Estudio Poliedral y Algoritmo Branch-and-Cut para el Problema de Coloreo Equitativo en Grafos

Daniel Esteban Severin

Tesis para optar al título de

Doctor de la Universidad de Buenos Aires

Directoras: Isabel Méndez-Díaz Graciela Nasini

Buenos Aires, Marzo de 2012

Resumen

- Problema de Coloreo Equitativo (PCEG):
  - Definición y aplicaciones
  - Diferencias con el Coloreo Clásico
  - Complejidad del problema
- Modelos de PLE para PCEG
- Estudio poliedral:
  - Adaptación de desigualdades válidas del Coloreo Clásico
  - Nuevas desigualdades válidas
- Algoritmo Branch-and-Cut $\longleftarrow$ `Ecopt`
  - Separación de desigualdades válidas
  - Heurísticas iniciales $\longleftarrow$ `TabuEqCol`
  - Heurísticas primales
  - Estrategia de branching
  - Rutina de enumeración
- Conclusiones y trabajos futuros

Definiciones básicas

Sea $G = (V, E)$ $\qquad V = \{1, \ldots, n\}$

Coloreo Clásico

k-coloreo $=$ partición de V en k conjuntos estables

$C_1,\ C_2,\ \ldots,\ C_k \quad \longleftarrow$ clases de color

Coloreo Equitativo

k-eqcol $=$ k-coloreo tal que:

- $||C_i| - |C_j|| \leq 1 \qquad \forall\ i, j = 1, \ldots, k$

o equivalentemente:

- $\lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \qquad \forall\ j = 1, \ldots, k$

Número cromático equitativo

$\chi_{eq}(G) = \text{mín}\{k : G \text{ admite un } k\text{-eqcol}\}$

- PCEG consiste en hallar $\chi_{eq}(G)$

Definiciones básicas

Sea $G = (V, E)$ $\qquad V = \{1, \ldots, n\}$

Coloreo Clásico

k-coloreo $=$ partición de V en k conjuntos estables

$C_1,\ C_2,\ \ldots,\ C_k \quad \longleftarrow \quad$ clases de color

Coloreo Equitativo

k-eqcol $=$ k-coloreo tal que:

- $||C_i| - |C_j|| \leq 1 \qquad \forall\ i, j = 1, \ldots, k$

o equivalentemente:

- $\lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \qquad \forall\ j = 1, \ldots, k$

Número cromático equitativo

$\chi_{eq}(G) = \min\{k : G \text{ admite un } k\text{-eqcol}\}$

- PCEG consiste en hallar $\chi_{eq}(G)$

Definiciones básicas

Sea $G = (V, E)$ $\qquad V = \{1, \ldots, n\}$

Coloreo Clásico

k-coloreo $=$ partición de V en k conjuntos estables

$C_1,\ C_2,\ \ldots,\ C_k \quad \longleftarrow$ clases de color

Coloreo Equitativo

k-eqcol $=$ k-coloreo tal que:

- $||C_i| - |C_j|| \leq 1 \qquad \forall\ i, j = 1, \ldots, k$

o equivalentemente:

- $\lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \qquad \forall\ j = 1, \ldots, k$

Número cromático equitativo

$\chi_{eq}(G) = \text{mín}\{k : G \text{ admite un } k\text{-eqcol}\}$

- PCEG consiste en hallar $\chi_{eq}(G)$

Coloreo clásico vs. Coloreo equitativo

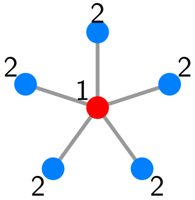
Coloreo clásico vs. Coloreo equitativo

$\chi(K_{1,5}) = 2$ $\qquad$ $\chi_{eq}(K_{1,5}) = 4$

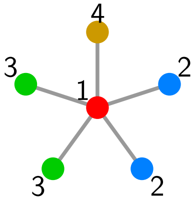
- En todo coloreo, el vértice central conforma una clase de color de tamaño 1
- En col. equitativo, las otras clases de color tienen tamaño a lo sumo 2
- En general:

$$\chi(K_{1,m}) = 2 \qquad \chi_{eq}(K_{1,m}) = \lceil m/2 \rceil + 1 \qquad \forall\ m$$

Coloreo clásico vs. Coloreo equitativo



$\chi(K_{1,5}) = 2$

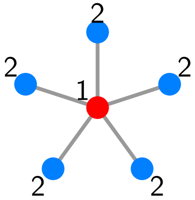


$\chi_{eq}(K_{1,5}) = 4$

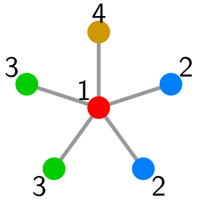
- En todo coloreo, el vértice central conforma una clase de color de tamaño 1
- En col. equitativo, las otras clases de color tienen tamaño a lo sumo 2
- En general:

$\chi(K_{1,m}) = 2 \qquad \chi_{eq}(K_{1,m}) = \lceil m/2 \rceil + 1 \qquad \forall\ m$

Coloreo clásico vs. Coloreo equitativo



$\chi(K_{1,5}) = 2$



$\chi_{eq}(K_{1,5}) = 4$

- En todo coloreo, el vértice central conforma una clase de color de tamaño 1
- En col. equitativo, las otras clases de color tienen tamaño a lo sumo 2
- En general:

$$\chi(K_{1,m}) = 2 \qquad \chi_{eq}(K_{1,m}) = \lceil m/2 \rceil + 1 \qquad \forall\ m$$

Coloreo clásico vs. Coloreo equitativo

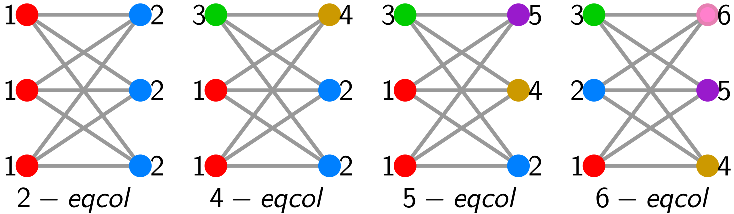
- Coloreo clásico $\longrightarrow$ G admite un k-coloreo $\forall\ k \geq \chi(G)$
- Coloreo equitativo $\longrightarrow$ G puede no admitir un k-eqcol tal que $k \geq \chi_{eq}(G)$

Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow$ G admite un k-coloreo $\forall\ k \geq \chi(G)$
- Coloreo equitativo $\longrightarrow$ G puede no admitir un k-eqcol tal que $k \geq \chi_{eq}(G)$

Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow$ G admite un k-coloreo $\forall\ k \geq \chi(G)$
- Coloreo equitativo $\longrightarrow$ G puede no admitir un k-eqcol tal que $k \geq \chi_{eq}(G)$



☞ $K_{3,3}$ no admite un 3-eqcol

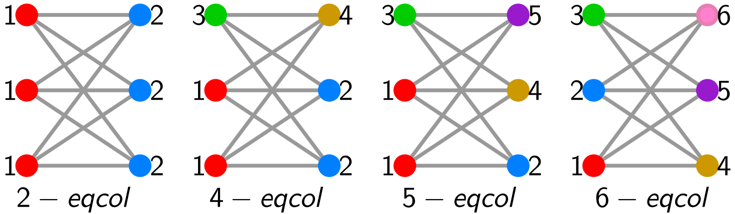
Conjunto de "saltos"

$$\mathscr{S}(G) = \{k \geq \chi_{eq}(G) : \nexists\ k\text{-eqcol en } G\}.$$

- *Ejemplo:* $\mathscr{S}(K_{3,3}) = \{3\}$

Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow$ G admite un k-coloreo $\forall\ k \geq \chi(G)$
- Coloreo equitativo $\longrightarrow$ G puede no admitir un k-eqcol tal que $k \geq \chi_{eq}(G)$



☞ $K_{3,3}$ no admite un 3-eqcol

Conjunto de "saltos"

$$\mathscr{S}(G) = \{k \geq \chi_{eq}(G) : \nexists\ k\text{-eqcol en } G\}.$$

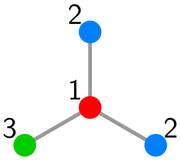
- *Ejemplo:* $\mathscr{S}(K_{3,3}) = \{3\}$

Coloreo clásico vs. Coloreo equitativo

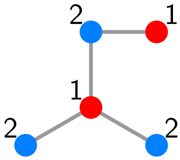
- Coloreo clásico $\longrightarrow \chi(G') \leq \chi(G) \;\; \forall \; G' \subset G$

Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow \chi(G') \leq \chi(G) \;\; \forall \; G' \subset G$
- Coloreo equitativo $\longrightarrow \chi_{eq}(G')$ puede ser mayor a $\chi_{eq}(G)$



$\chi_{eq}(G') = 3$



$\chi_{eq}(G) = 2$

Coloreo clásico vs. Coloreo equitativo

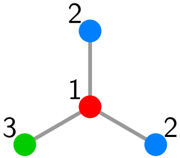
- Coloreo clásico $\longrightarrow$ Si G_1, G_2, ..., G_m son las componentes conexas de G entonces

$$\chi(G) = \text{máx}\{\chi(G_1), \chi(G_2), \ldots, \chi(G_m)\}$$

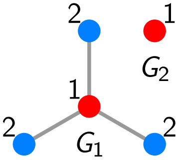
Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow$ Si G_1, G_2, ..., G_m son las componentes conexas de G entonces

$$\chi(G) = \text{máx}\{\chi(G_1), \chi(G_2), \ldots, \chi(G_m)\}$$



$\chi_{eq}(G_1) = 3$



$\chi_{eq}(G) = 2$

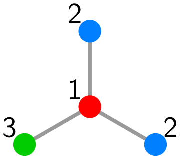
- Coloreo equitativo $\longrightarrow$ propiedad no se verifica

  ☞ no podemos restringirnos a grafos conexos

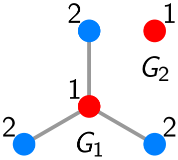
Coloreo clásico vs. Coloreo equitativo

- Coloreo clásico $\longrightarrow$ Si G_1, G_2, ..., G_m son las componentes conexas de G entonces

$$\chi(G) = \text{máx}\{\chi(G_1), \chi(G_2), \ldots, \chi(G_m)\}$$



$\chi_{eq}(G_1) = 3$



$\chi_{eq}(G) = 2$

- Coloreo equitativo $\longrightarrow$ propiedad no se verifica

  ☞ no podemos restringirnos a grafos conexos

Aplicaciones del PCEG

¿Existe algún interés práctico en resolver el PCEG?

