	90.5	32,736	60,192	45.6	0.0	**33,152**	1.3	44.9
hamming6-4	64	704	34.9	396	4,264	90.7	0.0	**1,887**	79.0	55.7
hamming8-2	256	31,616	96.9	800,624	1,558,810	48.6	0.3	**959,214**	16.5	38.5
hamming8-4	256	20,864	63.9	12,360	146,767	91.6	0.2	**42,622**	71.0	71.0
hamming10-2	1,024	518,656	99.0	-	26,113,600	-	6.6	**16,418,800**	-	37.1
hamming10-4	1,024	434,176	82.9	-	2,798,370	-	9.0	**620,286**	-	77.8
johnson8-2-4	28	210	55.6	192	1,079	82.2	0.0	**470**	59.1	56.5

Instances					UB_1			UB_2		
name	$\|V\|$	$\|E\|$	% μ	$\omega(G, \boldsymbol{c})$	val	% gap	time	val	% gap	% diff.
johnson8-4-4	70	1,855	76.8	6,552	29,569	77.8	0.0	**16,196**	59.5	45.2
johnson16-2-4	120	5,460	76.5	3,808	51,784	92.6	0.1	**14,381**	73.5	72.2
johnson32-2-4	496	107,880	87.9	-	454,263	-	3.6	**79,529**	-	82.5
keller4	171	9,435	64.9	6,745	143,427	95.3	0.1	**73,383**	90.8	48.8
keller5	776	225,990	75.2	-	2,693,370	-	7.2	**1,075,770**	-	60.1
keller6	3,361	4,619,898	81.8	-	24,531,900	-	399.9	**6,377,880**	-	74.0
MANN_a9	45	918	92.7	5,460	15,019	63.6	0.0	**8,532**	36.0	43.2
MANN_a27	378	70,551	99.0	-	2,390,210	-	0.7	**1,060,520**	-	55.6
MANN_a45	1,035	533,115	99.6	-	17,904,100	-	7.8	**7,161,940**	-	60.0
MANN_a81	3,321	5,506,380	99.9	-	184,688,000	-	117.8	**69,119,400**	-	62.6
p_hat300-1	300	10,933	24.4	3,321	77,256	95.7	0.2	**43,347**	92.3	43.9
p_hat300-2	300	21,928	48.9	31,564	291,594	89.2	0.3	**161,521**	80.5	44.6
p_hat300-3	300	33,390	74.4	63,390	719,293	91.2	0.5	**392,526**	83.9	45.4
p_hat500-1	500	31,569	25.3	4,764	200,342	97.6	0.5	**97,547**	95.1	51.3
p_hat500-2	500	62,946	50.5	63,870	797,998	92.0	1.1	**388,701**	83.6	51.3
p_hat500-3	500	93,800	75.2	-	1,901,550	-	1.3	**958,679**	-	49.6
p_hat700-1	700	60,999	24.9	5,185	344,531	98.5	1.3	**155,886**	96.7	54.8
p_hat700-2	700	121,728	49.8	-	1,399,730	-	3.4	**657,379**	-	53.0
p_hat700-3	700	183,010	74.8	-	3,427,090	-	4.9	**1,704,780**	-	50.3
p_hat1000-1	1,000	122,253	24.5	5,436	637,294	99.1	3.3	**260,628**	97.9	59.1
p_hat1000-2	1,000	244,799	49.0	-	2,607,790	-	7.3	**1,108,950**	-	57.5
p_hat1000-3	1,000	371,746	74.4	-	6,770,190	-	17.6	**3,071,040**	-	54.6
p_hat1500-1	1,500	284,923	25.3	7,135	1,377,910	99.5	11.4	**511,910**	98.6	62.8
p_hat1500-2	1,500	568,960	50.6	-	5,751,650	-	24.1	**2,369,350**	-	58.8
p_hat1500-3	1,500	847,244	75.4	-	14,909,000	-	24.6	**6,533,200**	-	56.2
san200_0.7_1	200	13,930	70.0	45,295	191,976	76.4	0.1	**76,586**	40.9	60.1
san200_0.7_2	200	13,930	70.0	15,073	122,917	87.7	0.1	**28,314**	46.8	77.0
san200_0.9_1	200	17,910	90.0	242,710	582,449	58.3	0.2	**373,655**	35.0	35.8
san200_0.9_2	200	17,910	90.0	178,468	578,346	69.1	0.1	**380,316**	53.1	34.2
san200_0.9_3	200	17,910	90.0	96,764	430,170	77.5	0.1	**235,758**	59.0	45.2
san400_0.5_1	400	39,900	50.0	7,442	129,393	94.2	0.6	**16,979**	56.2	86.9
san400_0.7_1	400	55,860	70.0	77,719	544,666	85.7	0.7	**144,594**	46.3	73.5
san400_0.7_2	400	55,860	70.0	44,155	413,269	89.3	0.6	**80,474**	45.1	80.5
san400_0.7_3	400	55,860	70.0	24,727	338,062	92.7	0.9	**74,846**	67.0	77.9
san400_0.9_1	400	71,820	90.0	-	1,766,750	-	0.8	**874,696**	-	50.5
san1000	1,000	250,500	50.2	10,661	376,843	97.2	8.1	**23,342**	54.3	93.8
sanr200_0.7	200	13,868	69.7	16,398	298,303	94.5	0.1	**177,123**	90.7	40.6
sanr200_0.9	200	17,863	89.8	-	632,064	-	0.2	**435,063**	-	31.2
sanr400_0.5	400	39,984	50.1	8,298	507,448	98.4	0.8	**260,041**	96.8	48.8
sanr400_0.7	400	55,869	70.0	22,791	1,082,260	97.9	0.6	**580,546**	96.1	46.4

