

5

Algoritmo Exato para Coloração de Partições

Este capítulo apresenta o primeiro algoritmo exato para o Problema de Coloração de Partições - PCP (do inglês *Partition Coloring Problem*) e sua aplicação na resolução de instâncias de min-RWA. Primeiramente, são apresentados trabalhos relacionados ao Problema de Coloração de Grafos - GCP (do inglês *Graph Coloring Problem*), que é uma especialização do PCP. Em seguida, as melhores estratégias encontradas na literatura de GCP são aplicadas para desenvolver um algoritmo exato para PCP. Esta pesquisa foi realizada em conjunto com Yuri A. Frota e também é apresentada em sua tese de doutorado [37].

5.1

Algoritmos para Coloração de Grafos

O Problema de Coloração de Grafos é bastante estudado na literatura. Dado um grafo não direcionado $G = (V, E)$, onde V é o conjunto de vértices e E é o conjunto de arestas, o GCP consiste em atribuir uma cor para cada vértice em V de modo que dois vértices que compartilham alguma aresta tenham cores diferentes. O objetivo do GCP é minimizar o número total de cores utilizadas para colorir todos os vértices de G .

A versão de decisão de GCP é um problema NP-completo [40]. Várias classes de problemas reais podem ser modelados como extensões do GCP, tais como problemas de construção de tabelas [18] e de alocação de frequência [26]. Devido à dificuldade de se resolver este problema de forma exata, a maioria dos trabalhos apresentados na literatura sugerem o uso de heurísticas [21, 35, 38, 39, 46, 53, 60, 62, 77].

Os primeiros trabalhos desenvolvidos para resolver GCP de forma exata propunham algoritmos enumerativos [10, 11, 12, 56, 58]. Entretanto, estes se mostraram ineficientes para resolver instâncias de médio porte. Os melhores resultados obtidos para o GCP, até o momento, são baseados em programação linear inteira. Mehrotra e Trick [67] propuseram um algoritmo de geração de colunas, enquanto Figueiredo [34] e Méndez-Díaz e Zabala [68] desenvolveram algoritmos do tipo *branch-and-cut*.

Recentemente, Campelo et al. [25] propuseram uma nova formulação para o GCP denominada *formulação por representantes*, que atenua a simetria tipicamente encontrada em formulações para GCP (exceção feita à formulação utilizada em [67]). Em vez de associar cores a vértices, esta formulação escolhe um vértice para ser o *representante* de todos os vértices coloridos com a mesma cor. Portanto, existem dois estados distintos para cada vértice: ou ele representa sua própria cor ou existe um outro vértice que representa a sua cor. Uma formulação por representantes assimétrica e desigualdades válidas para GCP foram propostas em [17]. Resultados experimentais em [24] mostraram que a formulação assimétrica por representantes obtém limites inferiores muito próximos da solução ótima para várias instâncias de GCP, especialmente aquelas onde o tamanho da clique máxima se aproxima do número cromático do grafo.

Seja um grafo não-direcionado $G = (V, E)$, como previamente definido. Dado um vértice $u \in V$, sua *anti-vizinhança* $A(u) = \{w \in V : (u, w) \notin E, w \neq u\}$ é definida como o conjunto dos vértices que não lhe são adjacentes em G . Por conveniência, também define-se $A'(u) = A(u) \cup \{u\}$. Além disso, denota-se por $E[S]$ o subconjunto de arestas de G que são induzidas pelo subconjunto $S \subseteq V$. Um vértice $v \in A(u)$ é dito *isolado* em $A(u)$ se $E[A(u)] = E[A(u) \setminus \{v\}]$, ou seja, o vértice v não tem vizinhos em $A(u)$.

A formulação por representantes do GCP utiliza variáveis binárias $y_{uv} = 1$ se o vértice u representa a cor do vértice v ; caso contrário $y_{uv} = 0$. Quando $y_{uu} = 1$ diz-se que o vértice u representa sua própria cor ou que u é um vértice representante. A formulação por representantes para coloração de grafos é dada pelo modelo:

$$\min \sum_{u \in V} y_{uu} \quad (5-1)$$

sujeito a:

$$\sum_{v \in A'(u)} y_{vu} = 1 \quad \forall u \in V \quad (5-2)$$

$$y_{uv} + y_{uw} \leq y_{uu} \quad \forall u \in V, \quad \forall (v, w) \in E \text{ tal que } v, w \in A(u) \quad (5-3)$$

$$y_{uv} \leq y_{uu} \quad \forall u \in V, \quad \forall v \in A(u) \text{ tal que } v \text{ está isolado em } A(u) \quad (5-4)$$

$$y_{uv} \in \{0, 1\} \quad \forall u \in V, \quad \forall v \in A'(u). \quad (5-5)$$

O número de variáveis y é igual a $|V| + \bar{m}$, onde \bar{m} é o número de arestas no grafo complementar de G . A função objetivo (5-1) minimiza o número de vértices representantes, ou seja, o número de cores utilizadas. A restrição (5-2) garante que cada vértice $u \in V$ é um vértice representante, ou é representado por algum outro vértice $v \in A(u)$. A restrição (5-3) impõe que

vértices adjacentes tenham representantes (cores) diferentes. A restrição (5-3), junto com a restrição (5-4), garante que um vértice só pode ser representado por vértices escolhidos como representantes.

A formulação acima foi refinada em [17] com o objetivo de reduzir sua simetria. Na *formulação assimétrica por representantes*, um vértice u só pode representar um vértice v se $I[u] < I[v]$, onde $I[u]$ e $I[v]$ são os índices dos vértices u e v , respectivamente. Sendo assim, o vértice representante de uma cor é necessariamente aquele com menor índice. Define-se $A_{<}(u) = \{v \in A(u) : I[v] < I[u]\}$ como a *anti-vizinhança-menor* do vértice u , ou seja, os vértices que podem representar o vértice u , e $A_{>}(u) = \{v \in A(u) : v > u\}$ como a *anti-vizinhança-maior* do vértice $u \in V$, ou seja, os vértices que não podem representar o vértice u . Por conveniência, também são definidos $A'_{>}(u) = A_{>}(u) \cup \{u\}$ e $A'_{<}(u) = A_{<}(u) \cup \{u\}$. Aplicando-se as definições acima, as restrições (5-2) a (5-5) são reescritas nas restrições (5-7) a (5-10).

Denota-se por $V^s = \{u \in V : A_{<}(u) = \emptyset\}$ o conjunto dos vértices cuja anti-vizinhança-menor é vazia, ou seja, os vértices que são necessariamente representantes. Portanto, para todos os vértices $u \in V^s$, $y_{uu} = 1$ em todas as soluções viáveis do problema. Essas variáveis podem ser removidas da formulação anterior. Define-se $\beta_u = 1$ se $u \in V^s$ e $\beta_u = y_{uu}$ caso contrário, obtendo-se então a nova formulação:

$$\min \sum_{u \in V} \beta_u \quad (5-6)$$

sujeito a:

$$\sum_{v \in A'_{<}(u)} y_{vu} = 1 \quad \forall u \in V \setminus V^s \quad (5-7)$$

$$y_{uv} + y_{uv} \leq \beta_u \quad \forall u \in V, \quad \forall (v, w) \in E \text{ tal que } v, w \in A_{>}(u) \quad (5-8)$$

$$y_{uv} \leq y_{uu} \quad \forall u \in V \setminus V^s, \quad \forall v \in A_{>}(u) \text{ tal que } v \text{ está isolado em } A_{>}(u) \quad (5-9)$$

$$y_{uv} \in \{0, 1\} \quad \forall u \in V, \quad \forall v \in A'_{>}(u), \quad (5-10)$$

5.2

Formulação por Representantes para Coloração de Partições

Nesta tese, é proposta uma extensão da formulação por representantes de [17] para o problema de coloração de partições. Assim como em [17, 25], um vértice é escolhido para ser o representante de todos os vértices com a mesma cor. No entanto, somente um vértice de cada componente da partição é colorido na formulação por representantes para PCP. Sendo assim, cada vértice está exatamente em um dos três estados: (i) colorido e representando todos os

vértices com a mesma cor, (ii) colorido e sendo representado por outro vértice com a mesma cor, ou (iii) não está colorido.

