

# Abordagem Gulosa para Coloração Harmônica de Grafos

Áulus Dauare Pinho

Orientador: *PhD. Márcio Costa Santos*

*Departamento de Ciencia da Computação*

*Universidade Federal de Minas Gerais*

Belo Horizonte, Brasil

aulus97@gmail.com

orcid.org/0000-0003-1845-9242

**Resumo**— Um dos problemas mais populares e amplamente estudados em Teoria de Grafos é o problema de coloração de grafos (PCG), que conta com amplas aplicações práticas na atualidade. Dentre as variantes do PCG encontra-se o Problema de Coloração Harmônica de Grafos (PCHG), que propõe colorir os vértices (nós) de um grafo com o menor número de cores possíveis de maneira que as arestas sejam únicas e distinguíveis pelo conjunto de cores dos vértices de suas pontas, ainda que nós adjacentes no grafo tenham as mesmas cores - o que difere do problema de coloração usual, pois gera uma coloração não-própria. A literatura sobre o Problema de Coloração Harmônica de Grafos ainda hoje carece de trabalhos em Pesquisa Operacional a respeito, assim como escasseiam soluções heurísticas dedicadas ao problema, sendo elas majoritariamente adaptações de soluções para o PCG. Em contraponto, este trabalho propõe uma heurística gulosa dedicada à resolução do PCHG e apresenta os resultados por ela obtidos frente a instâncias distintas de grafos de teste.

**Palavras-chave**—problema de coloração harmônica de grafos, heurística, algoritmo guloso, problema de coloração de grafos.

## I. INTRODUÇÃO

Na área da matemática, a *Teoria dos Grafos* é o campo de estudo sobre *grafos*, que são estruturas matemáticas utilizadas para modelar conjuntos de objetos definidos e as relações existentes entre eles. Nessa estrutura, os objetos são representados por nós ou vértices e, as relações entre entres, por arestas conectando os nós entre si. A *figura 1* ilustra um exemplo simples de grafo.

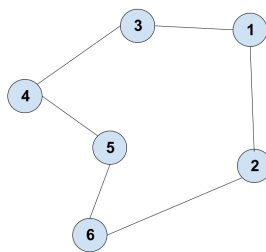


Fig. 1. Grafo simples.

Fonte: Elaborado pelo próprio autor (2023)

Desde os primeiros indícios do uso dessa estrutura em 1736 [8] até os dias atuais, os grafos têm sido utilizados em modelagens diversas para estudos da Ciência da Computação,

Matemática e outras áreas de conhecimento cujos problemas poderiam ser modelados de maneiras interessantes utilizando grafos. A utilização de grafos também é observada em aplicações comuns da atualidade, como em sistemas de navegação por GPS, gerenciamento de tráfego aéreo [3], análises e estruturação de redes sociais [5] [9], problemas de atribuição de tarefas [7], dentre outros.

Dentro da Teoria dos Grafos, um dos problemas mais conhecidos e estudados é o Problema de Coloração de Grafos (PCG). Consiste ele na atribuição de cores a cada um dos vértices de modo que cora-se todos os nós do grafo com o menor número de cores possíveis e desde que vértices adjacentes (ou vizinhos, por estarem conectados por uma aresta em comum) não possuam a mesma cor. Pode-se dizer que essa característica é chamada de coloração própria. Uma variante do PCG é o Problema de Coloração Harmônica de Grafos (PCHG) que, em inglês, é chamado de Harmonious Graph Coloring Problem. Nesta variante, as arestas do grafo são coloridas ou rotuladas e este rótulo é formado pelo par de cores dos vértices em suas pontas. O objetivo é utilizar o menor número possível de rótulos atribuídos a cada aresta de modo que cada aresta e respectivo rótulo sejam únicos ou distinguíveis no grafo.

O PCHG também possui potencial para ser aplicado a diversos problemas reais, conforme aponta [6]. Um exemplo de sua aplicação é o fato de ser uma extensão do problema de orientação de aviação [7]. Como se trata de um sistema crítico, qualquer risco de colisão entre aeronaves é totalmente indesejado e, naturalmente, isso demanda uma atenção também na estrutura de comunicação entre torres e na organização de vias aéreas. Grafos se mostram como interessantes abordagens para esse problema. Os vértices do grafo poderiam representar as torres faróis e as arestas as vias aéreas ligando as torres. Cada torre poderia ser representada por uma cor e cada via aérea por um rótulo composto pelo conjunto contendo as torres de início e fim. Seguindo as características do PCHG, cada via aérea representada por uma aresta deve conter um conjunto único de torres de modo que não haja confusão entre uma via aérea e outra já existente e representada no grafo [7]. A *figura 2*, proposta por [7], ilustra o exemplo citado.

O Problema de Coloração Harmônica de Grafos, foi enunciado há mais de 40 anos atrás por [4] e, ainda hoje, há uma

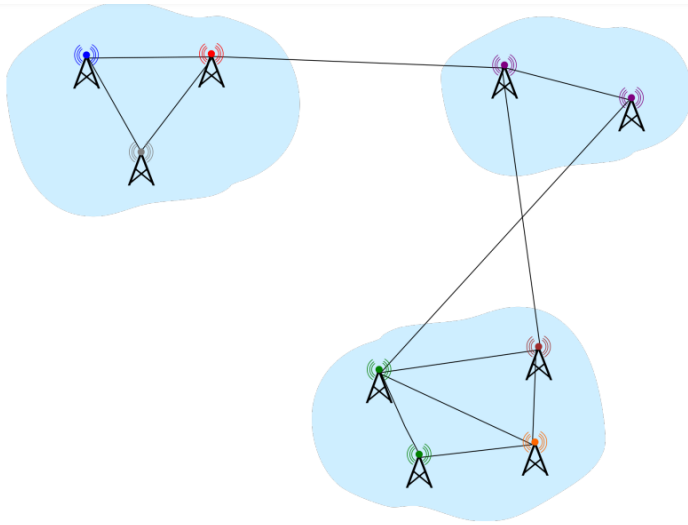


Fig. 2. Modelo em grafo para o Problema de Orientação de Aviação.  
Fonte: [7]

grande deficiência de trabalhos em Pesquisa Operacional e de algoritmos dedicados e escaláveis [7] para sua resolução.

Considerando as interessantes aplicações práticas nas quais o PCHG pode ser adotado [6], entende-se como de real importância o esforço por ampliar os estudos e desenvolver soluções dedicadas à coloração harmônica de grafos. As pesquisas em torno desta variante do PCG poderão abrir espaço para alargamento do saber na resolução de problemas atuais, assim como inspirar a modelagem e solução de outros problemas análogos, deriváveis ou redutíveis ao PCHG. Conforme aponta [7], o desenvolvimento de métodos eficientes para problemas de otimização combinatória poderá ser fomentado a partir da resolução do Problema de Coloração Harmônica de Grafos.

No intuito de começar a sanar a lacuna ainda existente na literatura, este trabalho propõe uma heurística gulosa dedicada ao Problema de Coloração Harmônica de Grafos (PCHG). Neste caso, esta proposta não se trata de uma adaptação de uma heurística do PCG, como o fazem a maioria dos poucos trabalhos sobre PCHG existentes na literatura, mas de propor uma abordagem dedicada ao mesmo.

### A. Objetivos

O principal objetivo deste trabalho é propor e desenvolver uma heurística gulosa aplicada ao Problema de Coloração Harmônica de Grafos. Os objetivos específicos são os que se seguem:

- Realizar um levantamento bibliográfico das estratégias computacionais seminais e atuais aplicadas ao Problema de Coloração Harmônica de Grafos;
- Propor uma heurística gulosa para a Coloração Harmônica de Grafos.
- Avaliar o desempenho e comportamento da heurística proposta para distintas instâncias de grafos de teste.

### B. Estrutura do trabalho

O restante deste trabalho se estrutura em 4 seções. A seção 2 apresenta uma revisão bibliográfica em torno das temáticas abordadas nesta monografia. A seção 3 contém o desenvolvimento do trabalho ao apresentar o método desenvolvido para implementação da heurística gulosa. A seção 4 contém os resultados da aplicação dos testes de comportamento e desempenho da heurística frente aos diversos grafos de entrada, apresentando, igualmente, uma comparação dos resultados obtidos com os valores referenciais. A seção 5, por fim, contém a conclusão deste trabalho e a proposição de temas para trabalhos futuros. Os apêndices A e B contendo tabelas e gráficos não apresentados nas seções de corpo deste trabalho encontram-se ao final deste.

## II. PRELIMINARES

### A. Teoria dos Grafos

O *grafo* - objeto central de estudo da Teoria dos Grafos - é uma estrutura matemática utilizada para modelar objetos e as relações entre eles. Um grafo  $G$  *simple* e *não direcionado* pode ser formalmente definido por  $G = (V, E)$ , como sendo composto por um conjunto  $V$  finito e não vazio de elementos denominados *vértices* ou *nós* e por um conjunto  $E$  de *arestas*, que são pares não ordenados de vértices distintos de  $V$ . Assim sendo, para dois vértices distintos quaisquer  $x$  e  $y$  de  $V$ , uma aresta  $e$  que conecte  $x$  e  $y$  é definida por  $e = xy$ , podendo esta aresta ser denotada por  $xy$  e ser descrita como uma aresta que incide em  $x$  e  $y$ , de modo que  $x$  e  $y$  são ditos extremos de  $e$  ou que  $x$  e  $y$  são *vértices adjacentes*. Formalmente,  $E \subseteq \{\{x, y\} \mid x, y \in V, x \neq y\}$ . Existem representações de grafos com arestas ligando dois vértices sem estabelecer uma direção a esta ligação - os chamados *grafos não direcionados*, assim como existem grafos com arestas estabelecendo uma direção na ligação entre os nós de suas extremidades - são os *grafos direcionados*.

Para um vértice  $x$  de  $V$ , diz-se que o *grau* de  $x$ , denotado por  $g(x)$ , é definido pelo número de arestas que incidem em  $x$  ou, de maneira mais simples, o número de arestas que contenham  $x$ . Nesse sentido, o *grau máximo* de um grafo, denotado por  $\Delta(G)$ , representa o maior número de arestas que incide em qualquer nó de um grafo  $G$ , sendo formalmente definido como  $\Delta(G) = \max\{g(x) \mid x \in V(G)\}$ . Analogamente, o *grau mínimo* de um grafo, denotado por  $\delta(G)$ , representa o menor número de arestas incidentes em qualquer vértice de  $G$ , definido formalmente como  $\delta(G) = \min\{g(x) \mid x \in V(G)\}$ .