- Asignación de aulas a asignaturas [Kubale & Furmańczyk 2005]
- Recolección de residuos [Tucker 1973]
- Esquemas de distribución paralela de datos en memoria [Das, Finocchi & Petreschi 2006]
- Derivación de cotas Chernoff-Hoeffding en suma de variables aleatorias [Pemmaraju 2001]

Aplicaciones del PCEG

¿Existe algún interés práctico en resolver el PCEG?

- Asignación de aulas a asignaturas [Kubale & Furmańczyk 2005]
- Recolección de residuos [Tucker 1973]
- Esquemas de distribución paralela de datos en memoria [Das, Finocchi & Petreschi 2006]
- Derivación de cotas Chernoff-Hoeffding en suma de variables aleatorias [Pemmaraju 2001]

Aplicaciones del PCEG

Recolección de residuos:

- Rutas de recolección previamente establecidas

$$\text{Rutas} \longrightarrow \text{Vértices}$$

- Dos rutas que cubren parcialmente un mismo tramo no pueden ser recorridas el mismo día

$$\text{Tramos en común} \longrightarrow \text{Aristas}$$

- Hay que asignar un día a cada ruta

$$\text{Días} \longrightarrow \text{Colores}$$

- Se deben recorrer cantidad similar de rutas cada día
  ☞ Restricción de Equidad

$$\chi_{eq}(G) = \text{Cant. mínima de días necesarios}$$

Aplicaciones del PCEG

Recolección de residuos:

- Rutas de recolección previamente establecidas

$$\text{Rutas} \longrightarrow \text{Vértices}$$

- Dos rutas que cubren parcialmente un mismo tramo no pueden ser recorridas el mismo día

$$\text{Tramos en común} \longrightarrow \text{Aristas}$$

- Hay que asignar un día a cada ruta

$$\text{Días} \longrightarrow \text{Colores}$$

- Se deben recorrer cantidad similar de rutas cada día
  ☞ Restricción de Equidad

$$\chi_{eq}(G) = \text{Cant. mínima de días necesarios}$$

Recolección de residuos:

- Rutas de recolección previamente establecidas

$$\text{Rutas} \longrightarrow \text{Vértices}$$

- Dos rutas que cubren parcialmente un mismo tramo no pueden ser recorridas el mismo día

$$\text{Tramos en común} \longrightarrow \text{Aristas}$$

- Hay que asignar un día a cada ruta

$$\text{Días} \longrightarrow \text{Colores}$$

- Se deben recorrer cantidad similar de rutas cada día

☞ Restricción de Equidad

$$\chi_{eq}(G) = \text{Cant. mínima de días necesarios}$$

Aplicaciones del PCEG

Recolección de residuos:

- Rutas de recolección previamente establecidas

$$\text{Rutas} \longrightarrow \text{Vértices}$$

- Dos rutas que cubren parcialmente un mismo tramo no pueden ser recorridas el mismo día

$$\text{Tramos en común} \longrightarrow \text{Aristas}$$

- Hay que asignar un día a cada ruta

$$\text{Días} \longrightarrow \text{Colores}$$

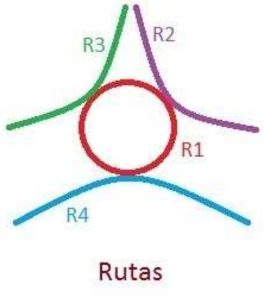
- Se deben recorrer cantidad similar de rutas cada día

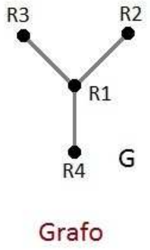
  ☞ Restricción de Equidad

$$\chi_{eq}(G) = \text{Cant. mínima de días necesarios}$$

Aplicaciones del PCEG

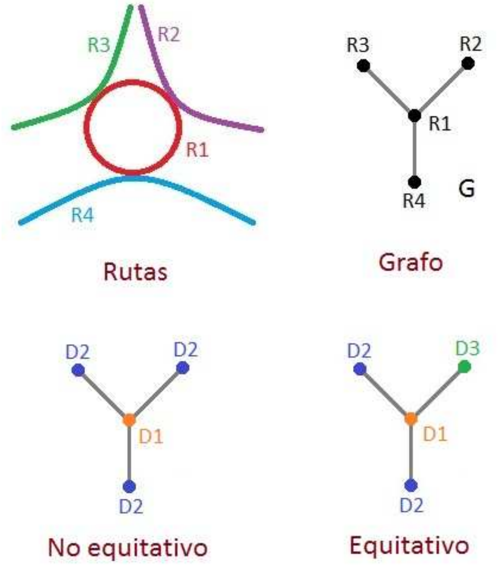
Recolección de residuos:





Aplicaciones del PCEG

Recolección de residuos:



Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales ⟶ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales ⟶ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
- *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
  ☞ buena herramienta para lidiar con problemas NP-difícil
  - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
  - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

Complejidad del PCEG

Aplicaciones reales $\longrightarrow$ Instancias de gran tamaño

El PCEG es NP-difícil [Furmańczyk & Kubale 2005]

- *Algoritmos heurísticos*: ☞ tiempo razonable
  ☞ no garantiza optimalidad
- *Algoritmos exactos*: ☞ solución óptima
  ☞ tiempo de búsqueda puede ser muy grande
  - *Alg. B&C*: Aprovechan estructura poliedral modelo PLE
    ☞ buena herramienta para lidiar con problemas NP-difícil
    - *Concorde*, algoritmo B&C para el TSP (instancia con 85900 ciudades) [Applegate, Bixby, Chvátal & Cook 2006]
    - Algoritmo B&C para el PCG [Méndez-Díaz & Zabala 2005]

Objetivo: Diseñar un algoritmo B&C competitivo para el PCEG

☞ Se requiere formular el problema con un modelo de PLE

[Méndez-Díaz & Zabala 2005]

$$x_{vj} = \begin{cases} 1 & \text{color } j \text{ asignado a } v, \\ 0 & \text{si no,} \end{cases} \qquad w_j = \begin{cases} 1 & x_{vj} = 1 \text{ para algún } v, \\ 0 & \text{si no.} \end{cases}$$

$$(x,w) \ \textit{coloreo} \quad \leftrightarrow \quad (x,w) \ \textit{binario tal que}:$$

$$\begin{aligned} &\sum_{j=1}^{n} x_{vj} = 1, && \forall\, v \in V \\ &x_{uj} + x_{vj} \leq w_j, && \forall\, uv \in E,\ j = 1,\dots,n \\ &w_{j+1} \leq w_j, && \forall\, j = 1,\dots,n \end{aligned}$$

- $\chi(G) = \min\{\sum_{j=1}^{n} w_j : (x,w) \ \textit{coloreo}\}$

[Méndez-Díaz & Zabala 2005]

$$x_{vj} = \begin{cases} 1 & \text{color } j \text{ asignado a } v, \\ 0 & \text{si no,} \end{cases} \qquad w_j = \begin{cases} 1 & x_{vj} = 1 \text{ para algún } v, \\ 0 & \text{si no.} \end{cases}$$

$$(x,w) \ \textit{coloreo} \quad \leftrightarrow \quad (x,w) \ \textit{binario tal que}:$$

$$\sum_{j=1}^{n} x_{vj} = 1, \qquad \forall\, v \in V$$

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\, uv \in E,\ j = 1, \dots, n$$

$$w_{j+1} \leq w_j, \qquad \forall\, j = 1, \dots, n$$

- $\chi(G) = \min\{\sum_{j=1}^{n} w_j : (x,w) \ \textit{coloreo}\}$

Formulación para el Problema de Coloreo Clásico

[Méndez-Díaz & Zabala 2005]

$$x_{vj} = \begin{cases} 1 & \text{color } j \text{ asignado a } v, \\ 0 & \text{si no,} \end{cases} \qquad w_j = \begin{cases} 1 & x_{vj} = 1 \text{ para algún } v, \\ 0 & \text{si no.} \end{cases}$$

$$(x, w) \ \textit{coloreo} \quad \leftrightarrow \quad (x, w) \ \textit{binario tal que}:$$

$$\begin{aligned} \sum_{j=1}^{n} x_{vj} &= 1, && \forall\ v \in V \\ x_{uj} + x_{vj} &\leq w_j, && \forall\ uv \in E,\ j = 1, \dots, n \\ w_{j+1} &\leq w_j, && \forall\ j = 1, \dots, n \end{aligned}$$

- $\chi(G) = \min\{\sum_{j=1}^{n} w_j : (x, w) \ \textit{coloreo}\}$

[Méndez-Díaz & Zabala 2005]

$$x_{vj} = \begin{cases} 1 & \text{color } j \text{ asignado a } v, \\ 0 & \text{si no,} \end{cases} \qquad w_j = \begin{cases} 1 & x_{vj} = 1 \text{ para algún } v, \\ 0 & \text{si no.} \end{cases}$$

$$(x, w) \ \textit{coloreo} \quad \leftrightarrow \quad (x, w) \ \textit{binario tal que}:$$

$$\begin{aligned} \sum_{j=1}^{n} x_{vj} &= 1, & & \forall\ v \in V \\ x_{uj} + x_{vj} &\leq w_j, & & \forall\ uv \in E,\ j = 1, \ldots, n \\ w_{j+1} &\leq w_j, & & \forall\ j = 1, \ldots, n \end{aligned}$$

