New Bounds and Constraint Programming Models for the Weighted Vertex Coloring Problem

Olivier Goudet
Cyril Grelier
David Lesaint

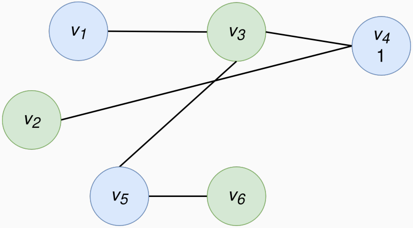
2023

Sommaire

Introduction

GCP - Graph coloring problem

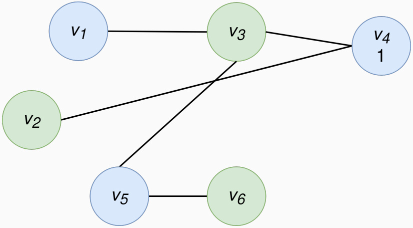
- Given an undirected graph $G = (V, E)$vertex set V, edge set E
- **Legal coloring** : partition $\{V_1, \ldots, V_k\}$ of V, such that no $(v_1, v_2) \in V_i$ are in E.
- **Objective** : find a legal coloring $S = \{V_1, \ldots, V_k\}$ with the minimum number of colors k.
- χ_G (chromatic number of the graph) : minimum number of color required to build a legal solution.



$$\chi_G = 2$$

GCP - Graph Coloring Problem

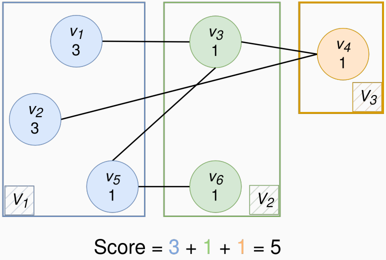
- Given an undirected graph $G = (V, E)$, a **(legal) coloring** is a partition $\{V_1, \ldots, V_k\}$ of V into independent sets ($\forall i = 1 \ldots k$, $\forall u, v \in V_i : \ (u, v) \notin E$).
- **Objective** : find a coloring $S = \{V_1, \ldots, V_k\}$ of G with the minimum number of colors k.
- Chromatic number of G (χ_G) : the minimum number of colors to color G.



$$\chi_G = 2$$

WVCP - Weighted Vertex Coloring Problem

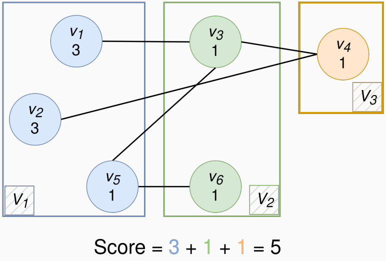
- WVCP instance $P = (G, w)$ defined by :
  - a graph $G = (V, E)$,
  - a function $w : V \mapsto \mathbb{N}^*$ assigning a strictly positive weight $w(v)$ to each vertex v.
- **Objective** : find a legal coloring $S = \{V_1, \ldots, V_k\}$ whose score $f(S) = \sum_{i=1}^{k} \max_{v \in V_i} w(v)$ is minimum.



- **Example of application** : scheduling on a Batch Machine with Job Compatibilities.

WVCP - Weighted Vertex Coloring Problem

- A WVCP instance is defined by a vertex-weighted graph (G, w) :
  - $G = (V, E)$ is an undirected graph
  - $w : V \mapsto \mathbb{N}^*$ is the weight function
- **Objective** : find a coloring $S = \{V_1, \ldots, V_k\}$ of G with a minimum score $f(S) = \sum_{i=1}^{k} \max_{v \in V_i} w(v)$.



- WVCP is NP-hard (WVCP≡GCP if w is constant).
- Example of application : scheduling on a batch machine with job compatibilities.

