On variations of the coloring problem

F. Bonomo, M. Cecowski, G. Durán and J. Marenco

CONICET / Departamento de Computación, Universidad de Buenos Aires
Argentina

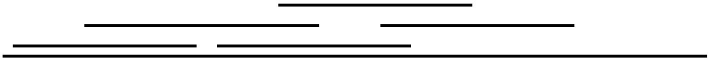
Università Tor Vergata, Marzo 2008

The classes allocation problem

Let us suppose the following problem: we have to assign classrooms to the courses of a semester. But, for example, if Algebra is on Monday from 14 to 17 and Graph Theory is on Monday from 16 to 19, then they cannot be assigned the same classroom.

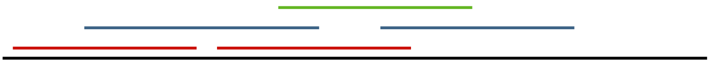
Model for the problem

- For each day of the week, we can place the schedule of courses as segments on a line, then two courses cannot share classroom when the segments have non-empty intersection.
- Now, we can think the classrooms as colors, so we have to give a color to each segment in such a way that no intersecting segments receive the same color.
- Finally, the number k of available classrooms is fixed (and usually less than the number of courses).



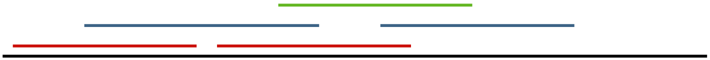
Model for the problem

- For each day of the week, we can place the schedule of courses as segments on a line, then two courses cannot share classroom when the segments have non-empty intersection.
- Now, we can think the classrooms as colors, so we have to give a color to each segment in such a way that no intersecting segments receive the same color.
- Finally, the number k of available classrooms is fixed (and usually less than the number of courses).



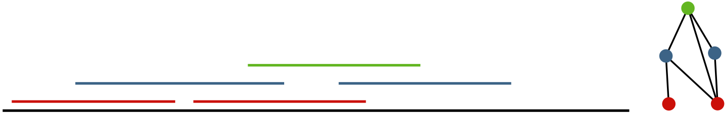
Model for the problem

- For each day of the week, we can place the schedule of courses as segments on a line, then two courses cannot share classroom when the segments have non-empty intersection.
- Now, we can think the classrooms as colors, so we have to give a color to each segment in such a way that no intersecting segments receive the same color.
- Finally, the number k of available classrooms is fixed (and usually less than the number of courses).



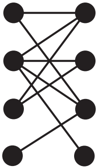
Interval graphs

- A graph is an interval graph if it is the intersection graph of a set of intervals over the real line. A unit interval graph is the intersection graph of a set of intervals of length one.

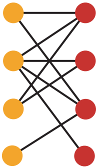


The k-coloring problem

- A coloring of a graph $G = (V, E)$ is a function $f : V \rightarrow \mathbb{N}$ such that $f(v) \neq f(w)$ whenever $vw \in E$.
- A k-coloring is a coloring f such that $f(v) \leq k$ for every $v \in V$.
- The vertex coloring problem, or k-coloring problem, takes as input a graph G and a natural number k, and consists in deciding whether G is k-colorable or not.



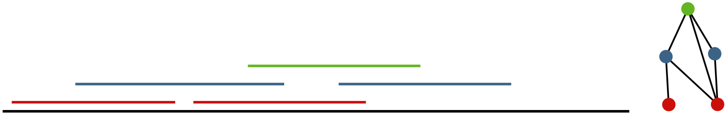
k=3



k=3

Solving k-coloring on interval graphs

For solving k-coloring on interval graphs, it is enough to apply a greedy algorithm in the order of the start point of the intervals. That is, taking the vertices by order of starting, give the minimum color not used by a colored neighbor.



If an interval $\mathcal{I}$ cannot be colored with a color at most k, then it has k colored neighbors $\mathcal{I}_1, \ldots, \mathcal{I}_k$ using colors $1, \ldots, k$, respectively. Since all of them start before $\mathcal{I}$ and intersect $\mathcal{I}$, the all pass through the starting point of $\mathcal{I}$. So, there is a clique of size $k+1$ in the graph.

Perfect graphs

- In the previous algorithm, we get that the graph is not k-colorable when we find a clique of size at least $k+1$.
- That happens because interval graphs are perfect.
- An alternative definition for perfect graphs is the following: "G is perfect when for every induced subgraph H of G and for every k, H is k-colorable if and only if every clique of H is k-colorable".

The classes allocation problem with capacities

Suppose now a more realistic situation: each classroom has a certain capacity and the courses have different numbers of students.

If we order the classrooms by capacity (decreasing), then the most popular course maybe needs the classrooms 1, 2 or 3, while a boring non-mandatory course can use classrooms 1, 2, . . . , 12.

This situation leads to define the μ-coloring problem.

The μ-coloring problem

- Given a graph G and a function $\mu : V \to \mathbb{N}$, G is μ-colorable if there exists a coloring f of G such that $f(v) \leq \mu(v)$ for every $v \in V$.

Goals

The first goal was to solve μ-coloring on interval graphs. But, in the meantime, we begin to study the problem in general. In particular, the notion of perfectness for μ-coloring.

M-perfect graphs

Analogously to the alternative definition of perfect graphs, we define M-perfect graphs as follows:

A graph G is M-perfect when for every induced subgraph H of G and for every function $\mu : V \rightarrow \mathbb{N}$, H is μ-colorable if and only if every clique of H is μ-colorable.

M-perfect graphs

Analogously to the alternative definition of perfect graphs, we define M-perfect graphs as follows:

A graph G is M-perfect when for every induced subgraph H of G and for every function $\mu : V \to \mathbb{N}$, H is μ-colorable if and only if every clique of H is μ-colorable.