Seja $P[u]$ o índice da componente da partição que contém o vértice $u \in V$, define-se $A_P(u) = \{v \in A(u) : P[u] \neq P[v]\}$, como a *anti-vizinhança particionada* de u , ou seja, os vértices na anti-vizinhança de u que estão em outra componente da partição. Por conveniência, também define-se $A'_P(u) = A_P(u) \cup \{u\}$. Um vértice $v \in A_P(u)$ é dito *isolado* em $A_P(u)$ se $E[A_P(u)] = E[A_P(u) \setminus \{v\}]$, ou seja, se v não tem vizinhos em $A_P(u)$. A formulação por representantes para PCP utiliza as variáveis binárias x_{uv} para todo $u \in V$ e para todo $v \in A'_P(u)$, de modo que $x_{uv} = 1$ se o vértice u representa a cor do vértices v e $x_{uv} = 0$, caso contrário. Se um vértice $v \in V$ não está colorido na solução do PCP, $x_{uv} = 0$ para todo $u \in A'_P(v)$. Sendo assim, o PCP pode ser formulado como o seguinte problema de programação inteira:

$$\min \sum_{u \in V} x_{uu} \quad (5-11)$$

sujeito a:

$$\sum_{u \in Q_p} \sum_{v \in A'_P(u)} x_{vu} = 1 \quad \forall p = 1, \dots, q \quad (5-12)$$

$$x_{uv} + x_{uw} \leq x_{uu} \quad \forall u \in V, \quad \forall (v, w) \in E \text{ com } v, w \in A_P(u) \text{ e } P[v] \neq P[w] \quad (5-13)$$

$$x_{uv} \leq x_{uu} \quad \forall u \in V, \quad \forall v \in A_P(u) \text{ tal que } v \text{ está isolado em } A_P(u) \quad (5-14)$$

$$x_{uv} \in \{0, 1\} \quad \forall u \in V, \quad \forall v \in A'_P(u). \quad (5-15)$$

O número de variáveis x é igual $|V| + \overline{m}_p$, onde \overline{m}_p é o número de arestas no grafo complementar de G_P cujos nós terminais estão em diferentes componentes da partição. A função objetivo (5-11) minimiza o número de vértices representantes, ou seja, o número de cores. A restrição (5-12) garante que cada componente $Q_p, p = 1, \dots, q$, tem pelo menos um dos seus vértices $u \in Q_p$ representado, ou por ele mesmo ($x_{uu} = 1$) ou por algum outro vértices v ($x_{vu} = 1$, com $v \neq u$) na sua anti-vizinhança particionada, ou seja, um vértice de outra componente que não compartilha uma aresta com u . A restrição (5-13) impõe que vértices adjacentes tenham representantes distintos e em conjunto com a restrição (5-14) garante que um vértice só pode ser representado por um vértice representante.

Para reduzir a simetria da formulação acima, aplica-se a mesma estratégia proposta em [17]. Estabelece-se que um vértice u só pode representar a cor do

vértice v se $P[u] < P[v]$, ou seja, se o índice da componente de u é menor que o índice da componente de v . Portanto, o vértice representante de uma cor é necessariamente o vértice que está na componente da partição com menor índice. Define-se $A_{P>}(u) = \{v \in A_P(u) : P[u] > P[v]\}$ como a *anti-vizinhança-maior particionada* de um vértice $u \in V$, ou seja, os vértices que não podem representar a cor do vértice u e $A_{P<}(u) = \{v \in A_P(u) : P[v] < P[u]\}$ como a *anti-vizinhança-menor particionada* do vértice u , ou seja, os vértices que podem representar a cor do vértice u (excluindo o próprio u). Por conveniência, também define-se $A'_{P>}(u) = A_{P>}(u) \cup \{u\}$ e $A'_{P<}(u) = A_{P<}(u) \cup \{u\}$. Aplicando-se as definições acima, as restrições (5-12) a (5-15) são reescritas nas restrições (5-17) a (5-20), respectivamente.

Um vértice que encontra-se sozinho em uma componente da partição é chamado de *vértice elementar*. Define-se $V^e \subseteq V$ como o conjunto de todos os vértices elementares em G_P , ou seja, o conjunto de vértices que sempre estão coloridos numa solução do PCP para G_P . Sendo assim, define-se $V^0 = \{u \in V^e : A_{P<}(u) = \emptyset\}$ como o conjunto dos vértices elementares cuja anti-vizinhança-menor particionada é vazia, ou seja, o conjunto dos vértices que sempre serão representantes. Define-se também Q^0 como o conjunto de componentes da partição que contém os vértices em V^0 . Como os vértices $v \in V^0$ são sempre representantes, $x_{vv} = 1$ em qualquer solução do PCP. Portanto, estas variáveis podem ser removidas da formulação e o número de variáveis x é reduzido a $\bar{m}_p + |V \setminus V^0|$. Sendo assim, a função objetivo (5-11) é reescrita como (5-16) na formulação assimétrica por representantes para coloração de partições:

$$\min \sum_{v \in V \setminus V^0} x_{vv} + |V^0| \quad (5-16)$$

sujeito a:

$$\sum_{u \in Q_p} \sum_{v \in A'_{P<}(u)} x_{vu} = 1 \quad \forall Q_p \in Q \setminus Q^0 \quad (5-17)$$

$$x_{uw} + x_{uw} \leq \beta_u \quad \forall u \in V, \quad \forall (v, w) \in E \text{ com } v, w \in A_{P>}(u) \text{ e } P[v] \neq P[w] \quad (5-18)$$

$$x_{uv} \leq x_{uu} \quad \forall u \in V, \quad \forall v \in A_{P>}(u) \text{ tal que } v \text{ está isolado em } A_{P>}(u) \quad (5-19)$$

$$x_{uv} \in \{0, 1\} \quad \forall u \in V, \quad \forall v \in A'_{P>}(u), \quad (5-20)$$

onde $\beta_u = 1$ se $u \in V^0$; caso contrário $\beta_u = x_{uu}$.

5.3

Branch-and-cut para Coloração de Partições

Nesta seção, descreve-se o algoritmo de *branch-and-cut* B&C-PCP, baseado na formulação assimétrica por representantes definida pelas equações (5-16) a (5-20). Desigualdades válidas são adicionadas progressivamente em cada subproblema da árvore de *branch-and-cut*, o que na maioria dos casos melhora o limite inferior fornecido pela relaxação das restrições de integralidade das variáveis x da formulação (Equação (5-20)).

