Universidade Estadual de Campinas
Instituto de Computação

RECOLORAÇÃO CONVEXA DE GRAFOS

Defesa de Mestrado

27 de Setembro de 2019

Ana Paula dos Santos Dantas
Orientador: Prof. Dr. Zanoni Dias
Coorientador: Prof. Dr. Cid Carvalho de Souza

Instituto de Computação - IC
Unicamp

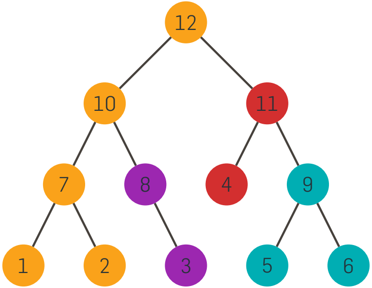
ROTEIRO

INTRODUÇÃO

COLORAÇÃO CONVEXA

→ **Coloração:** Função que atribui cores aos vértices de um grafo, independente de sua vizinhança

→ **Convexa:** Cada classe de cor induz um subgrafo conexo



Exemplo de uma coloração convexa

RECOLORAÇÃO CONVEXA

Dados um grafo e uma coloração qualquer, encontrar o menor número de vértices que precisam ser recoloridos, de modo a obter uma *coloração convexa*

RECOLORAÇÃO CONVEXA DE ÁRVORES

RECOLORAÇÃO CONVEXA DE ÁRVORES

Convex Tree Recoloring - CTR

- $\rightarrow$ Motivação: Estudo de árvores filogenéticas
  - ◇ Distância de Recoloração
  - ◇ Erros na construção da árvore ou na classificação
  - ◇ Indicam a ocorrência de convergências ou reversões
- $\rightarrow$ NP-difícil [1]
- $\rightarrow$ $(2+\epsilon)$-aproximação de Bar-Yehuda, Feldman e Rawitz [2]
- $\rightarrow$ Modelos PLI propostos por de Campêlo *et al.* [3] e Chopra *et al.* [4]

RECOLORAÇÃO CONVEXA DE ÁRVORES COM APENAS FOLHAS COLORIDAS

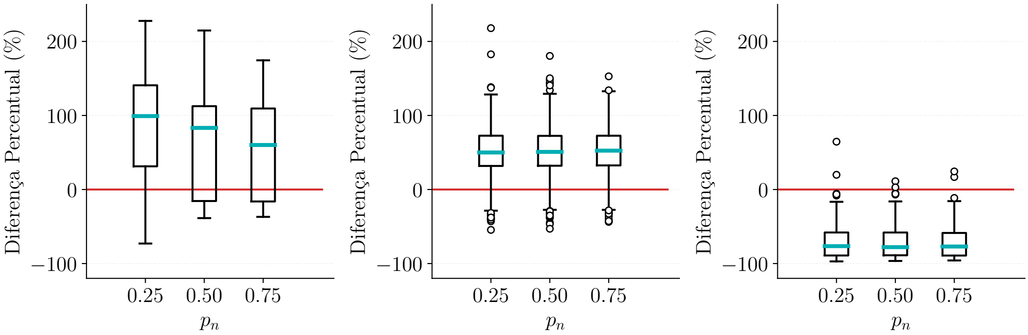
Convex Leaf-colored tree Recoloring - CTR

- $\rightarrow$ NP-difícil [1]
- $\rightarrow$ Algoritmo linear para identificar se precisa recolorir algum vértice [5]
- $\rightarrow$ Polinomial, quando temos até três vértices por cor [6]

RESULTADOS

- → Implementamos o modelo de Programação Linear Inteira para o problema em árvores
- → Criamos instâncias para a versão do problema com apenas folhas coloridas
  - ◇ Usam árvores filogenéticas
  - ◇ Possuem dois parâmetros de configuração p_c e p_n
  - ◇ O parâmetro p_c influencia no número de cores da instância
  - ◇ O parâmetro p_n influencia na quantidade de vértices que devem ser recoloridos
- → Executamos o modelo com instâncias dos dois problemas CTR e CLR

RESULTADOS



(a) $p_c = 0.005$ (b) $p_c = 0.05$ (c) $p_c = 0.5$

Distribuição das diferenças percentuais das instâncias do CTR e do CLR.

