Between coloring and list-coloring: μ-coloring

Flavia Bonomo   Mariano Cecowski

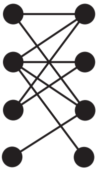
Departamento de Computación, Universidad de Buenos Aires, Argentina

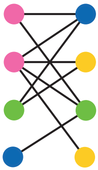
GRACO 2005

Graph coloring

Coloring a graph consists of giving a “color” (usually a number) to each vertex in such a way that adjacent vertices receive different colors.

Formally, a coloring of a graph $G = (V, E)$ is a function $f : V \to \mathbb{N}$ such that $f(v) \neq f(w)$ if v is adjacent to w.

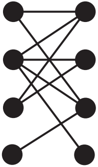




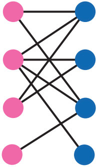
k-coloring

Given a graph $G = (V, E)$, a k-coloring of G is a coloring f for which $f(v) \leq k$ for every $v \in V$ (there are only k available colors).

A graph G is k-colorable if there is a k-coloring of G.



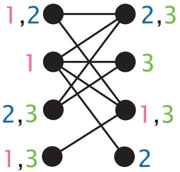
k=3

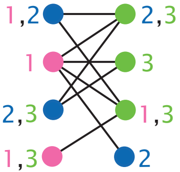


k=3

List-coloring

Given a graph $G = (V, E)$ and a finite list $L(v) \subseteq \mathbb{N}$ of colors for each vertex $v \in V$, G is list-colorable if there is a coloring f for which $f(v) \in L(v)$ for each $v \in V$ (Vizing, 1976).

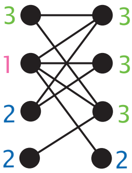


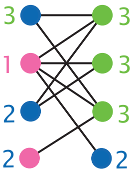


μ-coloring

Given a graph $G = (V, E)$ and a function $\mu : V \to \mathbb{N}$, a μ-coloring of G is a coloring f for which $f(v) \leq \mu(v)$ for each $v \in V$.

A graph G is μ-colorable if there is a μ-coloring of G.





k-coloring, list-coloring, μ-coloring

The μ-coloring problem lies between k-coloring and list-coloring.

- A trivial reduction from k-coloring to μ-coloring can be done defining $\mu(v) = k$ for every v.
- The reduction from μ-coloring to list-coloring can be done defining $L(v) = \{1, \ldots, \min\{\mu(v), |V(G)|\}\}$.

We show in this work that the betweenness is strict, that is, there is a class of graphs (bipartite graphs) for which μ-coloring is NP-complete while coloring is in P, and there is another class of graphs (cographs) for which list-coloring is NP-complete while μ-coloring is in P.

k-coloring, list-coloring, μ-coloring

The μ-coloring problem lies between k-coloring and list-coloring.

- A trivial reduction from k-coloring to μ-coloring can be done defining $\mu(v) = k$ for every v.
- The reduction from μ-coloring to list-coloring can be done defining $L(v) = \{1, \ldots, \min\{\mu(v), |V(G)|\}\}$.

We show in this work that the betweenness is strict, that is, there is a class of graphs (bipartite graphs) for which μ-coloring is NP-complete while coloring is in P, and there is another class of graphs (cographs) for which list-coloring is NP-complete while μ-coloring is in P.

k-coloring, list-coloring, μ-coloring

The μ-coloring problem lies between k-coloring and list-coloring.

- A trivial reduction from k-coloring to μ-coloring can be done defining $\mu(v) = k$ for every v.
- The reduction from μ-coloring to list-coloring can be done defining $L(v) = \{1, \ldots, \min\{\mu(v), |V(G)|\}\}$.

We show in this work that the betweenness is strict, that is, there is a class of graphs (bipartite graphs) for which μ-coloring is NP-complete while coloring is in P, and there is another class of graphs (cographs) for which list-coloring is NP-complete while μ-coloring is in P.

k-coloring, list-coloring, μ-coloring

The μ-coloring problem lies between k-coloring and list-coloring.

- A trivial reduction from k-coloring to μ-coloring can be done defining $\mu(v) = k$ for every v.
- The reduction from μ-coloring to list-coloring can be done defining $L(v) = \{1, \ldots, \min\{\mu(v), |V(G)|\}\}$.