M-perfect graphs are also perfect, because perfection is equivalent to M-perfection with μ restricted to constant functions. An example of a perfect graph being not M-perfect is the following. Define μ as in the figure. Clearly, every clique is μ-colorable, but the whole graph is not.

M-perfect graphs

The greedy coloring algorithm consists of successively color the vertices with the least possible color in a given order.

Theorem (Chvátal, 1984)

The greedy coloring algorithm gives an optimal coloring for cographs, independently of the order of the vertices.

Corollary 1

The greedy coloring algorithm applied to the vertices in non-decreasing order of μ gives a μ-coloring for a cograph, when it is μ-colorable.

Corollary 2

A graph is M-perfect if and only if it is P_4-free, that is, a cograph.

M-perfect graphs

The greedy coloring algorithm consists of successively color the vertices with the least possible color in a given order.

Theorem (Chvátal, 1984)

The greedy coloring algorithm gives an optimal coloring for cographs, independently of the order of the vertices.

Corollary 1

The greedy coloring algorithm applied to the vertices in non-decreasing order of μ gives a μ-coloring for a cograph, when it is μ-colorable.

Corollary 2

A graph is M-perfect if and only if it is P_4-free, that is, a cograph.

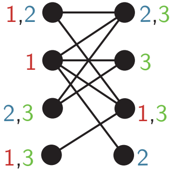
Common properties between M-perfect graphs and perfect graphs

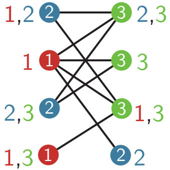
- Self-complementary class.
- Can be recognized in polynomial time.
- The corresponding coloring problem (μ-coloring, resp. k-coloring) can be solved in polynomial time.

The list-coloring problem

In order to take into account particular constraints arising in practical settings, more elaborate models of vertex coloring have been defined in the literature. One of such generalized models is the list-coloring problem, which considers a prespecified set of available colors for each vertex.

- Given a graph G and a finite list $L(v) \subseteq \mathbb{N}$ for each vertex $v \in V$, the list-coloring problem asks for a list-coloring of G, i.e., a coloring f such that $f(v) \in L(v)$ for every $v \in V$.





The list-coloring problem

- The list-coloring problem is NP-complete for perfect graphs, and is also NP-complete for many subclasses of perfect graphs, including cographs, split graphs, interval graphs, and bipartite graphs.
- Trees and complete graphs are two classes of graphs where the list-coloring problem can be solved in polynomial time. In the first case it can be solved using dynamic programming techniques [Jansen-Scheffler, 1997]. In the second case, the problem can be reduced to the maximum matching problem in bipartite graphs.
- In this case, a definition of perfection similar to that for μ-coloring, leads only to disjoint unions of complete graphs.

Relation between the problems

The μ-coloring problem particular case of list-coloring, where for each vertex $v \in V(G)$, the list $L(v) = \{i \in \mathbb{N} : 1 \leq i \leq \mu(v)\}$.

In turn, the k-coloring problem is a particular case of μ-coloring, where for each vertex $v \in V(G)$, it holds $\mu(v) = k$.

So, we can think those as polynomial reductions from k-coloring to μ-coloring and from μ-coloring to list-coloring that do not change the input graph.

Therefore, for any class of graphs $\mathcal{G}$, if list-coloring is polynomially solvable in $\mathcal{G}$ then so is μ-coloring; if μ-coloring is polynomially solvable in $\mathcal{G}$ then so is k-coloring; if k-coloring is NP-complete in $\mathcal{G}$, so is μ-coloring and if μ-coloring is NP-complete in $\mathcal{G}$, so is list-coloring.

Interesting classes

Class	coloring	list-col.
Complete bipartite	P	NP-c [Jansen-Scheffler, 1997]
Bipartite	P	NP-c [Hujter-Tuza, 1993]
Cographs	P	NP-c [Hujter-Tuza, Jansen-Scheffler, 1996]
Distance-hereditary	P	NP-c
Unit interval	P	NP-c [Marx, 2004]
Interval	P	NP-c
Complete split	P	NP-c [Jansen-Scheffler, 1997]
Split	P	NP-c
Line of $K_{n,n}$	P	NP-c [Colbourn, 1984]
Line of K_n	P	NP-c [Kubale, 1992]
Complement of bipartite	P	NP-c [Jansen, 1997]

"NP-c": NP-complete problem, "P": polynomial problem.

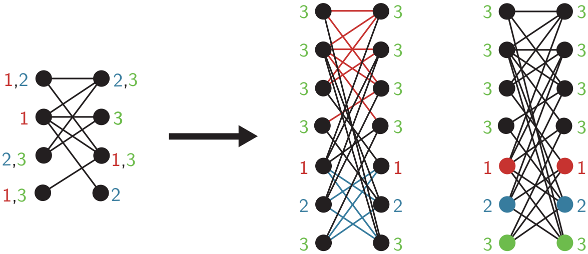
We were interested in studying the computational complexity of μ-coloring over different subclasses of graphs (most of them subclasses of perfect graphs) where vertex coloring is polynomially solvable and list-coloring is NP-complete or unknown.

μ-coloring on bipartite graphs

Theorem

μ-coloring is NP-complete for bipartite graphs.

The reduction is from bipartite list-coloring.



Bounds on the number of colors: cographs

Also as a corollary of Chvátal's Theorem, we have the following result.

Corollary

Let G be a cograph, and let μ be a function such that G is μ-colorable. Then G can be μ-colored using at most the first $\chi(G)$ colors.

This does not happen for bipartite graphs, even for trees. But some upper bounds can be proved.