Define-se um *passeio* como sendo uma sequência de vértices  $x \in V(G)$  tais que, se  $y$  e  $x$  forem nós consecutivos na sequência, então  $yx$  é uma aresta do grafo  $G$  e, define-se como uma aresta do passeio qualquer aresta  $yx \in E(G)$  tais que  $x$  seja um sucessor de  $y$  no passeio em questão. Um *caminho* em um grafo  $G$  é um passeio cuja sequência é formada por nós  $(x_0, x_1, \dots, x_k)$  de  $V(G)$  distintos entre si, ou seja, é um passeio com arestas distintas entre si. Um *ciclo* em um grafo é um caminho fechado, isto é, um caminho que começa e termina no mesmo vértice. Todo ciclo em um grafo deve possuir um

comprimento maior que 2 vértices e não podem haver arestas repetidas. Um grafo composto por  $n$  vértices e que forma um único ciclo é denotado por  $C_n$  e é chamado de *grafo circular*.

Um grafo  $G$  é denominado *completo* se, para todos e quaisquer dois vértices  $x$  e  $y$  de  $V(G)$ , exista uma aresta entre eles ( $\forall x, y \in V(G), (x, y) \in E(G)$ ). Um grafo completo  $G$  que possua  $n$  nós é denotado por  $K_n$ . Um grafo  $G$  é denominado *conexo* se, para quaisquer dois vértices  $x$  e  $y$  de  $V(G)$  exista um caminho que parta de  $x$  e chegue em  $y$ . Para um dado grafo  $G' = (V', E')$ , se  $V' \subseteq V$  e  $E' \subseteq E$ , então  $G'$  é denominado um *subgrafo* de  $G$ . E, se qualquer par de nós  $x, y$  de  $G$  forma uma aresta  $xy$  em  $G$  e também em  $G'$ , então  $G'$  é denominado um subgrafo *induzido*.

### B. Problemas de Coloração de Grafos

PCG Problemas de coloração de grafos (PCG) são muito populares na teoria de grafos [14] e, dentre os seus variados tipos, um dos mais estudados é o de coloração de vértices que, inclusive, é tomada como definição clássica do PCG. De acordo com essa definição, dado um grafo  $G$  e um número inteiro positivo  $k$ , busca-se uma coloração *própria* de  $G$  utilizando  $k$  cores. Uma coloração própria ocorre se todos os vértices adjacentes possuem cores distintas - formalmente, pode ser considerada uma função  $c$  de coloração, tal que  $c: V(G) \rightarrow N$  (em que  $N$  é um conjunto de inteiros positivos) tal que para quaisquer vértices distintos  $u$  e  $v$  de  $V(G)$ ,  $c(u) \neq c(v)$ . Se cada cor utilizada para coloração de vértices de  $G$  for uma das  $k$  cores definidas, então denomina-se essa como sendo uma *k-coloração* [1].

Um grafo  $G$  é dito *k-colorível* se  $G$  pode ser colorido a partir de um conjunto de  $k$  cores, isto é, se existe uma *k-coloração* de  $G$  [1]. O menor número  $k$  tal que um grafo  $G$  seja *k-colorível* é denominado *número cromático*, denotado por  $\chi(G)$ . Assim, um grafo  $G$  com número cromático  $k$  é denominado grafo *k-cromático*. Consequentemente, se o número cromático de um grafo  $G$  é  $k$ , isto é, se  $\chi(G) = k$ , então existe uma *k-coloração* de  $G$  mas não uma *(k-1)-coloração* [1].

Um outro tipo de coloração muito estudado em Teoria de Grafos são as colorações *distintivas*, que consistem em atribuir *rótulos* a vértices ou arestas como forma de coloração, de maneira que os objetos coloridos sejam distintos entre si. Rótulos podem ser, por exemplo, números ou conjuntos atribuídos aos objetos em que se aplique a rotulagem. Desse modo, uma coloração *vértice-distinguível* é aquela em que os vértices recebem rótulos únicos e distintos entre si. Analogamente, uma coloração *aresta-distinguível* é tal que os rótulos atribuídos a cada aresta sejam únicos e distintos entre si.

Outras definições interessantes porém, não necessárias ao momento, podem ser consultadas em [1].

Além das aplicações indicadas na seção Introdução deste trabalho, outras mais podem ser apresentadas a título de enriquecer o leitor mas sem a pretensão de esgotar as possibilidades existentes. Conforme apresentado por [9], algumas aplicações do PCG auxiliam na gestão de horários educacionais [27], de projetos [28], de canais de redes sem fio [29], de

competições esportivas [31] e de tarefas em pequenas linhas de montagem e sistemas máquinas em trabalho conjunto [30].

### C. Problemas de Coloração Harmônica de Grafos

PCHG O Problema de Coloração Harmônica de Grafos (PCHG) é uma variante de PCG e, mais especificamente, um tipo de coloração distintiva no qual os vértices adjacentes de um dado grafo  $G$  até podem ter as mesmas cores [4], provocando uma indução aresta-distinguível do grafo, tendo as arestas  $xy \in E(G)$  rótulos formados pelo conjunto  $\{cor(x), cor(y)\}$  das cores dos nós  $x, y \in V(G)$  de suas extremidades. A definição de [4] indica que a coloração harmônica de grafos é também uma coloração *não-própria*. A figura 3 ilustra uma representação [1] de uma coloração harmônica de um grafo. [17] nomeou de *Coloração Harmônica* o tipo de coloração harmônica que gera uma coloração própria: quando vértices adjacentes são proibidos de terem a mesma cor, restringindo um pouco mais a definição apresentada por [4]

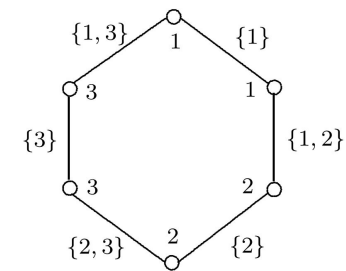


Fig. 3. 3-Coloração harmônica de um grafo.

Fonte: [1]

Analogamente ao número cromático para o PCG, o *número cromático harmônico*, denotado por  $\lambda(G)$ , é o menor inteiro positivo  $k$  para o qual se pode colorir um dado grafo  $G$  de modo que  $G$  possua uma *k-coloração harmônica* [1].

Estudos em torno do número cromático harmônico foram apresentados [18] [19] [20] [21] [22] desde a enunciação do PCHG em 1983 [4] e perduram até a atualidade [24], representando esforços para desenvolvimento dos estudos em torno desse conceito para diferentes tipos, classes [23] e disposições de grafos [25], de hipergrafos [26], etc.

## III. METODOLOGIA

A heurística dedicada ao PCHG foi construída através de uma abordagem gulosa. Abordagens como esta consistem, basicamente, em fazer escolhas localmente ótimas com a intenção de obter-se uma solução ótima global ao final de todas as iterações. Para ter-se uma ideia do funcionamento desta abordagem uma possível implementação dela poderia ser descrita da seguinte forma: dada uma lista  $k$  de cores possíveis, percorra cada vértice do grafo uma única vez, atribuindo a cada nó uma cor que não tenha sido atribuída a nenhum de seus vizinhos já coloridos e que seja a menor cor possível dentro da lista  $k$ . Ao final, obter-se-á uma lista com a menor quantidade de cores possíveis utilizadas para colorir todos os vértices do grafo.

A heurística dedicada ao PCHG proposta neste trabalho será referenciada pela sigla HG para abreviar o nome Heurística Gulosa. Para facilitar o entendimento do funcionamento completo da HG explicitado no Algoritmo 3, serão apresentadas sequencialmente suas principais partes e funções de modo a criar um fluxo didático de compreensão que culmine no panorama completo da HG.

Para seu devido funcionamento, a HG necessita seja definida uma ordem em que os vértices do grafo sejam visitados e coloridos. Tal ordenação foi definida de duas maneiras diferentes na HG. A primeira e, na execução da HG, usada por padrão, pode ser chamada de Ordenação por Saturação [10]. A segunda pode ser chamada de Ordenação por Grau (de Vértices). A primeira foi escolhida como método padrão de ordenação por se mostrar mais eficaz na obtenção de colorações menores [13] e eficientes do que a segunda, conforme se apreende dos trabalhos [1], [10], [11] e [12].

A Ordenação por Saturação propõe que todos os vértices sejam ordenados de maneira decrescente a partir do nó de maior grau até o menor atentando-se que, após cada vértice de grau máximo ser escolhido para ocupar uma posição na lista de ordenação, todos os seus vizinhos têm o valor de seus graus decrescidos em 1 unidade; esse processo acontece para todos os nós do grafo de forma que o próximo vértice de grau máximo a ser escolhido é sempre aquele que, após a atualização de graus ser feita, possua o grau atual máximo dentre os graus atuais de todos os nós disponíveis restantes. Esse tipo de ordenação é também chamado de Ordenação por Maior Grau Dinâmico. Para facilitar a compreensão, o Algoritmo 1 ilustra o funcionamento deste tipo de ordenação. Já a Ordenação por Grau propõe que todos os vértices sejam ordenados de maneira decrescente desde o nó de maior grau até o menor selecionando sempre, a cada iteração, o vértice de grau máximo dentre o conjunto de nós disponíveis restantes para escolha. Esse tipo de ordenação é mais simples e, nesse sentido, considerado um método ‘naive’.

---

#### Algorithm 1 Saturation Ordering

---

**Require:** lista  $N$  de todos os  $n$  vértices do grafo  $G$

**Ensure:** lista  $O$  com os  $n$  vértices ordenados

```

1:  $L \leftarrow \{1, 2, \dots, N\}$ .
2: for  $i = 1, \dots, N$  do
3:    $x \leftarrow VertexOfMaxDegree(L)$ .
4:    $O[i] \leftarrow x$ .
5:   for all  $j \in x.Neighbors()$  do
6:     if  $j.degree() \geq 1$  then
7:        $j.degree() \leftarrow j.degree() - 1$ .
8:     end if
9:   end for
10:   $L \leftarrow L - x$ .
11: end for
12: return  $O$ .
```

---

A etapa essencial para o devido funcionamento da HG é a de coloração. Nela, a abordagem gulosa é utilizada para execução de uma efetiva coloração harmônica do grafo  $G$  fornecido

como entrada para a HG. O algoritmo projetado para esse fim percorre sequencialmente a lista de vértices obtida na etapa anterior de ordenação e atribui a cada vértice a menor cor possível que atenda a todos os 2 requisitos descritos abaixo.