We show in this work that the betweenness is strict, that is, there is a class of graphs (bipartite graphs) for which μ-coloring is NP-complete while coloring is in P, and there is another class of graphs (cographs) for which list-coloring is NP-complete while μ-coloring is in P.

Applications

A typical application of coloring and list-coloring is the assignment of resources to users with temporal restrictions (two users cannot use the same resource at the same time).

- A problem in which all the users can use all the resources can be modelled as a k-coloring problem, where k is the number of resources and the graph represents the compatibility between users.
- A problem in which each user can use some of the resources can be modelled as a list-coloring problem.
- A problem in which the resources have an order (best to worst) and each user can use any resource "good enough" for him can be modelled as a μ-coloring problem.

Applications

A typical application of coloring and list-coloring is the assignment of resources to users with temporal restrictions (two users cannot use the same resource at the same time).

- A problem in which all the users can use all the resources can be modelled as a k-coloring problem, where k is the number of resources and the graph represents the compatibility between users.
- A problem in which each user can use some of the resources can be modelled as a list-coloring problem.
- A problem in which the resources have an order (best to worst) and each user can use any resource "good enough" for him can be modelled as a μ-coloring problem.

Applications

A typical application of coloring and list-coloring is the assignment of resources to users with temporal restrictions (two users cannot use the same resource at the same time).

- A problem in which all the users can use all the resources can be modelled as a k-coloring problem, where k is the number of resources and the graph represents the compatibility between users.
- A problem in which each user can use some of the resources can be modelled as a list-coloring problem.
- A problem in which the resources have an order (best to worst) and each user can use any resource "good enough" for him can be modelled as a μ-coloring problem.

Perfect graphs

- The chromatic number of a graph G is the minimum k such that G is k-colorable, and is denoted by $\chi(G)$.
- A complete of G is a subset of vertices pairwise adjacent. A clique is a complete not properly contained in any other.
- It is easy to see that $\chi(G)$ is at least the cardinality of a maximum clique of G, denoted by $\omega(G)$.
- A graph G is perfect (Berge, 1960) when $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect graphs

- The chromatic number of a graph G is the minimum k such that G is k-colorable, and is denoted by $\chi(G)$.
- A complete of G is a subset of vertices pairwise adjacent. A clique is a complete not properly contained in any other.
- It is easy to see that $\chi(G)$ is at least the cardinality of a maximum clique of G, denoted by $\omega(G)$.
- A graph G is perfect (Berge, 1960) when $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect graphs

- The chromatic number of a graph G is the minimum k such that G is k-colorable, and is denoted by $\chi(G)$.
- A complete of G is a subset of vertices pairwise adjacent. A clique is a complete not properly contained in any other.
- It is easy to see that $\chi(G)$ is at least the cardinality of a maximum clique of G, denoted by $\omega(G)$.
- A graph G is perfect (Berge, 1960) when $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect graphs

- The chromatic number of a graph G is the minimum k such that G is k-colorable, and is denoted by $\chi(G)$.
- A complete of G is a subset of vertices pairwise adjacent. A clique is a complete not properly contained in any other.
- It is easy to see that $\chi(G)$ is at least the cardinality of a maximum clique of G, denoted by $\omega(G)$.
- A graph G is perfect (Berge, 1960) when $\chi(H) = \omega(H)$ for every induced subgraph H of G.

Perfect graphs

Perfect graphs have very nice properties:

- They are a self-complementary class of graphs (Lovász, 1972).
- The k-coloring problem is solvable in polynomial time for perfect graphs (Grötschel, Lovász and Schrijver, 1981).
- They have been characterized by minimal forbidden subgraphs (Chudnovsky, Robertson, Seymour and Thomas, 2002).
- They can be recognized in polynomial time (Chudnovsky, Cornuéjols, Liu, Seymour and Vušković, 2003).

M-perfect graphs

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".

Analogously, 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.

In order to characterize M-perfect graphs, we need some results on μ-colorings and cliques.

M-perfect graphs

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".

Analogously, 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.

In order to characterize M-perfect graphs, we need some results on μ-colorings and cliques.

M-perfect graphs

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".

Analogously, 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 are also perfect, because perfection is equivalent to M-perfection with μ restricted to constant functions.