Bounds on the number of colors: cographs

Also as a corollary of Chvátal's Theorem, we have the following result.

Corollary

Let G be a cograph, and let μ be a function such that G is μ-colorable. Then G can be μ-colored using at most the first $\chi(G)$ colors.

This does not happen for bipartite graphs, even for trees. But some upper bounds can be proved.

Bounds on the number of colors: trees

Theorem 5

Let T be a tree, and let μ be a function such that T is μ-colorable. Then T can be μ-colored using at most the first $\log_2(|V(T)|) + 1$ colors.

There is a family $\{T_n\}_{n \in \mathbb{N}}$ of trees and $\{\mu_n\}_{n \in \mathbb{N}}$ of functions such that T_n requires n colors to be μ_n-colored, and it has 2^{n-1} vertices.

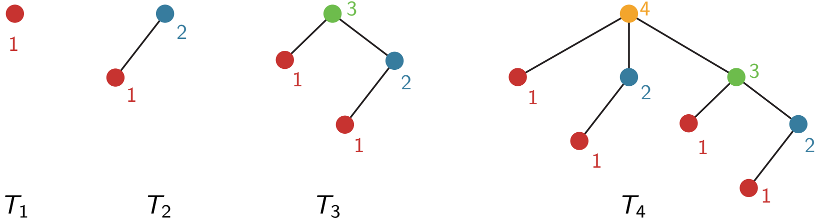
T_1 T_2 T_3 T_4

Bounds on the number of colors: trees

Theorem 5

Let T be a tree, and let μ be a function such that T is μ-colorable. Then T can be μ-colored using at most the first $\log_2(|V(T)|) + 1$ colors.

There is a family $\{T_n\}_{n\in\mathbb{N}}$ of trees and $\{\mu_n\}_{n\in\mathbb{N}}$ of functions such that T_n requires n colors to be μ_n-colored, and it has 2^{n-1} vertices.



Bounds on the number of colors: bipartite graphs

Theorem

Let B be a bipartite graph, and let μ be a function such that B is μ-colorable. Then B can be μ-colored using at most the first $\frac{(|V(B)|+2)}{2}$ colors.

There is a family $\{B_n\}_{n\in\mathbb{N}}$ of bipartite graphs and $\{\mu_n\}_{n\in\mathbb{N}}$ of functions such that B_n requires n colors to be μ_n-colored, and it has $2n-2$ vertices (if $n \geq 2$).

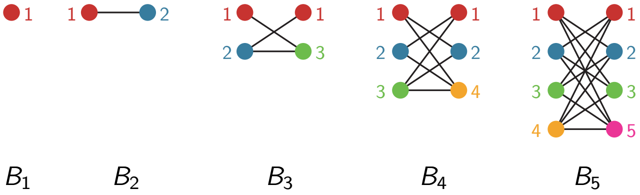
B_1 B_2 B_3 B_4 B_5

Bounds on the number of colors: bipartite graphs

Theorem

Let B be a bipartite graph, and let μ be a function such that B is μ-colorable. Then B can be μ-colored using at most the first $\frac{(|V(B)|+2)}{2}$ colors.

There is a family $\{B_n\}_{n\in\mathbb{N}}$ of bipartite graphs and $\{\mu_n\}_{n\in\mathbb{N}}$ of functions such that B_n requires n colors to be μ_n-colored, and it has $2n-2$ vertices (if $n \geq 2$).



μ-coloring on interval graphs

Theorem

The μ-coloring problem over interval graphs is NP-complete.

Its proof is based on the NP-completeness of the coloring problem on circular-arc graphs [Garey-Johnson-Miller-Papadimitriou, 1980].

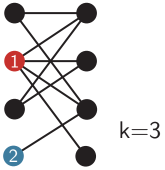
Qualche dubbio fino a qua ?

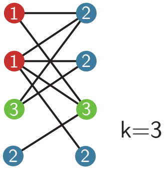
The precoloring extension problem

Another particular case of list-coloring is the following.

- The precoloring extension (PrExt) problem takes as input a graph $G = (V, E)$, a subset $W \subseteq V$, a coloring f' of W, and a natural number k, and consists in deciding whether G admits a k-coloring f such that $f(v) = f'(v)$ for every $v \in W$ or not [Biro-Hujter-Tuza, 1992].

In other words, a prespecified vertex subset is colored beforehand, and our task is to extend this partial coloring to a valid k-coloring of the whole graph.





The precoloring extension problem

The precoloring extension problem is also a particular case of list-coloring and a generalization of k-coloring. But there is not direct relation with μ-coloring (of course there are reductions among both problems because they are both NP-complete, but these reductions change the input graph).

It is NP-complete on unit interval graphs [Marx, 2004], bipartite graphs [Hujter-Tuza, 1993] and line graphs of complete bipartite graphs $K_{n,n}$ [Colbourn, 1984].

It is solvable in polynomial time on split graphs [Hujter-Tuza, 1996], complements of bipartite graphs [Hujter-Tuza, 1996] and cographs [Hujter-Tuza, Jansen-Scheffler, 1996].

Precoloring extension on split graphs

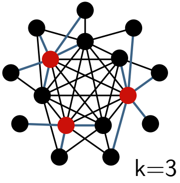
A split graph is a graph whose vertex set can be partitioned into a complete graph K and a stable set S. A split graph is said to be complete if its edge set includes all possible edges between K and S.

It is not difficult to solve the precoloring extension problem on split graphs, and it can be done in polynomial time. Instead, μ-coloring is NP-complete on this class.

μ-coloring is NP-complete for split graphs

Proof: It is used a reduction from the dominating set problem on split graphs, which is NP-complete (A. Bertossi, 1984).