Seja  $i$  o vértice a ser colorido na iteração atual tal que  $i \in O$  e  $O$  é uma lista de ordenação tal que  $|O| = |V(G)|$ , seja  $K_i$  o conjunto de vizinhos de  $i$  tal que  $|K_i| \leq |V(G) - 1|$  e seja  $j$  a menor cor candidata para colorir  $i$ , temos:

- 1)  $cor(y) \neq j, \forall y \in K_x, \forall x \in K_i$ , ou seja, os  $|K_i|$  nós vizinhos ao vértice  $i$  não podem ter seus respectivos  $|K_i|$  vizinhos coloridos com a cor  $j$  em análise;
- 2) Não pode haver no grafo  $G$  nenhuma aresta com rótulos  $\{j, cor(x)\}$  ou  $\{cor(x), j\}$  já existentes para nenhum  $x \in K_i$ ;

Para registro das cores (rótulos) das arestas do grafo  $G$ , utilizou-se uma matriz  $|V(G)| \times |V(G)|$  booleana capaz de representar a existência (*true*) ou não (*false*) de um rótulo  $\{x, y\}$  tais que  $x = cor\ i$  e  $y = cor\ j$  de toda aresta  $A_{(i,j)}$  no grafo  $G$ . Lembrando que o grafo  $G$  fornecido de entrada na HG é não-direcionado, tem-se que, toda vez que o último vértice ainda não colorido de uma aresta  $A_{(i,j)} \in G$  recebe uma cor, a matriz de cores de arestas é atualizada e, as posições  $matriz[x][y]$  e  $matriz[y][x]$  são definidas como *true*, de modo a representar que a aresta que liga os vértices  $i$  e  $j$  foi colorida (rotulada) pelo par de cores  $\{x, y\}$  dos vértices de suas pontas - conceito previamente apresentado na seção de Introdução deste trabalho. A presença de um *false* em quaisquer das posições  $x$  e  $y$  da matriz indicam que nenhuma aresta  $A_{(i,j)} \in G$  foi colorida com o par de cores  $\{x, y\}$ .

O Algoritmo 2 apresenta o funcionamento da função de coloração implementada na HG. A propriedade *blocking\_colors* refere-se ao controle de cores explicitado no requisito 1 do antepenúltimo parágrafo.

Passadas as etapas de ordenação e de coloração harmônica do grafo  $G$  de entrada, é fundamental que se verifique a validade da coloração obtida de modo a conferir se os princípios de coloração harmônica foram respeitados. Para tanto, a HG conta com uma função específica para esta etapa.

A partir de uma matriz  $|V(G)| \times |V(G)|$  booleana inicializada em *true* em todas as suas posições, a função de verificação percorre uma única vez cada um dos  $|V(G)|$  vértices do grafo  $G$  não direcionado e, para cada vizinho  $j$  de um vértice  $i$  em visita na iteração atual, é feita uma inversão do valor booleano da matriz em ambas as posições  $(x, y)$  e  $(y, x)$  para  $x = cor\ i$  e  $y = cor\ j$  que representam a aresta  $A_{(i,j)} \in G$  em visita. Considerando que a função de coloração da HG descrita anteriormente utiliza a restrição de não repetição de cores de arestas de maneira intrínseca ao processo de coloração e considerando também que o registro da coloração de cada aresta persiste sem sofrer quaisquer alterações posteriores ao instante em que é definida, a única maneira de uma cor de aresta se repetir é se houver algum erro de memória do sistema operacional. Tal erro seria algo imprevisível e alheio ao funcionamento da HG que teve todos os seus *memory leaks* identificados, solucionados e prevenidos (vide seção de Memory leaks). Matematicamente, se as restrições do PCHG

---

**Algorithm 2** Harmonious Coloring

---

**Require:** lista  $O$  com os  $|V(G)|$  vértices ordenados**Ensure:** matriz  $M$  ( $|V(G)| \times |V(G)|$ ) com as  $E(G)$  arestas coloridas harmonicamente

```
1:  $M_{(i,j)} \leftarrow false \forall \{i,j\} \in |V(G)|$ .
2: for all  $i \in O$  do
3:    $j \leftarrow 1$ .
4:   while  $found\_color\_flag = false$  do
5:     if  $i.Neighbors() = 0$  then
6:        $found\_color\_flag \leftarrow true$ .
7:     else
8:       for all  $k \in i.Neighbors()$  do
9:         if  $k.blocking\_colors(j) = true$  then
10:           $found\_color\_flag \leftarrow false$ .
11:          break while loop.       $\triangleright$  requisite 1
                                   violated!
12:        end if
13:        if  $M_{(j,k.color)} = true$ 
14:          OR
15:           $M_{(k.color,j)} = true$ 
16:        then
17:           $found\_color\_flag \leftarrow false$ .
18:          break while loop.       $\triangleright$  requisite 2
                                   violated!
19:        end if
20:      end for
21:    end if
22:     $found\_color\_flag \leftarrow false$ .
23:    if  $i.Neighbors() > 0$  then
24:      for all  $k \in i.Neighbors()$  do
25:         $k.blocking\_colors(i.color) \leftarrow true$ 
26:        if  $isColored(k) = true$  then
27:           $edgeColorExistence(i.color, k.color)$ 
28:             $\leftarrow$ 
29:           $edgeColorExistence(k.color, i.color)$ 
30:             $\leftarrow true$ .
31:        end if
32:      end for
33:    end if
34:  end while
35: end for
36: return  $M$ .
```

---

foram devidamente respeitadas pelo algoritmo de coloração, o número de inversões booleanas feitas em todas as posições da matriz de verificação é exatamente 2. Essa quantidade é verificável intuitivamente pois que se cada vértice  $i \in V(G)$  é visitado uma única vez e se cada aresta  $A_{(i,j)}$  incidente em  $i$  é visitada somente uma vez quando  $i$  está sendo visitado e somente uma vez quando seu vizinho  $j$  estiver sendo visitado então o número de visitas a uma aresta  $A_{(i,j)}$  de um grafo  $G$  não direcionado é exatamente 2. A função de verificação, ao fazer exatas duas visitas em cada aresta  $A_{(i,j)}$  do grafo  $G$ , realiza exatas duas inversões nos valores booleanos das posições matriciais  $(x, y)$  e  $(y, x)$  para  $x = \text{cor } i$  e  $y = \text{cor } j$ , o

que, na prática, não altera o valor original que estas posições possuíam quando foram inicializadas no início de execução da função verificadora. Através dessa propriedade pode-se efetuar a verificação quanto à corretude de coloração harmônica de arestas obtida, uma vez que, se cada cor de aresta foi devidamente usada uma única vez, o número de visitas a cada aresta  $A_{(i,j)}$  do grafo  $G$  assim como o número de inversões provocadas nas correspondentes posições da matriz verificadora  $(x, y)$  e  $(y, x)$  para  $x = \text{cor } i$  e  $y = \text{cor } j$  serão exatamente 2. Por consequência, o valor final booleano apresentado na matriz de verificação nas respectivas posições  $(x, y)$  e  $(y, x)$  permanecerão *true*. Se, portanto, o valor final booleano de qualquer posição da matriz de verificação for diferente do valor original *true*, deduz-se que a cor (rótulo) de aresta representado por essa posição foi usada mais de uma vez ou, então, houve algum erro de memória inesperado.

Explicitadas as funções principais que compõem a HG, torna-se mais compreensível o entendimento de seu funcionamento completo apresentado no Algoritmo 3.

---

**Algorithm 3** Harmonious Graph Coloring Greedy Heuristic (HG)

---

**Require:** grafo  $G$  não direcionado**Ensure:** matriz  $M$  ( $|V(G)| \times |V(G)|$ ) com as  $E(G)$  arestas coloridas harmonicamente, número  $X$  de cores de arestas obtida

```
1:  $G \leftarrow readGraph()$ .
2:  $OrderType \leftarrow Saturation\_Order$  OR
3:    $userChoice(Saturation\_Order, Naive\_Order)$ .
    $\triangleright$   $Saturation\_Order$  is default value above.
4:  $O \leftarrow defineNodesOrder(OrderType, |V(G)|)$ .
5:  $M \leftarrow colorGraph(O)$ .
6:  $X \leftarrow getNumberOfEdgeColors()$ .
7: if  $isSolutionCorrect(M) = true$  then
8:   return  $M, X$ .
9: else
10:  return  $Error$ .
11: end if
```

---

### A. Complexidade Computacional

Considerando a implementação da HG apresentada em partes ao longo de toda esta seção e, diretamente, no Algoritmo 3, a análise de sua complexidade computacional de tempo pode ser expressa a partir das considerações abaixo. Seja um grafo  $G$  não direcionado e defina  $n = |V(G)|$ ,  $m = |E(G)|$ , tem-se que:

- a etapa de inicialização do grafo e leitura do arquivo de entrada na HG possui uma complexidade de pior caso de  $O(n^2)$ , pois que:
  - a etapa independente de inicialização do grafo  $G$  é realizada em  $O(n^2)$  iterações;
  - a etapa independente de leitura do arquivo de entrada contendo as informações do grafo  $G$  de entrada requer, no máximo, um número linear equivalente

a  $O(m)$  iterações, por causa da disposição das informações no arquivo de entrada;

- a etapa ordenação de vértices possui uma complexidade de pior caso de  $O(n \log n)$  em quaisquer dos tipos de ordenação aplicados;
- a etapa de coloração harmônica do grafo  $G$  possui uma complexidade de pior caso de  $O(n^2)$ ;
- a etapa de verificação da solução obtida possui uma complexidade de pior caso de  $O(n^2)$  por percorrer sequencialmente duas vezes todas as arestas do grafo  $G$ ;
- a etapa final interna de desalocação de toda a memória utilizada na função *main* da HG tem complexidade de pior caso  $O(n^2)$  considerando a necessidade de alocação dinâmica de estruturas bidimensionais de tamanho máximo igual a  $O(n^2)$ , o que, de acordo com a lógica de funcionamento das funções de alocação dinâmica das bibliotecas padrão utilizadas, consomem  $O(n)$  iterações máximas para desalocação das mesmas.

Assim, a complexidade computacional de tempo total da HG fica estabelecida em  $O(n^2)$ .

Considerando, igualmente, a implementação da HG apresentada nesta seção, a análise de sua complexidade computacional de espaço total da HG fica estabelecida em  $O(n^2)$ , uma vez que todas as estruturas são dinamicamente alocadas e, dentre elas, as maiores são as que representam o grafo  $G$  e a matriz de cores de arestas, resultando na complexidade final acima informada.

#### IV. RESULTADOS

Foram conduzidos testes para avaliar o desempenho da HG frente a diferentes instâncias de grafos comparando os resultados com valores padrão obtidos para tais instâncias. A HG foi implementada em linguagem  $C$  e executada em uma máquina Ubuntu 20.04.2 LTS, Intel® Core™ i7-6700, 8 x 3.40GHz, 16GB DDR4 RAM, 1 TB Disk, Mesa Intel® HD Graphics 530 (SKL GT2), arquitetura 64-bit com ordem de bytes *Little Endian*. Mais detalhes sobre as especificações da máquina encontram-se apresentadas no Apêndice A.

##### A. Memory leaks

Ressalta-se que a HG que teve todos os seus memory leaks identificados, solucionados e prevenidos com o auxílio da ferramenta valgrind e das respectivas flags `-s -leak-check=full -show-leak-kinds=all`. Para efeitos de reprodutibilidade, segue abaixo o resultado dessa averiguação para uma das instâncias testadas.