In order to characterize M-perfect graphs, we need some results on μ-colorings and cliques.

M-perfect graphs

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".

Analogously, 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.

In order to characterize M-perfect graphs, we need some results on μ-colorings and cliques.

Minimal colorings on cographs

- We say that a coloring f is minimal when for every vertex v, and every color $i < f(v)$, v has a neighbor w_i with color $f(w_i) = i$. Every k-coloring or μ-coloring can be transformed into a minimal one.
- A cograph is a P_4-free graph (a graph with no induced P_4).

Theorem 1

Let G be a cograph and x a vertex of G. Let f be a minimal coloring of $G - x$. If f cannot be extended to G coloring x with a color at most T then there is a complete in the neighborhood of x of size T using all the colors between 1 and T.

Minimal colorings on cographs

- We say that a coloring f is minimal when for every vertex v, and every color $i < f(v)$, v has a neighbor w_i with color $f(w_i) = i$. Every k-coloring or μ-coloring can be transformed into a minimal one.
- A cograph is a P_4-free graph (a graph with no induced P_4).

Theorem 1

Let G be a cograph and x a vertex of G. Let f be a minimal coloring of $G - x$. If f cannot be extended to G coloring x with a color at most T then there is a complete in the neighborhood of x of size T using all the colors between 1 and T.

Minimal colorings on cographs

- We say that a coloring f is minimal when for every vertex v, and every color $i < f(v)$, v has a neighbor w_i with color $f(w_i) = i$. Every k-coloring or μ-coloring can be transformed into a minimal one.
- A cograph is a P_4-free graph (a graph with no induced P_4).

Theorem 1

Let G be a cograph and x a vertex of G. Let f be a minimal coloring of $G - x$. If f cannot be extended to G coloring x with a color at most T then there is a complete in the neighborhood of x of size T using all the colors between 1 and T.

Characterization

Theorem 2

If G is a graph, the following are equivalent:

1. G is a cograph
2. G is M-perfect
3. for every function $\mu : V \to \mathbb{N}$, G is μ-colorable if and only if every clique of G is μ-colorable.

It follows from this equivalence that M-perfect graphs are a self-complementary class of graphs (G is M-perfect iff $\overline{G}$ is) and can be recognized in linear time (Corneil, Perl and Stewart, 1984).

Characterization

Theorem 2

If G is a graph, the following are equivalent:

1. G is a cograph
2. G is M-perfect
3. for every function $\mu : V \to \mathbb{N}$, G is μ-colorable if and only if every clique of G is μ-colorable.

It follows from this equivalence that M-perfect graphs are a self-complementary class of graphs (G is M-perfect iff $\overline{G}$ is) and can be recognized in linear time (Corneil, Perl and Stewart, 1984).

Sketch of proof

Theorem 2

If G is a graph, the following are equivalent:

1. G is a cograph
2. G is M-perfect
3. for every function $\mu : V \to \mathbb{N}$, G is μ-colorable if and only if every clique of G is μ-colorable.

($3 \Leftrightarrow 2$) It is not difficult to prove.
($2 \Rightarrow 1$) Define μ as in the figure. Clearly, every clique is μ-colorable, but the whole graph is not.

($1 \Rightarrow 2$) This proof is strongly based on Theorem 1.

Complexity issues: cographs

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

Theorem 4

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.

A little improvement in the greedy algorithm allows us to find a non μ-colorable clique when the graph is not μ-colorable.

Jansen and Scheffler (1997) proved that list-coloring is NP-complete for cographs, hence μ-coloring is "easier" than list-coloring, unless P=NP.

Complexity issues: cographs

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

Theorem 4

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.

A little improvement in the greedy algorithm allows us to find a non μ-colorable clique when the graph is not μ-colorable.

Jansen and Scheffler (1997) proved that list-coloring is NP-complete for cographs, hence μ-coloring is "easier" than list-coloring, unless P=NP.

Complexity issues: cographs

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

Theorem 4

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.

A little improvement in the greedy algorithm allows us to find a non μ-colorable clique when the graph is not μ-colorable.

Jansen and Scheffler (1997) proved that list-coloring is NP-complete for cographs, hence μ-coloring is “easier” than list-coloring, unless P=NP.

Complexity issues: cographs

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

Theorem 4

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.