- $\chi(G) = \min\{\sum_{j=1}^{n} w_j : (x, w) \ \textit{coloreo}\}$

Formulación para el Problema de Coloreo Clásico

- En un grafo conexo, todo v es adyacente a algún u, luego

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ j = 1, \ldots, n$$

asegura que v no puede pintarse con j cuando $w_j = 0$

- Si v es vértice aislado,

$$x_{vj} = 1 \qquad w_j = 0$$

es factible

- El modelo se puede extender a grafos no conexos con

$$x_{vj} \leq w_j, \qquad \forall\ v \ aislado,\ j = 1, \ldots, n$$

- En un grafo conexo, todo v es adyacente a algún u, luego

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ j = 1, \ldots, n$$

asegura que v no puede pintarse con j cuando $w_j = 0$

- Si v es vértice aislado,

$$x_{vj} = 1 \qquad w_j = 0$$

es factible

- El modelo se puede extender a grafos no conexos con

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1, \ldots, n$$

Formulación para el Problema de Coloreo Clásico

- En un grafo conexo, todo v es adyacente a algún u, luego

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ j = 1, \ldots, n$$

asegura que v no puede pintarse con j cuando $w_j = 0$

- Si v es vértice aislado,

$$x_{vj} = 1 \qquad w_j = 0$$

es factible

- El modelo se puede extender a grafos no conexos con

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1, \ldots, n$$

Modelando la equidad I

- Sea (x, w) k-coloreo
  - (x, w) k-eqcol $\iff \lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \quad \forall\, j = 1, \ldots, k$
- Sea (x, w) coloreo
  - $|C_j| = \sum_{v \in V} x_{vj}$
  - (x, w) k-coloreo $\iff \begin{cases} w_k - w_{k+1} = 1, \\ w_j - w_{j+1} = 0, \ \forall\, j \neq k \end{cases}$

Restricción de equidad:

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

Modelando la equidad I

- Sea (x, w) k-coloreo
  - (x, w) k-eqcol $\iff \lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \quad \forall\, j = 1, \ldots, k$
- Sea (x, w) coloreo
  - $|C_j| = \sum_{v \in V} x_{vj}$
  - (x, w) k-coloreo $\iff \begin{cases} w_k - w_{k+1} = 1, \\ w_j - w_{j+1} = 0, \ \forall\, j \neq k \end{cases}$

Restricción de equidad:

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

Modelando la equidad I

- Sea (x, w) k-coloreo
  - (x, w) k-eqcol $\iff \lfloor n/k \rfloor \leq |C_j| \leq \lceil n/k \rceil \;\; \forall\, j = 1, \dots, k$
- Sea (x, w) coloreo
  - $|C_j| = \sum_{v \in V} x_{vj}$
  - (x, w) k-coloreo $\iff \begin{cases} w_k - w_{k+1} = 1, \\ w_j - w_{j+1} = 0, \; \forall\, j \neq k \end{cases}$

Restricción de equidad:

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\, j = 1, \dots, n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \dots, n$$

Formulación *ECF* para el PCEG

(x, w) *eqcol* $\leftrightarrow$ (x, w) *binario tal que* :

$$\sum_{j=1}^{n} x_{vj} = 1, \qquad \forall\ v \in V$$

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ uv \in E,\ j = 1, \ldots, n$$

$$w_{j+1} \leq w_j, \qquad \forall\ j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\ j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\ j = 1, \ldots, n$$

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1, \ldots, n$$

- $\chi_{eq}(G) = \text{mín}\{\sum_{j=1}^{n} w_j : (x, w)\ eqcol\}$

Formulación *ECF* para el PCEG

$$(x,w)\ eqcol \quad \leftrightarrow \quad (x,w)\ binario\ tal\ que:$$

$$\sum_{j=1}^{n} x_{vj} = 1, \qquad \forall\ v \in V$$

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ uv \in E,\ j = 1,\ldots,n$$

$$w_{j+1} \leq w_j, \qquad \forall\ j = 1,\ldots,n$$

$$\sum_{v\in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\ j = 1,\ldots,n$$

$$\sum_{v\in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\ j = 1,\ldots,n$$

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1,\ldots,n$$

- $\chi_{eq}(G) = \text{mín}\{\sum_{j=1}^{n} w_j : (x,w)\ eqcol\}$

Formulación ECF_2 para el PCEG

(x, w) *eqcol* $\leftrightarrow$ (x, w) *es binario y satisface* :

$$\sum_{j=1}^{n} x_{vj} = 1, \qquad \forall\ v \in V$$

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ uv \in E,\ j = 1, \ldots, n$$

$$w_{j+1} \leq w_j, \qquad \forall\ j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\ j = 1, \ldots, n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\ j = 1, \ldots, n$$

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1, \ldots, n$$

$$\left.\begin{array}{r} x_{vj} = 0,\ \forall\ 1 \leq v < j \leq n \\ x_{vj} \leq \sum_{u=j-1}^{v-1} x_{uj-1},\ \forall\ 2 \leq j < v \leq n \end{array}\right\} \text{[Méndez-Díaz \& Zabala 2005]}$$

☞ elimina soluciones equivalentes

Formulación ECF_2 para el PCEG

$$(x,w)\ eqcol \quad \leftrightarrow \quad (x,w)\ es\ binario\ y\ satisface:$$

$$\sum_{j=1}^{n} x_{vj} = 1, \qquad \forall\ v \in V$$

$$x_{uj} + x_{vj} \leq w_j, \qquad \forall\ uv \in E,\ j = 1,\dots,n$$

$$w_{j+1} \leq w_j, \qquad \forall\ j = 1,\dots,n$$

$$\sum_{v \in V} x_{vj} \geq \sum_{k=j}^{n} \left\lfloor \frac{n}{k} \right\rfloor (w_k - w_{k+1}), \qquad \forall\ j = 1,\dots,n$$

$$\sum_{v \in V} x_{vj} \leq \sum_{k=j}^{n} \left\lceil \frac{n}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\ j = 1,\dots,n$$

$$x_{vj} \leq w_j, \qquad \forall\ v\ aislado,\ j = 1,\dots,n$$

$$\left.\begin{array}{r} x_{vj} = 0,\ \forall\ 1 \leq v < j \leq n \\ x_{vj} \leq \displaystyle\sum_{u=j-1}^{v-1} x_{uj-1},\ \forall\ 2 \leq j < v \leq n \end{array}\right\}$$ [Méndez-Díaz & Zabala 2005]

☞ elimina soluciones equivalentes

Modelando la equidad II

Lema

Sea (x, w) k-coloreo

$$\left.\begin{array}{r}(x,w)\ k\text{-eqcol}\\ |C_1| \geq |C_2| \geq \ldots \geq |C_k|\end{array}\right\} \Longleftrightarrow |C_j| = \left\lceil \frac{n-j+1}{k} \right\rceil \quad \forall\, j$$

- Es suficiente considerar k-eqcols tales que

$$|C_1| \geq |C_2| \geq \ldots \geq |C_k|$$

☞ elimina soluciones equivalentes

$$\sum_{v \in V} x_{vj} = \sum_{k=j}^{n} \left\lceil \frac{n-j+1}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

Lema

Sea (x, w) k-coloreo

$$\left.\begin{array}{r} (x,w)\ k\text{-eqcol} \\ |C_1| \geq |C_2| \geq \ldots \geq |C_k| \end{array}\right\} \Longleftrightarrow |C_j| = \left\lceil \frac{n-j+1}{k} \right\rceil \quad \forall\, j$$

- Es suficiente considerar k-eqcols tales que

$$|C_1| \geq |C_2| \geq \ldots \geq |C_k|$$

☞ elimina soluciones equivalentes

$$\sum_{v \in V} x_{vj} = \sum_{k=j}^{n} \left\lceil \frac{n-j+1}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

Modelando la equidad II

Lema

Sea (x, w) k-coloreo

$$\left.\begin{array}{r} (x,w)\ k\text{-eqcol} \\ |C_1| \geq |C_2| \geq \ldots \geq |C_k| \end{array}\right\} \Longleftrightarrow |C_j| = \left\lceil \frac{n-j+1}{k} \right\rceil \quad \forall\, j$$

- Es suficiente considerar k-eqcols tales que

$$|C_1| \geq |C_2| \geq \ldots \geq |C_k|$$

☞ elimina soluciones equivalentes

$$\sum_{v \in V} x_{vj} = \sum_{k=j}^{n} \left\lceil \frac{n-j+1}{k} \right\rceil (w_k - w_{k+1}), \qquad \forall\, j = 1, \ldots, n$$

Formulación ECF_1 para el PCEG

$$(x, w)\ eqcol \quad \leftrightarrow \quad (x, w)\ es\ binario\ y\ satisface:$$

$$\begin{aligned}
&\sum_{j=1}^{n} x_{vj} = 1, && \forall\ v \in V \\
&x_{uj} + x_{vj} \leq w_j, && \forall\ uv \in E,\ j = 1, \ldots, n \\
&w_{j+1} \leq w_j, && \forall\ j = 1, \ldots, n \\
&\sum_{v \in V} x_{vj} = \sum_{k=j}^{n} \left\lceil \frac{n-j+1}{k} \right\rceil (w_k - w_{k+1}), && \forall\ j = 1, \ldots, n \\
&x_{vj} \leq w_j, && \forall\ v\ aislado,\ j = 1, \ldots, n
\end{aligned}$$

- $\chi_{eq}(G) = \min\{\sum_{j=1}^{n} w_j : (x, w)\ eqcol\}$

Performance de las formulaciones

- Branch-and-Bound puro
- 50 instancias aleatorias de 40 vértices
- Tiempo máx: 2 horas

Densidad	Instancias resueltas			Prom. rel. lineal			Prom. tiempo		
	ECF	ECF_1	ECF_2	ECF	ECF_1	ECF_2	ECF	ECF_1	ECF_2
10 %	10	10	10	2	2	**2.85**	1.1	1.4	**0.5**
30 %	5	0	**10**	2	2	**3.85**	4112	–	**162**
50 %	0	0	**9**	2	2	**4.63**	–	–	**1919**
70 %	0	0	**3**	2	2	**8.34**	–	–	**3813**
90 %	0	1	**5**	2	2	**17.30**	–	6769	**821**

- ECF_2 presenta el mejor comportamiento

Performance de las formulaciones

- Branch-and-Bound puro
- 50 instancias aleatorias de 40 vértices
- Tiempo máx: 2 horas

Densidad	Instancias resueltas			Prom. rel. lineal			Prom. tiempo		
	ECF	ECF_1	ECF_2	*ECF*	ECF_1	ECF_2	*ECF*	ECF_1	ECF_2
10 %	10	10	10	2	2	**2.85**	1.1	1.4	**0.5**
30 %	5	0	**10**	2	2	**3.85**	4112	–	**162**
50 %	0	0	**9**	2	2	**4.63**	–	–	**1919**
70 %	0	0	**3**	2	2	**8.34**	–	–	**3813**
90 %	0	1	**5**	2	2	**17.30**	–	6769	**821**