RESULTADOS

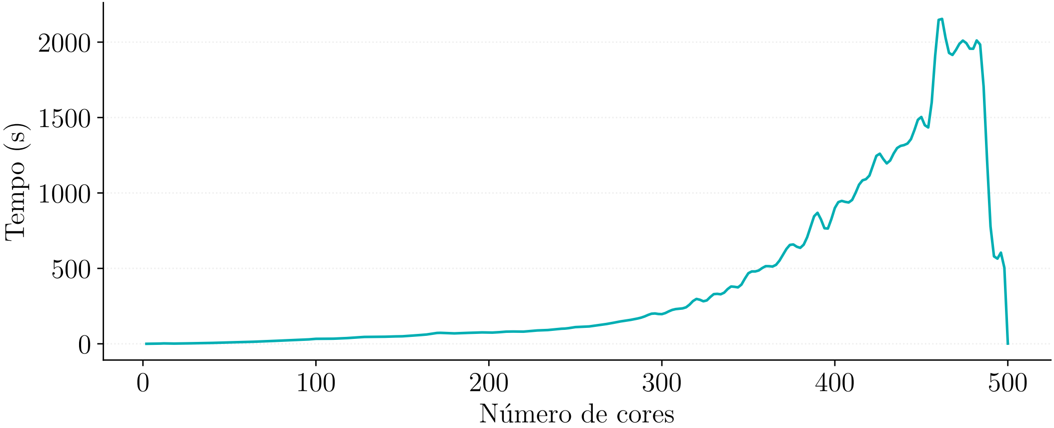
Análise dos resultados

- $\rightarrow$ Diferença no número de cores usadas nas instâncias com $p_c = 0.5$
- $\rightarrow$ Diferença na distribuição de vértices por classe de cor
- $\rightarrow$ Não haviam instâncias do CLR que poderiam ser expandidas sem recolorações das folhas

Novas instâncias

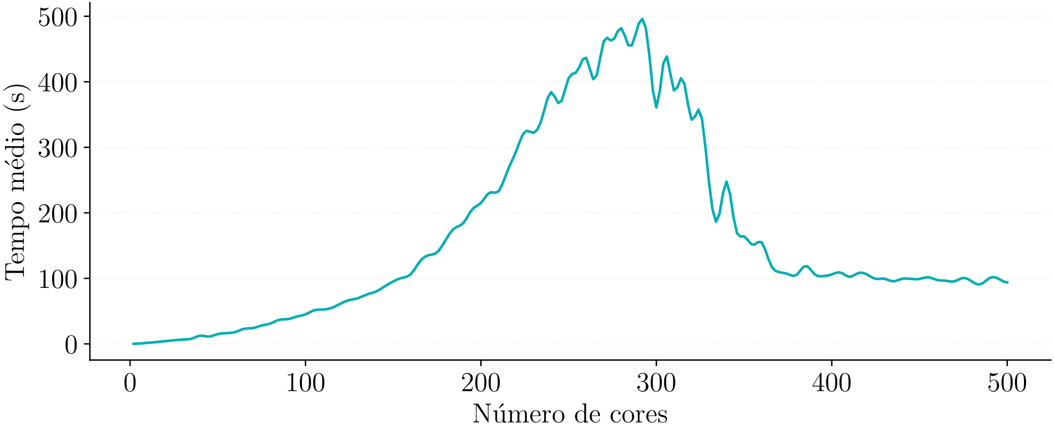
- $\rightarrow$ Criamos novas instâncias específicas para o CLR
- $\rightarrow$ Instâncias criadas variando o tamanho k das classes de cores ($k \in \{2, 3, 4, 5, 6, 7, 8, 9, 10, 15, \ldots, 500\}$)
  - $\diamond$ Conjunto 1: folha i recebe a cor $(i \,\%(k+1))$
  - $\diamond$ Conjunto 2: folha i recebe uma cor aleatória dentre as k, onde cada cor tem a mesma probabilidade de ser escolhida