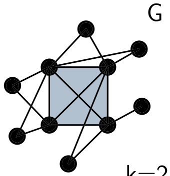
An instance of the dominating set problem on split graphs is given by a split graph G and an integer $k \geq 1$, and consists in deciding if there exists a subset D of $V(G)$, with $|D| \leq k$, and such that every vertex of $V(G)$ either belongs to D or has a neighbor in D. Such a set is called a dominating set.



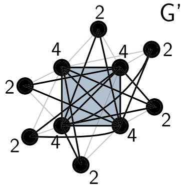
dominating set

Let G be a split graph and $k \geq 0$; $V(G) = K \cup I$, K is a complete and I is a stable set. We may assume G with no isolated vertices and $k \leq |K|$.

- We will construct a split graph G' and a function $\mu : V(G') \to \mathbb{N}$ such that G' is μ-colorable if and only if G admits a dominating set of cardinality at most k:
  - $V(G') = K \cup I$
  - K is a complete and I is a stable set in G'
  - for $v \in K$ and $w \in I$, $vw \in E(G')$ iff $vw \notin E(G)$
  - $\mu(v) = |K|$ for $v \in K$ and $\mu(w) = k$ for $w \in I$.

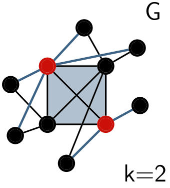


instance of split dominating set



instance of split μ-coloring

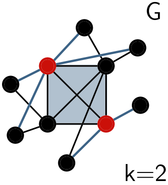
- Suppose first that G admits a dominating set D with $|D| \leq k$. Since G has no isolated vertices, G admits such a set $D \subseteq K$.



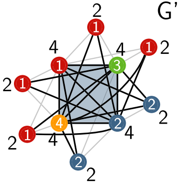
dominating set

- Let us define a μ-coloring of G' as follows:
  - color the vertices of D using different colors from 1 to $|D|$
  - color the remaining vertices of K using different colors from $|D| + 1$ to $|K|$
  - for each vertex w in I, choose w' in D such that $ww' \in E(G)$ and color w with the color used by w'.

- Suppose first that G admits a dominating set D with $|D| \leq k$. Since G has no isolated vertices, G admits such a set $D \subseteq K$.



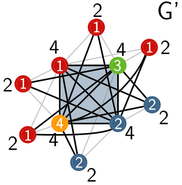
dominating set



μ-coloring

- Let us define a μ-coloring of G' as follows:
  - color the vertices of D using different colors from 1 to $|D|$
  - color the remaining vertices of K using different colors from $|D| + 1$ to $|K|$
  - for each vertex w in I, choose w' in D such that $ww' \in E(G)$ and color w with the color used by w'.

- Suppose now that G' is μ-colorable, and let $c : V(G') \to \mathbb{N}$ be a μ-coloring of G'. Since $\mu(v) = |K|$ for every $v \in K$ and K is complete in G', it follows that $c(K) = \{1, \ldots, |K|\}$.

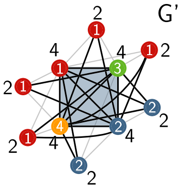


μ-coloring

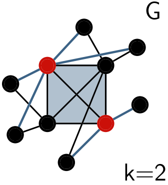
- Since $k \leq |K|$, for each vertex $w \in I$ there is a vertex $w' \in K$ such that $c(w) = c(w') \leq k$. Then $ww' \notin E(G')$, so $ww' \in E(G)$. Thus the set $\{v \in K : c(v) \leq k\}$ is a dominating set of G of size k.

$\square$

- Suppose now that G' is μ-colorable, and let $c : V(G') \to \mathbb{N}$ be a μ-coloring of G'. Since $\mu(v) = |K|$ for every $v \in K$ and K is complete in G', it follows that $c(K) = \{1, \ldots, |K|\}$.



μ-coloring



dominating set

- Since $k \leq |K|$, for each vertex $w \in I$ there is a vertex $w' \in K$ such that $c(w) = c(w') \leq k$. Then $ww' \notin E(G')$, so $ww' \in E(G)$. Thus the set $\{v \in K : c(v) \leq k\}$ is a dominating set of G of size k.

□

Split graphs

At this moment, this is the only class that we know where the computational complexity of μ-coloring and precoloring extension is different, unless $\mathrm{P} = \mathrm{NP}$.

Line graphs

Considering these coloring variations applied to edge coloring, we have the following results.

Theorem

The μ-coloring problem over line graphs of complete graphs and complete bipartite graphs is NP-complete.

Theorem

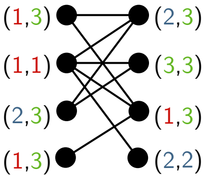
The precoloring extension problem over line graphs of complete graphs is NP-complete.

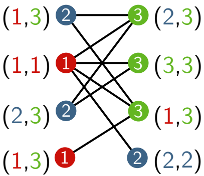
All these proofs are based on the NP-completeness of precoloring extension on line graphs of complete bipartite graphs.

Qualche dubbio fino a qua ?

The (γ, μ)-coloring problem

- Given a graph G and functions $\gamma, \mu : V \to \mathbb{N}$ such that $\gamma(v) \leq \mu(v)$ for every $v \in V$, we say that G is (γ, μ)-colorable if there exists a coloring f of G such that $\gamma(v) \leq f(v) \leq \mu(v)$ for every $v \in V$.
- Note that, given a list-coloring instance, then it can be viewed as a (γ, μ)-coloring instance by reordering the colors when the vertex-color incidence matrix given by the lists satisfies the consecutive 1's property. It can be decided in polynomial time.