- ECF_2 presenta el mejor comportamiento

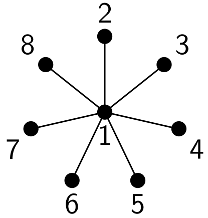
Dificultades de ECF_2

Dificultades de ECF_2

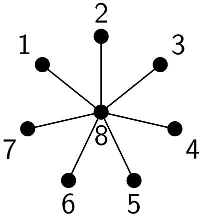
- *Poliedros*:

$$\mathcal{ECP}, \mathcal{ECP}_1, \mathcal{ECP}_2 = conv(\text{soluciones enteras } ECF, ECF_1, ECF_2)$$

- Propiedades de $\mathcal{ECP}_2$ depende del *etiquetado* de sus vértices



$dim(\mathcal{ECP}_2) = 23$



$dim(\mathcal{ECP}_2) = 28$

- Desigualdades válidas de $\mathcal{ECP}$ también son válidas para $\mathcal{ECP}_1$ y $\mathcal{ECP}_2$, pues

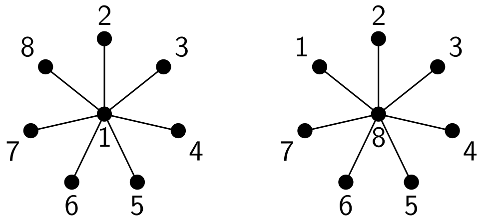
$$\mathcal{ECP} \supset \mathcal{ECP}_1 \qquad \mathcal{ECP} \supset \mathcal{ECP}_2$$

Dificultades de ECF_2

- *Poliedros*:

$$\mathcal{ECP}, \mathcal{ECP}_1, \mathcal{ECP}_2 = conv(\text{soluciones enteras } ECF, ECF_1, ECF_2)$$

- Propiedades de $\mathcal{ECP}_2$ depende del *etiquetado* de sus vértices



$dim(\mathcal{ECP}_2) = 23$ $\qquad$ $dim(\mathcal{ECP}_2) = 28$

- Desigualdades válidas de $\mathcal{ECP}$ también son válidas para $\mathcal{ECP}_1$ y $\mathcal{ECP}_2$, pues

$$\mathcal{ECP} \supset \mathcal{ECP}_1 \qquad \mathcal{ECP} \supset \mathcal{ECP}_2$$

El poliedro $\mathcal{ECP}$

Dimensión de $\mathcal{ECP}$

$$dim(\mathcal{ECP}) = n^2 - (\chi_{eq} + |\mathscr{S}| + 1)$$

Sistema minimal de ecuaciones:

$$\begin{aligned}
&\sum_{j=1}^{n} x_{vj} = 1, && \forall\ v \in V \\
&w_j = 1, && \forall\ j = 1, \dots, \chi_{eq} \\
&w_j = w_{j+1}, && \forall\ j \in \mathscr{S} \quad \longleftarrow \text{Conj. de saltos} \\
&\sum_{v \in V} x_{vn} = w_n.
\end{aligned}$$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

Desigualdades Clique [Méndez-Díaz & Zabala 2005]

Sea $j \leq n-1$, Q clique maximal de G tal que $|Q| \geq 2$.

$$\sum_{v \in Q} x_{vj} \leq w_j$$

define una **faceta** de $\mathcal{CP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

Desigualdades Clique [Méndez-Díaz & Zabala 2005]

Sea $j \leq n-1$, Q clique maximal de G tal que $|Q| \geq 2$.

$$\sum_{v \in Q} x_{vj} \leq w_j$$

define una **faceta** de $\mathcal{CP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

Desigualdades Clique [Méndez-Díaz & Zabala 2005]

Sea $j \leq n-1$, Q clique maximal de G tal que $|Q| \geq 2$.

$$\sum_{v \in Q} x_{vj} \leq w_j$$

define una **faceta** de $\mathcal{CP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

Desigualdades Clique

Sea $j \leq n-1$, Q clique maximal de G tal que $|Q| \geq 2$.

$$\sum_{v \in Q} x_{vj} \leq w_j$$

define una **faceta** de $\mathcal{ECP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

Desigualdades Block [Méndez-Díaz & Zabala 2005]

Sea $v \in V$, $j \leq n-1$.

$$\sum_{k=j}^{n} x_{vj} \leq w_j$$

es **válida** para $\mathcal{CP}$. Si G admite un $(j-1)$-eqcol, entonces define una **faceta** de $\mathcal{CP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$

- $\mathcal{CP} = conv$(modelo coloreo clásico)
- $\mathcal{CP} \supset \mathcal{ECP} \longrightarrow$ desig. válidas de $\mathcal{CP}$ también válidas en $\mathcal{ECP}$ ☞ analizamos su dimensión como caras de $\mathcal{ECP}$

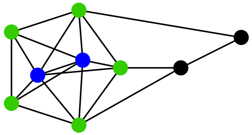
Desigualdades Block

Sea $v \in V$, $j \leq n-1$.

$$\sum_{k=j}^{n} x_{vj} \leq w_j$$

es **válida** para $\mathcal{ECP}$. Si G admite un $(j-1)$-eqcol, entonces define una **faceta** de $\mathcal{ECP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$



$j \leq n-1$
$S \subset V$ con $\alpha(S) = 2$
$Q = \{q : q \in S,\ S \subset N[q]\}$

Desigualdades (S, Q, j)-2-rango

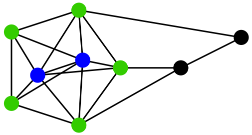
$$\sum_{v \in S \setminus Q} x_{vj} + 2 \sum_{v \in Q} x_{vj} \leq 2w_j$$

es **válida** para $\mathcal{ECP}$. Si

- $|Q| \geq 2$
- ninguna componente conexa de $\overline{G[S \setminus Q]}$ es bipartita
- $\forall\ v \in V \setminus S,\ Q \cup \{v\}$ no es una clique

define una **faceta** de $\mathcal{ECP}$.

Desigualdades válidas heredadas de $\mathcal{CP}$



$j \leq n - 1$
$S \subset V$ con $\alpha(S) = 2$
$Q = \{q : q \in S,\ S \subset N[q]\}$

Desigualdades (S, Q, j)-2-rango

$$\sum_{v \in S \setminus Q} x_{vj} + 2 \sum_{v \in Q} x_{vj} \leq 2w_j$$

es **válida** para $\mathcal{ECP}$. Si

- $|Q| \geq 2$
- ninguna componente conexa de $\overline{G[S \setminus Q]}$ es bipartita
- $\forall\ v \in V \setminus S$, $Q \cup \{v\}$ no es una clique

define una **faceta** de $\mathcal{ECP}$.

Nuevas desigualdades válidas de $\mathcal{ECP}$

Desigualdades J-color

- J = conjunto de colores
- b_{Jk} = máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

$$b_{Jk} = \text{máximo valor de} \sum_{j \in J} \sum_{v \in V} x_{vj} \text{ en un } k\text{-eqcol}$$

Desigualdades J-color

- J = conjunto de colores
- b_{Jk} = máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

$$b_{Jk} = \text{máximo valor de} \sum_{j \in J} \sum_{v \in V} x_{vj} \text{ en un } k\text{-eqcol}$$

Desigualdades J-color

- $J =$ conjunto de colores
- $b_{Jk} =$ máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

$$b_{Jk} = \text{máximo valor de } \sum_{j \in J} \sum_{v \in V} x_{vj} \text{ en un } k\text{-eqcol}$$

Desigualdades J-color

- J = conjunto de colores
- b_{Jk} = máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

Entonces
$$\sum_{j\in J}\sum_{v\in V} x_{vj} \leq \sum_{k=1}^{n} b_{Jk}(w_k - w_{k+1})$$
es **válida** para $\mathcal{ECP}$.

Si existe un matching M del complemento de G tal que $2 \leq |M| \leq \lfloor \frac{n-1}{2} \rfloor$, $|J| = 2|M| - r - 1$ con $r \in \{1,2\}$ y J contiene todos los colores mayores que $n - |M|$ entonces la desigualdad J-color define **faceta** de $\mathcal{ECP}$.

- $b_{Jk} = |J \cap \{1,\ldots,k\}|\lfloor \frac{n}{k} \rfloor + \min\{|J \cap \{1,\ldots,k\}|, n - k\lfloor \frac{n}{k} \rfloor\}$

☞ fácil de calcular

Desigualdades J-color

- J = conjunto de colores
- b_{Jk} = máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

Entonces
$$\sum_{j \in J} \sum_{v \in V} x_{vj} \leq \sum_{k=1}^{n} b_{Jk}(w_k - w_{k+1})$$
es **válida** para $\mathcal{ECP}$.

Si existe un matching M del complemento de G tal que $2 \leq |M| \leq \lfloor \frac{n-1}{2} \rfloor$, $|J| = 2|M| - r - 1$ con $r \in \{1, 2\}$ y J contiene todos los colores mayores que $n - |M|$ entonces la desigualdad J-color define **faceta** de $\mathcal{ECP}$.

- $b_{Jk} = |J \cap \{1, \ldots, k\}| \lfloor \frac{n}{k} \rfloor + \min\{|J \cap \{1, \ldots, k\}|, n - k\lfloor \frac{n}{k} \rfloor\}$

  ☞ fácil de calcular

Nuevas desigualdades válidas de $\mathcal{ECP}$

Desigualdades J-color

- $J =$ conjunto de colores
- $b_{Jk} =$ máxima cantidad de vértices que pueden colorearse con colores de J en un k-eqcol

Entonces
$$\sum_{j \in J} \sum_{v \in V} x_{vj} \leq \sum_{k=1}^{n} b_{Jk}(w_k - w_{k+1})$$
es **válida** para $\mathcal{ECP}$.
Si existe un matching M del complemento de G tal que $2 \leq |M| \leq \lfloor \frac{n-1}{2} \rfloor$, $|J| = 2|M| - r - 1$ con $r \in \{1, 2\}$ y J contiene todos los colores mayores que $n - |M|$ entonces la desigualdad J-color define **faceta** de $\mathcal{ECP}$.