5.3. Results for RANDOM instances

In this section, we present the results of the tests conducted on the RANDOM instances.

Figure 6 displays the percentage gap of the two upper bounds across different instance sizes, ranging from 10 to 90 vertices, using a box plot representation. Each box summarizes the distribution of gap values for all instances of the same size, showing the median, interquartile range, and potential outliers. Specifically, each boxplot is based on 90 instances with densities ranging from 10% to 90%, and 6 different colorings per instance, resulting in a total of 540 upper bound evaluations.

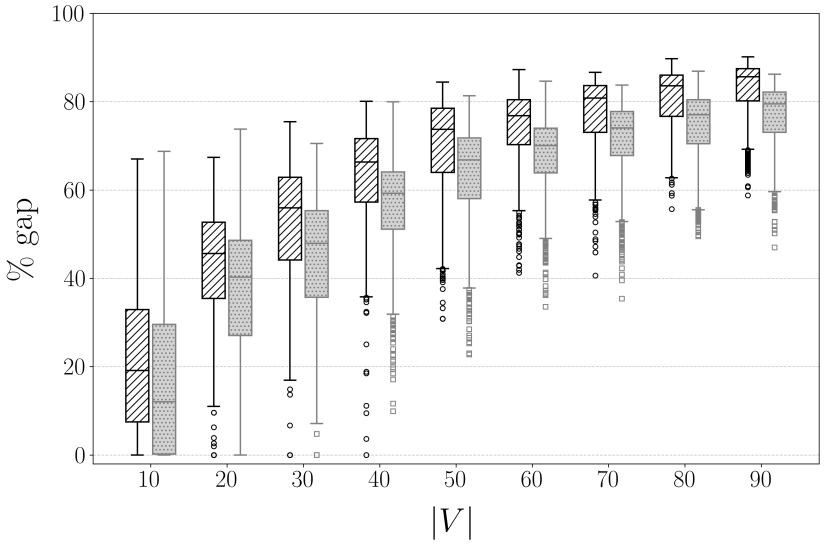
The results reveal that the median gap for both UB_1 and UB_2 is significantly large even for relatively small instances (i.e., for $|V| \geq 20$), confirming the general weakness of these upper bounds, as previously observed on the DIMACS benchmarks. On average, the percentage gap is 62.5% for UB_1 and 56.5% for UB_2, with extreme cases approaching 90%.

Moreover, the percentage gap shows a clear upward trend with increasing graph size, highlighting the growing difficulty of obtaining tight upper bounds as $|V|$ increases. Interestingly, the variability in gap values decreases for larger instances, as indicated by the shrinking interquartile range. This suggests that the performance of both upper bounds, in terms of deviation from $\omega(G, \boldsymbol{c})$, becomes more consistent as the graph size grows.