Hierarchy of coloring problems

- The classical vertex coloring problem is clearly a special case of μ-coloring and precoloring extension, which in turn are special cases of (γ, μ)-coloring.
- Furthermore, (γ, μ)-coloring is a particular case of list-coloring.
- These observations imply that all the problems in this hierarchy are polynomially solvable in those graph classes where list-coloring is polynomial and, on the other hand, all the problems are NP-complete in those graph classes where vertex coloring is NP-complete.

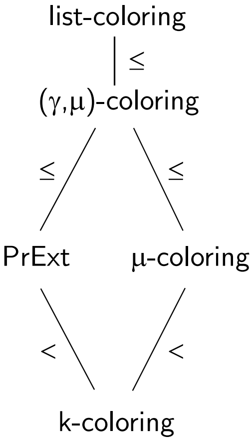


Table of complexities by now

Class	coloring	PrExt	μ-col.	(γ, μ)-col.	list-col.
Complete bipartite	P	P	P	?	NP-c
Bipartite	P	NP-c	NP-c	NP-c	NP-c
Cographs	P	P	P	?	NP-c
Distance-hereditary	P	?	?	?	NP-c
Interval	P	NP-c	NP-c	NP-c	NP-c
Unit interval	P	NP-c	?	NP-c	NP-c
Complete split	P	P	P	?	NP-c
Split	P	P	NP-c	NP-c	NP-c
Line of $K_{n,n}$	P	NP-c	NP-c	NP-c	NP-c
Line of K_n	P	NP-c	NP-c	NP-c	NP-c
Complement of bipartite	P	P	?	?	NP-c

"NP-c": NP-complete problem, "P": polynomial problem, "?": open problem.

Applications

This model is applicable to situations of resource assignment with incompatibilities, where each user has minimum quality requirements but bounded money to spend on it.

(γ, μ)-coloring is polynomial for complete bipartite graphs

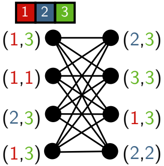
Proof: The following is a combinatorial algorithm that solves (γ, μ)-coloring in polynomial time for complete bipartite graphs.

Let $G = (V, E)$ be a complete bipartite graph, with bipartition $V_1 \cup V_2$, and let $\gamma, \mu : V \to \mathbb{N}$ such that $\gamma(v) \leq \mu(v)$ for every $v \in V$.

We have to consider two cases:

(i) There exists a vertex v such that $\gamma(v) = \mu(v)$.

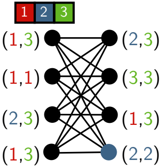
(ii) For every vertex v, $\gamma(v) < \mu(v)$.



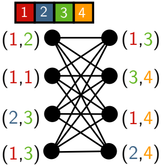
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



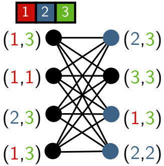
Example 2:



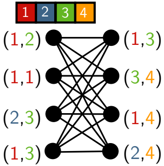
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



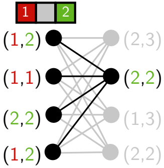
Example 2:



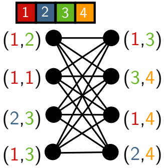
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



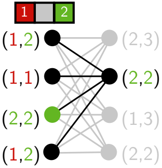
Example 2:



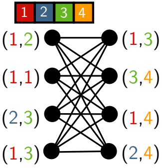
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



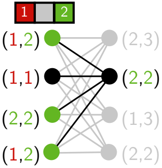
Example 2:



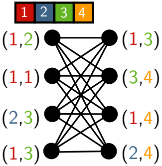
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



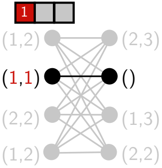
Example 2:



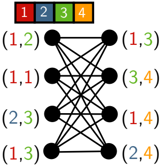
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



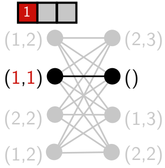
Example 2:



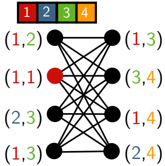
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



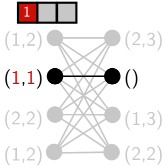
Example 2:



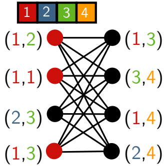
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:



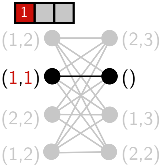
Example 2:



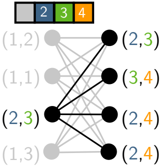
Case (i):

- If $\gamma(v) = \mu(v)$, the vertex v must be colored with color $\mu(v)$. Suppose $v \in V_2$. Since G is complete bipartite, no vertex of V_1 can use color $\mu(v)$.
- So, we can color with color $\mu(v)$ every vertex w of V_2 such that $\gamma(w) \leq \mu(v) \leq \mu(w)$ without affecting the feasibility of the problem.
- Then we remove those vertices and remove the color $\mu(v)$ from the universe of colors (we renumber the remaining colors so that they are still consecutive numbers).
- If some vertex of V_1 remains with no available color, the original graph was not (γ, μ)-colorable. Otherwise, we repeat this procedure until reaching either a coloring, or the non-colorability, or the case (ii).

Example 1:

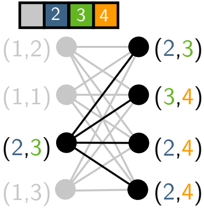


Example 2:



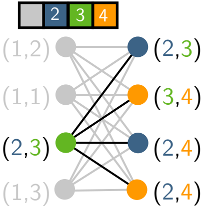
Case (ii):

- If for every vertex v, $\gamma(v) < \mu(v)$, then every vertex has among its possible colors at least an odd color and an even color.
- So the graph is (γ, μ)-colorable, we can color the vertices of V_1 with odd colors and the vertices of V_2 with even colors.