- $b_{Jk} = |J \cap \{1, \ldots, k\}| \lfloor \frac{n}{k} \rfloor + \text{mín}\{|J \cap \{1, \ldots, k\}|, n - k \lfloor \frac{n}{k} \rfloor\}$
  ☞ fácil de calcular

Nuevas desigualdades válidas de $\mathcal{ECP}$

Desigualdades (S, J)-color

$S \subset V$, $J \subset \{1, \ldots, n-2\}$ tales que $|S| = |J| + 1$

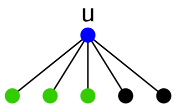
$$\sum_{j \in J} \sum_{v \in S} x_{vj} + \sum_{v \in V} x_{vn-1} \leq \sum_{k=1}^{n} b_{SJk}(w_k - w_{k+1})$$

donde $d_{Jk} = |J \cap \{1, \ldots, k\}|$ y

$$b_{SJk} = \begin{cases} \text{mín}\left\{|S|, |J|\alpha(S), d_{Jk}\lfloor \frac{n}{k} \rfloor + \text{mín}\{d_{Jk}, n - k\lfloor \frac{n}{k} \rfloor\}\right\}, & \text{si } k \leq n-2 \\ |S| + 1, & \text{si } k = n-1 \\ |S|, & \text{si } k = n \end{cases}$$

☞ similar a J-color: considera subconjunto S en vez de todo V

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

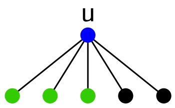
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

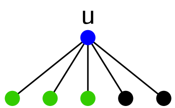
γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

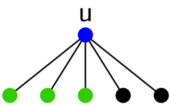
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

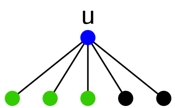
γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

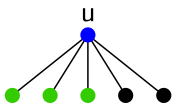
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

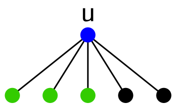
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- (x, w) k-eqcol
  - $k < j \longrightarrow w_j = 0 \longrightarrow$ ambos miembros en cero ✓
  - $k \geq j \longrightarrow w_j = 1$
    - $x_{uj} = 1 \longrightarrow$ ambos miembros valen γ_{jS} ✓
    - $x_{ut} = 1,\ t < j \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{jS}$ ✓
    - $x_{ut} = 1,\ t > j \longrightarrow k \geq t \longrightarrow \sum_{v \in S} x_{vj} \leq \gamma_{tS}$ ✓

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n-1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

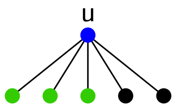
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- $\gamma_{kS} = \text{mín}\{\lceil n/\chi_{eq} \rceil, \lceil n/k \rceil, \alpha(S)\}$
- $\tilde{\gamma}_{kN(u)} = \text{mín}\{\lceil n/\underline{\chi_{eq}} \rceil, \lceil n/k \rceil, \overline{\alpha(N(u))}\}$ ←versión relajada

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq n - 1$

$u \in V$

$S \subset N(u)$ y no es una clique

γ_{kS} = máx. cant. de vértices de S que pueden pintarse con color j en un k-eqcol

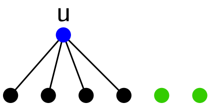
Desigualdades (u, j, S)-subvecindad

$$\gamma_{jS} x_{uj} + \sum_{v \in S} x_{vj} + \sum_{k=j+1}^{n} (\gamma_{jS} - \gamma_{kS}) x_{uk} \leq \gamma_{jS} w_j$$

es **válida** para $\mathcal{ECP}$.

- $\gamma_{kS} = \text{mín}\{\lceil n/\chi_{eq} \rceil, \lceil n/k \rceil, \alpha(S)\}$
- $\tilde{\gamma}_{kN(u)} = \text{mín}\{\lceil n/\chi_{eq} \rceil, \lceil n/k \rceil, \alpha(N(u))\}$ ←versión relajada

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq \lfloor n/2 \rfloor$

$u \in V$ tal que $N(u)$ no es una clique

Idea: (x, w) j-eqcol

$$|C_j| = \sum_{v \in V \setminus N[u]} x_{vj} + x_{uj} \geq \left\lfloor \frac{n}{j} \right\rfloor x_{uj}$$

o equivalentemente

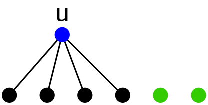
$$\left(\left\lfloor \frac{n}{j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} \leq 0$$

Desigualdades (u, j)-fuera-vecindad

$$\left(\left\lfloor \frac{n}{t_j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} + \sum_{k=t_j+1}^{n} b_{jk} x_{uk} \leq \sum_{k=t_j+1}^{n} b_{jk} (w_k - w_{k+1})$$

es **válida** para $\mathcal{ECP}$ ($t_j = \text{máx}\{j, \chi_{eq}\}$ y $b_{jk} = \lfloor n/t_j \rfloor - \lfloor n/k \rfloor$)

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq \lfloor n/2 \rfloor$
$u \in V$ tal que $N(u)$ no es una clique

Idea: (x, w) j-eqcol

$$|C_j| = \sum_{v \in V \setminus N[u]} x_{vj} + x_{uj} \geq \left\lfloor \frac{n}{j} \right\rfloor x_{uj}$$

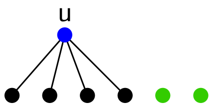
o equivalentemente

$$\left(\left\lfloor \frac{n}{j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} \leq 0$$

Desigualdades (u, j)-fuera-vecindad

$$\left(\left\lfloor \frac{n}{t_j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} + \sum_{k=t_j+1}^{n} b_{jk} x_{uk} \leq \sum_{k=t_j+1}^{n} b_{jk} (w_k - w_{k+1})$$

es **válida** para $\mathcal{ECP}$ ($t_j = \max\{j, \chi_{eq}\}$ y $b_{jk} = \lfloor n/t_j \rfloor - \lfloor n/k \rfloor$)



$j \leq \lfloor n/2 \rfloor$

$u \in V$ tal que $N(u)$ no es una clique

Idea: (x, w) j-eqcol

$$|C_j| = \sum_{v \in V \setminus N[u]} x_{vj} + x_{uj} \geq \left\lfloor \frac{n}{j} \right\rfloor x_{uj}$$

o equivalentemente

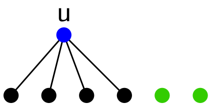
$$\left(\left\lfloor \frac{n}{j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} \leq 0$$

Desigualdades (u, j)-fuera-vecindad

$$\left(\left\lfloor \frac{n}{t_j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} + \sum_{k=t_j+1}^{n} b_{jk} x_{uk} \leq \sum_{k=t_j+1}^{n} b_{jk}(w_k - w_{k+1})$$

es **válida** para $\mathcal{ECP}$ ($t_j = \max\{j, \chi_{eq}\}$ y $b_{jk} = \lfloor n/t_j \rfloor - \lfloor n/k \rfloor$)

Nuevas desigualdades válidas de $\mathcal{ECP}$



$j \leq \lfloor n/2 \rfloor$
$u \in V$ tal que $N(u)$ no es una clique

Idea: (x, w) j-eqcol

$$|C_j| = \sum_{v \in V \setminus N[u]} x_{vj} + x_{uj} \geq \left\lfloor \frac{n}{j} \right\rfloor x_{uj}$$

o equivalentemente

$$\left(\left\lfloor \frac{n}{j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} \leq 0$$

Desigualdades (u, j)-fuera-vecindad

$$\left(\left\lfloor \frac{n}{t_j} \right\rfloor - 1 \right) x_{uj} - \sum_{v \in V \setminus N[u]} x_{vj} + \sum_{k=t_j+1}^{n} b_{jk} x_{uk} \leq \sum_{k=t_j+1}^{n} b_{jk} (w_k - w_{k+1})$$

es **válida** para $\mathcal{ECP}$ ($t_j = \max\{j, \chi_{eq}\}$ y $b_{jk} = \lfloor n/t_j \rfloor - \lfloor n/k \rfloor$)

Desigualdades (u, j, k, Q)-clique-vecindad

$u \in V$, clique Q de G tal que $Q \cap N[u] = \varnothing$, números j, k tales que $k = \left\lceil \dfrac{n}{\left\lceil \frac{n}{k-1} \right\rceil - 1} \right\rceil$, $3 \leq k \leq \alpha(N(u)) + 1$ y $1 \leq j \leq \left\lceil \dfrac{n}{k-1} \right\rceil - 1$

$$(k-1)x_{uj} + \sum_{l=\lceil \frac{n}{k-1} \rceil}^{n-2} \left(k - \left\lceil \frac{n}{l} \right\rceil\right) x_{ul} + (k-1)\left(x_{un-1} + x_{un}\right) + \sum_{v \in N(u) \cup Q} x_{vj}$$
$$+ \sum_{v \in V \setminus \{u\}} \left(x_{vn-1} + x_{vn}\right) \leq \sum_{l=j}^{n} b_{ul}(w_l - w_{l+1})$$

donde

$$b_{ul} = \begin{cases} \min\{\lceil n/l \rceil, \alpha(N(u)) + 1\}, & \text{si } j \leq l \leq \lceil n/k \rceil - 1 \\ k, & \text{si } \lceil n/k \rceil \leq l \leq n-2 \\ k+1, & \text{si } l \geq n-1 \end{cases}$$

Nuevas desigualdades válidas de $\mathcal{ECP}$

Familias de desigualdades válidas:

- J-color
- (S, J)-color
- (u, j, S)-subvecindad
- (u, j)-fuera-vecindad
- (u, j, k, Q)-clique-vecindad

- $\mathscr{F}$ = cara de la desigualdad
  - $\mathscr{F}$ es de dimensión alta
    - ☞ $dim(\mathscr{F}) \sim o(dim(\mathcal{ECP}))$
  - condiciones suficientes para que $\mathscr{F}$ sea faceta
    - ☞ ejemplos

Nuevas desigualdades válidas de $\mathcal{ECP}$

Familias de desigualdades válidas:

- J-color
- (S, J)-color
- (u, j, S)-subvecindad
- (u, j)-fuera-vecindad
- (u, j, k, Q)-clique-vecindad

- $\mathscr{F}$ = cara de la desigualdad
  - $\mathscr{F}$ es de dimensión alta
    ☞ $dim(\mathscr{F}) \sim o(dim(\mathcal{ECP}))$
  - condiciones suficientes para que $\mathscr{F}$ sea faceta
    ☞ ejemplos

Separación

Familia de desig. válidas $\longrightarrow$ Rutina de separación

"más complejas"	"cantidad polinomial"
Heurísticas golosas	Enumeración directa
Clique	Block
2-rango	Subvecindad
J-color	Fuera-vecindad
(S, J)-color	Clique-vecindad