##### B. Instâncias

Foram utilizadas instâncias de duas fontes diferentes. O primeiro grupo de instâncias foi proveniente do conjunto de testes *DIMACS Challenge* [2]. O conjunto obtido contou com 54 instâncias de grafo com exemplares variando o número de vértices entre 11 e 496. O segundo grupo foi um conjunto de 100 instâncias de grafos gerados aleatoriamente [10] com número de vértices entre 10 e 100 e

---

**Instância DIMACS usada:** `'queen14_14.col'`

**Comando usado:** `valgrind -s --leak-check=full --show-leak-kinds=all ./greedy_pchg < ./queen14_14.col`

**Resposta obtida após execução do comando acima:**

```
==14885== Memcheck, a memory error detector
==14885== Copyright (C) 2002-2017, and GNU GPL'd,
           by Julian Seward et al.
==14885== Using Valgrind-3.18.1 and LibVEX; rerun with
           -h for copyright info
==14885== Command: ./greedy_pchg
==14885==
```

Harmonious Coloring solved with 196 vertex colors!

```
==14885==
==14885== HEAP SUMMARY:
==14885== in use at exit: 0 bytes in 0 blocks
==14885== total heap usage: 17,938 allocs, 17,938 frees,
           675,950 bytes allocated
==14885==
==14885== All heap blocks were freed – no leaks are
           possible
==14885==
==14885== ERROR SUMMARY: 0 errors from 0 contexts
           (suppressed: 0 from 0)
```

---

densidade de arestas definidas a partir de probabilidades de 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 e 0,9.

Os valores padrão (os melhores já alcançados) ou os ótimos de solução de 30 das 54 instâncias *DIMACS* [2] foram obtidas a partir de um estudo conduzido pelo Professor Orientador deste trabalho que ainda não teve sua versão preliminar submetida e, por isso, não há referência ainda. As demais soluções padrão ou ótimas deste grupo não foram encontradas. Os valores padrão (os melhores já alcançados) ou os ótimos de solução das 100 instâncias de grafos aleatórios foram igualmente obtidas a partir do mesmo estudo ainda não publicado pelo Professor Orientador deste trabalho. A experiência do Professor Orientador deste trabalho - que já orientou e participou de outros estudos sobre o PCHG [7] [15], além de publicar a respeito de outros temas de estudo da Teoria de Grafos, como por exemplo o *Problema de Cobertura Máxima Local* [16] - foi fundamental para garantir uma maior congruência na condução desta etapa de teste.

Destaca-se que os valores padrão ou ótimos obtidos são para soluções ao PCG, ou seja, a comparação de resultados neste trabalho é feita entre os resultados da HG, que solucionou as instâncias colorindo-as nos padrões do PCHG, e os resultados padrão, que são a solução das instâncias para colorações nos padrões do PCG. Essa diferença, exigiu uma alteração estratégica no funcionamento da HG. Nela, a restrição do

PCHG que permite que um único par de vértices em todo o grafo receba a mesma cor de modo que a aresta que os liga possua o rótulo  $\{cor\ x, cor\ x\}$  foi retirada para a execução dos testes. Essa propriedade do PCHG é responsável por gerar uma coloração não-própria no grafo, conforme apresentado na seção Problemas de Coloração de Grafos. Com sua retirada, as colorações geradas pela HG passam a ser do tipo *próprias*, apesar de atenderem a todos os requisitos do PCHG. Esse novo comportamento da HG é idêntico ao esperado de um solucionador do PCHG com a diferença de ser mais estrito, na medida em que força com que todos os vértices adjacentes tenham cores distintas. Essa escolha estratégica favorece uma paridade mais realista e justa às comparações com as instâncias de teste utilizadas. Além disso, a comparação sendo feita nestas condições realça e evidencia ainda mais os melhores e os piores desempenhos que a HG alcançou.

No Algoritmo 2, a mudança necessária para implementar a retirada da restrição do PCHG explicada no parágrafo acima consiste em adicionar um parâmetro dentro da condicional presente entre as linhas 13 e 16, de modo que o código desta parte fica assim implementado:

---

```

if  $M_{(j,k.color)} = true$ 
  OR
   $M_{(k.color,j)} = true$ 
  OR
   $j = k.color$ 
then
   $found\_color\_flag \leftarrow false.$ 
  break while loop.
end if =0

```

▷ requisite 2 violated!

---

Para os leitores interessados em observar os resultados da HG quando a restrição de geração de colorações não-próprias é mantida, as tabelas com estes resultados específicos também encontram-se disponíveis no Apêndice A.

### C. Comparativo entre Soluções e Tempos

O comparativo entre as soluções obtidas encontra-se apresentado nas tabelas e gráficos abaixo e, por motivos de conveniência de leitura, outras tabelas e gráficos adicionais encontram-se nos Apêndices A e B.

Nos resultados abaixo dispostos, as soluções medem o número de cores obtidas para colorir os vértices e os tempos estão medidos em segundos. A lista abaixo apresenta o significado de alguns termos importantes:

- *Sol - HG* e *Time - HG*: representa as soluções e os tempos obtidos pela HG;
- *Sol - Std* e *Time - Std*: representa as soluções padrão e respectivos tempos;
- *Gap - Std*: representa a diferença relativa entre a solução padrão (a melhor já alcançada) e o valor ótimo conhecido, quando o *Gap* é 0 significa que o valor de solução é, de fato, o ótimo para aquela instância;

- %: representa a probabilidade definida para a densidade dos grafos aleatórios gerados;
- *TL*: indica que o tempo limite de 1800 segundos definido pelos autores das soluções padrão de referência foi extrapolado para obtenção da solução a que se refere.

#### 1) Instâncias aleatórias:

A apresentação dos resultados selecionados e respectivas análises de instâncias de grafos aleatórios é tema desta subseção. As instâncias aleatórias totalizaram uma quantidade de 500, compreendendo 5 versões de instâncias para cada uma das 10 quantidades de nós e respectivas 10 probabilidades. Em outras palavras, haviam 5 versões de instâncias distintas de 10 vértices e probabilidade 0.05, outras 5 versões distintas de 10 vértices e probabilidade 0.1, e assim por diante para todas as quantidades de nós e respectivas probabilidades. Ao final, obteve-se o resultado de todas as 500 instâncias executadas pela HG e calculou-se uma média de resultado e tempo obtido para cada tipo de instância de número de vértices e probabilidades diferentes. Obteve-se, portanto, 100 resultados compilados. Os valores padrão vinham em quantidade de 100 e já eram resultado da média das 5 versões distintas de cada tipo de instância.

TABELA I  
RESULTADOS DE INSTANCIAS ALEATÓRIAS  
DE PROBABILIDADE 0.05

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0,10	5,8	4,8	0	0,001	0,001
20	0,10	9,6	7,4	0	0,001	0,081
30	0,10	15,4	11	0	0,001	5,325
40	0,10	20,6	14,6	7,3	0,001	1096,094
50	0,10	26,8	19,4	10,8	0,001	1248,21
60	0,10	32,6	24,8	25,8	0,001	TL
70	0,10	37,4	30,2	35,3	0,001	TL
80	0,10	43,8	35,6	39,9	0,001	TL
90	0,10	49,6	41,4	36,3	0,001	TL
100	0,10	57,6	48,6	34,5	0,004	TL

TABELA II  
RESULTADOS DE INSTANCIAS ALEATÓRIAS  
DE PROBABILIDADE 0.1

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0,10	5	4,8	0	0,001	0,001
20	0,10	9,8	7,4	0	0,001	0,081
30	0,10	15,2	11	0	0,001	5,325
40	0,10	20,4	14,6	7,3	0,001	1096,094
50	0,10	26,4	19,4	10,8	0,001	1248,21
60	0,10	31,8	24,8	25,8	0,001	TL
70	0,10	37	30,2	35,3	0,001	TL
80	0,10	41,8	35,6	39,9	0,001	TL
90	0,10	49,8	41,4	36,3	0,001	TL
100	0,10	58	48,6	34,5	0,004	TL

As tabelas I a IV apresentadas são uma amostra dos resultados e tempos obtidos de instâncias de grafos aleatórios para algumas das probabilidades. Essa amostra seleciona alguns resultados para destacar detalhes interessantes no comparativo



TABELA III  
RESULTADOS DE INSTANCIAS ALEATORIAS  
DE PROBABILIDADE 0.3

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0,30	7,8	7,2	0	0,001	0,001
20	0,30	15,4	14,4	0	0,001	0,003
30	0,30	24,6	23,4	0	0,001	0,002
40	0,30	35,4	34,6	0	0,001	0,001
50	0,30	45,2	45,2	0	0,001	0,001
60	0,30	57	56,4	0	0,001	0,001
70	0,30	68,8	68,4	0	0,001	0,001
80	0,30	78,2	78,2	0	0,001	0,001
90	0,30	89,8	89,8	0	0,001	0,001
100	0,30	99,4	99,4	0	0,001	0,001

TABELA IV  
RESULTADOS DE INSTANCIAS ALEATORIAS  
DE PROBABILIDADE 0.9

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0,90	10	10	0	0,001	0
20	0,90	10	10	0	0,001	0
30	0,90	20	20	0	0,001	0
40	0,90	20	20	0	0,001	0,001
50	0,90	30	30	0	0,001	0,001
60	0,90	30	30	0	0,001	0,001
70	0,90	40	40	0	0,001	0,001
80	0,90	40	40	0	0,002	0,001
90	0,90	50	50	0	0,002	0,001
100	0,90	50	50	0	0,002	0,001

entre soluções e tempos de execução. Os dados de todas as instâncias encontram-se disponíveis no Apêndice A.

O leitor poderá notar que, nos casos das instâncias para as quais os valores de tempo *Time-Std* foram grandes ou tiveram uma grande discrepância com o tempo apresentado pela HG, os resultados apresentados pela HG para tais instâncias aparenta ser maior que os resultados padrão. Essa aparente discrepância, porém, advém tão somente do fator de escala dos gráficos, sem qualquer diferença real para os demais resultados cujos valores alcançados pela HG foram sempre muito próximos ou idênticos aos padrões.

Os gráficos a seguir apresentam os resultados obtidos para todas as probabilidades de grafos de uma mesma quantidade de vértices. Apenas alguns gráficos estão dispostos abaixo, sendo o restante apresentado complementarmente no Apêndice B. A seleção aqui visou destacar comportamentos interessantes apenas.

Os gráficos mostram um comportamento muito similar para todos os casos. Não foi diferente para com os outros apresentados no Apêndice B. Destaca-se que para nenhuma das instâncias a HG obteve colorações de número inferior aos valores padrão. Esse comportamento era esperado visto que as restrições sobre as quais a HG opera são mais rígidas que as do PCG.

Observou-se que a HG teve desempenho competitivo frente aos valores padrão disponíveis, estando, em 62% dos casos, equiparada a estes e, nos 38% restantes, demonstrando resultados piores que os padrões.