Figure 7 illustrates the influence of graph density on the percentage gap of the two upper bounds, focusing specifically on instances with $|V| = 100$, using a box plot representation. Each group consists of 10 instances, and 6 different colorings are considered for each instance, thus resulting in a total of 60 observations per group. Subfigure (a) shows results for densities from 10% to 90%, while Subfigure (b) focuses on high-density values from 91% to 99%. The results show that both upper bounds perform worst at moderate densities (below 70%). Conversely, for denser graphs, a monotonic decrease in the median gap is observed, with the lowest values—around 10%—attained for highly dense instances ($\mu = 0.99$). As the box plots reveal, both UB_1 and UB_2 exhibit weak performance across most density levels, with the median gap exceeding 40% for nearly all density values. Notably, sparse graphs appear to be more challenging than dense ones: for 10%-dense instances, both bounds reach a median gap close to 80%. Finally, and in line with the results observed for the DIMACS instances, the box plots confirm that UB_2 consistently outperforms UB_1 in terms of percentage gap, regardless of the edge density or size of the graph.

Figure 8 illustrates the percentage difference between the two upper bounds across different instance sizes ($|V|$) on randomly generated graphs. It can be noticed that, as $|V|$ increases from 10 to 90 vertices, there is an upward trend in the median difference, accompanied by increasing variability. This indicates that the bounds tend to diverge more in larger instances, in favor of UB_2. Nonetheless, results on this kind of instances reveal a less pronounced advantage for UB_2 in terms of solution quality, with respect to

Figure 6: Box plot representation of the percentage gap of the two upper bounds on RANDOM instances, grouped by number of vertices from 10 to 90. Box plots for UB_1 are filled with a diagonal line pattern, while those for UB_2 use a light gray color with a dotted pattern.



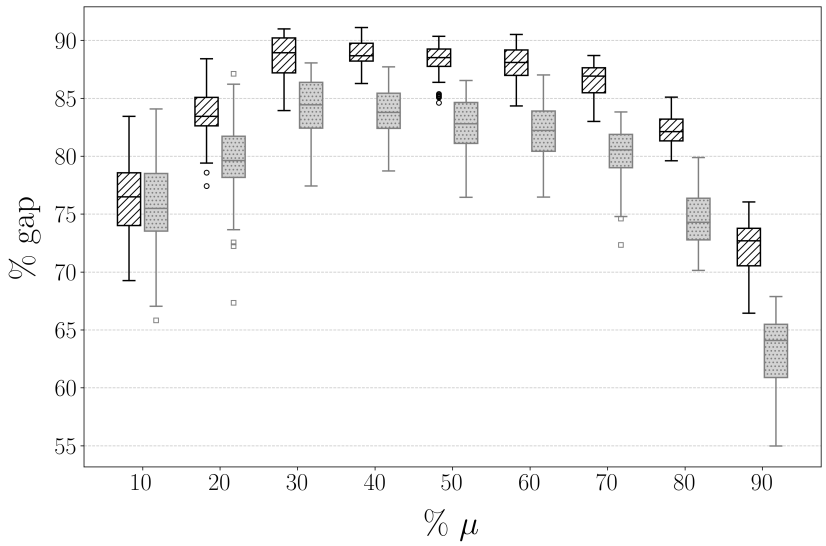
those on the DIMACS instances. Indeed, even though across all the instance sizes UB_2 is observed to be smaller, on average, than UB_1, such difference is below 10% for small graphs (up to 20 vertices).

Figure 9 focuses on RANDOM instances with $|V| = 100$ and illustrates the relationship between graph density and the percentage difference between the two upper bounds, using a box plot representation. Each group corresponds to 10 instances, and 6 different colorings are considered for each instance, resulting in a total of 60 observations per group. The box plots reveal a pattern of increasing difference up to approximately 70% density, followed by a slight decline. The right subfigure highlights high-density instances, with density values ranging from 91% to 99%. In this range, a clear decreasing trend emerges: the median percentage difference drops from around 23.5% at 91% density to nearly 2% at 99%.