State of the art on WVCP

Exact methods :

- 2-Phase [Malaguti et al., 2009] : generation of independent sets and optimization
- MWSS [Cornaz et al., 2017] : Mixed Integer Linear Programming

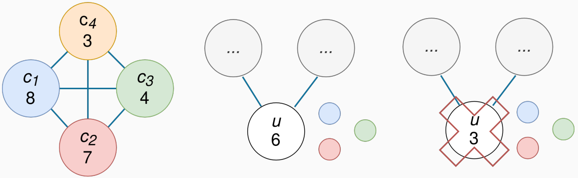
Metaheuristics : (LS : local search)

- R-GRASP [Prais and Ribeiro, 2000] : Iterated Greedy Algorithm with LS
- AFISA [Sun et al., 2018] : LS with adaptive management of weights
- RedLS [Wang et al., 2020] : Reduction and LS with weights on edges
- ILS-TS [Nogueira et al., 2021] : Iterated LS with grenade operator
- DLMCOL [Goudet et al., 2022] : Memetic Algorithm with deep learning for the crossover selection
- MCTS [Grelier et al., 2022, Grelier et al., 2023] : Monte Carlo Tree Search with LS

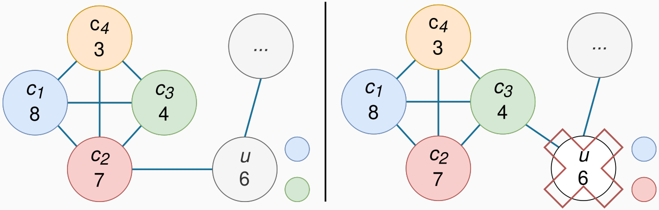
Vertex Reduction Rules and Iterative Reduction Procedure

First atomic reduction rule R1 : improvement on the rule of [Wang et al., 2020]

- Reduction rule R0 [Wang et al., 2020] :

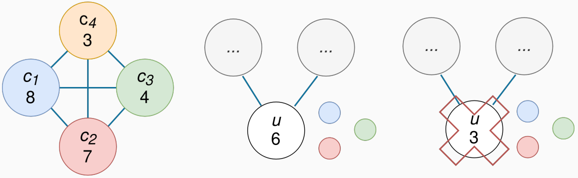


- Reduction rule R1 takes into account that u may have neighbors in the clique C :

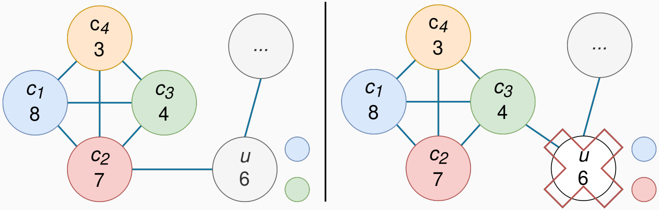


First vertex reduction rule R1

- Given a clique C and $u \notin C$, the reduction rule (R0) proposed by [Wang et al., 2020] removes u if ...

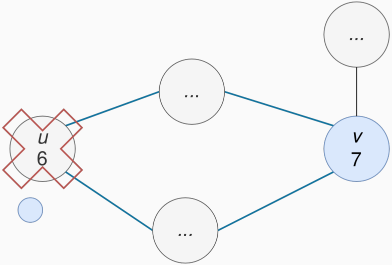


- Our reduction rule (R1) also takes into account that u may have neighbors in C



Second vertex reduction rule R2

- Second reduction rule (R2) adapted from a reduction operator proposed by [Cheeseman et al., 1991] for GCP.



Iterative reduction procedure

- Extract 1 clique of maximum weight per vertex using FastWClq algorithm [Cai and Lin, 2016].
- Sort vertices by increasing order of weights and apply R1 and R2 on each vertex.
- Iterate until no vertex can be removed.
- When a solution is found for the reduced graph, it is possible to obtain a solution of the same score for the original instance by coloring each vertex of the list L of removed vertices with a greedy algorithm in the reverse order of arrival in L.

Impact of reductions on benchmark instances

instance	$\|V\|$	density	R0	R1	R1+R2	Iterated	time(s)
DSJC125.1g	125	0.1	0	0	0	0	0.03
DSJC125.5g	125	0.5	0	0	0	0	0.26
DSJC125.9g	125	0.9	0	0	0	0	3.22
DSJR500.1	500	0.03	78	80	80	**256**	1.32
GEOM110	110	0.11	6	9	9	**23**	0.09
inithx.i.1	864	0.05	469	574	596	**683**	19.45
le450_15a	450	0.08	28	28	28	**30**	1.38
le450_25b	450	0.08	90	90	90	**105**	2.26
mulsol.i.5	186	0.23	28	53	75	**82**	1.16
queen10_10	100	0.59	0	0	0	0	0.08
p42	138	0.12	1	1	1	**3**	0.1
r30	301	0.09	0	0	0	0	0.48

Table 1 – Number of vertices removed by the different reduction rules.