- Separación de desig. Clique y Block es el de [Méndez-Díaz & Zabala 2005]
- Se probaron procedimientos para separar desigualdades (S, J)-color, pero éstas no cortan casi nunca

Separación

Familia de desig. válidas $\longrightarrow$ Rutina de separación

"más complejas"	"cantidad polinomial"
Heurísticas golosas	Enumeración directa
Clique	Block
2-rango	Subvecindad
J-color	Fuera-vecindad
(S, J)-color	Clique-vecindad

- Separación de desig. Clique y Block es el de [Méndez-Díaz & Zabala 2005]
- Se probaron procedimientos para separar desigualdades (S, J)-color, pero éstas no cortan casi nunca

Separación

Familia de desig. válidas $\longrightarrow$ Rutina de separación

"más complejas"	"cantidad polinomial"
Heurísticas golosas	Enumeración directa
Clique	Block
2-rango	Subvecindad
J-color	Fuera-vecindad
(S, J)-color	Clique-vecindad

- Separación de desig. Clique y Block es el de [Méndez-Díaz & Zabala 2005]
- Se probaron procedimientos para separar desigualdades (S, J)-color, pero éstas no cortan casi nunca

Separación

Familia de desig. válidas $\longrightarrow$ Rutina de separación

“más complejas”	“cantidad polinomial”
Heurísticas golosas	Enumeración directa
Clique	Block
2-rango	Subvecindad
J-color	Fuera-vecindad
(S, J)-color	Clique-vecindad

- Separación de desig. Clique y Block es el de [Méndez-Díaz & Zabala 2005]
- Se probaron procedimientos para separar desigualdades (S, J)-color, pero éstas no cortan casi nunca

Familia de desig. válidas $\longrightarrow$ Rutina de separación

"más complejas" "cantidad polinomial"

Heurísticas golosas	Enumeración directa
Clique	Block
2-rango	Subvecindad
J-color	Fuera-vecindad
~~(S, J)-color~~	Clique-vecindad

- Separación de desig. Clique y Block es el de [Méndez-Díaz & Zabala 2005]
- Se probaron procedimientos para separar desigualdades (S, J)-color, pero éstas no cortan casi nunca

- *Mejor combinación*:
  Estrategia $= \{\texttt{clique}, \texttt{block}, \texttt{2-rango}, \texttt{J}-\texttt{color}\}$

- Comparación:
  - Branch-and-Bound vs. Cut-and-Branch
  - 50 instancias aleatorias
  - 20 iteraciones del alg. de plano de corte
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. res.		Prom. gap rel.		Prom. nodos		Prom. tiempo	
		B&B	C&B	B&B	C&B	B&B	C&B	B&B	C&B
90	10 %	10	10	0 %	0 %	3399	**2211**	**30.9**	35.4
70	30 %	10	10	0 %	0 %	55080	**42927**	2405	**1776**
70	50 %	0	**2**	11.5 %	**9.8 %**	–	**51621**	–	**6192**
70	70 %	1	**6**	10.6 %	**3.6 %**	**19079**	19237	**1497**	3233
70	90 %	10	10	0 %	0 %	39382	**1267**	1113	**68**

- *Mejor combinación*:
  Estrategia $= \{$`clique`, `block`, `2-rango`,`J`$-$`color`$\}$
- Comparación:
  - Branch-and-Bound vs. Cut-and-Branch
  - 50 instancias aleatorias
  - 20 iteraciones del alg. de plano de corte
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. res.		Prom. gap rel.		Prom. nodos		Prom. tiempo	
		B&B	C&B	B&B	C&B	B&B	C&B	B&B	C&B
90	10 %	10	10	0 %	0 %	3399	**2211**	**30.9**	35.4
70	30 %	10	10	0 %	0 %	55080	**42927**	2405	**1776**
70	50 %	0	**2**	11.5 %	**9.8 %**	–	**51621**	–	**6192**
70	70 %	1	**6**	10.6 %	**3.6 %**	**19079**	19237	**1497**	3233
70	90 %	10	10	0 %	0 %	39382	**1267**	1113	**68**

Performance del alg. de plano de corte

- *Mejor combinación*:
  Estrategia $= \{\texttt{clique}, \texttt{block}, \texttt{2-rango}, \texttt{J}-\texttt{color}\}$
- Comparación:
  - Branch-and-Bound vs. Cut-and-Branch
  - 50 instancias aleatorias
  - 20 iteraciones del alg. de plano de corte
  - Tiempo máx: 2 horas

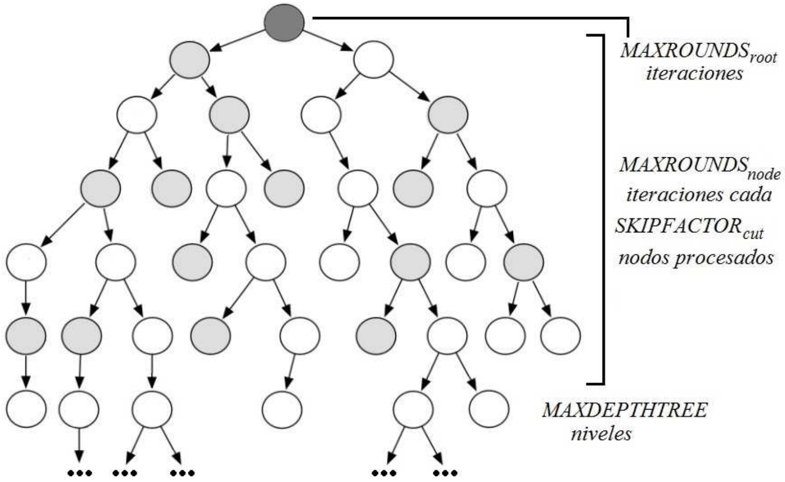
Vért.	Dens.	Inst. res.		Prom. gap rel.		Prom. nodos		Prom. tiempo	
		B&B	C&B	B&B	C&B	B&B	C&B	B&B	C&B
90	10 %	10	10	0 %	0 %	3399	**2211**	**30.9**	35.4
70	30 %	10	10	0 %	0 %	55080	**42927**	2405	**1776**
70	50 %	0	**2**	11.5 %	**9.8 %**	–	**51621**	–	**6192**
70	70 %	1	**6**	10.6 %	**3.6 %**	**19079**	19237	**1497**	3233
70	90 %	10	10	0 %	0 %	39382	**1267**	1113	**68**

Performance del alg. de plano de corte

- *Mejor combinación*:
  Estrategia $= \{\texttt{clique}, \texttt{block}, \texttt{2-rango}, \texttt{J}-\texttt{color}\}$
- Comparación:
  - Branch-and-Bound vs. Cut-and-Branch
  - 50 instancias aleatorias
  - 20 iteraciones del alg. de plano de corte
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. res.		Prom. gap rel.		Prom. nodos		Prom. tiempo	
		B&B	C&B	B&B	C&B	B&B	C&B	B&B	C&B
90	10 %	10	10	0 %	0 %	3399	**2211**	**30.9**	35.4
70	30 %	10	10	0 %	0 %	55080	**42927**	2405	**1776**
70	50 %	0	**2**	11.5 %	**9.8 %**	–	**51621**	–	**6192**
70	70 %	1	**6**	10.6 %	**3.6 %**	19079	**6928**	1497	**841**
70	90 %	10	10	0 %	0 %	39382	**1267**	1113	**68**

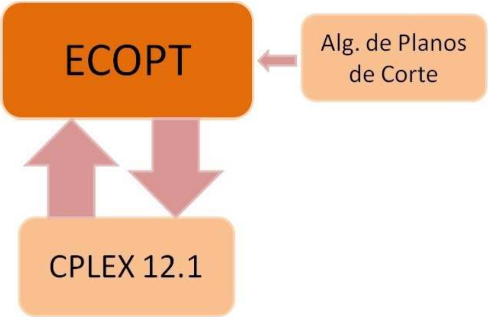
Algoritmo de plano de corte



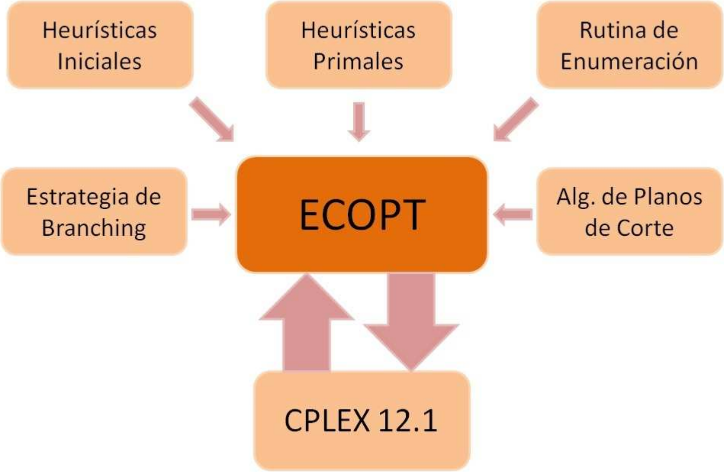
Mejores: $MAXROUNDS_{root} = 20$ $\quad MAXROUNDS_{node} = 1$

$SKIPFACTOR_{cut} = 2$ $\quad MAXDEPTHTREE = 50$

Algoritmo Branch-and-Cut



Algoritmo Branch-and-Cut



- TabuCol: [Hertz & de Werra 1987]
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-coloreo
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

1. v va de V_i a V_j $\leftarrow \begin{cases} V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

TabuEqCol

- `TabuCol`: [Hertz & de Werra 1987]
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-coloreo
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

    1. v va de V_i a V_j $\leftarrow$ $\begin{cases} V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

TabuEqCol

- `TabuCol`: [Hertz & de Werra 1987]
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-coloreo
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

    1. v va de V_i a V_j $\leftarrow \begin{cases} V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

TabuEqCol

- TabuCol: [Hertz & de Werra 1987]
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-coloreo
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

1. v va de V_i a V_j $\leftarrow \begin{cases} V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

TabuEqCol

- TabuEqCol:
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$ $/ \ ||V_i| - |V_j|| \leq 1$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-eqcol
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

1. v va de V_i a V_j $\leftarrow \begin{cases} |V_i| = |V_j| + 1, \\ V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