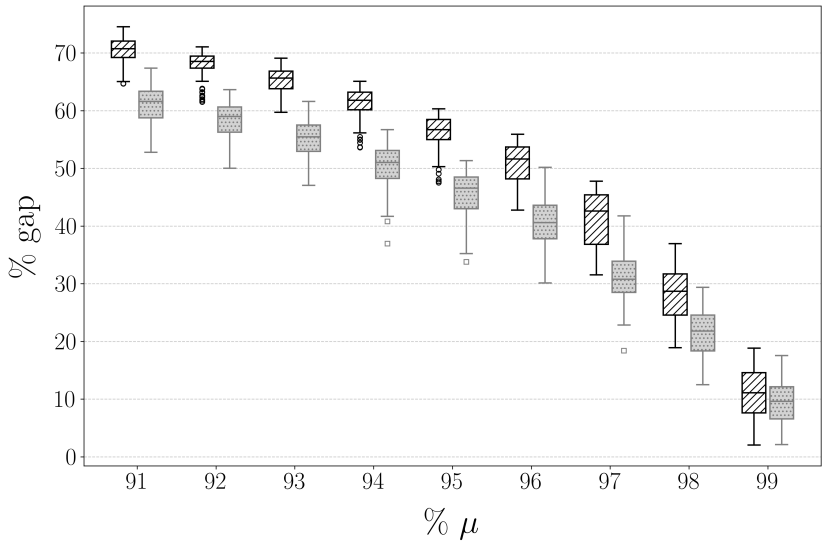
These results indicate that the strength of UB_1 and UB_2, as well as the percentage difference between them, are influenced by both graph size and density. The most pronounced differences between the two upper bounds, along with the largest percentage gap values, are observed in larger, moderately dense instances. This behavior can be attributed to the fact that, while both upper bounds tend to weaken as the number of vertices increases and around densities of approximately 70%, UB_1 exhibits reduced robustness compared to UB_2 as instance complexity grows. In summary, although both bounds are expected to perform poorly on large, moderately dense graphs, UB_1 appears to suffer more significant degradation.

Finally, it is worth to notice that the value of UB_1 is invariant under different orderings

Figure 7: Box plot representation of the percentage gap of the two upper bound on RANDOM instances with $|V| = 100$, grouped by edge density. Box plots for UB_1 are filled with a diagonal line pattern, while those for UB_2 use a light gray color with a dotted pattern.

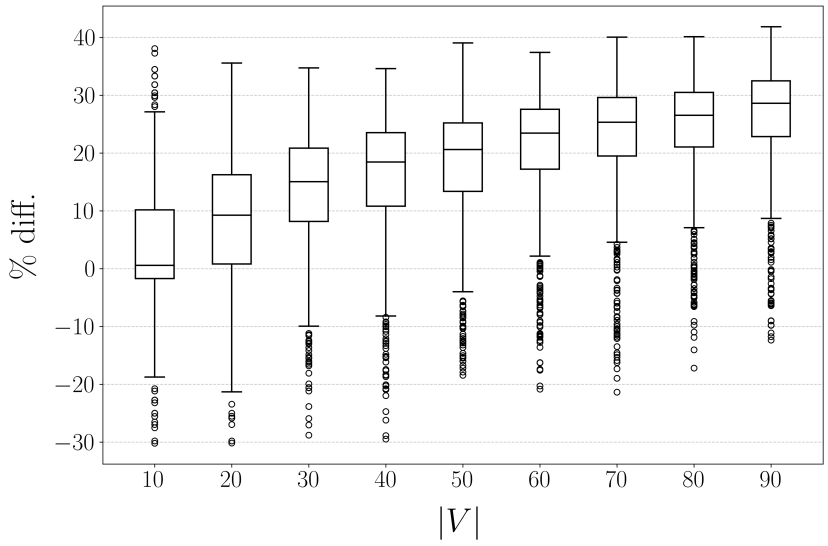


(a)



(b)

Figure 8: Box plot representation of the percentage difference of the two upper bounds on RANDOM instances, grouped by number of vertices from 10 to 90.