Primeiramente, são apresentadas as técnicas de pré-processamento utilizadas para reduzir o tamanho dos subproblemas que são resolvidos em cada nó da árvore de *branch-and-cut*. Depois, descreve-se a estratégia de particionamento utilizada para decompor um problema de PCP em dois subproblemas, de modo que a solução ótima do problema original encontre-se em um dos dois subproblemas. Em seguida, descreve-se a heurística utilizada para gerar soluções viáveis para cada subproblema. Finalmente, apresentam-se as desigualdades válidas que são utilizadas no algoritmo de planos de corte desenvolvido para melhorar o limite inferior fornecido pela relaxação linear.

5.3.1

Pré-processamento

O pré-processamento é usado para reduzir o tamanho do grafo particionado $G_P = (V, E, Q)$ antes de aplicar-se o algoritmo de *branch-and-cut* para PCP. Espera-se que quanto menor o tamanho do grafo particionado, menos difícil seja o problema de coloração de partições resultante. O pré-processamento está dividido em três etapas listadas a seguir.

Primeiramente, todas as arestas $(u, v) \in E$ com $P[u] = P[v]$ são removidas de G_P , já que somente um dos dois vértices u e v está colorido em uma solução viável do PCP.

Em seguida, os vértices elementares (aqueles que estão sozinhos em suas respectivas componentes) que estão conectados a todos os outros vértices em G_P são removidos do grafo. Esses vértices necessariamente terão uma cor diferente de todos os outros vértices no grafo em qualquer solução viável do PCP. Conseqüentemente, eles podem ser removidos do grafo e o número total de vértices removidos é adicionado ao valor da função objetivo obtido ao final do algoritmo de *branch-and-cut*.

Por fim, a relaxação linear no nó raiz da árvore de *branch-and-cut* é resolvida e todas as componentes $Q_p \in Q$, que contém pelo menos um vértice $v \in Q_p$ com grau menor que o limite inferior fornecido pela relaxação linear, são removidas de G_P . Isto é possível porque o número de cores na solução

ótima do PCP será maior que o número de vizinhos de v . Portanto, ao final do algoritmo de *branch-and-cut*, o vértice v pode ser colorido com qualquer uma das cores na solução que não é usada por algum dos seus vizinhos.

5.3.2

Estratégia de Decomposição

A estratégia de decomposição empregada por B&C-PCP consiste em decompor um problema de coloração de partições em dois subproblemas, também de coloração de partições, de tal forma que a solução ótima de um dos dois subproblemas seja a mesma solução ótima do problema original. A estratégia empregada é baseada naquela proposta em [67] para GCP. Dado um grafo particionado $G_P = (V, E, Q)$, para cada par de componentes $Q_i, Q_j \in Q$, tais que $i < j$, definem-se as seguintes operações:

- DIFFER(i, j): colorir as componentes Q_i e Q_j com diferentes cores; e
- SAME(i, j): colorir as componentes Q_i e Q_j com a mesma cor.

A primeira operação é implementada através da inserção de uma aresta para cada par de vértices não adjacentes nas componentes Q_i e Q_j . Já a operação SAME(i, j) é implementada substituindo as duas componentes Q_i e Q_j por uma nova componente da partição. Esta última contém um vértice w_{uv} para cada par de vértices $u \in Q_i$ e $v \in A_P(u) \cap Q_j$, sendo que o conjunto de vizinhos de w_{uv} é a união dos conjuntos de vizinhos de u e v .

Quando estas duas operações são aplicadas ao grafo particionado G_P , elas criam dois novos grafos particionados G_1 e G_2 , tais que a solução ótima do PCP sobre G_P é a melhor dentre as soluções ótimas do PCP sobre G_1 e G_2 . A Figura 5.1 ilustra a decomposição de um grafo particionado G_P em dois grafos G_1 e G_2 , onde G_1 foi gerado através da aplicação da operação SAME(1, 2) e G_2 foi gerado pela operação DIFFER(1, 2).

5.3.3

Limites Superiores

A heurística de busca tabu TS-PCP [70] descrita na seção 2.5 é utilizada para calcular um limite superior para o custo da solução ótima do PCP em cada nó da árvore de *branch-and-cut*. No entanto, ao contrário de [70] a heurística construtiva *onestepCD* [63] não é utilizada para gerar a solução inicial de TS-PCP.

A nova heurística construtiva baseia-se no fato de que, devido a estratégia de decomposição empregada no B&C-PCP, o grafo particionado $\overline{G}_P = (\overline{V}, \overline{E}, \overline{Q})$ em cada nó da árvore de *branch-and-cut* diferencia-se do grafo particionado

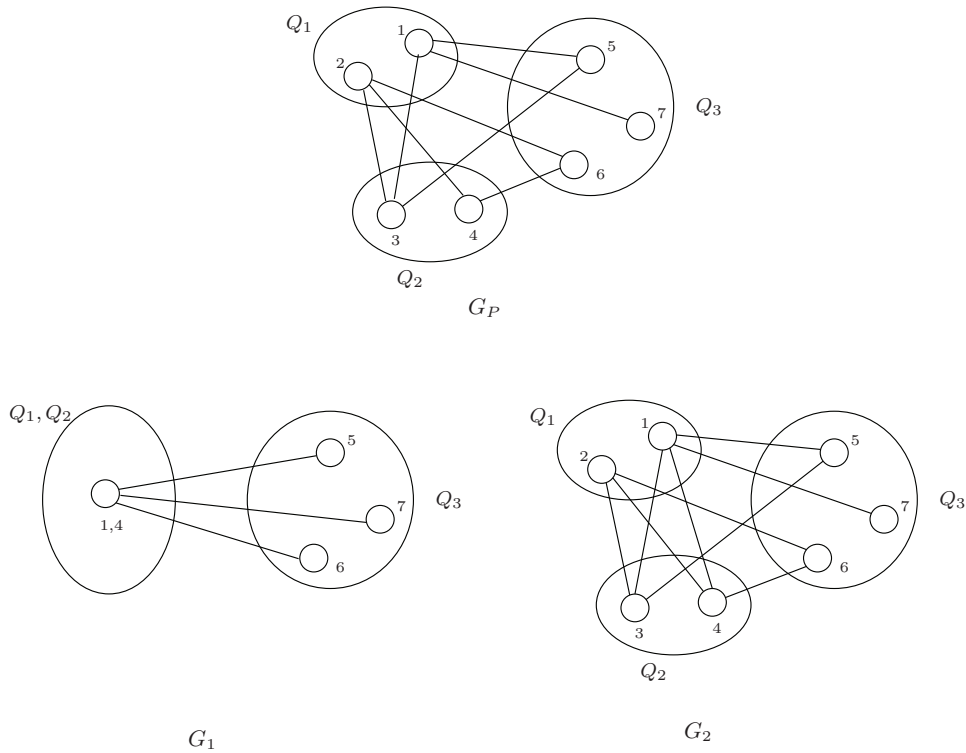


Figura 5.1: Decomposição do PCP sobre o grafo G_P em dois subproblemas de coloração de partições definidos sobre os grafos G_1 e G_2 .

do nó pai por no máximo duas componentes da partição. A heurística parte de uma solução parcial que é igual a coloração encontrada por TS-PCP para o grafo do nó pai, exceto pelas duas componentes \bar{Q}_i e \bar{Q}_j que foram alteradas pela aplicação das operações $\text{SAME}(i, j)$ ou $\text{DIFFER}(i, j)$. Sendo assim, a solução parcial tem no máximo duas componentes não coloridas. Em seguida, um vértice e uma cor são escolhidos para cada componente não colorida utilizando-se o mesmo critério guloso empregado pela heurística *onestepCD* [63]. Portanto, em cada nó da árvore de branch-and-cut, TS-PCP parte de uma solução inicial com no máximo duas cores a mais que a solução encontrada no nó pai.