RESULTADOS



Média de tempo do Conjunto 1

RESULTADOS



Média de tempo do Conjunto 2

RECOLORAÇÃO CONVEXA DE GRAFOS

RECOLORAÇÃO CONVEXA DE GRAFOS

Convex Recoloring - CR

- → Versão do problema de recoloração na qual a classe do grafo não é especificada
- → NP-difícil[7]
- → Não pode ser aproximado com um fator logarítmico [3]
- → Modelo PLI de Campêlo *et al.* [3]
- → Modelo PLI com geração de colunas de Moura [8]

GRASP

- *Greedy Randomized Adaptive Search Procedure*
- Heurística iterativa
- Consiste de duas fases em cada iteração
    - Fase 1: constrói uma solução gulosa aleatorizada
    - Fase 2: busca por um ótimo local
- Estrutura permite a combinação com outras heurísticas
- Poucos parâmetros que precisam de ajustes

CONSTRUINDO UMA SOLUÇÃO

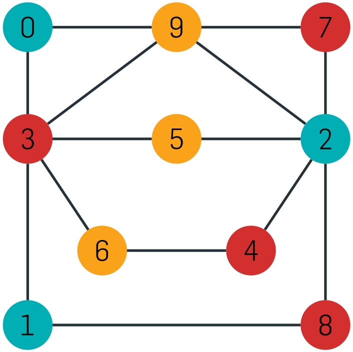
Algoritmo base para resolver uma instância do CR:

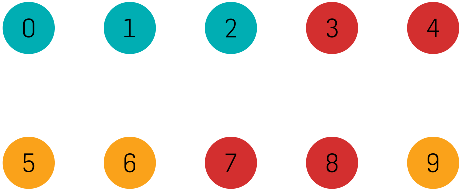
- Contraia todas as arestas que conectam vértices de mesma cor
- Enquanto houver uma cor não convexa, faça:
  - Selecione e ordene vértices candidatos
  - Crie a lista restrita de candidatos (RCL)
  - Sorteie e troque a cor de um vértice da RCL
  - Contraia todas as arestas que conectam vértices de mesma cor

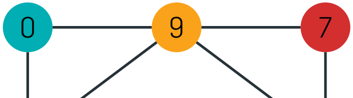
Criamos dois conjuntos de critérios gulosos para selecionar, ordenar e recolorir os vértices candidatos.

CONSTRUINDO UMA SOLUÇÃO

Conjunto de critérios **RATIO**



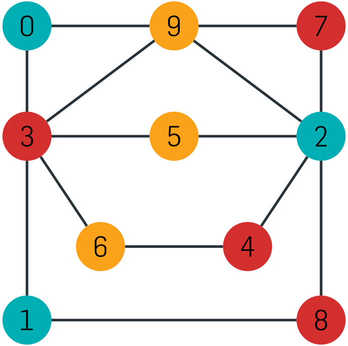


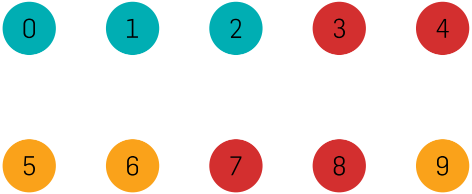


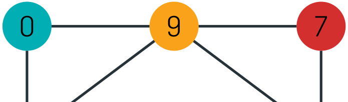
$w=1$	$w=1$	$w=1$	$w=1$	$w=1$
$r=0.5$	$r=1.0$	$r=0.6$	$r=0.6$	$r=0.5$