Upper bounds on the score and the number of colors

New upper bounds on the score and number of colors for WVCP

- [Demange et al., 2007] : $\Delta(G) + 1$ is an upper-bound on the number of colors needed to solve P to optimality.
- **New upper bounds :**

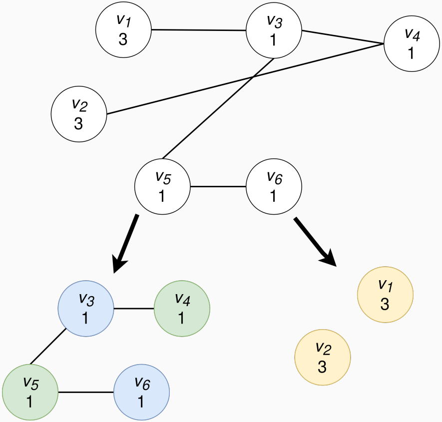
Theorem

Given a WVCP *instance* $P = (G, w)$ *with* $G = (V, E)$ *and an optimal solution* $S^* = \{V_1, \ldots, V_k\}$ *of* P *corresponding to a partition of* V *into* k *non-empty independent sets, then* $k \leq \sum_{w \in W} \chi_{G_w}$ *and* $f(S^*) \leq \sum_{w \in W} w \times \chi_{G_w}$.

With :

- $W = \{w(v) \mid v \in V\}$ the set of weight values used in G.
- $G_w = (V_w, E_w)$ the subgraph of G induced by weight w.
- χ_{G_w} the chromatic number of G_w.

Example



$\chi_{G_1} = 2$

$\chi_{G_3} = 1$

$$k \leq \chi_{G_1} + \chi_{G_3} = 3$$

$$f(S^*) \leq 1 * \chi_{G_1} + 3 * \chi_{G_3} = 5$$

Instance	$\|V'\|$	density	h_W	$\Delta+1$	colors bounds lb	colors bounds ub	score bounds lb	score bounds ub
DSJC125.1g	125	0.1	0.04	24	4	**14**	19	42
DSJC125.5g	125	0.5	0.04	76	10	**34**	42	105
DSJC125.9g	125	0.9	0.04	121	32	**72**	124	220
DSJR500.1	244	0.03	0.08	26	12	26	166	477
GEOM110	87	0.11	0.11	20	9	20	65	151
inithx.i.1	181	0.05	0.1	169	54	**78**	569	800
le450_15a	420	0.08	0.05	99	15	**61**	206	628
le450_25b	345	0.08	0.06	108	25	**73**	307	735
mulsol.i.5	104	0.23	0.18	88	31	**58**	367	574
queen10_10	100	0.59	0.19	36	10	36	153	420
p42	135	0.12	0.46	25	14	25	2466	8108
r30	301	0.09	0.76	35	19	35	9816	104285

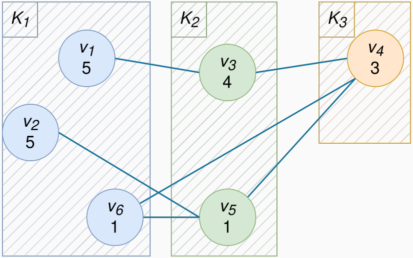
Table 2 – Lower and upper bounds on score and colors.

Three Constraint Programming Models for WVCP

Definitions

- P_κ denotes the problem of determining the existence of a solution to P that uses a number of colors smaller than or equal to κ.
- Solutions to P_κ are modeled as maps $s : [V] \mapsto K$ where $K = \{1, \ldots, \kappa\}$.
- A total ordering $\geq_w$ over V is defined which is consistent with the descending order of weights ($u \geq_w v \rightarrow w(u) \geq w(v)$ for $u, v \in V$).
- A solution is d-sorted if non-empty colors start from rank 1 and are sorted consistently with the ordering $\geq_w$ of their dominant vertices.
- $\mathcal{S}_{P_\kappa}$: set of d-solutions using a number of colors smaller than κ.