5.3.4 Desigualdades Válidas

Os limites inferiores fornecidos pela relaxação da restrição de integralidade das variáveis x na formulação assimétrica por representantes para PCP podem ser ruins. Sendo assim, nesta tese generalizam-se as duas famílias de desigualdades válidas descritas em [17, 25] com o objetivo de melhorar o valor da relaxação linear. Um subconjunto $V_P \subseteq V$ é dito particionado se $P[u] \neq P[v]$ para todo $u, v \in V_P$, ou seja, se cada um dos seus vértices pertence a uma componente da partição diferente. Define-se um *conjunto independente parti-*

cionado como um conjunto particionado cujos vértices não compartilham nenhuma aresta em G_P . Exemplos de um conjunto particionado e de um conjunto independente particionado são apresentados na Figura 5.2(a) e Figura 5.2(b), respectivamente.

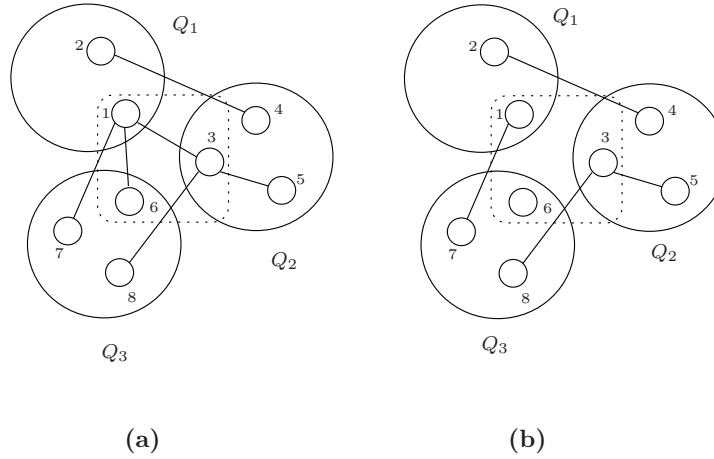


Figura 5.2: (a) Conjunto particionado e (b) Conjunto independente particionado.

Os *cortes internos* para PCP são derivados do número mínimo de cores necessárias para colorir um buraco ímpar ou um anti-buraco ímpar no grafo G_P . Um buraco ímpar é um circuito induzido de comprimento ímpar maior ou igual a 5, enquanto um anti-buraco ímpar é o complemento de um buraco ímpar. A fórmula dos cortes internos é descrita em (5-21), onde $H \subseteq V^e$ induz um buraco ou anti-buraco ímpar em G_P e $\chi(G[H])$ é o número cromático do subgrafo $G[H]$ induzido em G_P por H . Somente buracos e anti-buracos ímpares compostos apenas por vértices elementares são considerados na geração dos cortes internos, porque estes vértices necessariamente serão coloridos numa solução viável do PCP.

$$\sum_{v \in H \setminus V^0} x_{vv} + |H \cap V^0| + \sum_{v \in H \setminus V^0, w \in A_{P<}(v) \setminus H} x_{vw} \geq \chi(G[H]). \quad (5-21)$$

Teorema 1 *Se $H \subseteq V^e$ induz um buraco ou anti-buraco ímpar particionado em G_P , então (5-21) é uma desigualdade válida para a formulação assimétrica por representantes para PCP.*

Prova: O primeiro somatório do lado esquerdo de (5-21) é igual ao número de vértices em H que são representantes. O segundo somatório do lado esquerdo da equação é igual ao número de vértices em H que são representados por um vértice que não está em H . Em qualquer coloração particionada de G_P , cada cor usada em um vértice de H contribui com uma unidade ao primeiro ou ao

segundo somatório de (5-21). Como pelo menos $\chi(G[H])$ cores são necessárias para colorir $G[H]$, então a desigualdade (5-21) é válida para a formulação assimétrica por representantes para PCP. \square

Os cortes externos são derivados do número de vértices de um subconjunto $K \subseteq A_{P>}(u)$ que podem ser representados por um vértice $u \in V$. A fórmula dos cortes externos é descrita em (5-22), onde α_v é a cardinalidade do maior conjunto independente particionado do grafo $G[K]$ induzido em G_P por K que contém o vértice $v \in K$.

$$\sum_{v \in K} \frac{x_{uv}}{\alpha_v} \leq \beta_u. \quad (5-22)$$

Teorema 2 Para cada vértice $u \in V$ e qualquer conjunto particionado não vazio $K \subseteq A_{P>}(u)$, a desigualdade (5-22) é válida para a formulação assimétrica por representantes para PCP.

Prova: Se u não é um vértice representante em uma dada solução viável para PCP, então $\beta_u = 0$ e $x_{uv} = 0$ para todo $v \in A_{P>}(u)$. Portanto, a desigualdade (5-22) é válida quando u não é um vértice representante. No caso onde u é um vértice representante, $\beta_u = 1$, $x_{uv} = 1$ para todo $v \in W$ e $x_{uw} = 0$ para todo $w \in G[K] \setminus W$, onde $W \subseteq G[K]$ é o conjunto independente particionado composto pelos vértices representados por u na solução. Seja $\delta = \min_{v \in W} \{\alpha_v\}$. Então, $|W| \leq \delta$, pois $|W| \leq |K|$ e, portanto, não pode ser maior que o maior conjunto independente particionado de seus vértices em $G[K]$. Logo,

$$\sum_{v \in K} \frac{x_{uv}}{\alpha_v} = \sum_{v \in W} \frac{1}{\alpha_v} \leq \sum_{v \in W} \frac{1}{\delta} = \frac{|W|}{\delta} \leq 1.$$

Conseqüentemente, a desigualdade (5-22) também é válida para o caso onde u é um vértice representante. \square

Uma clique K do grafo G_P é dita particionada se $P[u] \neq P[v]$ para todo $u, v \in K$, ou seja, se cada um dos seus vértices pertence a uma componente da partição diferente. O Corolário 1 abaixo é derivado do Teorema 2 e do fato de que $\alpha_v = 1$, para todo $v \in K$.

Corolário 1 Para cada $u \in V$ e qualquer clique particionada maximal $K \subseteq A_{P>}(u)$,

$$\sum_{v \in K} x_{uv} \leq \beta_u \quad (5-23)$$

é uma desigualdade válida para a formulação assimétrica por representantes para PCP.

Define-se que um buraco ímpar (resp. anti-buraco ímpar) H é particionado se todos os seus vértices pertencem a diferentes componentes da partição. O Teorema 2 é válido para todo conjunto particionado não vazio. Conseqüentemente, ele também é válido para buracos e anti-buracos ímpares particionados. Nos dois casos, o valor de α_v , para todo $v \in H$, é igual à cardinalidade α_H do maior conjunto independente em $G[H]$, onde $\alpha_H = \lfloor |H|/2 \rfloor$ para buracos ímpares e $\alpha_H = 2$ para anti-buracos ímpares. Portanto, o lado esquerdo da equação (5-22) pode ser reescrito como

$$\sum_{v \in H} \frac{x_{uv}}{\alpha_v} = \sum_{v \in H} \frac{x_{uv}}{\alpha_H} = \frac{\sum_{v \in H} x_{uv}}{\alpha_H}$$

para todo $u \in V$ e $H \subseteq A_{P>}(u)$. Logo,

Corolário 2 Para todo $u \in V$ e qualquer buraco ou anti-buraco ímpar particionado $H \subseteq A_{P>}(u)$,

$$\sum_{v \in H} x_{uv} \leq \alpha_H \beta_u \tag{5-24}$$

é uma desigualdade válida para a formulação assimétrica por representantes para PCP.