TabuEqCol

- TabuEqCol:
  - *Espacio de soluciones*: particiones $(V_1, V_2, \ldots, V_k)$ $/ \; ||V_i| - |V_j|| \leq 1$
  - *Objetivo*: que cada V_i sea estable
    - $\sum_{i=1}^{k} |E(V_i)| = 0 \iff (V_1, V_2, \ldots, V_k)$ es un k-eqcol
    - Minimizar $\sum_{i=1}^{k} |E(V_i)|$
  - *Búsqueda de próxima solución*:

    $i, \quad j, \quad v \in V_i$ (conflictivo)

    1. v va de V_i a V_j $\leftarrow \begin{cases} |V_i| = |V_j| + 1, \\ V_i' = V_i \backslash \{v\}, \\ V_j' = V_j \cup \{v\}. \end{cases}$

    o

    2. Intercambiar v con u $\leftarrow \begin{cases} u \in V_j, \\ V_i' = V_i \cup \{u\} \backslash \{v\}, \\ V_j' = V_j \cup \{v\} \backslash \{u\}. \end{cases}$

TabuEqCol: Resultados computacionales

- Comparación con Naive [Kubale & Furmańczyk 2005]
- Límite de tiempo de TabuEqCol: 30 segundos
- Límite de iteraciones de TabuEqCol: 50000

- 25 instancias aleatorias de 100 vértices
  - TabuEqCol logra mejores soluciones en 96 % de los casos respecto de Naive
  - En una instancia la reducción es de un 53 % (Naive: 32-eqcol, TabuEqCol: 15-eqcol)
  - Reducción más pronunciada en grafos de densidad media
    ☞ son las más difíciles de resolver y es justo donde se requieren cotas de mejor calidad ☺

- 62 instancias de prueba (Kneser + COLORLIB/DIMACS)
  - **Optimalidad**: Naive ⟶ 10 % de los casos
    TabuEqCol ⟶ 18 % de los casos
  - *Diferencia entre cota inf. y cota sup. de* **una unidad**: De las instancias restantes, Naive ⟶ 14 % de los casos
    TabuEqCol en el 41 % de los casos

TabuEqCol: Resultados computacionales

- Comparación con `Naive` [Kubale & Furmańczyk 2005]
- Límite de tiempo de `TabuEqCol`: 30 segundos
- Límite de iteraciones de `TabuEqCol`: 50000

- 25 instancias aleatorias de 100 vértices
  - `TabuEqCol` logra mejores soluciones en 96 % de los casos respecto de `Naive`
  - En una instancia la reducción es de un 53 % (`Naive`: 32-eqcol, `TabuEqCol`: 15-eqcol)
  - Reducción más pronunciada en grafos de densidad media
    ☞ son las más difíciles de resolver y es justo donde se requieren cotas de mejor calidad ☺

- 62 instancias de prueba (Kneser + COLORLIB/DIMACS)
  - **Optimalidad**: `Naive` ⟶ 10 % de los casos
    `TabuEqCol` ⟶ 18 % de los casos
  - *Diferencia entre cota inf. y cota sup. de* **una unidad**: De las instancias restantes, `Naive` ⟶ 14 % de los casos
    `TabuEqCol` en el 41 % de los casos

TabuEqCol: Resultados computacionales

- Comparación con `Naive` [Kubale & Furmańczyk 2005]
- Límite de tiempo de `TabuEqCol`: 30 segundos
- Límite de iteraciones de `TabuEqCol`: 50000

- 25 instancias aleatorias de 100 vértices
  - `TabuEqCol` logra mejores soluciones en 96 % de los casos respecto de `Naive`
  - En una instancia la reducción es de un 53 % (`Naive`: 32-eqcol, `TabuEqCol`: 15-eqcol)
  - Reducción más pronunciada en grafos de densidad media
    ☞ son las más difíciles de resolver y es justo donde se requieren cotas de mejor calidad ☺

- 62 instancias de prueba (Kneser + COLORLIB/DIMACS)
  - **Optimalidad**: `Naive` $\longrightarrow$ 10 % de los casos
    `TabuEqCol` $\longrightarrow$ 18 % de los casos
  - *Diferencia entre cota inf. y cota sup. de* **una unidad**: De las instancias restantes, `Naive` $\longrightarrow$ 14 % de los casos
    `TabuEqCol` en el 41 % de los casos

$$(x^*, w^*) \textit{ fraccionaria} \longrightarrow (x, w) \textit{ eqcol}$$

- *Esquemas*:

HeurPrim$_1$	HeurPrim$_2$
Se arma coloreo parcial con la siguiente regla: $x^*_{vj} = 1 \longrightarrow v$ se pinta de j	
(v, j) / x^*_{vj} máximo	j / $\lvert\{u : x^*_{uj} \notin \mathbb{Z}\}\rvert$ mínimo v / x^*_{vj} máximo
v se pinta de j	

- *Experiencia computacional*:
  - Pruebas computacionales evidencian que es conveniente habilitar ambas heurísticas en todos los nodos del árbol
  - Mejor que heurística por defecto de CPLEX en instancias de media a alta densidad

Heurísticas Primales

$$(x^*, w^*) \textit{ fraccionaria} \longrightarrow (x, w) \textit{ eqcol}$$

- *Esquemas*:

$\texttt{HeurPrim}_1$	$\texttt{HeurPrim}_2$
Se arma coloreo parcial con la siguiente regla: $x^*_{vj} = 1 \longrightarrow v$ se pinta de j	
(v, j) / x^*_{vj} máximo	j / $\lvert\{u : x^*_{uj} \notin \mathbb{Z}\}\rvert$ mínimo v / x^*_{vj} máximo
v se pinta de j	

- *Experiencia computacional*:
  - Pruebas computacionales evidencian que es conveniente habilitar ambas heurísticas en todos los nodos del árbol
  - Mejor que heurística por defecto de CPLEX en instancias de media a alta densidad

Heurísticas Primales

$$(x^*, w^*) \textit{ fraccionaria} \longrightarrow (x, w) \textit{ eqcol}$$

- *Esquemas*:

$\texttt{HeurPrim}_1$	$\texttt{HeurPrim}_2$
Se arma coloreo parcial con la siguiente regla: $x^*_{vj} = 1 \longrightarrow v$ se pinta de j	
(v, j) / x^*_{vj} máximo	j / $\lvert\{u : x^*_{uj} \notin \mathbb{Z}\}\rvert$ mínimo v / x^*_{vj} máximo
v se pinta de j	

- *Experiencia computacional*:
  - Pruebas computacionales evidencian que es conveniente habilitar ambas heurísticas en todos los nodos del árbol
  - Mejor que heurística por defecto de CPLEX en instancias de media a alta densidad

Estrategia de Branching

- Dada (x^*, w^*) fraccionaria, elegir variable $x^*_{vj} \notin \mathbb{Z}$
- $x^* \in \mathbb{Z} \Rightarrow w^* \in \mathbb{Z}$

Estrategia de Branching

- Dada (x^*, w^*) fraccionaria, elegir variable $x^*_{vj} \notin \mathbb{Z}$
- $x^* \in \mathbb{Z} \Rightarrow w^* \in \mathbb{Z}$

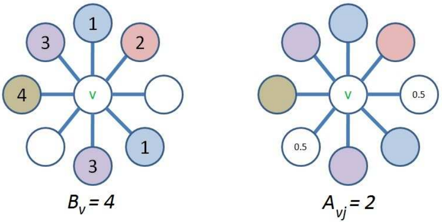
Estrategia de Branching

- Dada (x^*, w^*) fraccionaria, elegir variable $x^*_{vj} \notin \mathbb{Z}$
- $x^* \in \mathbb{Z} \Rightarrow w^* \in \mathbb{Z}$
- *Criterio*: elijo v según vecindad más colorida

$$B_v = |\{k \in \{1,\ldots,n\} \ : \ x^*_{uk} = 1, \ \forall \ u \in N(v)\}| \ \ \text{máximo}$$

elijo j según vecindad más fraccionaria

$$A_{vj} = |\{u \in N(v) \ : \ x^*_{uj} \notin \mathbb{Z}\}| \ \ \text{máximo}$$



- *Experiencia computacional*: De 6 a 8 veces más rápido que la estrategia de branching por defecto de CPLEX

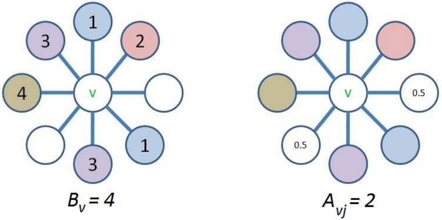
Estrategia de Branching

- Dada (x^*, w^*) fraccionaria, elegir variable $x^*_{vj} \notin \mathbb{Z}$
- $x^* \in \mathbb{Z} \Rightarrow w^* \in \mathbb{Z}$
- *Criterio*: elijo v según vecindad más colorida

$$B_v = |\{k \in \{1, \ldots, n\} \ : \ x^*_{uk} = 1, \ \forall \ u \in N(v)\}| \ \text{ máximo}$$

elijo j según vecindad más fraccionaria

$$A_{vj} = |\{u \in N(v) \ : \ x^*_{uj} \notin \mathbb{Z}\}| \ \text{ máximo}$$



- *Experiencia computacional*: De 6 a 8 veces más rápido que la estrategia de branching por defecto de CPLEX

Evaluación final de ECOPT

- Comparamos:
  - `TabuEqCol` + CPLEX 12.1 (estrategia por defecto)
  - `TabuEqCol` + ECOPT
  - B&C-LF_2 [Bahiense,Frota,Maculan,Noronha&Ribeiro 2009]
- Instancias aleatorias
  - 100 instancias (10 por línea)
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. resueltas			Prom. gap relativo			Prom. tiempo		
		CPX	ECO	LF_2	CPX	ECO	LF_2	CPX	ECO	LF_2
70	10 %	100 %	100 %	100 %	0 %	0 %	0 %	**3.7**	**3.7**	109
	30 %	60 %	**100 %**	0 %	4.8 %	**0 %**	18.3 %	1932	**16.5**	–
	50 %	0 %	**100 %**	0 %	8 %	**0 %**	8.5 %	–	**124**	–
	70 %	50 %	90 %	**100 %**	3.7 %	0.5 %	**0 %**	1447	329	**273**
	90 %	100 %	100 %	100 %	0 %	0 %	0 %	**4.7**	5.2	11
80	10 %	100 %	100 %	?	0 %	0 %	?	6.9	**6.5**	?
	30 %	10 %	**100 %**	?	12.1 %	**0 %**	?	4200	**204**	?
	50 %	0 %	0 %	?	11.6 %	**7.7 %**	?	–	–	?
	70 %	0 %	**40 %**	?	7 %	**3.5 %**	?	–	**3227**	?
	90 %	100 %	100 %	?	0 %	0 %	?	366	**41**	?