Example of d-sorted solution



Primal model for P_κ

minimize x^o *s.t.*

$$x^o \in \{\max_{v_i \in V}(w(v_i)), \ldots, \sum_{v_i \in V} w(v_i)\} \quad \text{(P1)}$$

$$\forall v_i \in U : x_i^U \in K \quad \text{(P2)}$$

$$\forall k \in K : x_k^K \in 2^U \quad \text{(P3)}$$

$$\forall k \in K : x_k^D \in U \quad \text{(P4)}$$

$$\text{INT\_SET\_CHANNEL}([x_k^K | k \in K], [x_i^U | v_i \in U]) \quad \text{(P5)}$$

$$\forall k \in K : x_{|V|+k}^U = k \quad \text{(P6)}$$

$$\forall \{v_i, v_j\} \in E : x_i^U \neq x_j^U \quad \text{(P7)}$$

$$\forall k \in K : x_k^D = \min(x_k^K) \quad \text{(P8)}$$

$$x^o = \sum_{k \in K} w[x_k^D] \quad \text{(P9)}$$

$$\text{STRICTLY\_INCREASING}(x^D) \quad \text{(P10)}$$

Experimental Settings

- Intel Xeon ES 2630, 2.66 GHz.
- OR-Tools [Perron and Furnon, 2022] solver.
- Heuristics *first-fail* combined with domain bisection.
- Time limit of 1 hour for each run on a single CPU.

Primal model results and impact of pre-computed bounds

		primal		primal ub color		primal all bounds	
instance	BKS	score	time(s)	score	time(s)	score	time(s)
DSJC125.1g	23	**23***	862	**23***	435	**23***	451
DSJC125.5gb	240	270	tl	270	tl	270	tl
DSJC125.5g	71	78	tl	78	tl	78	tl
DSJC125.9g	169*	176	tl	176	tl	176	tl
DSJR500.1	169	187	tl	177	tl	169	tl
GEOM110	68*	69	tl	**68***	1893	**68***	1729
inithx.i.1	569*	569	tl	569	tl	**569***	54
le450_15a	212	245	tl	234	tl	234	tl
le450_25b	307	307	tl	307	tl	**307***	322
mulsol.i.5	367*	367	tl	367	tl	**367***	31
queen10_10	162	170	tl	169	tl	169	tl
p42	2466*	2480	tl	2466	tl	**2466***	2908
r30	9816*	9831	tl	9831	tl	9831	tl
nb bks reached		101/188		105/188		107/188	
nb optim		72/188		75/188		95/188	

Table 3 – Primal model results and impact of pre-computed bounds.

Compact solutions

- A solution is compact, if the color value of each vertex cannot be reduced.
- Proposal of an algorithm, g_{P_κ}, compacting any d-sorted solution without deteriorating its score.

Theorem

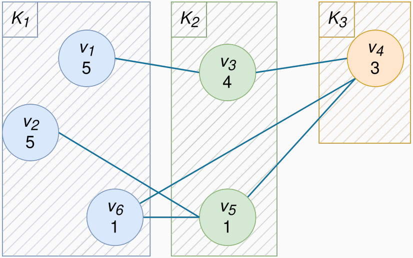
Let P_κ be a satisfiable WVCP *instance. There exists $g_{P_\kappa} : \mathcal{S}_{P_\kappa} \mapsto \mathcal{S}_{P_\kappa}$ such that, for all $s \in \mathcal{S}_{P_\kappa}$, $g_{P_\kappa}(s)$ is compact, $f(g_{P_\kappa}(s)) \leq f(s)$ and $g_{P_\kappa}(g_{P_\kappa}(s)) = g_{P_\kappa}(s)$.*

Corollaries

1. Reduction of the domain of each variable v to $\{1, \dots, min(\kappa, \Delta(v) + 1)\}$.
2. New global constraint to achieve compactness

Example of compact solution

- This d-sorted solution is compact since neither v_3, v_4 nor v_5 may be left-shifted.
- If (v_3, v_4) and (v_5, v_6) were not part of the graph, then v_5 and v_4 could be left-shifted to colors K_1 and K_2 respectively to compact the solution.