5.3.5 Separação dos Cortes Violados

Nesta tese é proposto um algoritmo de plano de cortes baseado nas desigualdades válidas (5-21), (5-23) e (5-24). Este algoritmo iterativamente (i) resolve a relaxação linear da formulação assimétrica por representantes para PCP, (ii) identifica cortes externos e internos violados, (iii) adiciona os cortes violados à formulação e (iv) resolve a relaxação linear do novo problema. Dada a solução da relaxação linear, a separação dos cortes violados na relaxação linear consiste em encontrar cliques, buracos ímpares, e anti-buracos ímpares particionados no grafo $G_P = (V, E, Q)$ associado a cada nó da árvore de *branch-and-cut*.

Separação de Cortes Externos Baseados em Cliques

Sejam $G^u = (V^u, E^u)$ o subgrafo induzido em G pela anti-vizinhança-maior particionada $V^u = A_{P>}(u)$ do vértice $u \in V$ e \bar{x}_{uv} o valor da variável x_{uv} na solução ótima da relaxação linear da formulação (5-16) a (5-20), para todo $v \in A'_P(u)$. O algoritmo para separação dos cortes externos baseados em cliques consiste em encontrar uma clique particionada $K \subseteq V^u$ tal que

$\sum_{v \in K} \bar{x}_{uv} > \beta_u$. Ele é executado uma vez para cada vértice $u \in V$ tal que $\bar{x}_{uu} > 0$.

Uma heurística GRASP [32, 33, 83, 84, 85] foi utilizada na identificação das cliques em G^u . A heurística é um procedimento iterativo composto de duas fases: uma *fase de construção* e uma *fase de busca local*. A fase de construção gera uma solução inicial que pode ser posteriormente melhorada na fase de busca local. A heurística tenta encontrar uma clique particionada $C \subseteq V^u$ que maximiza a função $\sum_{v \in K} \bar{x}_{uv}$.

A fase de construção é iniciada com uma clique vazia e constrói uma clique particionada, adicionando um vértice de cada vez. Dados uma clique particionada $C \subseteq V^u$ e o conjunto de vértices $\delta(C) = \{w \in V^u \setminus C \mid C \cup \{w\} \text{ é uma clique particionada de } G\}$, a cada iteração, um vértice $w \in \delta(C)$ é escolhido com probabilidade igual a $\bar{x}_{uw} / (\sum_{z \in \delta(C)} \bar{x}_{uz})$. O vértice escolhido é inserido em C e o conjunto $\delta(C)$ é atualizado. Este procedimento é repetido até que $\delta(C) = \emptyset$.

Na fase de busca local, a vizinhança $\gamma(C)$ é definida como o conjunto de todas as cliques particionadas obtidas pela troca de um vértice $v \in C$ com outro vértice $u \in \delta(C \setminus \{v\})$. O algoritmo é iniciado com a clique particionada fornecida pela fase de construção e iterativamente substitui a solução corrente pela solução com melhor custo na sua vizinhança. A busca local termina quando nenhuma solução melhor é encontrada na vizinhança da solução corrente.

A heurística GRASP é encerrada após $10 \cdot |V^u|$ iterações terem sido executadas sem que a clique particionada de maior peso tenha sido atualizada. Então, as $|V^u|$ cliques com maior peso encontradas durante a execução da heurística são selecionadas e gera-se um corte externo para cada uma delas. Por fim, os cortes violados são adicionados à formulação assimétrica por representantes para PCP definida pelas desigualdades (5-16) a (5-20).

Separação de Cortes Externos Baseados em Buracos e Anti-buracos Ímpares

Para todo $u \in V$ tal que $\bar{x}_{uu} > 0$, a separação de cortes externos baseados em buracos e anti-buracos ímpares consiste em encontrar um buraco ou anti-buraco ímpar particionado $H \in V^u$ tal que $\sum_{v \in H} \bar{x}_{uv} > \alpha_H \beta_u$. Para identificação dessas estruturas foi utilizado uma generalização do algoritmo de Hoffman e Padberg [47] para encontrar cortes baseados em buracos ímpares violados em $G^u = (V^u, E^u)$. O mesmo algoritmo é aplicado ao grafo complementar a G^u para encontrar cortes externos baseados em anti-buracos ímpares violados.

O algoritmo está dividido em duas etapas. Na primeira etapa, constrói-se um grafo em camadas da seguinte forma. Um vértice $r \in V^u$ é escolhido aleatoriamente. Em seguida, associa-se a cada vértice $w \in V^u$ um valor h_w igual ao menor número de arestas no caminho de w a r . Por definição, $h_r = 0$. O valor de h para os vizinhos de r é igual a 1, e assim por diante. A Figura 5.3 ilustra um grafo com os valores de h para cada um dos vértices. Para quaisquer dois vértices w e z com $h_w = h_z \geq 2$, se existe um caminho mínimo p_w de w para r e outro p_z de z para r que não contenham vértices em comum, então existe um buraco ímpar que contém w , z e r .

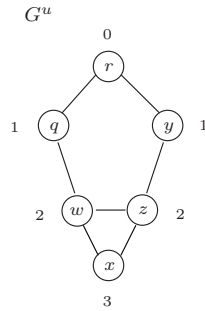


Figura 5.3: Grafo com os respectivos valores de h .

Na segunda etapa, atribui-se pesos $t_{wz} = 2 - \bar{x}_{uw} - \bar{x}_{uz}$ para toda aresta $(w, z) \in E^u$. Em seguida, para cada aresta $(w, z) \in E^u$ com $h_w = h_z \geq 2$ e $P[w] \neq P[z] \neq P[r]$, busca-se o caminho mais curto de z para r e o caminho mais curto de w para r . Os caminhos devem ser formados exclusivamente por vértices de componentes diferentes. Se os dois caminhos existirem, então encontrou-se um buraco ímpar particionado que pode ser usado para gerar cortes externos. Caso contrário, o algoritmo continua a partir da próxima aresta em E^u . Este algoritmo é repetido $0.4 \cdot |V^u|$ vezes, partindo de diferentes vértices r escolhidos aleatoriamente.

Separação de Cortes Internos Baseados em Buracos e Anti-buracos Ímpares

A separação de cortes internos baseados em buracos e anti-buracos ímpares consiste em encontrar um buraco ou anti-buraco ímpar particionado $H \in V$ tal que

$$\sum_{v \in H \setminus V^0} \bar{x}_{vv} + |H \cap V^0| + \sum_{v \in H \setminus V^0, w \in A_{P^<}(v) \setminus H} \bar{x}_{wv} < \chi(G[H]).$$

O algoritmo utilizado é similar ao algoritmo de separação descrito na seção anterior. No entanto, todos os vértices do grafo G_P são considerados na busca.

O mesmo algoritmo é aplicado no grafo complementar de G^P para encontrar cortes internos baseados em anti-buracos ímpares violados.