Case (ii):

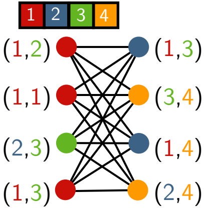
- If for every vertex v, $\gamma(v) < \mu(v)$, then every vertex has among its possible colors at least an odd color and an even color.
- So the graph is (γ, μ)-colorable, we can color the vertices of V_1 with odd colors and the vertices of V_2 with even colors.



□

Case (ii):

- If for every vertex v, $\gamma(v) < \mu(v)$, then every vertex has among its possible colors at least an odd color and an even color.
- So the graph is (γ, μ)-colorable, we can color the vertices of V_1 with odd colors and the vertices of V_2 with even colors.



□

Choosability

Case (ii) of the previous algorithm can be generalized: if for each vertex v of G we have that $\mu(v) - \gamma(v) + 1 \geq \chi(G)$, then the graph is (γ, μ)-colorable.

Choosability

Case (ii) of the previous algorithm can be generalized: if for each vertex v of G we have that $\mu(v) - \gamma(v) + 1 \geq \chi(G)$, then the graph is (γ, μ)-colorable.

Just take a coloring c of G with $\chi(G)$ colors, and then for each vertex v of G, assign to it a color $c'(v)$ such that $\gamma(v) \leq c'(v) \leq \mu(v)$ and $c'(v) \equiv c(v) \mod \chi(G)$. □

Choosability

Case (ii) of the previous algorithm can be generalized: if for each vertex v of G we have that $\mu(v) - \gamma(v) + 1 \geq \chi(G)$, then the graph is (γ, μ)-colorable.

Just take a coloring c of G with $\chi(G)$ colors, and then for each vertex v of G, assign to it a color $c'(v)$ such that $\gamma(v) \leq c'(v) \leq \mu(v)$ and $c'(v) \equiv c(v) \mod \chi(G)$. $\square$

For list-coloring, a graph G is said t-choosable if for any list assignment such that $|L(v)| \geq t$ for every v in G, G is list-colorable.

A famous conjecture says that if G is claw-free, then G is $\chi(G)$-choosable. It is open even for line graphs.

Choosability

Some advances on this conjecture are done:

Theorem (Gavlin, 1995)

If G is the line graph of a bipartite graph, then G is $\chi(G)$-choosable.

Theorem (Gravier and Maffray, 1998)

If $\alpha(G) \leq 2$, then G is $\chi(G)$-choosable.

Theorem (Gravier and Maffray, 2004)

If G is a claw-free perfect 3-colorable graph, then G is 3-choosable.

The last proof is based on the decomposition of claw-free perfect graphs by Chvátal and Sbihi.

Choosability

Inductively, we can prove this upper bound for the choice number ch(G) of a graph G.

$$\mathrm{ch}(G) \leq \max_{H \subseteq G} \delta(H) + 1$$

Choosability

Inductively, we can prove this upper bound for the choice number ch(G) of a graph G.

$$\mathrm{ch}(G) \leq \max_{H \subseteq G} \delta(H) + 1$$

Let $c = \max_{H \subseteq G} \delta(H) + 1$. Suppose that every vertex of G has a list of size c. Let $v \in G$ with $d(v) = \delta(G) < c$. Since $c \geq \max_{H \subseteq G - v} \delta(H) + 1$, by inductive hypothesis $G - v$ is c-choosable, so there exists a coloring for this list assignment. Since $d(v) < c$, v has a free color in its list, so the coloring can be extended to G. $\square$

Choosability

Using the previous upper bound, it is easy to prove that planar graphs are 6-choosable, since every planar graph H has a vertex of degree at most 5, that is, $\delta(H) \leq 5$. So, for a planar graph G,

$$\mathrm{ch}(G) \leq \max_{H \subseteq G} \delta(H) + 1 \leq 6$$

Choosability

Using the previous upper bound, it is easy to prove that planar graphs are 6-choosable, since every planar graph H has a vertex of degree at most 5, that is, $\delta(H) \leq 5$. So, for a planar graph G,

$$\mathrm{ch}(G) \leq \max_{H \subseteq G} \delta(H) + 1 \leq 6$$

Nevertheless, a better bound is known.

Theorem (Thomassen, 1994)

If G is a planar graph, then G is 5-choosable.

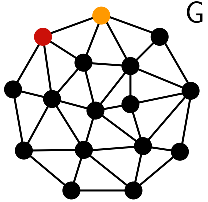
The 5-colorability of planar graphs is a corollary of this theorem, but its proof is very different, since it is not easy to “interchange” colors in a list-coloring context. Instead, the possibility of having vertices with different list sizes is exploited in order to strength the statement and prove it by induction.

Choosability

Thomassen proves by induction the following stronger result.

Theorem

Let G be a planar graph. Suppose that every inner face of G is triangular, and its outer face is bounded by a cycle $v_1 \ldots v_k v_1$. Suppose that $L(v_1) = \{1\}$, $L(v_2) = \{2\}$, $|L(v_i)| \geq 3$ for $i = 3, \ldots, k$ and $|L(w)| \geq 5$ for every other w in G. Then G is list-colorable with this list assignment.



Qualche dubbio fino a qua ?

(γ, μ)-coloring on complete split graphs

Theorem

The (γ, μ)-coloring problem in complete split graphs can be solved in polynomial time.

Let $G = (V, E)$ be a complete split graph with partition $V = K \cup I$, where K is a complete graph and I is a stable set. For $0 < j \leq i \leq \mu_{\max}$, let $L_{i,j} = |\{v \in K : j \leq \gamma(v) \text{ and } \mu(v) \leq i\}|$.