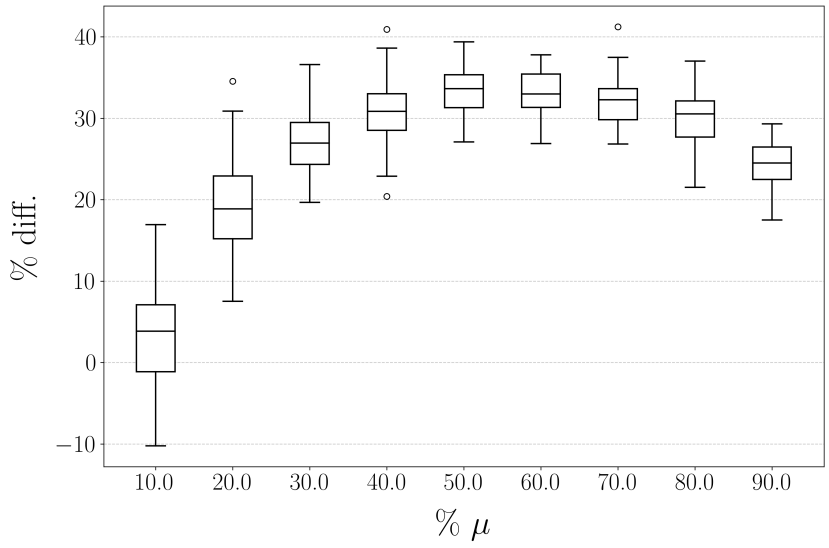


of the independent sets in the coloring, whereas UB_2 can vary significantly depending on the order (with deviations of up to 20% with respect to its average value), as shown in Tables 3 and 4 of the additional computational results published on the GitHub repository linked to this work. However, our experiments do not reveal any clear pattern that would allow for consistently arranging the independent sets to minimize the value of UB_2.

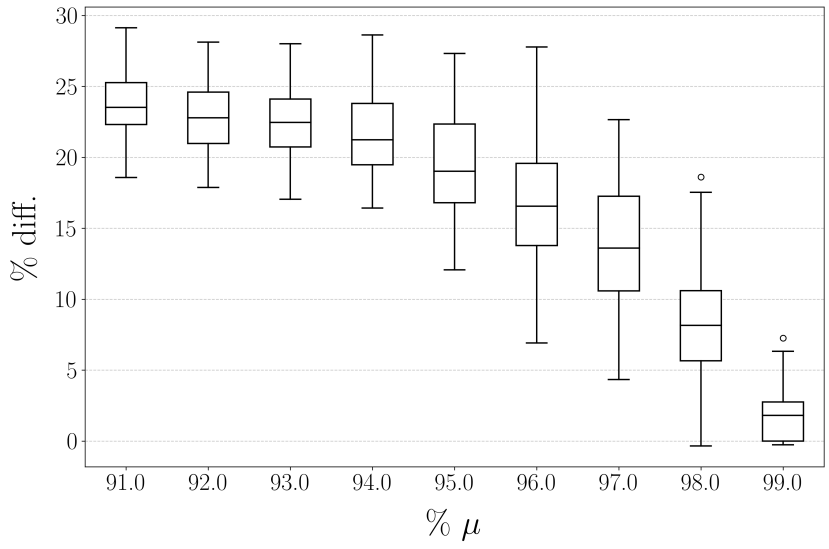
6. Conclusions

In this study, we conducted the first theoretical and computational comparison of the two main upper bounds on $\omega(G, \boldsymbol{c})$, the optimal value of the EWMCP, proposed independently by San Segundo et al. [19] and Shimizu et al. [22]. Our theoretical analysis shows that neither upper bound admits a performance guarantee: for each, there exist instances where its ratio to the optimal EWMCP value is unbounded. Moreover, we identified families of instances where one upper bound can be arbitrarily better than the other and vice versa. In particular, the bound proposed by Shimizu et al. [22] appears to perform better on instances where large independent sets are connected by a few heavy edges, while the one introduced by San Segundo et al. [19] tends to yield tighter values when the coloring includes a larger number of independent sets. These observations were corroborated by our empirical analysis, showing that the two upper bounds behave differently depending on the graph's structure. While they perform similarly on small or sparse instances, the upper bound by Shimizu et al. [22] proves tighter on large, moderately dense graphs, especially around 70% density. This pattern emerges consistently across both randomly generated graphs and the DIMACS benchmark set, highlighting the importance of accounting for graph size and density

Figure 9: Box plot representation of the percentage difference of the two upper bounds on RANDOM instances with $|V| = 100$, grouped by edge density.



(a)



(b)

when selecting or designing bounding strategies for the EWMCP.