Primeiramente, constrói-se um grafo em camadas enraizado em um vértice $r \in V$ escolhido aleatoriamente. Em seguida, atribui-se pesos

$$t_{wz} = \sum_{u \in A'_{P>}(w) \setminus H} \bar{x}_{uw} + \sum_{u \in A'_{P>}(z) \setminus H} \bar{x}_{uz}$$

para toda aresta $(w, z) \in E$. Então, para cada aresta $(w, z) \in E$ com $h_w = h_z \geq 2$ e $P[w] \neq P[z] \neq P[r]$, busca-se o caminho mais curto de z para r e o caminho mais curto de w para r . Os caminhos devem ser formados exclusivamente por vértices de componentes diferentes. Se os dois caminhos existirem, então encontrou-se um buraco ímpar particionado que pode ser usado para gerar cortes internos. Caso contrário, o algoritmo continua a partir da próxima aresta em E . Este algoritmo é repetido $0.4 \cdot |V|$ vezes, partindo de diferentes vértices r escolhidos aleatoriamente.

5.4

Experimentos Computacionais

O algoritmo B&C-PCP descrito na Seção 5.3 foi implementado em C++ e compilado com a versão v3.41 do compilador Linux/GNU. A relaxação linear da formulação por representantes assimétrica (5-16) a (5-20) foi resolvida pelo XPRESS versão 2005-a. Os experimentos foram executados em um computador AMD-Atlon 1.8 GHz com 1 Gigabyte de memória RAM. O algoritmo de *branch-and-cut* encerra sua execução quando a solução ótima é encontrada ou após duas horas de processamento. A qualidade do algoritmo é avaliada em função do *desvio relativo* $([UB-LB]/LB)$ e do *desvio absoluto* $(UB-LB)$ entre o melhor limite inferior UB e o melhor limite inferior LB para o custo da solução ótima do PCP, calculados durante a execução do algoritmo.

O primeiro experimento investiga o comportamento de B&C-PCP para grafos particionados gerados aleatoriamente com 20 a 120 vértices. Cada componente da partição tem exatamente dois vértices e a densidade das arestas é 0,5. Cinco grafos foram gerados para cada número de vértices. O algoritmo B&C-PCP foi executado três vezes para cada grafo, com diferentes sementes para o gerador de números aleatórios utilizado pelas heurísticas de separação. Como a separação dos cortes não é determinística, o número de subproblemas resolvidos por B&C-PCP até encontrar a solução ótima pode variar em cada execução. As médias, sobre as 15 execuções para cada tamanho de grafo, dos limites superiores e inferiores para o número ótimo de cores obtidos até duas horas de execução, são plotadas na Figura 5.4. B&C-PCP encontrou a solução

ótima para todas as instâncias com até 80 vértices. Com relação às outras instâncias, a diferença entre os limites superiores e inferiores foi sempre de uma única cor.

No segundo experimento, geraram-se aleatoriamente grafos com 90 vértices particionados em 45 componentes, com exatamente dois vértices cada uma. Variou-se a densidade de arestas entre 0,1 e 0,9. Cinco instâncias foram geradas para cada valor de densidade de arestas. O algoritmo B&C-PCP foi executado três vezes para cada instância com diferentes sementes para o gerador de números aleatórios. A média dos limites inferiores e superiores para o número de cores na solução ótima das 15 execuções para cada tamanho de grafo é plotada na Figura 5.5. Pode-se observar que a diferença entre os limites superiores e inferiores nunca foi maior que um, nas instâncias que não foram resolvidas dentro do limite de duas horas. O número de instâncias cuja solução ótima foi encontrada em menos de duas horas de processamento para cada valor de densidade de arestas é apresentado na Tabela 5.1. Pode-se observar que este número cresce com a densidade das arestas. Isto se deve ao fato de que grafos com maior densidade de arestas têm cliques de maior cardinalidade, o que leva a melhores limites inferiores. As instâncias mais difíceis são aquelas com densidade de arestas entre 0,3 e 0,5.

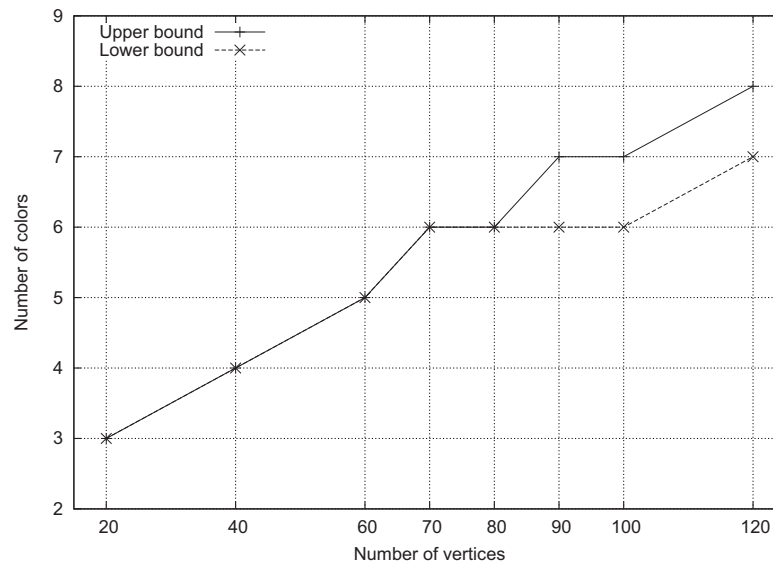


Figura 5.4: Média dos limites inferiores e superiores para o número ótimo de cores variando-se o número de vértices.

A Tabela 5.2 mostra a contribuição dos cortes externos e internos para a relaxação linear da formulação assimétrica por representantes para PCP nas instâncias usadas nos dois experimentos apresentados acima. As primeiras três colunas mostram o número de vértices, de arestas e de componentes,

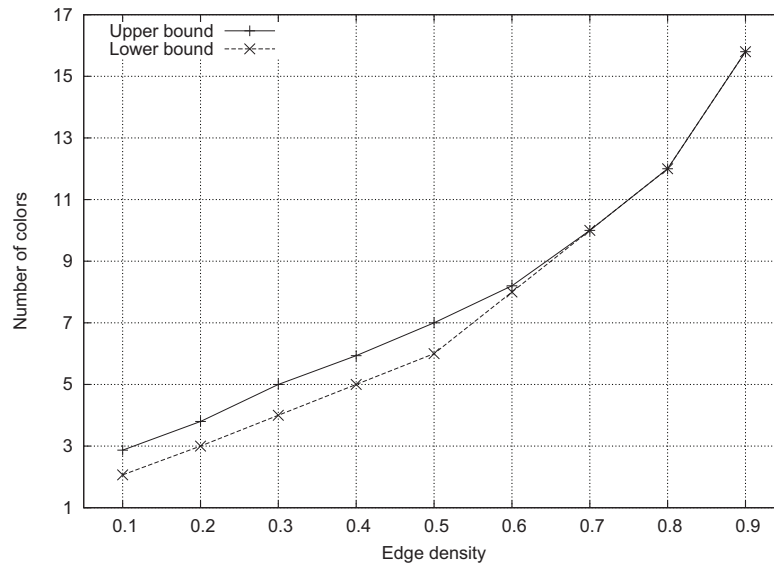


Figura 5.5: Média dos limites inferiores e superiores para o número ótimo de cores variando-se a densidade das arestas.

respectivamente, para cada grupo de cinco instâncias geradas aleatoriamente em ambos experimentos. O valor médio da relaxação linear é apresentado na quarta coluna. As duas últimas colunas apresentam a melhoria relativa na relaxação linear obtida pela adição dos (i) cortes externos e (ii) dos cortes externos e internos em conjunto. O valor da melhoria é calculado dividindo-se o valor médio da relaxação linear sem cortes pelo valor médio da relaxação linear após a aplicação dos cortes. Pode-se observar que os cortes externos melhoram a relaxação linear sem cortes em 32.93%, enquanto os cortes internos melhoram muito pouco os resultados obtidos após a aplicação dos cortes externos.