Solutions for random graphs of 10 nodes and 'p' probability

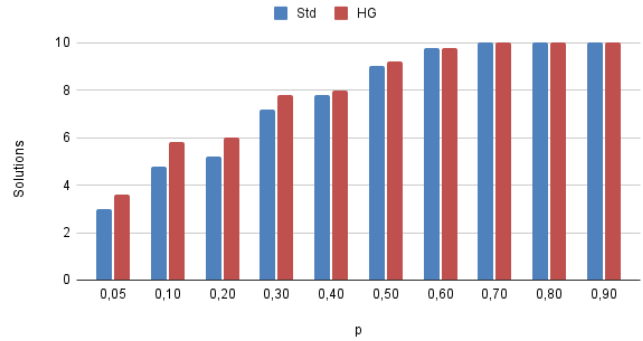


Fig. 4. Gráficos de soluções da HG para grafos de 10 vértices

Solutions for random graphs of 20 nodes and 'p' probability

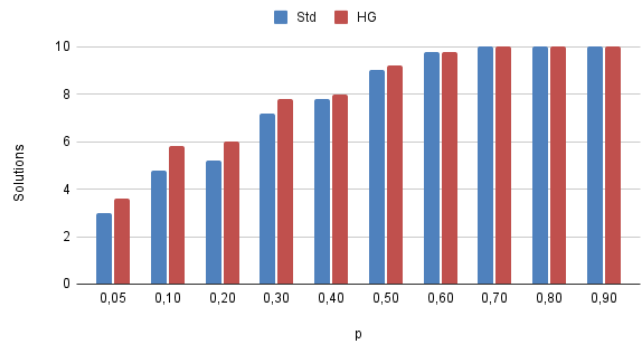


Fig. 5. Gráficos de soluções da HG para grafos de 20 vértices

Solutions for random graphs of 30 nodes and 'p' probability

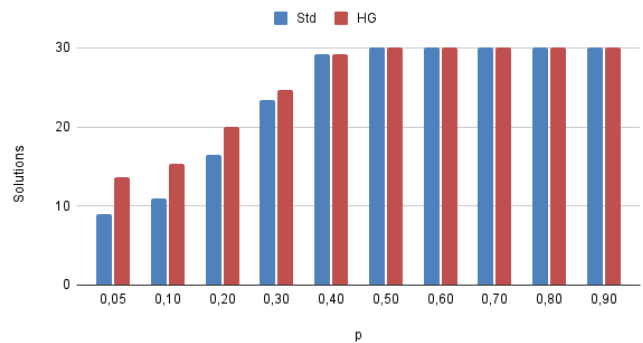


Fig. 6. Gráficos de soluções da HG para grafos de 30 vértices

## 2) Instâncias DIMACS [2]:

A apresentação de resultados selecionados e respectivas análises das instâncias DIMACS [2] é tema desta subseção.



Solutions for random graphs of 100 nodes and 'p' probability

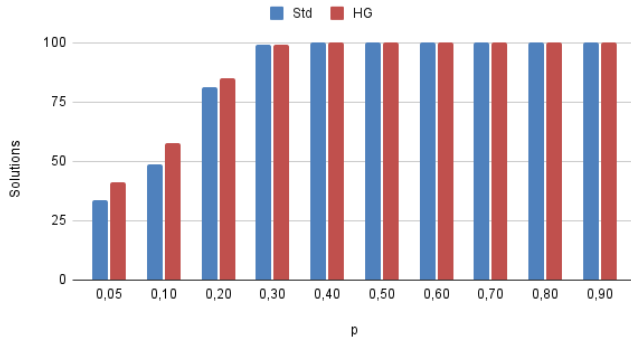


Fig. 7. Gráficos de soluções da HG para grafos de 100 vértices

Ao total, as instâncias *DIMACS* [2] para teste totalizaram 54. No entanto, conforme informado no início desta super-seção, apenas 30 destas instâncias contavam com valores padrão para comparação. Os resultados das outras 24 instâncias seguem apresentadas no Apêndice A. Seguem também apresentadas no Apêndice A a tabela com os resultados alcançados pela HG quando a restrição original do PCHG para geração de uma coloração não-própria é contemplada.

N° Colors

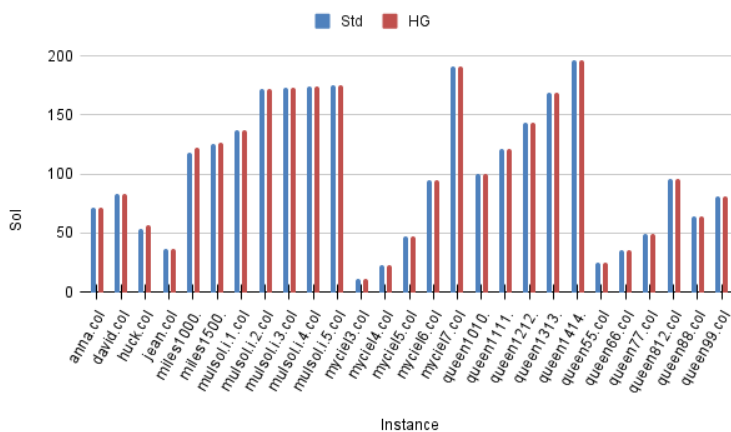


Fig. 8. Gráficos de soluções da HG para todas as instâncias *DIMACS* [2]

O desempenho da HG manteve-se pareado aos valores padrão para quase todas as instâncias. As exceções são para 3 casos em que a HG performou pior que os resultados padrão (apresentadas na tabela V). Conclui-se que para esse conjunto de instâncias, a HG tem quase sempre um desempenho igual às soluções padrão.

TABELA V  
INSTÂNCIAS *DIMACS* [2] PARA AS QUAIS A HG TEVE  
DESEMPENHO INFERIOR AOS VALORES PADRÃO

Instance	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
huck.col	57	54	0,0	0,001	4,627
miles1000.col	122	118	23,7	0,004	TL
miles1500.col	127	126	0,0	0,005	19,976

## V. CONCLUSÃO E TRABALHOS FUTUROS

### A. Conclusão

Neste trabalho abordou-se o Problema de Coloração Harmônica de Grafos (PCHG), propondo-se uma abordagem heurística gulosa a partir de estudos da literatura corrente sobre o PCHG. Foi realizado de um estudo dos fundamentos teóricos relacionados a Teoria de Grafos e, mais especificamente, o Problema de Coloração de Grafos e o Problema de Coloração Harmônica de Grafos, assim como um levantamento bibliográfico das estratégias computacionais seminais e atuais aplicadas ao PCHG. Foi implementado uma Heurística Gulosa (HG) dedicada ao PCHG, aplicando-se as restrições deste tipo de coloração diretamente ao funcionamento dos algoritmos que compõem a heurística, culminando em complexidades assintóticas de tempo e espaço de  $O(N^2)$ . Testes foram realizados com dois conjuntos diferentes de instâncias distintas de grafos e, em sequência, foi feita uma avaliação do desempenho e comportamento da HG em comparação com valores padrão ou ótimos obtidos para quase todas as instâncias de teste utilizadas. Os resultados dos testes indicam que o desempenho da HG se iguala aos valores padrões na maioria dos casos, obtendo alguns resultados melhores e outros piores que os valores de referência. Nas instâncias de grafos aleatórios, a HG apresentou resultados iguais ou melhores que os padrões em 62% dos casos ao passo que, nas instâncias *DIMACS* [2], esse mesmo desempenho foi observado em 90% dos casos.

### B. Trabalhos Futuros

Para futuros trabalhos, novos métodos de ordenação de vértices podem ser testados no intuito de melhorar o desempenho da heurística apresentada neste trabalho para as instâncias em que a HG foi pior que os valores padrão. Outra estratégia pode ser a combinação de métodos de ordenação na busca pelos melhores resultados finais. Heurísticas dedicadas ao PCHG que implementam outras abordagens no lugar da gulosa também são propostas interessantes de serem adotadas em futuros estudos.

## AGRADECIMENTOS

Agradeço ao Professor Orientador deste trabalho Márcio Santos pela constante paciência, cuidado, compreensão e suporte a mim ofertado durante todo o processo de orientação. Agradeço ao Departamento de Ciências da Computação e à Universidade Federal de Minas Gerais pela oportunidade e facilitação de meu desenvolvimento e aprendizado. Agradeço aos meus pais, irmão e minha noiva pelo compreensivo amparo em que me envolveram por toda a vida. A Deus agradeço por tudo e todos em meu caminho.

## REFERÊNCIAS

- [1] G. Chartrand and P. Zhang, *Chromatic Graph Theory*. London, England: CRC Press, 2019.
- [2] Dimacs, 'DIMACS Implementation Challenges', 1995. [Online]. Available: <http://dimacs.rutgers.edu/Challenges/>.
- [3] N. Barnier and P. Brisset, 'Graph Coloring for Air Traffic Flow Management', *Annals of Operations Research*, vol. 130, no. 1, pp. 163–178, Aug. 2004.
- [4] J. E. Hopcroft and M. S. Krishnamoorthy, 'On the harmonious coloring of graphs', *SIAM J. Algebr. Discrete Methods*, vol. 4, no. 3, pp. 306–311, Sep. 1983.
- [5] T. R. Jensen and B. Toft, *Graph Coloring Problems*. New York: Wiley-Interscience, 2011.
- [6] M. V. Kavitha and R. Govindarajan, 'On A Harmonious Colouring Graphs And Its Applications', *IOSR Journal of Mathematics*, vol. 4, pp. 13–17, 2012.
- [7] H. C. de Oliveira, *Coloração harmônica e grafos: uma abordagem utilizando programação inteira*. 2019.
- [8] A. Ostroski and L. Menoncin, 'Aplicações práticas da teoria dos grafos', Pato Branco: XIII ERMAC, 2009.
- [9] M. R. Vieira, *Heurística matemática aplicada ao problema da coloração de grafos*. 2018.
- [10] D. Brélaz, 'New methods to color the vertices of a graph', *Commun. ACM*, vol. 22, no. 4, pp. 251–256, Apr. 1979.
- [11] R. Lewis, *Guide to graph colouring*, 2nd ed. Cham, Switzerland: Springer Nature, 2021.
- [12] M. Kubale, Ed., *Graph Colorings*. Providence, RI: American Mathematical Society, 2004.
- [13] R. Lewis, 'DSatur Algorithm for Graph Coloring - GeeksforGeeks — [geeksforgeeks.org](https://www.geeksforgeeks.org/)', 2023. [Online]. Available: <https://www.geeksforgeeks.org/dsatur-algorithm-for-graph-coloring/>.
- [14] N. Alon, 'Restricted colorings of graphs. Surveys in combinatorics, v. 187, p. 1–33, 1993.
- [15] J. Araújo, A. Martins, and M. Santos, "Coloracao Harmoniosa", in *Anais do VII Encontro de Teoria da Computação, Niterói, 2022*, pp. 121–124, doi: <https://doi.org/10.5753/etc.2022.223216>.
- [16] M. C. Santos, H. Luss, D. Nace, and M. Poss, 'Proportional and maxmin fairness for the sensor location problem with chance constraints', *Discrete Appl. Math.*, vol. 261, pp. 316–331, May 2019.
- [17] J. Mitchem, 'On the harmonious chromatic number of a graph', *Discret. Math.*, vol. 74, pp. 151–157, 1989.
- [18] K. Edwards, 'A new upper bound for the harmonious chromatic number', *J. Graph Theory*, vol. 29, no. 4, pp. 257–261, Dec. 1998.
- [19] I. Krasikov and Y. Roditty, 'Bounds for the harmonious chromatic number of a graph', *J. Graph Theory*, vol. 18, pp. 205–209, 1994.
- [20] S.-M. Lee and J. Mitchem, 'An upper bound for the harmonious chromatic number of a graph', *J. Graph Theory*, vol. 11, no. 4, pp. 565–567, Dec. 1987.
- [21] Z. Lu, 'On an upper bound for the harmonious chromatic number of a graph', *J. Graph Theory*, vol. 15, no. 4, pp. 345–347, Sep. 1991.
- [22] Z. Miller and D. Pritikin, 'The harmonious coloring number of a graph', *Discrete Math.*, vol. 93, no. 2–3, pp. 211–228, Nov. 1991.
- [23] M. Huilgol and S. Vaidyanathan, 'On the harmonious coloring of certain classes of graphs', *Journal of Combinatorics, Information System Sciences*, vol. 41, pp. 17–29, 01 2016.
- [24] R. Marinescu-Ghemeci, C. Obreja, and A. Popa, 'Approximate and exact results for the harmonious chromatic number', 2021.
- [25] J.-B. Liu, M. Arockiaraj, and A. Nelson, 'Tight bounds on 1-harmonious coloring of certain graphs', *Symmetry (Basel)*, vol. 11, no. 7, p. 917, Jul. 2019.
- [26] S. Czerwiński, 'On harmonious coloring of hypergraphs', 2023.
- [27] G. A. Neufeld and J. Tartar, 'Graph coloring conditions for the existence of solutions to the timetable problem', *Commun. ACM*, vol. 17, no. 8, pp. 450–453, Aug. 1974.
- [28] N. Zufferey, 'Graph coloring Tabu search for project scheduling', in *Advanced Computational Methods for Knowledge Engineering*, Cham: Springer International Publishing, 2015, pp. 107–118.
- [29] M. P. Lima, T. B. Rodrigues, R. F. Alexandre, R. H. C. Takahashi, and E. G. Carrano, 'Using evolutionary algorithms for channel assignment in 802.11 networks', in *2014 IEEE Symposium on Computational Intelligence for Communication Systems and Networks (CICOMMS)*, Orlando, FL, USA, 2014.
- [30] A. Kouider, H. A. Haddadene, S. Ourari, and A. Oulamara, 'Mixed integer linear programs and tabu search approach to solve mixed graph coloring for unit-time job shop scheduling', in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Gothenburg, Sweden, 2015.
- [31] A. Drexel and S. Knust, 'Sports league scheduling: Graph- and resource-based models', *Omega*, vol. 35, no. 5, pp. 465–471, Oct. 2007.