CONSTRUINDO UMA SOLUÇÃO

Conjunto de critérios **UNION**







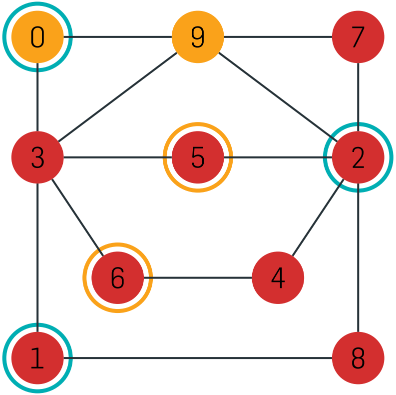
$c=1$ $c=1$ $c=1$ $c=1$ $c=1$

BUSCA LOCAL

- → Ideia base: Retornar um vértice para a sua classe de cor original
- → Desenvolvemos três vizinhanças
  - ◇ vizinhança SIMPLES
  - ◇ vizinhança ESTENDEDIDA
  - ◇ **vizinhança de TROCA**

BUSCA LOCAL

Vizinhança **SWAP**



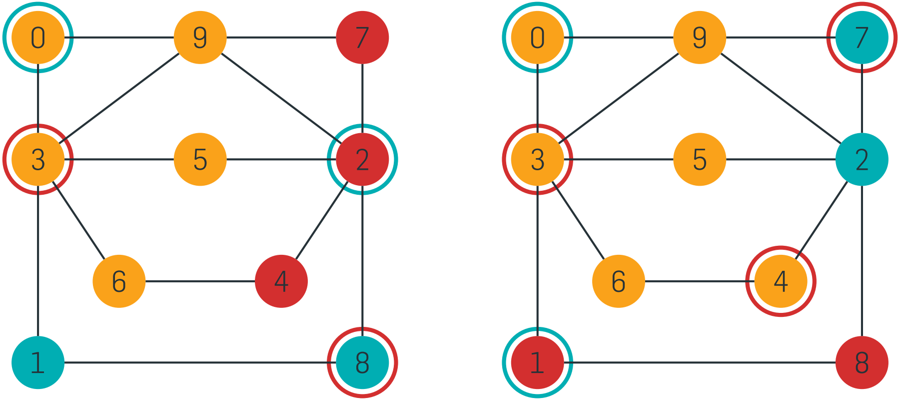
PÓS-PROCESSAMENTO

→ Criamos dois procedimentos para pós-processamento

- ◇ Mutações, aplicado ao final de cada iteração do GRASP
- ◇ **ELITE**, aplicado ao final da execução do GRASP

PÓS-PROCESSAMENTO

Soluções **ELITE**



RESULTADOS

Benchmark de instâncias

- → Grafos conexos Erdos-Renyi
  - ◇ $n \in \{10, 20, \ldots, 100\}$, onde n é o número de vértices
  - ◇ $p \in \{0.1, 0.2, \ldots, 0.5\}$, onde p é a probabilidade de adicionarmos uma aresta ao grafo
- → Usamos uma coloração própria para a coloração inicial [9]
- → Aplicamos um balanceamento nas classes de cores

Experimentos Computacionais

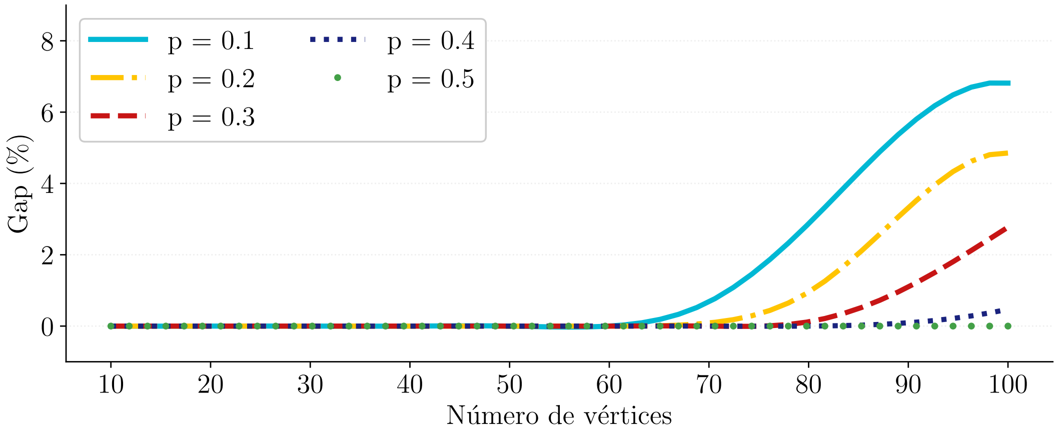
- → Implementamos o modelo de Programação Linear Inteira (PLI) de Campêlo *et al.* [3]
- → Implementamos as versões da heurística GRASP, com as vizinhanças e os precedimentos de pós-processamento