Enforcing solution compactness

- **Definition** : Let y be an integer domain variable and $[x_1, \ldots, x_n]$ be a vector of positive integer domain variables ($n \geq 0$). $\text{MAX\_LEFT\_SHIFT}(y, [x_1, \ldots, x_n])$ holds iff $y = \min_{k=1..n+1}(\{k \mid \forall i = 1..n : x_i \neq k\})$.
- **New global constraint for the primal model** :

$$\forall v_i \in V : \text{MAX\_LEFT\_SHIFT}(x_i^U, [x_j^U | v_j \in N(v_i)]) \qquad \text{(P11)}$$

Implementation of max_left_shift

Decomposition of MAX_LEFT_SHIFT using constraints (M1-M3) with global constraint NVALUE [Bessiere et al., 2006] :

$$\text{MAX\_LEFT\_SHIFT}(y, [x_1, \ldots, x_n]) \equiv$$

$$\forall i \in \{1, \ldots, n\} : z_i \in \{0, \ldots, n+1\} \quad \text{(M1)}$$

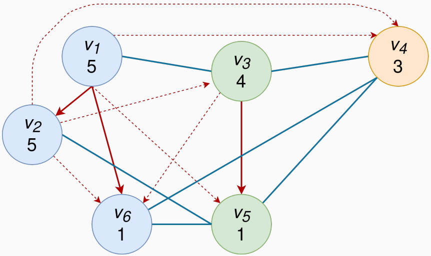
$$\forall i \in \{1, \ldots, n\} : z_i = (y > x_i) \times x_i \quad \text{(M2)}$$

$$\text{NVALUE}(y, [0, z_1, \ldots, z_n]) \quad \text{(M3)}$$

Impact of this symmetry breaking rule on the results

		primal		primal + P11	
instance	BKS	score	time(s)	score	time(s)
DSJC125.1g	23	**23***	862	**23***	628
DSJC125.5g	71	78	tl	78	tl
DSJC125.9g	169*	176	tl	176	tl
DSJR500.1	169	187	tl	173	tl
GEOM110	68*	69	tl	**68***	53
inithx.i.1	569*	569	tl	569	tl
le450_15a	212	245	tl	235	tl
le450_25b	307	307	tl	310	tl
mulsol.i.5	367*	367	tl	367	tl
queen10_10	162	170	tl	170	tl
p42	2466*	2480	tl	2480	tl
r30	9816*	9831	tl	9831	tl
nb BKS reached		101/188		102/188	
nb optim		72/188		76/188	

Dual graph from [Cornaz and Jost, 2008]



Primal

$Score_P = w(v_1) + w(v_3) + w(v_4) = 12$

Dual

$Score_D = w(v_1) + w(v_3) + w(v_4) = 7$

$$Score_P + Score_D = \sum_{i=1}^{n} w(v_i) = 19$$

- **Set of arcs of the dual graph** : $\overrightarrow{E^c} = \{ij \mid v_i, v_j \in V \wedge \{v_i, v_j\} \notin E \wedge v_i \geq_w v_j\}$.
- **Solution in the dual model** : a set of simplicial stars that span disjoint subsets of nodes.

Dual model for P_κ - adaptation from [Cornaz et al., 2017]

maximize y^o *s.t.*

$$\forall ij \in \overrightarrow{E^c} : y^A_{ij} \in \{0,1\} \tag{D1}$$

$$y^o \in \{0, \dots, \sum_{v_i \in V} (w(v_i))\} \tag{D2}$$

$$y^o = \sum_{ij \in \overrightarrow{E^c}} w(v_j) \times y^A_{ij} \tag{D3}$$

$$\forall ij, ik \in \overrightarrow{E^c} \ s.t. \ \{jk, kj\} \cap \overrightarrow{E^c} = \emptyset :$$

$$y^A_{ij} + y^A_{ik} \leq 1 \tag{D4}$$

$$\forall ij, jk \in \overrightarrow{E^c} : y^A_{ij} + y^A_{jk} \leq 1 \tag{D5}$$