APÊNDICE A  
TABELAS

TABELA VI  
ESPECIFICAÇÕES DETALHADAS DA MÁQUINA USADA PARA EXECUÇÃO  
DA HG E INSTÂNCIAS DE TESTES APRESENTADAS NA SUBSEÇÃO *Instâncias*

<b>Memoria:</b>	15.5 GiB
<b>Processador:</b>	Intel® Core™ i7-6700 CPU @ 3.40GHz × 8
<b>Gráficos:</b>	Mesa Intel® HD Graphics 530 (SKL GT2)
<b>Capacidade de Disco:</b>	1.0 TB
<b>SO:</b>	Ubuntu 20.04.2 LTS
<b>Arquitetura:</b>	x86_64
<b>Modo(s) operacional da CPU:</b>	32-bit, 64-bit
<b>Ordem dos bytes:</b>	Little Endian
<b>Address sizes:</b>	39 bits physical, 48 bits virtual
<b>CPU(s):</b>	8
<b>Lista de CPU(s) on-line:</b>	0-7
<b>Thread(s) per núcleo:</b>	2
<b>Núcleo(s) por soquete:</b>	4
<b>Soquete(s):</b>	1
<b>Nó(s) de NUMA:</b>	1
<b>ID de fornecedor:</b>	GenuineIntel
<b>Família da CPU:</b>	6
<b>Modelo:</b>	94
<b>Nome do modelo:</b>	Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz
<b>Step:</b>	3
<b>CPU MHz:</b>	3400.000
<b>CPU MHz máx.:</b>	4000.0000
<b>CPU MHz mín.:</b>	800.0000
<b>BogoMIPS:</b>	6799.81
<b>Virtualização:</b>	VT-x
<b>cache de L1d:</b>	128 KiB
<b>cache de L1i:</b>	128 KiB
<b>cache de L2:</b>	1 MiB
<b>cache de L3:</b>	8 MiB
<b>CPU(s) de nó NUMA:</b>	0-7

TABELA VII  
 RESULTADOS COMPARATIVOS PARA INSTÂNCIAS *DIMACS* [2]

<b>Instance</b>	<b>Sol - HG</b>	<b>Sol - Std</b>	<b>Gap - Std</b>	<b>Time - HG</b>	<b>Time - Std</b>
anna.col	72	72	0.0	0.001	23.774
david.col	83	83	0.0	0.001	0.061
games120.col	66	0	100.0	0.001	TL
huck.col	57	54	0.0	0.001	4.627
jean.col	37	37	0.0	0.001	22.437
miles1000.col	122	118	23.7	0.004	TL
miles1500.col	127	126	0.0	0.005	19.976
miles250.col	72	0	100.0	0.001	TL
miles500.col	86	0	100.0	0.002	TL
miles750.col	108	0	100.0	0.002	TL
multsol.i.1.col	137	137	0.0	0.002	0.558
multsol.i.2.col	172	172	0.0	0.002	0.101
multsol.i.3.col	173	173	0.0	0.002	0.066
multsol.i.4.col	174	174	0.0	0.002	0.066
multsol.i.5.col	175	175	0.0	0.002	0.067
myciel3.col	11	11	0.0	0.001	0.000
myciel4.col	23	23	0.0	0.001	0.000
myciel5.col	47	47	0.0	0.001	0.000
myciel6.col	95	95	0.0	0.001	0.000
myciel7.col	191	191	0.0	0.002	0.001
queen1010.col	100	100	0.0	0.002	0.000
queen1111.col	121	121	0.0	0.002	0.000
queen1212.col	144	144	0.0	0.002	0.001
queen1313.col	169	169	0.0	0.003	0.001
queen1414.col	196	196	0.0	0.004	0.001
queen55.col	25	25	0.0	0.001	0.000
queen66.col	36	36	0.0	0.001	0.000
queen77.col	49	49	0.0	0.001	0.000
queen812.col	96	96	0.0	0.002	0.000
queen88.col	64	64	0.0	0.001	0.000
queen99.col	81	81	0.0	0.002	0.001

TABELA VIII  
 RESULTADOS COMPARATIVOS PARA INSTÂNCIAS ALEATÓRIAS DE GRAFOS

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std	V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0.05	3.6	3.0	0.0	0.001	0.001	10	0.50	9.2	9	0	0.001	0.001
20	0.05	7.6	5.6	0.0	0.001	0.071	20	0.50	19.6	19.6	0	0.001	0.001
30	0.05	13.6	9.0	0.0	0.001	101.553	30	0.50	30	30	0	0.001	0
40	0.05	18.2	10.6	13.3	0.001	TL	40	0.50	40	40	0	0.001	0.001
50	0.05	22.2	13.8	24.7	0.001	TL	50	0.50	50	50	0	0.001	0.001
60	0.05	25	16.6	36.9	0.001	TL	60	0.50	60	60	0	0.001	0.001
70	0.05	29.2	20.4	46.0	0.001	TL	70	0.50	70	70	0	0.001	0.001
80	0.05	34.4	26.0	56.1	0.001	TL	80	0.50	80	80	0	0.001	0.001
90	0.05	40.2	30.0	53.9	0.001	TL	90	0.50	90	90	0	0.002	0.001
100	0.05	41.2	33.6	70.1	0.003	TL	100	0.50	100	100	0	0.002	0.001
10	0.10	5.8	4.8	0.0	0.001	0.001	10	0.60	9.8	9.8	0	0.001	0
20	0.10	9.6	7.4	0.0	0.001	0.081	20	0.60	20	20	0	0.001	0
30	0.10	15.4	11.0	0.0	0.001	5.325	30	0.60	30	30	0	0.001	0
40	0.10	20.6	14.6	7.3	0.001	1096.094	40	0.60	40	40	0	0.001	0.001
50	0.10	26.8	19.4	10.8	0.001	1248.21	50	0.60	50	50	0	0.001	0.001
60	0.10	32.6	24.8	25.8	0.001	TL	60	0.60	60	60	0	0.001	0.001
70	0.10	37.4	30.2	35.3	0.001	TL	70	0.60	70	70	0	0.001	0.001
80	0.10	43.8	35.6	39.9	0.001	TL	80	0.60	80	80	0	0.001	0.001
90	0.10	49.6	41.4	36.3	0.001	TL	90	0.60	90	90	0	0.001	0.001
100	0.10	57.6	48.6	34.5	0.004	TL	100	0.60	100	100	0	0.002	0.001
10	0.20	6	5.2	0.0	0.001	0.002	10	0.70	10	10	0	0.001	0
20	0.20	12.6	10.8	0.0	0.001	0.041	20	0.70	20	20	0	0.001	0
30	0.20	20	16.4	0.0	0.001	0.67	30	0.70	30	30	0	0.001	0
40	0.20	27.2	22.8	0.0	0.001	1.979	40	0.70	40	40	0	0.001	0
50	0.20	36	29.6	0.0	0.001	0.963	50	0.70	50	50	0	0.001	0.001
60	0.20	45.4	40.6	0.0	0.001	0.211	60	0.70	60	60	0	0.001	0.001
70	0.20	54.8	48.0	0.0	0.001	0.295	70	0.70	70	70	0	0.001	0.001
80	0.20	62.4	58.0	0.0	0.001	0.101	80	0.70	80	80	0	0.001	0.001
90	0.20	73	69.6	0.0	0.001	0.011	90	0.70	90	90	0	0.002	0.001
100	0.20	85.2	81.4	0.0	0.001	0.027	100	0.70	100	100	0	0.002	0.001
10	0.30	7.8	7.2	0.0	0.001	0.001	10	0.80	10	10	0	0.001	0
20	0.30	15.4	14.4	0.0	0.001	0.003	20	0.80	20	20	0	0.001	0
30	0.30	24.6	23.4	0.0	0.001	0.002	30	0.80	30	30	0	0.001	0
40	0.30	35.4	34.6	0.0	0.001	0.001	40	0.80	40	40	0	0.001	0
50	0.30	45.2	45.2	0.0	0.001	0.001	50	0.80	50	50	0	0.001	0.001
60	0.30	57	56.4	0.0	0.001	0.001	60	0.80	60	60	0	0.001	0.001
70	0.30	68.8	68.4	0.0	0.001	0.001	70	0.80	70	70	0	0.001	0.001
80	0.30	78.2	78.2	0.0	0.001	0.001	80	0.80	80	80	0	0.001	0.001
90	0.30	89.8	89.8	0.0	0.001	0.001	90	0.80	90	90	0	0.002	0.001
100	0.30	99.4	99.4	0.0	0.001	0.001	100	0.80	100	100	0	0.003	0.001
10	0.40	8	7.8	0.0	0.001	0.001	10	0.90	10	10	0	0.001	0
20	0.40	19	19.0	0.0	0.001	0.001	20	0.90	20	20	0	0.001	0
30	0.40	29.2	29.2	0.0	0.001	0.001	30	0.90	30	30	0	0.001	0
40	0.40	39.6	39.6	0.0	0.001	0.001	40	0.90	40	40	0	0.001	0.001
50	0.40	49.6	49.6	0.0	0.001	0.001	50	0.90	50	50	0	0.001	0.001
60	0.40	60	60.0	0.0	0.001	0.001	60	0.90	60	60	0	0.001	0.001
70	0.40	70	70.0	0.0	0.001	0.001	70	0.90	70	70	0	0.001	0.001
80	0.40	80	80.0	0.0	0.001	0.001	80	0.90	80	80	0	0.002	0.001
90	0.40	90	90.0	0.0	0.001	0.001	90	0.90	90	90	0	0.002	0.001
100	0.40	100	100.0	0.0	0.001	0.001	100	0.90	100	100	0	0.002	0.001