RESULTADOS

Detalhes de Implementação

- → Fixamos o tempo máximo de execução do PLI em 30 minutos
- → Usamos o número de iterações $(2n^2)$ como critério de parada do GRASP
- → Implementações feitas em `C++` (compilador `g++` 5.4)
- → Resolvedor de PLI IBM CPLEX (versão 12.8)
- → Experimentos feitos em uma máquina Intel[R] Core[TM] i$7 3.40$GHz com 32GB de memória e sistema operacional Ubuntu 18.04 LTS

RESULTADOS - PLI



O *gap* considerado aqui é a proporção entre a solução ótima da relaxação e a melhor solução inteira

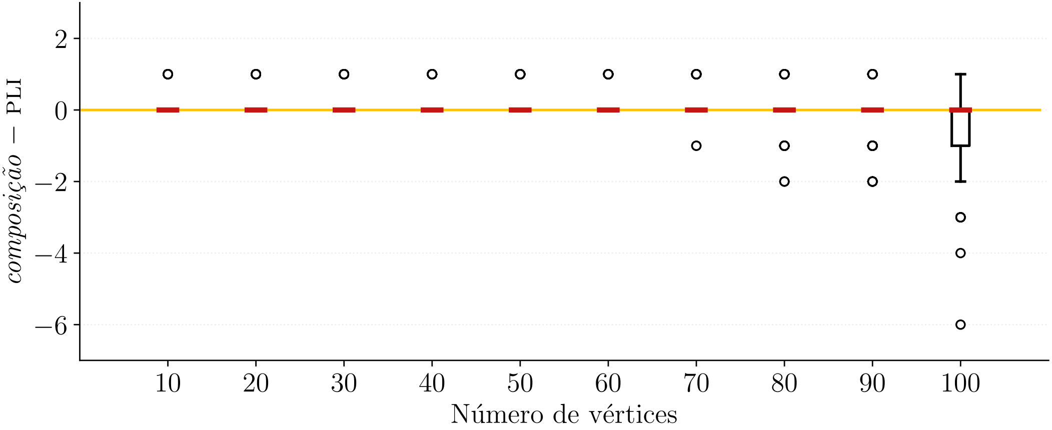
RESULTADOS - GRASP

→ Comparamos para cada conjunto de critério (*ratio* e *union*)
  - as versões puramente gulosas e **aleatorizada** sem busca local
  - as versões sem busca local e com busca local usando as vizinhanças SIMPLES, ESTENDIDA e **de TROCA**
  - as versões com as melhores buscas locais e os procedimentos de pós-processamento Mutações e **ELITE**

→ Verificamos se uma composição dos conjuntos *ratio* e *union* obtêm uma melhora significativa nas soluções
  - duas execuções independentes de cada conjunto