- Comparamos:
  - `TabuEqCol` + CPLEX 12.1 (estrategia por defecto)
  - `TabuEqCol` + ECOPT
  - B&C-LF$_2$ [Bahiense,Frota,Maculan,Noronha&Ribeiro 2009]
- Instancias aleatorias
  - 100 instancias (10 por línea)
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. resueltas			Prom. gap relativo			Prom. tiempo		
		CPX	ECO	LF$_2$	CPX	ECO	LF$_2$	CPX	ECO	LF$_2$
70	10 %	100 %	100 %	100 %	0 %	0 %	0 %	**3.7**	**3.7**	109
	30 %	60 %	**100 %**	0 %	4.8 %	**0 %**	18.3 %	1932	**16.5**	–
	50 %	0 %	**100 %**	0 %	8 %	**0 %**	8.5 %	–	**124**	–
	70 %	50 %	90 %	**100 %**	3.7 %	0.5 %	**0 %**	1447	329	**273**
	90 %	100 %	100 %	100 %	0 %	0 %	0 %	**4.7**	5.2	11
80	10 %	100 %	100 %	?	0 %	0 %	?	6.9	**6.5**	?
	30 %	10 %	**100 %**	?	12.1 %	**0 %**	?	4200	**204**	?
	50 %	0 %	0 %	?	11.6 %	**7.7 %**	?	–	–	?
	70 %	0 %	**40 %**	?	7 %	**3.5 %**	?	–	**3227**	?
	90 %	100 %	100 %	?	0 %	0 %	?	366	**41**	?

Evaluación final de ECOPT

- Comparamos:
  - `TabuEqCol` + CPLEX 12.1 (estrategia por defecto)
  - `TabuEqCol` + ECOPT
  - B&C-LF$_2$ [Bahiense,Frota,Maculan,Noronha&Ribeiro 2009]
- Instancias aleatorias
  - 100 instancias (10 por línea)
  - Tiempo máx: 2 horas

Vért.	Dens.	Inst. resueltas			Prom. gap relativo			Prom. tiempo		
		CPX	ECO	LF$_2$	CPX	ECO	LF$_2$	CPX	ECO	LF$_2$
	10 %	100 %	100 %	100 %	0 %	0 %	0 %	**3.7**	**3.7**	109
	30 %	60 %	**100 %**	0 %	4.8 %	**0 %**	18.3 %	1932	**16.5**	–
70	50 %	0 %	**100 %**	0 %	8 %	**0 %**	8.5 %	–	**124**	–
	70 %	50 %	90 %	**100 %**	3.7 %	0.5 %	**0 %**	1447	329	**273**
	90 %	100 %	100 %	100 %	0 %	0 %	0 %	**4.7**	5.2	11
	10 %	100 %	100 %	?	0 %	0 %	?	6.9	**6.5**	?
	30 %	10 %	**100 %**	?	12.1 %	**0 %**	?	4200	**204**	?
80	50 %	0 %	0 %	?	11.6 %	**7.7 %**	?	–	–	?
	70 %	0 %	**40 %**	?	7 %	**3.5 %**	?	–	**3227**	?
	90 %	100 %	100 %	?	0 %	0 %	?	366	**41**	?

Comparaciones finales

- Instancias de prueba (Kneser + COLORLIB/DIMACS)
  - 51 instancias
  - Tiempo máx: 4 horas
- Instancias resueltas por B&C-LF$_2$ no fueron obstáculo para ECOPT. En muchas de ellas, B&C-LF$_2$ tardó entre 1 y 2 órdenes de magnitud más que ECOPT
  Ejemplo (`5_FullIns_3`): LF$_2$ $\rightarrow$ 268 seg, ECOPT $\rightarrow$ 3 seg
- ECOPT vs. CPLEX
  - En (ii) se evalúa *tiempo*
    ☞ hay empate si diferencia es de 1 segundo
  - En (i) se evalúa *gap relativo*

	CPLEX gana	ECOPT gana	Hay empate	TOTAL
(i) Inst. que ninguno resuelve	2	5	3	10
(ii) Inst. que ambos resuelven	6	9	22	37
(iii) Inst. que alguno resuelve	0	4	0	4
Total de instancias evaluadas	8	15	25	51

Comparaciones finales

- Instancias de prueba (Kneser + COLORLIB/DIMACS)
  - 51 instancias
  - Tiempo máx: 4 horas
- Instancias resueltas por B&C-LF$_2$ no fueron obstáculo para ECOPT. En muchas de ellas, B&C-LF$_2$ tardó entre 1 y 2 órdenes de magnitud más que ECOPT
  Ejemplo (`5_FullIns_3`): LF$_2$ $\rightarrow$ 268 seg, ECOPT $\rightarrow$ 3 seg
- ECOPT vs. CPLEX
  - En (ii) se evalúa *tiempo*
    ☞ hay empate si diferencia es de 1 segundo
  - En (i) se evalúa *gap relativo*

	CPLEX gana	ECOPT gana	Hay empate	TOTAL
(i) Inst. que ninguno resuelve	2	5	3	10
(ii) Inst. que ambos resuelven	6	9	22	37
(iii) Inst. que alguno resuelve	0	4	0	4
Total de instancias evaluadas	8	15	25	51

- ECOPT vs. CPLEX vs. B&C-LF$_2$
  - ECOPT logra resolver instancias de mayor dificultad:
    - instancias aleatorias $\begin{cases} \text{mayor cantidad de vértices} \\ \text{densidad media} \end{cases}$
    - instancias de prueba que ya eran un desafío para PCG
  - ECOPT reduce el gap en la mayoría de los casos
- Heurística inicial `TabuEqCol`
  - Además de significar una componente importante por la calidad de las cotas que brinda a ECOPT, es independientemente una herramienta eficiente para generar coloreos equitativos cuando el objetivo no es la optimalidad

Conclusiones

- ECOPT vs. CPLEX vs. B&C-LF$_2$
  - ECOPT logra resolver instancias de mayor dificultad:
    - instancias aleatorias $\begin{cases} \text{mayor cantidad de vértices} \\ \text{densidad media} \end{cases}$
    - instancias de prueba que ya eran un desafío para PCG
  - ECOPT reduce el gap en la mayoría de los casos

- Heurística inicial `TabuEqCol`
  - Además de significar una componente importante por la calidad de las cotas que brinda a ECOPT, es independientemente una herramienta eficiente para generar coloreos equitativos cuando el objetivo no es la optimalidad

Conclusiones

- ECOPT vs. CPLEX vs. B&C-LF$_2$
  - ECOPT logra resolver instancias de mayor dificultad:
    - instancias aleatorias $\begin{cases} \text{mayor cantidad de vértices} \\ \text{densidad media} \end{cases}$
    - instancias de prueba que ya eran un desafío para PCG
  - ECOPT reduce el gap en la mayoría de los casos

- Heurística inicial `TabuEqCol`
  - Además de significar una componente importante por la calidad de las cotas que brinda a ECOPT, es independientemente una herramienta eficiente para generar coloreos equitativos cuando el objetivo no es la optimalidad

Trabajos futuros

- Mejorar ECOPT:
  - Nuevas familias de desigualdades válidas que puedan ser utilizadas como cortes
  - Implementar ECOPT en una arquitectura multiprocesador
- Resultados teóricos:
  - Describir por desigualdades lineales a $\mathcal{ECP}$ en familias particulares de grafos
  - Determinar familias de grafos para las cuales el PCEG es polinomial
- Problema de Coloreo Acotado:

  ☞ tamaño de clases acotados por constante h

  - Adaptar "estudio poliedral" para resolver este problema

    ☞ reemplazar restricciones de equidad por:

$$\sum_{v \in V} x_{vj} \leq h w_j, \qquad \forall\ 1 \leq j \leq n$$

Trabajos futuros

- Mejorar ECOPT:
  - Nuevas familias de desigualdades válidas que puedan ser utilizadas como cortes
  - Implementar ECOPT en una arquitectura multiprocesador
- Resultados teóricos:
  - Describir por desigualdades lineales a $\mathcal{ECP}$ en familias particulares de grafos
  - Determinar familias de grafos para las cuales el PCEG es polinomial
- Problema de Coloreo Acotado:

  ☞ tamaño de clases acotados por constante h

  - Adaptar "estudio poliedral" para resolver este problema

    ☞ reemplazar restricciones de equidad por:

$$\sum_{v \in V} x_{vj} \le h w_j, \qquad \forall\, 1 \le j \le n$$

Trabajos futuros

- Mejorar ECOPT:
  - Nuevas familias de desigualdades válidas que puedan ser utilizadas como cortes
  - Implementar ECOPT en una arquitectura multiprocesador
- Resultados teóricos:
  - Describir por desigualdades lineales a $\mathcal{ECP}$ en familias particulares de grafos
  - Determinar familias de grafos para las cuales el PCEG es polinomial
- Problema de Coloreo Acotado:

  ☞ tamaño de clases acotados por constante h

  - Adaptar "estudio poliedral" para resolver este problema

    ☞ reemplazar restricciones de equidad por:

$$\sum_{v \in V} x_{vj} \leq h w_j, \qquad \forall\, 1 \leq j \leq n$$

Trabajos futuros

- Mejorar ECOPT:
  - Nuevas familias de desigualdades válidas que puedan ser utilizadas como cortes
  - Implementar ECOPT en una arquitectura multiprocesador
- Resultados teóricos:
  - Describir por desigualdades lineales a $\mathcal{ECP}$ en familias particulares de grafos
  - Determinar familias de grafos para las cuales el PCEG es polinomial
- Problema de Coloreo Acotado:

  ☞ tamaño de clases acotados por constante h

  - Adaptar "estudio poliedral" para resolver este problema

    ☞ reemplazar restricciones de equidad por:

$$\sum_{v \in V} x_{vj} \leq h w_j, \qquad \forall\, 1 \leq j \leq n$$

Gracias!

- a mis directoras
- a mi papá, mi hermana y mi novia
- a los del proyecto de Optimización Combinatoria de Rosario
- al Depto. de Computación de la UBA