$$\forall hj, ij \in \overrightarrow{E^c} : y^A_{hj} + y^A_{ij} \leq 1 \tag{D6}$$

$$\forall v_i \in V : z^V_i \in \{0,1\} \tag{D7}$$

$$\forall v_i \in T : z^V_i = 1 - \max_{(h,i) \in \overrightarrow{E^c}} (y^A_{hi}) \tag{D8}$$

$$\forall v_i \in V \setminus T : z^V_i = 1 \tag{D9}$$

$$\sum z^V_i \leq \kappa \tag{D10}$$

Joint model

Joint Model = Primal + Dual + J1-J4 channeling constraints.

minimize x^o *s.t.*

$$\forall ij \in \overrightarrow{E^c} : y^A_{ij} \leq (x^U_i = x^U_j) \quad \text{(J1)}$$

$$\mathrm{GCC}([x^D_k \mid k \in K],\ V,\ [z^V_i \mid v_i \in V]) \quad \text{(J2)}$$

$$x^o + y^o = \sum_{v_i \in V} w(v_i) \quad \text{(J3)}$$

$$\forall v_i \in V, v_j \in \overline{N(v_i)} \ s.t.\ v_j \geq_w v_i :$$

$$\left(\bigwedge_{v_h \in N(v_i) \cap \overline{N(v_j)}} x^U_h \neq x^U_j \right) \Rightarrow x^U_i \leq x^U_j \quad \text{(J4)}$$

Results of the different CP models

instance	BKS	primal score	primal time(s)	primal + P11 score	primal + P11 time(s)	dual score	dual time(s)	joint + J4 score	joint + J4 time(s)
DSJC125.1g	23	**23***	862	**23***	628	26	tl	24	tl
DSJC125.5g	71	78	tl	78	tl	84	tl	78	tl
DSJC125.9g	169*	176	tl	176	tl	**169***	56	**169***	380
DSJR500.1	169	187	tl	173	tl	187	tl	186	tl
GEOM110	68*	69	tl	**68***	53	73	tl	**68***	741
inithx.i.1	569*	569	tl	569	tl	569	tl	**569***	1923
le450_15a	212	245	tl	235	tl	250	tl	-	tl
le450_25b	307	307	tl	310	tl	314	tl	-	tl
mulsol.i.5	367*	367	tl	367	tl	367	tl	**367***	203
queen10_10	162	170	tl	170	tl	177	tl	172	tl
p42	2466*	2480	tl	2480	tl	2517	tl	**2466***	673
r30	9816*	9831	tl	9831	tl	9831	tl	9831	tl
nb BKS reached		101/188		102/188		79/188		112/188	
nb optim		72/188		76/188		68/188		100/188	

Table 5 – Results of the different CP models.

New optimality proofs

We ran the CP models with pre-computed bounds during 1h in parallel on 10 threads.

instance	$\|V\|$	BKS	score	time(s)	instance	$\|V\|$	BKS	score	time(s)
DSJC125.1gb	125	90	**90***	25	myciel7gb	191	109	**109***	69
DSJC125.1g	125	23	**23***	11	myciel7g	191	29	**29***	241
DSJR500.1	500	169	**169***	66	queen9_9g	81	41	**41***	509
myciel6gb	95	94	**94***	17	queen10_10g	100	43	**43***	820
myciel6g	95	26	**26***	17	le450 25b	450	307	**307***	322

Table 6 – New optimality proofs for difficult benchmark instances.

Conclusion

- Iterative reduction procedure and new upper bounds on the score and the number of colors.
  $\rightarrow$ Reduce the search space.
- Three competitive and complementary CP models.
- 10 new optimality proofs for difficult benchmark instances.
- Future work : investigate possible hybridizations of the CP models with metaheuristics.

Thank you for your attention !