A little improvement in the greedy algorithm allows us to find a non μ-colorable clique when the graph is not μ-colorable.

Jansen and Scheffler (1997) proved that list-coloring is NP-complete for cographs, hence μ-coloring is "easier" than list-coloring, unless P=NP.

Complexity issues: cographs

A nice corollary of this, is the following known fact (Chvátal, 1984).

Corollary

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

Complexity issues: bipartite graphs

Hujter and Tuza (1993) proved that list-coloring is NP-complete for bipartite graphs. The same holds for μ-coloring.

Theorem 4

The μ-coloring problem is NP-complete for bipartite graphs.

Coloring is trivially in P for bipartite graphs, hence μ-coloring is "harder" than coloring, unless P=NP.

Complexity issues: bipartite graphs

Hujter and Tuza (1993) proved that list-coloring is NP-complete for bipartite graphs. The same holds for μ-coloring.

Theorem 4

The μ-coloring problem is NP-complete for bipartite graphs.

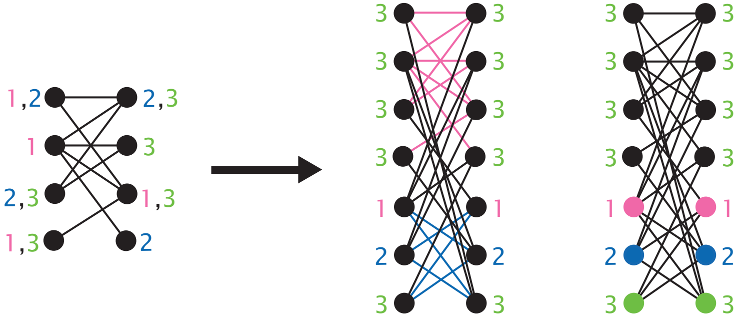
Coloring is trivially in P for bipartite graphs, hence μ-coloring is "harder" than coloring, unless P=NP.

Sketch of proof

Theorem 4

μ-coloring is NP-complete for bipartite graphs.

The reduction is from bipartite list-coloring.



Bounds on the number of colors: cographs

As a corollary of Theorem 3, 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 we can prove some upper bounds.

Bounds on the number of colors: cographs

As a corollary of Theorem 3, 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 we can prove some upper bounds.

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.

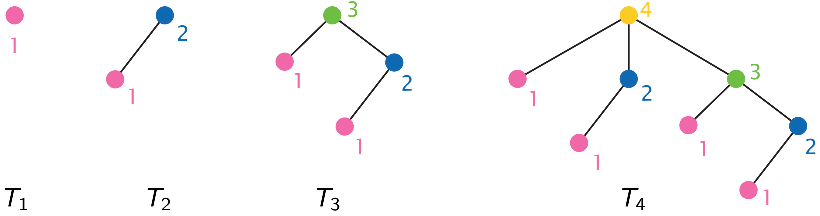
T_1 T_2 T_3 T_4

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 6

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 $n \geq 2$).

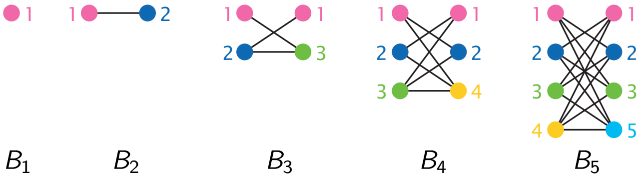
B_1 B_2 B_3 B_4 B_5

Bounds on the number of colors: bipartite graphs

Theorem 6

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 $n \geq 2$).



Summary

- A new problem of coloring is defined, μ-coloring, lying between k-coloring and list-coloring.
- It is proved that this betweenness is strict, unless P=NP.
- The concept of perfection is translated to μ-colorings, giving a new characterization of cographs in terms of M-perfection.
- It is shown that M-perfect graphs share some nice properties with perfect graphs: a graph is M-perfect if and only if its complement is M-perfect, they have polynomial time recognition, they have a simple characterization by forbidden subgraphs, and the μ-coloring problem can be solved polynomially on M-perfect graphs, as the k-coloring problem can be solved polynomially on perfect graphs.
- Bounds on the maximum number of colors needed to μ-coloring a graph are shown for trees, bipartite graphs and cographs.