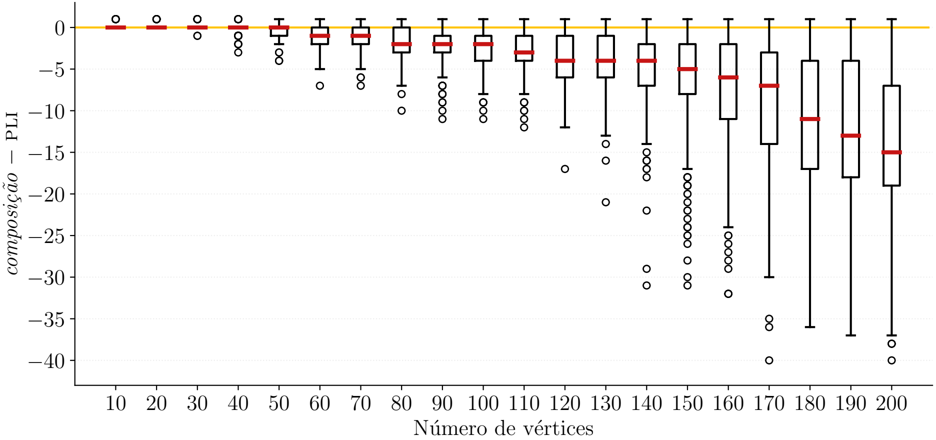
→ Todos os resultados foram validados com testes estatísticos

RESULTADOS - GRASP



Melhor solução do GRASP comparada com a melhor solução do modelo PLI

RESULTADOS - GRASP



Melhor solução do GRASP comparada com a melhor solução do modelo PLI, quando dados os mesmos recursos

RECOLORAÇÃO CONVEXA RESTRITA

RECOLORAÇÃO CONVEXA RESTRITA

Minimum Restricted Recoloring Problem - MRRP

→ Versão do problema de recoloração onde o conjunto de vértices é dividido em dois:
  → vértices clientes
  → vértices servidores
→ Vértices clientes não podem ser recoloridos, apenas descoloridos.

RECOLORAÇÃO CONVEXA RESTRITA

→ NP-difícil[10]

→ $(2 + \epsilon)$-aproximação para grafos com *treewidth* limitado de Kammer e Tholey [10]

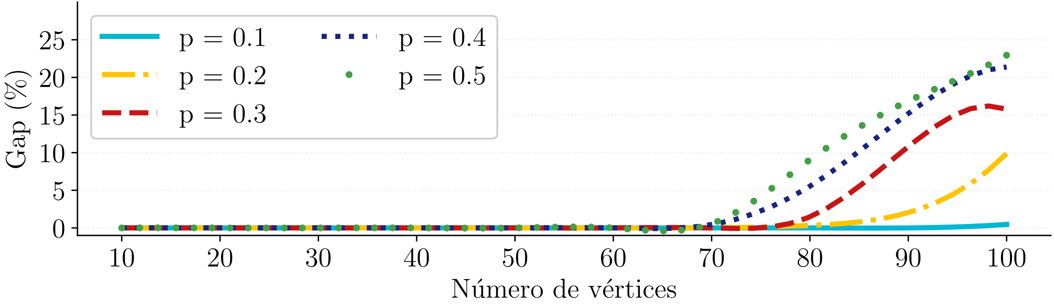
→ Casos polinomiais em grafos com *treewidth* limitado:

- ◇ Até três vértices por cor [10]
- ◇ Número de cores da ordem de $\log(n)$ [10]

RESULTADOS

- → Adaptamos o modelo PLI de Modelo PLI de Campêlo *et al.* [3]
- → Modificamos todas as versões da heurística GRASP
- → Adicionamos à cada instância um conjunto de vértices com papel de roteadores
  - ◇ os vértices roteadores formam um conjunto dominante do grafo da instância
  - ◇ a mesma coloração inicial das instâncias do CR
- → Executamos os experimentos no mesmo ambiente computacional