We reduce the problem of finding a (γ, μ)-coloring of G to a linear programming feasibility problem.

For $j = 1, \ldots, \mu_{\max}$, we define the integer variable x_j to be the number of colors from the set $\{1, \ldots, j\}$ assigned to vertices of K and, based on this definition, we consider the following linear program.

(γ, μ)-coloring on complete split graphs

$$
\begin{aligned}
x_0 &= 0 && && (1)\\
x_{j+1} - x_j &\geq 0 && \forall\, j \in \{0, \ldots, \mu_{\max} - 1\} && (2)\\
x_{j+1} - x_j &\leq 1 && \forall\, j \in \{0, \ldots, \mu_{\max} - 1\} && (3)\\
x_i - x_{j-1} &\geq L_{i,j} && \forall\, i, j : 0 < j \leq i \leq \mu_{\max} && (4)\\
x_{\mu(v)} - x_{\gamma(v)-1} &\leq \mu(v) - \gamma(v) && \forall\, v \in I && (5)
\end{aligned}
$$

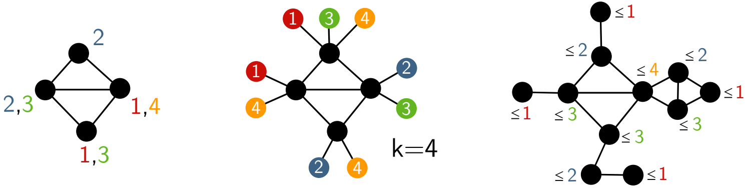
$L_{i,j} = |\{v \in K : j \leq \gamma(v) \text{ and } \mu(v) \leq i\}|$ and x_j is the number of colors from the set $\{1, \ldots, j\}$ assigned to vertices of K.

The constraint matrix is totally unimodular and some things can be assumed which make it of polynomial size.

It can be proved that this model solves the problem, using Hall's condition for maximum matchings on bipartite graphs.

General results

Since all the problems are NP-complete in the general case, there are also polynomial-time reductions from list-coloring to precoloring extension and μ-coloring. An example is shown in the figure, where we can see a list-coloring instance and its corresponding precoloring extension and μ-coloring instances.



These reductions involve changes in the graph, but are closed within some graph classes. This fact allows us to prove the following general results.

General results

Theorem

Let $\mathcal{F}$ be a family of graphs with minimum degree at least two. Then list-coloring, (γ, μ)-coloring and precoloring extension are polynomially equivalent in the class of $\mathcal{F}$-free graphs.

Theorem

Let $\mathcal{F}$ be a family of graphs satisfying the following property: for every graph G in $\mathcal{F}$, no connected component of G is complete, and for every cutpoint v of G, no connected component of $G \setminus v$ is complete. Then list-coloring, (γ, μ)-coloring, μ-coloring and precoloring extension are polynomially equivalent in the class of $\mathcal{F}$-free graphs.

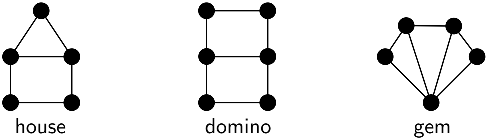
Note: for the proof of the second theorem, it is used the first theorem and then the second reduction but from (γ, μ)-coloring to μ-coloring, so an old vertex and its new neighbors form a complete.

Distance-hereditary graphs

Theorem

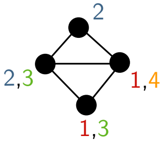
The (γ, μ)-coloring, precoloring extension and μ-coloring problems are NP-complete on distance-hereditary graphs.

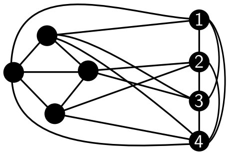
Distance-hereditary graphs are equivalent to $\{$house, domino, gem, $\{C_n\}_{n\geq 5}\}$-free. So, this result is a direct corollary of the previous general theorem and the NP-completeness of list-coloring on cographs.



Reduction from μ-coloring to k-coloring

There is a reduction also from list-coloring (and, in particular, μ-coloring) to k-coloring: let (G, L) be an instance of list-coloring, $\{1, \ldots, k\} = \bigcup_{v \in G} L(v)$. Build a graph G' by adding a complete graph with vertices $w_1, \ldots, w_k$ and making a vertex v of G adjacent to w_i iff $i \notin L(v)$. Then, G' will be k-colorable iff G admits an L-coloring.

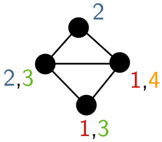


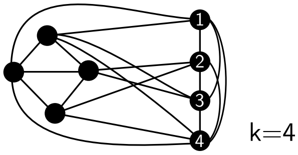


k=4

Reduction from μ-coloring to k-coloring

There is a reduction also from list-coloring (and, in particular, μ-coloring) to k-coloring: let (G, L) be an instance of list-coloring, $\{1, \ldots, k\} = \bigcup_{v \in G} L(v)$. Build a graph G' by adding a complete graph with vertices $w_1, \ldots, w_k$ and making a vertex v of G adjacent to w_i iff $i \notin L(v)$. Then, G' will be k-colorable iff G admits an L-coloring.

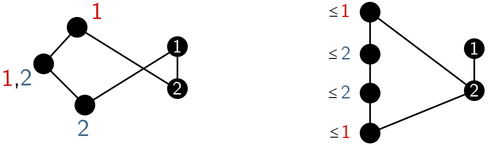




If the graph obtained by this reduction is perfect or belongs to some class where k-coloring is polynomial, then we can solve the list-coloring instance in polynomial time.