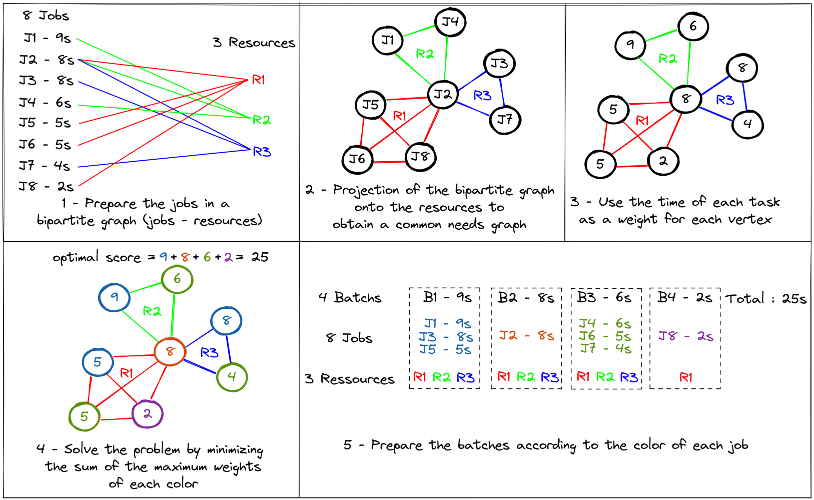
Questions ?

technical appendix and source code :

https://github.com/Cyril-Grelier/gc_wvcp_cp

Example of WVCP concrete application

Scheduling on a Batch Machine with Job Compatibilities



Bessiere, C., Hebrard, E., Hnich, B., Kiziltan, Z., and Walsh, T. (2006).
Filtering algorithms for the nvalueconstraint.
Constraints, 11(4) :271–293.

Cai, S. and Lin, J. (2016).
Fast solving maximum weight clique problem in massive graphs.
In *In Proceedings of the 25th International Joint Conference on Artificial intelligence*, pages 568–574.

Cheeseman, P., Kanefsky, B., and Taylor, W. M. (1991).
Where the really hard problems are.
In *Proceedings of the 12th International Joint Conference on Artificial intelligence - Volume 1*, page 331–337.

Bibliographie ii

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

Cornaz, D. and Jost, V. (2008).
A one-to-one correspondence between colorings and stable sets.
Operations Research Letters, 36(6) :673–676.

Demange, M., Werra, D. d., Monnot, J., and Paschos, V. T. (2007).
Time slot scheduling of compatible jobs.
Journal of Scheduling, 10(2) :111–127.

Bibliographie iii

Goudet, O., Grelier, C., and Hao, J.-K. (2022).
A deep learning guided memetic framework for graph coloring problems.
Knowledge-Based Systems, 258 :109986.

Grelier, C., Goudet, O., and Hao, J.-K. (2022).
On monte carlo tree search for weighted vertex coloring.
In Pérez Cáceres, L. and Verel, S., editors, *Evolutionary Computation in Combinatorial Optimization*, volume 13222 of *Lecture Notes in Computer Science*, pages 1–16.

Grelier, C., Goudet, O., and Hao, J.-K. (2023).
Monte carlo tree search with adaptive simulation : A case study on weighted vertex coloring.
In *Evolutionary Computation in Combinatorial Optimization : 23rd European Conference, EvoCOP 2023*, pages 98–113. Springer.

Malaguti, E., Monaci, M., and Toth, P. (2009).
Models and heuristic algorithms for a weighted vertex coloring problem.
Journal of Heuristics, 15(5) :503–526.

Nogueira, B., Tavares, E., and Maciel, P. (2021).
Iterated local search with tabu search for the weighted vertex coloring problem.
Computers & Operations Research, 125 :105087.

Perron, L. and Furnon, V. (2022).
Or-tools v9.4.
`https://developers.google.com/optimization/`.
Accessed : 2023-05-25.

Bibliographie v

Prais, M. and Ribeiro, C. C. (2000).
Reactive grasp : An application to a matrix decomposition problem in tdma traffic assignment.
INFORMS Journal on Computing, 12(3) :164–176.

Sun, W., Hao, J.-K., Lai, X., and Wu, Q. (2018).
Adaptive feasible and infeasible tabu search for weighted vertex coloring.
Information Sciences, 466 :203–219.

Wang, Yiyuanand Cai, S., Pan, S., Li, X., and Yin, M. (2020).
Reduction and local search for weighted graph coloring problem.
In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2433–2441.