RESULTADOS - PLI

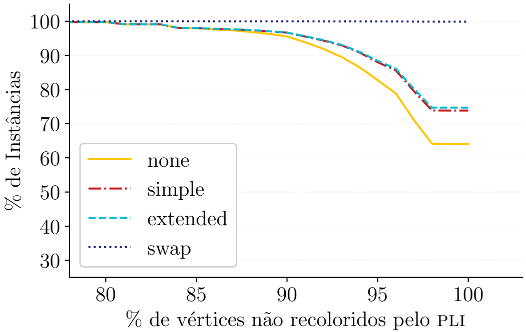


O *gap* considerado aqui é a proporção entre a solução ótima da relaxação e a melhor solução inteira

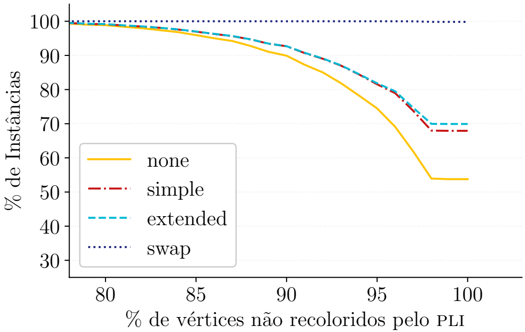
RESULTADOS - GRASP

- → Comparamos para cada conjunto de critério (*ratio* e *union*)
  - ◇ as versões puramente gulosas e **aleatorizada** sem busca local
  - ◇ as versões sem busca local e com busca local usando as vizinhanças SIMPLES, ESTENDIDA e **de TROCA**
  - ◇ as versões com as melhores buscas locais e os procedimentos de pós-processamento Mutações e ELITE
- → Verificamos se uma composição dos conjuntos *ratio* e *union* obtêm uma melhora significativa nas soluções
  - ◇ duas execuções independentes de cada conjunto
- → Todos os resultados foram validados com testes estatísticos

RESULTADOS - GRASP



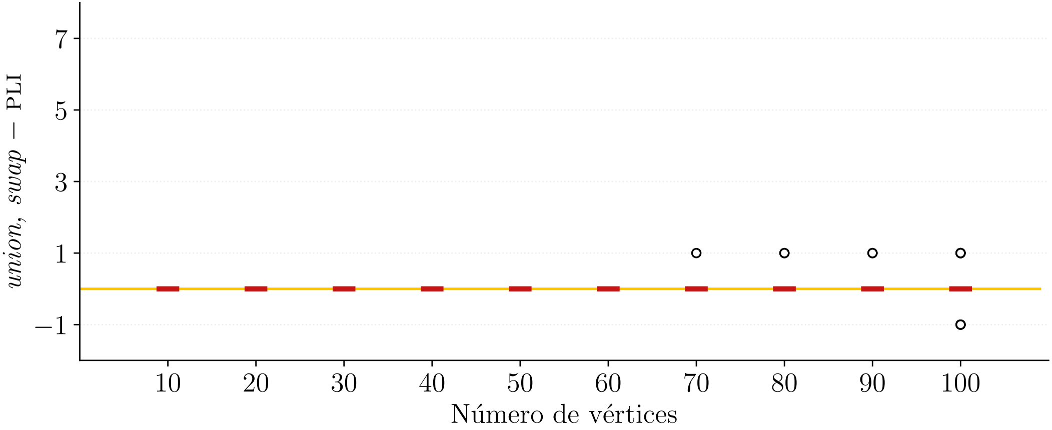
(a) *ratio*



(b) *union*

Figura 24: Comparando as soluções dadas pelo GRASP com o PLI (MRRP).

RESULTADOS - GRASP



Melhor solução do GRASP comparada com a melhor solução do modelo PLI

CONCLUSÕES

CONCLUSÕES

Estudamos três versões do problema de Recoloração Convexa

→ Recoloração Convexa em Árvores (CTR)
→ Recoloração Convexa de Grafos (CR)
→ Recoloração Convexa Restrita (MRRP)

Recoloração Convexa em Árvores (CTR)

→ Analisamos uma subversão do problema onde apenas as folhas são coloridas inicialmente (CLR).
→ Implementamos um modelo da literatura
→ Propusemos novos conjuntos de instâncias
→ Verificamos que o modelo também é adequado para resolver instâncias do versão CLR

CONCLUSÕES

Recoloração Convexa de Grafos (CR)

→ Propusemos uma heurística GRASP para o CR
→ Implementamos um modelo PLI da literatura
→ Propusemos um conjunto de instâncias
→ Analisamos as diversas versões da nossa heurística
→ Verificamos que a heurística retorna uma solução melhor que a do modelo PLI para um número significativo de instâncias
→ Parte dos resultados para esse problema foram publicadas no artigo *“A GRASP for the Convex Recoloring Problem”* [?].