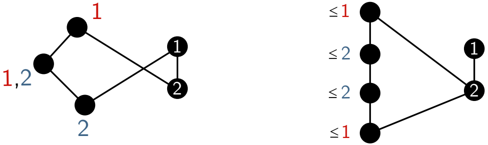
μ-perfect graphs

For example, these instances of list-coloring an μ-coloring lead to imperfect graphs.



μ-perfect graphs

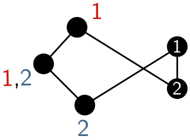
For example, these instances of list-coloring an μ-coloring lead to imperfect graphs.

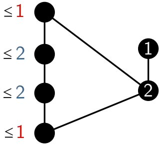


We can call a graph L-perfect or μ-perfect if the instance (G, L) or (G, μ), respectively, leads to a perfect graph.

μ-perfect graphs

For example, these instances of list-coloring an μ-coloring lead to imperfect graphs.





We can call a graph *L*-perfect or μ-perfect if the instance (G, L) or (G, μ), respectively, leads to a perfect graph.

Surprisingly, a graph is μ-perfect for every μ if and only if it is a cograph, or, equivalently, an M-perfect graph!

Also for list-coloring, a graph is L-perfect for every L if and only if it is the disjoint union of complete graphs.

Review: complexity table for coloring problems

Class	coloring	PrExt	μ-col.	(γ, μ)-col.	list-col.
Complete bipartite	P	P	P	P	NP-c
Bipartite	P	NP-c	NP-c	NP-c	NP-c
Cographs	P	P	P	?	NP-c
Distance-hereditary	P	NP-c	NP-c	NP-c	NP-c
Interval	P	NP-c	NP-c	NP-c	NP-c
Unit interval	P	NP-c	?	NP-c	NP-c
Complete split	P	P	P	P	NP-c
Split	P	P	NP-c	NP-c	NP-c
Line of $K_{n,n}$	P	NP-c	NP-c	NP-c	NP-c
Line of K_n	P	NP-c	NP-c	NP-c	NP-c
Complement of bipartite	P	P	?	?	NP-c
Block and cacti	P	P	P	P	P

"NP-c": NP-complete problem, "P": polynomial problem, "?": open problem.

Review: complexity table for coloring problems

Class	coloring	PrExt	μ-col.	(γ, μ)-col.	list-col.
Complete bipartite	P	P	P	P	NP-c
Bipartite	P	NP-c	NP-c	NP-c	NP-c
Cographs	P	P	P	?	NP-c
Distance-hereditary	P	NP-c	NP-c	NP-c	NP-c
Interval	P	NP-c	NP-c	NP-c	NP-c
Unit interval	P	NP-c	?	NP-c	NP-c
Complete split	P	P	P	P	NP-c
Split	P	P	NP-c	NP-c	NP-c
Line of $K_{n,n}$	P	NP-c	NP-c	NP-c	NP-c
Line of K_n	P	NP-c	NP-c	NP-c	NP-c
Complement of bipartite	P	P	?	?	NP-c
Block and cacti	P	P	P	P	P

"NP-c": NP-complete problem, "P": polynomial problem, "?": open problem.

As this table shows, unless P $=$ NP, μ-coloring and precoloring extension are strictly more difficult than vertex coloring, list-coloring is strictly more difficult than (γ, μ)-coloring and (γ, μ)-coloring is strictly more difficult than precoloring extension.

Review: complexity table for coloring problems

Class	coloring	PrExt	μ-col.	(γ, μ)-col.	list-col.
Complete bipartite	P	P	P	P	NP-c
Bipartite	P	NP-c	NP-c	NP-c	NP-c
Cographs	P	P	P	?	NP-c
Distance-hereditary	P	NP-c	NP-c	NP-c	NP-c
Interval	P	NP-c	NP-c	NP-c	NP-c
Unit interval	P	NP-c	?	NP-c	NP-c
Complete split	P	P	P	P	NP-c
Split	P	P	NP-c	NP-c	NP-c
Line of $K_{n,n}$	P	NP-c	NP-c	NP-c	NP-c
Line of K_n	P	NP-c	NP-c	NP-c	NP-c
Complement of bipartite	P	P	?	?	NP-c
Block and cacti	P	P	P	P	P

"NP-c": NP-complete problem, "P": polynomial problem, "?": open problem.

It remains as an open problem to know if there exists any class of graphs such that (γ, μ)-coloring is NP-complete and μ-coloring can be solved in polynomial time. Among the classes considered in this work, the candidate classes are cographs, unit interval and complement of bipartite.

Review: complexity table for coloring problems

Class	coloring	PrExt	μ-col.	(γ, μ)-col.	list-col.
Complete bipartite	P	P	P	P	NP-c
Bipartite	P	NP-c	NP-c	NP-c	NP-c
Cographs	P	P	P	?	NP-c
Distance-hereditary	P	NP-c	NP-c	NP-c	NP-c
Interval	P	NP-c	NP-c	NP-c	NP-c
Unit interval	P	NP-c	?	NP-c	NP-c
Complete split	P	P	P	P	NP-c
Split	P	P	NP-c	NP-c	NP-c
Line of $K_{n,n}$	P	NP-c	NP-c	NP-c	NP-c
Line of K_n	P	NP-c	NP-c	NP-c	NP-c
Complement of bipartite	P	P	?	?	NP-c
Block and cacti	P	P	P	P	P

"NP-c": NP-complete problem, "P": polynomial problem, "?": open problem.

For split graphs, precoloring extension can be solved in polynomial time, whereas μ-coloring is NP-complete. It remains as an open problem to find a class of graphs where the converse holds. Among the classes considered in this work, the candidate class is unit interval.

Review: hierarchy of coloring problems