These findings open several lines for future research, including the design of hybrid upper bounds that leverage the complementary strengths of the two methods based on instance characteristics, as well as their effective integration into exact algorithms such as branch-and-bound frameworks.

References

[1] L. A. Agapito, M. Fornari, D. Ceresoli, A. Ferretti, S. Curtarolo, and M. B. Nardelli. Accurate tight-binding hamiltonians for two-dimensional and layered materials. *Physical Review B*, 93(12):125137, 2016.

[2] D. Brélaz. New methods to color the vertices of a graph. *Communications of the ACM*, 22(4):251–256, 1979.

[3] S. Coniglio, F. Furini, and P. San Segundo. A new combinatorial branch-and-bound algorithm for the knapsack problem with conflicts. *European Journal of Operational Research*, 289(2):435–455, 2021.

[4] D. Cornaz, F. Furini, and E. Malaguti. Solving vertex coloring problems as maximum weight stable set problems. *Discrete Applied Mathematics*, 217:151–162, 2017.

[5] D. Delle Donne, F. Furini, E. Malaguti, and R. Wolfler Calvo. A branch-and-price algorithm for the minimum sum coloring problem. *Discrete Applied Mathematics*, 303:39–56, 2021.

[6] DIMACS. 2nd dimacs implementation challenge – NP-Hard Problems: Maximum clique, graph coloring, and satisfiability. `http://archive.dimacs.rutgers.edu/Challenges/`, 2017.

[7] F. Furini, I. Ljubić, S. Martin, and P. San Segundo. The maximum clique interdiction problem. *European Journal of Operational Research*, 277(1):112–127, 2019.

[8] F. Furini, I. Ljubić, P. San Segundo, and Y. Zhao. A branch-and-cut algorithm for the edge interdiction clique problem. *European Journal of Operational Research*, 294(1):54–69, 2021.

[9] M. R. Gary and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. WH Freeman and Company, 1979.

[10] L. Gouveia and P. Martins. Solving the maximum edge-weight clique problem in sparse graphs with compact formulations. *EURO Journal on Computational Optimization*, 3(1):1–30, 2015.

[11] S. Hosseinian, D. B. M. M. Fontes, and S. Butenko. A nonconvex quadratic optimization approach to the maximum edge weight clique problem. *Journal of Global Optimization*, 72:219–240, 2018.

[12] R. M. Karp. Reducibility among combinatorial problems. In *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*, pages 219–241. Springer, 2010.

[13] T. L. Lei and R. L. Church. On the unified dispersion problem: Efficient formulations and exact algorithms. *European Journal of Operational Research*, 241(3):622–630, 2015.

[14] T. Ma and L. J. Latecki. Maximum weight cliques with mutex constraints for video object segmentation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–677, 2012.

[15] S. Mattia. Reformulations and complexity of the clique interdiction problem by graph mapping. *Discrete Applied Mathematics*, 354:48–57, 2024.

[16] O. A. Prokopyev, N. Kong, and D. L. Martinez-Torres. The equitable dispersion problem. *European Journal of Operational Research*, 197(1):59–67, 2009.

[17] P. San Segundo and J. Artieda. A novel clique formulation for the visual feature matching problem. *Applied Intelligence*, 43:325–342, 2015.

[18] P. San Segundo, F. Matia, D. Rodriguez-Losada, and M. Hernando. An improved bit parallel exact maximum clique algorithm. *Optimization Letters*, 7(3):467–479, 2013.

[19] P. San Segundo, S. Coniglio, F. Furini, and I. Ljubić. A new branch-and-bound algorithm for the maximum edge-weighted clique problem. *European Journal of Operational Research*, 278(1):76–90, 2019.

[20] P. San Segundo, F. Furini, and J. Artieda. A new branch-and-bound algorithm for the maximum weighted clique problem. *Computers & Operations Research*, 110: 18–33, 2019.

[21] S. Shimizu, K. Yamaguchi, and S. Masuda. A branch-and-bound based exact algorithm for the maximum edge-weight clique problem. In *Computational Science/Intelligence & Applied Informatics*, pages 27–47. Springer, 2019.

[22] S. Shimizu, K. Yamaguchi, and S. Masuda. A maximum edge-weight clique extraction algorithm based on branch-and-bound. *Discrete Optimization*, 37: 100–583, 2020.