TABELA IX  
 RESULTADOS COMPARATIVOS PARA INSTÂNCIAS *DIMACS* [2] NO CONTEXTO  
 EM QUE A RESTRIÇÃO PARA GERAÇÃO DE COLORAÇÕES NÃO-PRÓPRIAS FOI  
 MANTIDA DENTRO DO FUNCIONAMENTO DA HG

Instance	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
anna.col	71	72	0.0	0.001	23.774
david.col	82	83	0.0	0.001	0.061
games120.col	66	0	100.0	0.001	TL
huck.col	56	54	0.0	0.001	4.627
jean.col	36	37	0.0	0.001	22.437
miles1000.col	122	118	23.7	0.004	TL
miles1500.col	127	126	0.0	0.005	19.976
miles250.col	71	0	100.0	0.001	TL
miles500.col	86	0	100.0	0.002	TL
miles750.col	108	0	100.0	0.002	TL
mulsol.i.1.col	137	137	0.0	0.002	0.558
mulsol.i.2.col	172	172	0.0	0.002	0.101
mulsol.i.3.col	173	173	0.0	0.002	0.066
mulsol.i.4.col	174	174	0.0	0.002	0.066
mulsol.i.5.col	175	175	0.0	0.002	0.067
myciel3.col	8	11	0.0	0.001	0.000
myciel4.col	19	23	0.0	0.001	0.000
myciel5.col	42	47	0.0	0.001	0.000
myciel6.col	89	95	0.0	0.001	0.000
myciel7.col	184	191	0.0	0.002	0.001
queen1010.col	100	100	0.0	0.002	0.000
queen1111.col	121	121	0.0	0.002	0.000
queen1212.col	144	144	0.0	0.002	0.001
queen1313.col	169	169	0.0	0.003	0.001
queen1414.col	196	196	0.0	0.004	0.001
queen55.col	25	25	0.0	0.001	0.000
queen66.col	36	36	0.0	0.001	0.000
queen77.col	49	49	0.0	0.001	0.000
queen812.col	96	96	0.0	0.002	0.000
queen88.col	64	64	0.0	0.001	0.000
queen99.col	81	81	0.0	0.002	0.001

TABELA X

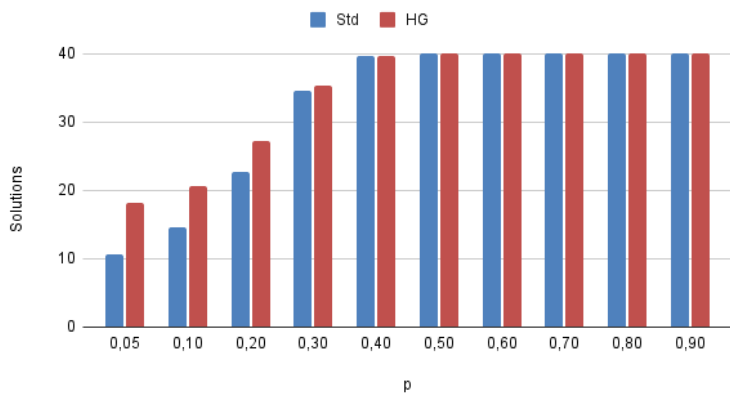
RESULTADOS COMPARATIVOS PARA INSTÂNCIAS ALEATÓRIAS DE GRAFOS PARA O CONTEXTO EM QUE A RESTRIÇÃO PARA GERAÇÃO DE COLORAÇÕES NÃO-PRÓPRIAS FOI MANTIDA DENTRO DO FUNCIONAMENTO DA HG

V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std	V(G)	%	Sol - HG	Sol - Std	Gap - Std	Time - HG	Time - Std
10	0.05	3	3.0	0.000	0.001	0.001	10	0.50	8.4	9	0	0.001	0.001
20	0.05	7	5.6	0.000	0.001	0.071	20	0.50	19.2	19.6	0	0.001	0.001
30	0.05	13	9.0	0.000	0.001	101.553	30	0.50	29.8	30	0	0.001	0
40	0.05	19	10.6	13.300	0.001	TL	40	0.50	40	40	0	0.001	0.001
50	0.05	23	13.8	24.700	0.001	TL	50	0.50	50	50	0	0.001	0.001
60	0.05	26	16.6	36.900	0.001	TL	60	0.50	60	60	0	0.001	0.001
70	0.05	30	20.4	46.000	0.001	TL	70	0.50	70	70	0	0.001	0.001
80	0.05	35	26.0	56.100	0.001	TL	80	0.50	80	80	0	0.001	0.001
90	0.05	40	30.0	53.900	0.001	TL	90	0.50	90	90	0	0.002	0.001
100	0.05	43	33.6	70.100	0.003	TL	100	0.50	100	100	0	0.002	0.001
10	0.10	5	4.8	0.000	0.001	0.001	10	0.60	9.6	9.8	0	0.001	0
20	0.10	10	7.4	0.000	0.001	0.081	20	0.60	20	20	0	0.001	0
30	0.10	15	11.0	0.000	0.001	5.325	30	0.60	30	30	0	0.001	0
40	0.10	20	14.6	7.300	0.001	1096.094	40	0.60	40	40	0	0.001	0.001
50	0.10	26	19.4	10.800	0.001	1248.21	50	0.60	50	50	0	0.001	0.001
60	0.10	32	24.8	25.800	0.001	TL	60	0.60	60	60	0	0.001	0.001
70	0.10	37	30.2	35.300	0.001	TL	70	0.60	70	70	0	0.001	0.001
80	0.10	42	35.6	39.900	0.001	TL	80	0.60	80	80	0	0.001	0.001
90	0.10	50	41.4	36.300	0.001	TL	90	0.60	90	90	0	0.001	0.001
100	0.10	58	48.6	34.500	0.004	TL	100	0.60	100	100	0	0.002	0.001
10	0.20	5	5.2	0.000	0.001	0.002	10	0.70	10	10	0	0.001	0
20	0.20	12	10.8	0.000	0.001	0.041	20	0.70	20	20	0	0.001	0
30	0.20	19	16.4	0.000	0.001	0.67	30	0.70	30	30	0	0.001	0
40	0.20	27	22.8	0.000	0.001	1.979	40	0.70	40	40	0	0.001	0
50	0.20	35	29.6	0.000	0.001	0.963	50	0.70	50	50	0	0.001	0.001
60	0.20	44	40.6	0.000	0.001	0.211	60	0.70	60	60	0	0.001	0.001
70	0.20	53	48.0	0.000	0.001	0.295	70	0.70	70	70	0	0.001	0.001
80	0.20	61	58.0	0.000	0.001	0.101	80	0.70	80	80	0	0.001	0.001
90	0.20	72	69.6	0.000	0.001	0.011	90	0.70	90	90	0	0.002	0.001
100	0.20	85	81.4	0.000	0.001	0.027	100	0.70	100	100	0	0.002	0.001
10	0.30	7	7.2	0.000	0.001	0.001	10	0.80	10	10	0	0.001	0
20	0.30	14.6	14.4	0	0.001	0.003	20	0.80	20	20	0	0.001	0
30	0.30	23.6	23.4	0	0.001	0.002	30	0.80	30	30	0	0.001	0
40	0.30	35.2	34.6	0	0.001	0.001	40	0.80	40	40	0	0.001	0
50	0.30	44.4	45.2	0	0.001	0.001	50	0.80	50	50	0	0.001	0.001
60	0.30	56.8	56.4	0	0.001	0.001	60	0.80	60	60	0	0.001	0.001
70	0.30	68	68.4	0	0.001	0.001	70	0.80	70	70	0	0.001	0.001
80	0.30	78	78.2	0	0.001	0.001	80	0.80	80	80	0	0.001	0.001
90	0.30	89.4	89.8	0	0.001	0.001	90	0.80	90	90	0	0.002	0.001
100	0.30	98.8	99.4	0	0.001	0.001	100	0.80	100	100	0	0.003	0.001
10	0.40	7.6	7.8	0	0.001	0.001	10	0.90	10	10	0	0.001	0
20	0.40	18	19	0	0.001	0.001	20	0.90	20	20	0	0.001	0
30	0.40	28	29.2	0	0.001	0.001	30	0.90	30	30	0	0.001	0
40	0.40	39.6	39.6	0	0.001	0.001	40	0.90	40	40	0	0.001	0.001
50	0.40	49.4	49.6	0	0.001	0.001	50	0.90	50	50	0	0.001	0.001
60	0.40	60	60	0	0.001	0.001	60	0.90	60	60	0	0.001	0.001
70	0.40	70	70	0	0.001	0.001	70	0.90	70	70	0	0.001	0.001
80	0.40	80	80	0	0.001	0.001	80	0.90	80	80	0	0.002	0.001
90	0.40	90	90	0	0.001	0.001	90	0.90	90	90	0	0.002	0.001
100	0.40	100	100	0	0.001	0.001	100	0.90	100	100	0	0.002	0.001