No terceiro experimento, foram geradas instâncias de coloração de partições a partir de instâncias de GCP disponíveis na literatura. Cada componente da partição é formada por exatamente um vértice. Neste caso, o PCP se reduz a um GCP. Os resultados obtidos por **B&C-PCP** são comparados com aqueles obtidos pelo algoritmo de *branch-and-cut* proposto em [68] para o GCP. Os resultados computacionais para as instâncias mais difíceis em [68] (ver Tabela 7 em [68]) são apresentadas na Tabela 5.3. Para cada instância, as três primeiras colunas mostram o nome, o número de vértices e o número de arestas no grafo, respectivamente. As três colunas seguintes apresentam o limite superior, o limite inferior e o desvio relativo fornecido por **B&C-PCP**, respectivamente. As últimas três colunas trazem as mesmas informações para o algoritmo de *branch-and-cut* em [68], após duas horas de processamento em uma estação de trabalho Sun ULTRA (ver Tabela 7 em [68]). Pode-se observar

Densidade	Ótimos
0,1	5
0,2	3
0,3	0
0,4	1
0,5	0
0,6	12
0,7	15
0,8	15
0,9	15

Tabela 5.1: Número de instâncias resolvidas exatamente em duas horas de processamento para diferentes valores da densidade de arestas.

que os desvios relativos de **B&C-PCP** são menores que os do algoritmo em [68] nas instâncias com cliques de maior cardinalidade (tais como as instâncias DSJC), enquanto ocorre o oposto nas instâncias com cliques de menor cardinalidade (tais como as instâncias Queen e Mycielsky). Isto se deve ao fato de que os cortes de clique utilizados no **B&C-PCP** são mais fortes do que aqueles propostos em [68]. Para os outros grupos de instâncias, o desvio relativo dos dois algoritmos foi muito similar.

No quarto e no quinto experimentos foram geradas instâncias de PCP a partir de instâncias de min-RWA. Primeiramente, são consideradas redes com topologias em anéis [51]. Três redes com 10, 15 e 20 nós foram geradas. As matrizes de tráfego foram geradas aleatoriamente, com probabilidade ρ de existir uma requisição de caminho ótico entre um par de nós variando de 0,1 a 1,0 em passos de 0,1. Para cada número de nós e cada valor de ρ , cinco matrizes de tráfego foram geradas. Os enlaces são bidirecionais e as requisições de caminhos óticos não são necessariamente simétricas, ou seja, o número de requisições de caminhos óticos de um vértice i para um vértice j pode ser diferente do número de requisições de j para i . Cada instância de min-RWA é transformada em uma instância de PCP sobre um grafo $G_P = (V, E, Q)$ construído conforme descrito a seguir. Cada componente em Q corresponde a uma requisição de caminho ótico e contém um vértice $v \in V$ para cada possível rota que o caminho ótico correspondente pode usar. O grafo G_P possui uma aresta $e \in E$ entre cada par de vértices cujas rotas associadas compartilham uma fibra ótica da rede. No caso de redes em anel, existem exatamente duas rotas alternativas entre os dois vértices terminais de um caminho ótico, uma delas no sentido horário e a outra no sentido anti-horário. Conseqüentemente, todas as componentes em Q possuem exatamente dois vértices. Como todas as possíveis rotas para cada caminho ótico são consideradas, esta transformação

V	densidade	Q	relaxação		melhoria relativa (%)	
			linear	externos	externos e internos	
20	0,5	2	2,33	1,87	1,87	
40	0,5	2	2,83	18,62	18,40	
60	0,5	2	3,21	31,97	32,37	
70	0,5	2	3,41	35,47	35,50	
80	0,5	2	3,56	40,94	40,96	
90	0,5	2	3,68	44,51	44,54	
100	0,5	2	3,76	46,80	46,80	
120	0,5	2	3,94	52,96	52,97	
90	0,1	2	1,28	27,95	27,97	
90	0,2	2	1,67	35,81	35,89	
90	0,3	2	2,18	43,08	43,08	
90	0,4	2	2,80	45,28	45,32	
90	0,5	2	3,68	44,51	44,54	
90	0,6	2	4,92	40,83	40,84	
90	0,7	2	6,66	32,78	32,79	
90	0,8	2	9,65	15,68	15,72	
90	0,9	2	14,77	0,78	0,78	
			Média	32,93	32,96	

Tabela 5.2: Contribuição dos cortes externos e internos a relaxação linear.

garante que à solução ótima da instância do PCP corresponde à solução ótima da respectiva instância de min-RWA.

O algoritmo B&C-PCP foi aplicado a cada uma das 150 instâncias descritas acima. Os resultados computacionais são apresentados na Tabela 5.4. As duas primeiras colunas mostram o número de nós na rede em anel e o valor de ρ . As três colunas seguintes apresentam o número médio de vértices, de arestas e de componentes das cinco instâncias de PCP correspondentes. As colunas seguintes mostram, respectivamente, o número médio e máximo de subproblemas resolvidos na árvore de *branch-and-cut*, o valor médio e máximo do desvio absoluto, o valor médio e máximo do desvio relativo e o número de instâncias resolvidas de forma exata, dentre as cinco instâncias com o mesmo valor de ρ . Dentro do tempo limite de duas horas, foram encontradas as soluções ótimas para todas as instâncias de PCP geradas a partir de redes em anel com 10 e 15 nós e para 40 dentre as 50 instâncias geradas a partir de redes em anel com 20 nós. Os desvios relativos nas instâncias não resolvidas dentro do limite de duas horas de processamento não foram maiores do que 6%.

Em seguida, considerou-se a rede NSFnet [51] com 14 vértices e 21 enlaces. Foram geradas diferentes matrizes de tráfego variando-se a probabilidade ρ de existir uma requisição de caminho ótico entre um par de nós no intervalo 0,1 a 1,0 em passos de 0,1. Como no experimento anterior, os enlaces são bidirecionais e as requisições de caminhos óticos não são necessariamente