CONCLUSÕES

Recoloração Convexa Restrita (MRRP)

→ Adaptamos a heurística GRASP e modelo PLI

→ A heurística retornou soluções com o mesmo custo que as soluções do modelo para 99% das instâncias, mesmo sem pós-processamento

Trabalhos Futuros

→ Estudar o problema em árvores filogenéticas que representam mais de uma característica por vez

→ Melhorar a heurística GRASP, combinando-a com outras técnicas

→ Estudar a versão do problema MRRP onde cada classe de cor deve induzir no máximo k componentes conexas

REFERÊNCIAS

[1] S. Moran and S. Snir, "Convex Recolorings of Strings and Trees: Definitions, Hardness Results and Algorithms," *Journal of Computer and System Sciences*, vol. 74, no. 5, pp. 850–869, 2008.

[2] R. Bar-Yehuda, I. Feldman, and D. Rawitz, "Improved Approximation Algorithm for Convex Recoloring of Trees," *Theory of Computing Systems*, vol. 43, no. 1, pp. 3–18, 2008.

[3] M. B. Campêlo, K. R. Lima, P. F. S. Moura, and Y. Wakabayashi, "Polyhedral Studies on the Convex Recoloring Problem," *Electronic Notes in Discrete Mathematics*, vol. 44, pp. 233–238, 2013.

[4] S. Chopra, B. Filipecki, K. Lee, M. Ryu, S. Shim, and M. V. Vyve, "An Extended Formulation of the Convex Recoloring Problem on a Tree," *Mathematical Programming*, vol. 165, no. 2, pp. 529–548, 2017.

[5] E. H. Bachoore and H. L. Bodlaender, "Convex Recoloring of Leaf-Colored Trees," Tech. Rep. UU-CS-2006-010, Department of Information and Computing Sciences, Utrecht University, 2006.

[6] I. A. Kanj and D. Kratsch, "Convex Recoloring Revisited: Complexity and Exact Algorithms," in *Proceedings of the 15th International Computing and Combinatorics Conference (COCOON'2009)*, Lecture Notes in Computer Science, (Berlin, Heidelberg), pp. 388–397, Springer Berlin Heidelberg, 2009.

[7] S. Moran, S. Snir, and W.-K. Sung, "Partial Convex Recolorings of Trees and Galled Networks: Tight Upper and Lower Bounds," *ACM Transactions on Algorithms*, vol. 7, no. 4, p. 42, 2011.

[8] P. F. S. Moura, *Graph Colorings and Digraph Subdivisions*.
PhD thesis, Instituto de Matemática e Estatística, Universidade de São Paulo, 2017.

[9] J. A. Bondy and U. S. R. Murty, *Graph Theory*, vol. 244 of *Graduate Texts in Mathematics*.
Springer-Verlag London, 1st ed., 2011.

[10] F. Kammer and T. Tholey, “The Complexity of Minimum Convex Coloring,” *Discrete Applied Mathematics*, vol. 160, no. 6, pp. 810–833, 2012.

Universidade Estadual de Campinas
Instituto de Computação

RECOLORAÇÃO CONVEXA DE GRAFOS

Defesa de Mestrado

27 de Setembro de 2019

Ana Paula dos Santos Dantas
Orientador: Prof. Dr. Zanoni Dias
Coorientador: Prof. Dr. Cid Carvalho de Souza

Instituto de Computação - IC
Unicamp