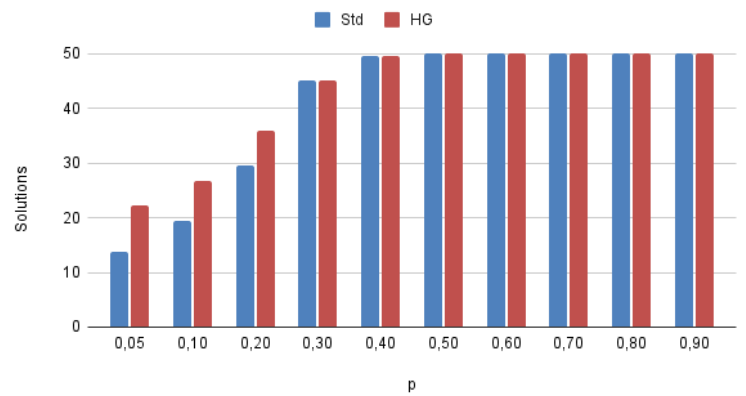
APÊNDICE B  
GRÁFICOS

Solutions for random graphs of 40 nodes and 'p' probability



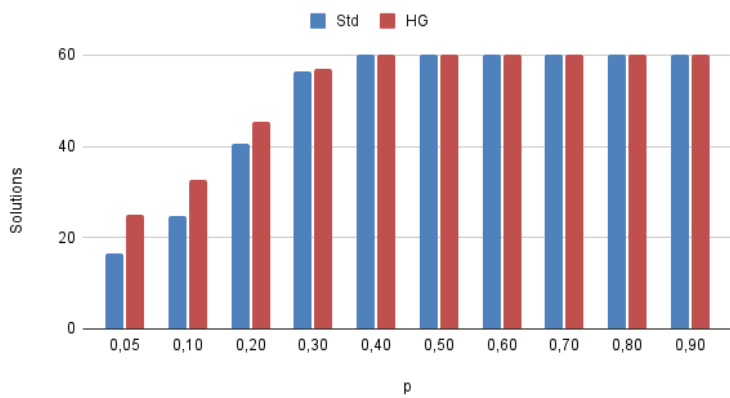
(a) Fig 9. Gráficos de soluções da HG para grafos de 40 vértices

Solutions for random graphs of 50 nodes and 'p' probability



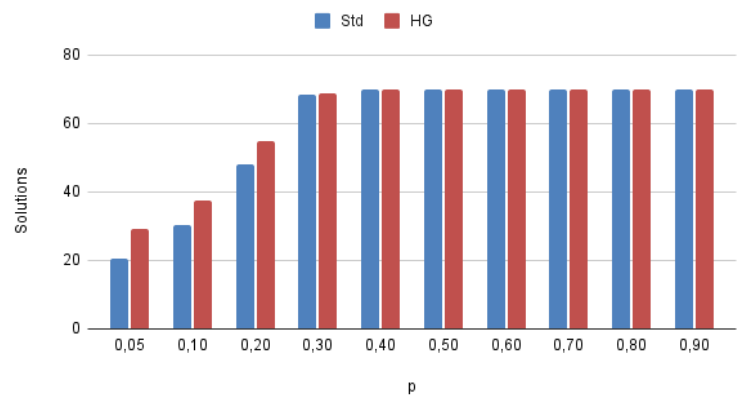
(b) Fig 10. Gráficos de soluções da HG para grafos de 50 vértices

Solutions for random graphs of 60 nodes and 'p' probability



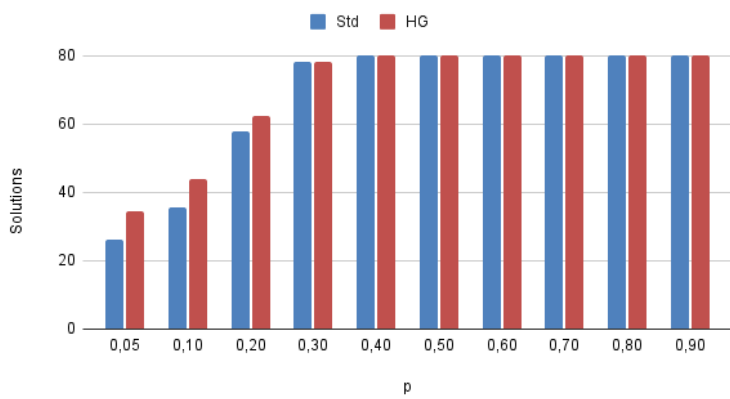
(c) Fig 11. Gráficos de soluções da HG para grafos de 60 vértices

Solutions for random graphs of 70 nodes and 'p' probability



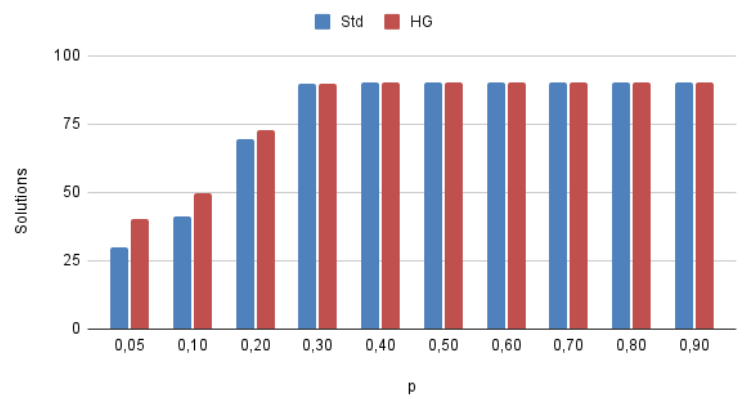
(d) Fig 12. Gráficos de soluções da HG para grafos de 70 vértices

Solutions for random graphs of 80 nodes and 'p' probability



(e) Fig 13. Gráficos de soluções da HG para grafos de 80 vértices

Solutions for random graphs of 90 nodes and 'p' probability



(f) Fig 14. Gráficos de soluções da HG para grafos de 90 vértices

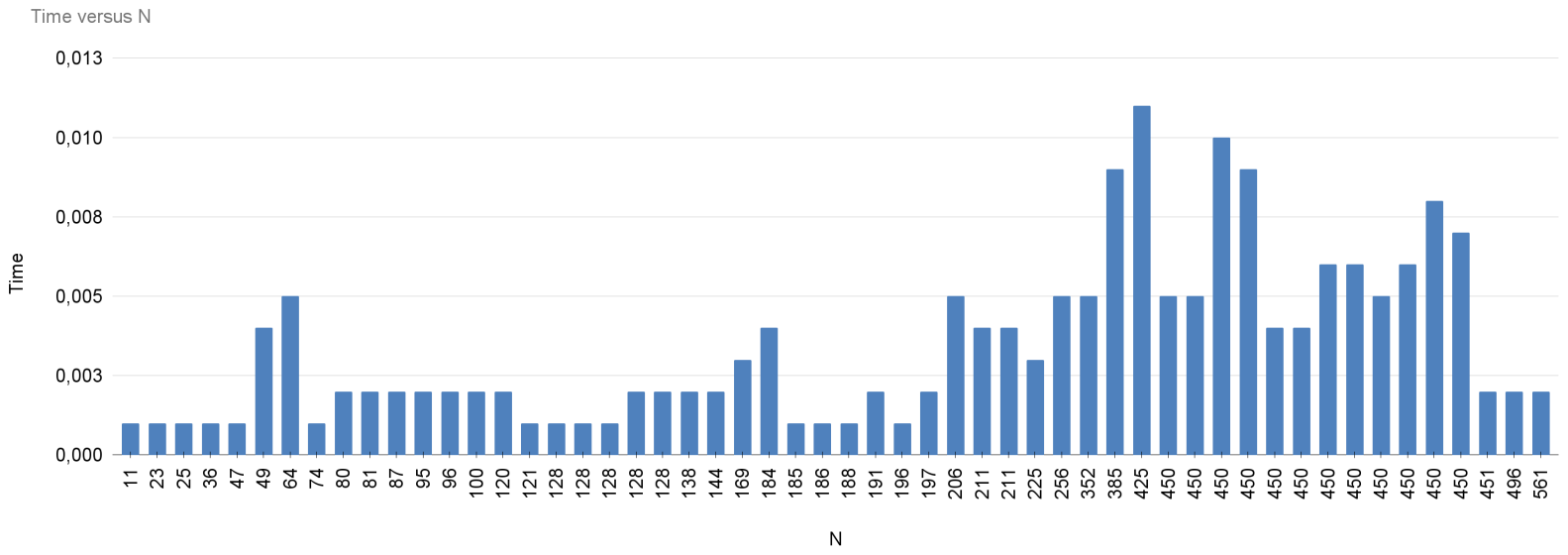


Fig. 9. Gráfico comparativo do tempo gasto pela HG na execução das instâncias *DIMACS* [2] utilizando a mesma base do gráfico abaixo (Fig. 10)

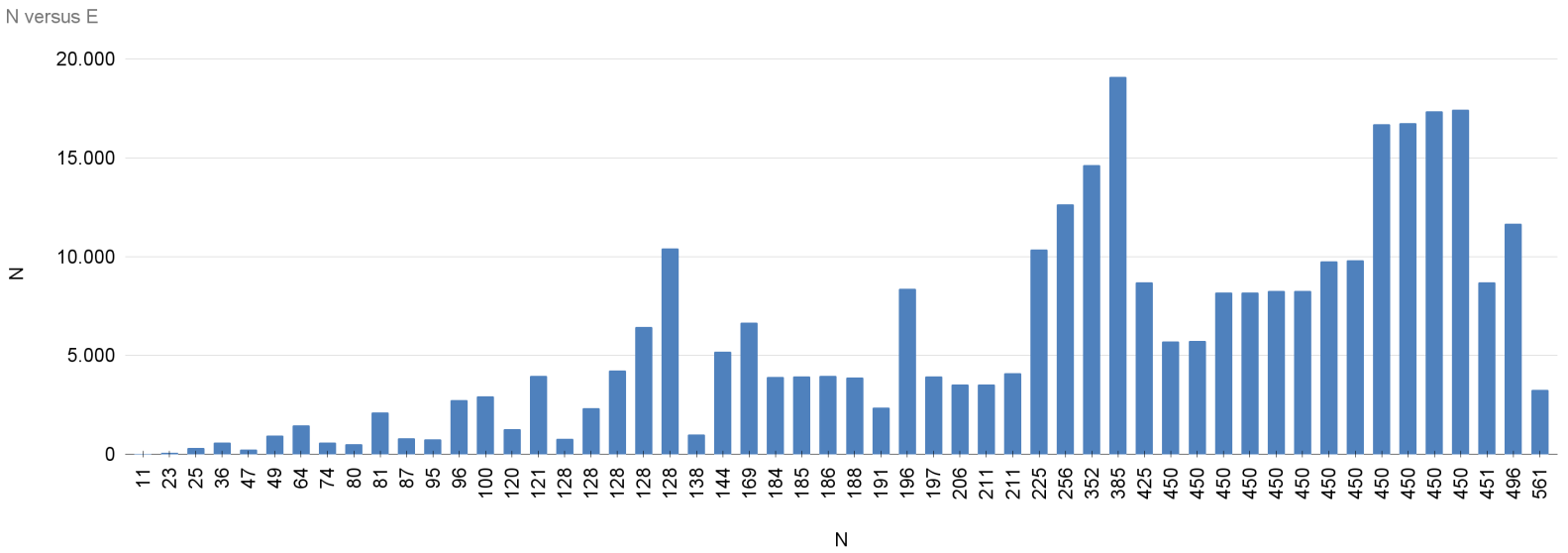


Fig. 10. Gráfico comparativo da quantidade de arestas de cada instância *DIMACS* [2] utilizando a mesma base do gráfico acima (Fig. 9)