Instância	V	E	B&C-PCP			Branch-and-cut [68]		
			UB	LB	desvio (%)	UB	LB	desvio (%)
DSJC125.5	125	3891	17	14	21,4	20	13	53,8
DSJC125.9	125	6961	43	43	0,0	47	42	11,9
DSJC250.1	250	3218	9	5	80,0	9	5	80,0
DSJC250.5	250	15668	29	15	93,3	36	13	176,9
DSJC250.9	250	27897	72	71	1,4	88	47	87,2
DSJR500.1c	500	121275	85	85	0,0	88	47	87,2
queen.9_9	81	2112	10	9	11,1	10	9	11,1
myciel6	95	755	7	4	75,0	7	5	40,0
myciel7	191	2360	8	4	100,0	8	5	60,0
1-Insertions-5	202	1227	6	3	100,0	6	4	50,0
1-Insertions-6	607	6337	7	3	133,3	6	4	50,0
2-Insertions-4	149	541	5	3	66,7	5	4	25,0
2-Insertions-5	597	3936	6	3	100,0	6	3	100,0
3-Insertions-4	281	1046	5	3	66,7	5	3	66,7
4-Insertions-3	79	156	4	3	33,3	4	3	33,3
4-Insertions-4	475	1795	5	3	66,7	5	3	66,7
1-FullIns-5	282	3247	6	4	50,0	6	4	50,0
2-FullIns-4	212	1621	6	5	20,0	6	5	20,0
2-FullIns-5	852	12201	7	5	40,0	7	5	40,0
3-FullIns-4	405	3524	7	6	16,7	7	6	16,7
3-FullIns-5	2030	33751	8	6	33,3	8	6	33,3
4-FullIns-4	690	6650	8	7	14,3	8	7	14,3
Média			51,06			53,37		

Tabela 5.3: Instâncias de coloração de vértices.

simétricas. Cinco instâncias de min-RWA foram geradas para cada valor de ρ . Cada instância de min-RWA foi transformada em uma instância de PCP pelo procedimento 2-EDR [70] descrito na Seção 2.5.1. Primeiramente, 2-EDR calcula duas rotas alternativas para cada caminho ótico. Então, uma instância de PCP é construída contendo um vértice para cada rota alternativa e uma aresta entre cada par de vértices cujas respectivas rotas compartilham uma fibra ótica da rede. Os vértices associados ao mesmo caminho ótico são colocados na mesma componente da partição. Neste caso, a solução ótima do PCP pode não ser a solução ótima da instância de min-RWA correspondente, já que 2-EDR não considera todas as possíveis rotas para cada caminhos ótico.

O algoritmo B&C-PCP foi aplicado a cada uma das 50 instâncias descritas acima. Os resultados computacionais são apresentados na Tabela 5.5. A primeira coluna mostra a probabilidade de existência de um caminho ótico entre dois pares de vértices na rede. As três colunas seguintes apresentam o número médio de vértices, de arestas e de componentes das cinco instâncias de PCP geradas com o mesmo valor de ρ . As colunas seguintes mostram o número médio e máximo de subproblemas resolvidos na árvore de *branch-and-cut*, o

valor médio e máximo do desvio absoluto, o valor médio e máximo do desvio relativo e o número de instâncias resolvidas de forma exata no tempo limite de duas horas, dentre as cinco instâncias com o mesmo valor de ρ . Todas as instâncias de PCP geradas a partir de instâncias de min-RWA com ρ menor ou igual a 0,5 foram resolvidas no nó raiz da árvore de *branch-and-cut*. Apesar das instâncias com ρ maior que 0,5 serem mais difíceis, mais da metade delas foram resolvidas de forma exata. Os desvios relativos nas instâncias não resolvidas de forma exata dentro do limite de duas horas de processamento não foram maiores do que 5%.

Nós	ρ	$ V $	$ E $	$ Q $	subproblemas média(max)	desvio absoluto médio(max)	desvio relativo (%) médio(max)	ótimos
10	0,1	17,6	63,6	8,8	1,0 (1)	- (-)	- (-)	5
10	0,2	35,6	242,8	17,8	1,2 (2)	- (-)	- (-)	5
10	0,3	49,6	477,2	24,8	1,0 (1)	- (-)	- (-)	5
10	0,4	67,6	886,2	33,8	1,0 (1)	- (-)	- (-)	5
10	0,5	83,2	1342,6	41,6	1,0 (1)	- (-)	- (-)	5
10	0,6	102,4	2024,0	51,2	1,0 (1)	- (-)	- (-)	5
10	0,7	127,2	3154,0	63,6	1,0 (1)	- (-)	- (-)	5
10	0,8	145,2	4119,2	72,6	1,0 (1)	- (-)	- (-)	5
10	0,9	164,4	5297,4	82,2	1,0 (1)	- (-)	- (-)	5
10	1,0	180,0	6360,0	90,0	1,0 (1)	- (-)	- (-)	5
15	0,1	47,2	457,8	23,6	1,0 (1)	- (-)	- (-)	5
15	0,2	92,8	1723,4	46,4	1,0 (1)	- (-)	- (-)	5
15	0,3	136,4	3729,4	68,2	1,0 (1)	- (-)	- (-)	5
15	0,4	178,0	6358,2	89,0	1,2 (2)	- (-)	- (-)	5
15	0,5	215,6	9316,8	107,8	1,0 (1)	- (-)	- (-)	5
15	0,6	255,6	13076,8	127,8	1,4 (3)	- (-)	- (-)	5
15	0,7	303,2	18492,6	151,6	1,0 (1)	- (-)	- (-)	5
15	0,8	339,6	23151,4	169,8	1,0 (1)	- (-)	- (-)	5
15	0,9	381,6	29285,2	190,8	1,2 (2)	- (-)	- (-)	5
15	1,0	420,0	35490,0	210,0	1,0 (1)	- (-)	- (-)	5
20	0,1	78,4	1251,0	39,2	2,2 (7)	- (-)	- (-)	5
20	0,2	155,2	4893,2	77,6	1,4 (3)	- (-)	- (-)	5
20	0,3	232,0	10925,6	116,0	8,0 (20)	0,4 (1)	2 (6)	3
20	0,4	308,4	19319,4	154,2	2,4 (8)	0,2 (1)	1 (4)	4
20	0,5	383,2	29848,8	191,6	1,0 (1)	- (-)	- (-)	5
20	0,6	454,0	41860,8	227,0	1,2 (2)	- (-)	- (-)	5
20	0,7	540,8	59507,2	270,4	1,0 (1)	- (-)	- (-)	5
20	0,8	612,8	76398,0	306,4	1,0 (1)	- (-)	- (-)	5
20	0,9	689,2	96702,8	344,6	1,0 (1)	0,4 (1)	1 (2)	3
20	1,0	760,0	117420,0	380,0	1,0 (1)	1,4 (2)	3 (4)	0

Tabela 5.4: Resultados computacionais para instâncias de min-RWA associadas a três redes com topologia em anel.

ρ	$ V $	$ E $	$ Q $	subproblemas média(max)	desvio absoluto médio(max)	desvio relativo (%) médio(max)	ótimos
0,1	26,4	36,0	22,2	1 (1)	- (-)	- (-)	5
0,2	52,6	149,0	39,0	1 (1)	- (-)	- (-)	5
0,3	79,0	359,6	58,6	1 (1)	- (-)	- (-)	5
0,4	103,0	634,4	75,4	1 (1)	- (-)	- (-)	5
0,5	125,2	947,6	92,0	1 (1)	- (-)	- (-)	5
0,6	151,2	1410,4	109,6	24,2 (75)	0,4 (1)	4 (11)	3
0,7	181,0	2093,8	130,4	4,0 (16)	0,2 (1)	2 (10)	4
0,8	204,6	2638,2	146,6	8,6 (18)	0,6 (1)	5 (9)	2
0,9	224,8	3118,2	165,6	4,6 (8)	0,6 (1)	5 (8)	2
1,0	248,8	3885,6	182,0	2,2 (4)	0,4 (1)	3 (8)	3

Tabela 5.5: Resultados computacionais para instâncias de RWA associadas com a rede NSFnet.