

## Uma Heurística Baseada em Coloração de Arestas para o Problema de Programação de Tabelas para Competições Esportivas

Tiago Januario e Sebastián Urrutia  
Departamento de Ciência da Computação  
Universidade Federal de Minas Gerais  
Belo Horizonte – MG – Brasil

{januario, surrutia}@dcc.ufmg.br

### RESUMO

O objetivo deste trabalho é o de caracterizar uma nova vizinhança para o Problema de Programação de Tabelas para Competições Esportivas. Experimentalmente avaliamos o seu desempenho utilizando o Problema do Torneio com Viagens com Estádios Predefinidos como um estudo de caso. A nova vizinhança é descrita em termos de Teoria dos Grafos e a sua corretude é provada.

**PALAVRAS CHAVE.** Problema do Torneio com Viagens, Coloração de Arestas, Vizinhanças, Teoria dos Grafos.

### 1. Introdução

Em um torneio *Single Round Robin* (SRR) entre  $n$  equipes, cada par de equipes enfrentam-se exatamente uma vez em  $n - 1$  rodadas. Esse tipo de torneio possui um relacionamento um-para-um com o problema de coloração de aresta de grafos completos com o número mínimo de cores. Podemos modelar um torneio SRR por meio de um grafo completo de tal forma que os vértices representam as  $n$  equipes, cada aresta  $e = (v_i, v_j)$  representa a partida entre as equipes  $v_i$  e  $v_j$ , e cada cor representa a rodada em que a partida é disputada.

O programação de uma tabela para um torneio SRR é dividida em duas partes. Primeiramente, nós temos o quadro de jogos que determina em qual rodada cada partida será disputada. Em seguida, nós temos o conjunto de padrões casa-fora, também chamado de *Home-Away Pattern Set* (HAP). O conjunto HAP determina em que condição, casa ou fora, cada equipe jogará em cada rodada. Para representar o conjunto HAP, utiliza-se uma matriz  $H = h_{tr}$  de dimensões  $m \times (n - 1)$ , onde  $h_{tr} = +$  (respectivamente,  $h_{tr} = -$ ) quando a equipe  $t$  disputa em casa (respectivamente, disputa fora) uma partida na rodada  $r$ . Combinados, o quadro de jogos e o conjunto HAP definem uma tabela esportiva. Neste trabalho, o quadro de jogos é modelado por meio da coloração de arestas de grafos completos.

O Problema do Torneio com Viagens com Estádios Predefinidos (PTVEP), introduzido por (Melo et al., 2009), é uma variação do Problema do Torneio com Viagens (PTV) proposto por (Easton et al., 2001). O PTVEP consiste em programar um torneio SRR no qual o local de cada partida encontra-se predeterminado, ou seja, o conjunto HAP é fixo e conhecido de antemão. Dada uma matriz de distâncias de dimensões  $n \times n$ , dado que cada equipe parte do seu local de origem no início do torneio e que essa retornará para casa depois de disputar a sua última partida, o objetivo do problema é o de minimizar a

distância total percorrida pelas equipes de modo que nenhuma equipe poderá jogar mais de três jogos consecutivos em casa ou três jogos consecutivos fora de casa. Uma extensa revisão literária sobre os Problemas de Programação de Tabelas para Competições Esportivas com suas aplicações e definições pode ser encontrada em (Kendall et al., 2010) e em (Rasmussen and Trick, 2008).

Uma heurística ILS (do inglês, *iterated local search*) para o PTVEP é descrita em (Costa et al., 2012). As soluções iniciais são obtidas a partir de dois algoritmos para coloração de arestas em grafos completos: o algoritmo de Vizing (Vizing, 1964) e o método do Polígono (de Werra, 1981). Os autores mostraram que, em algumas situações, o método do Polígono não deve ser utilizado na obtenção de soluções iniciais e que o algoritmo de Vizing oferece uma melhor alternativa, por sua vez, produzindo melhores resultados. (Costa et al., 2012).

Neste artigo, nós apresentamos uma nova vizinhança para o Problema de Programação de Tabelas para Competições Esportivas, chamada de Rotação de Rodadas (*RR*). Utilizando o PTVEP como um estudo de caso, nós mostramos, através da exploração da vizinhança *RR*, que independentemente do algoritmo para a geração de soluções iniciais ou instância do problema, resultados obtidos são melhores que os publicados até o momento.

Todos os grafos considerados neste trabalho são completos, não orientados, finitos, simples e com um número par de vértices. Até o final desta seção iremos nos basear nas definições apresentadas em (Diestel, 2006), (Gabow et al., 1985) e (Misra and Gries, 1992) aos quais nós sugerimos a leitura para maiores detalhes.

Um grafo é um par ordenado  $G = (V, E)$  em que os  $n$  elementos de  $V$  são seus vértices e os  $m$  elementos de  $E$  são suas arestas. A cor  $c$  está presente em um vértice  $v$  de um grafo  $G$  se qualquer aresta que incide em  $v$  tem tal cor. Duas arestas de um grafo são ditas adjacentes se elas compartilham um vértice. Um conflito em uma atribuição de cores é a existência de quaisquer arestas adjacentes com a mesma cor. Uma coloração própria é uma coloração das arestas de um grafo sem conflitos.

Uma coloração própria das arestas de um grafo  $G$  com o número mínimo de cores é uma função de coloração  $c : E \rightarrow C$  em que  $c(e_i) \neq c(e_j)$  para quaisquer arestas adjacentes  $e_i$  e  $e_j$ , em que os elementos de  $C$  são chamados de cores disponíveis e a cardinalidade de  $C$  é mínima.

Um 1-fator em um grafo  $G$  é um subconjunto de arestas de  $G$  onde cada vértice aparece exatamente uma vez, ou seja, um emparelhamento perfeito. Uma 1-fatoração é uma partição do conjunto de arestas de  $G$  em 1-fatores. Duas fatorações de  $G$ ,  $F = \{f_1, f_2, \dots, f_k\}$  e  $H = \{h_1, h_2, \dots, h_k\}$ , são chamadas de isomórficas se existe uma função  $\phi$  que mapeia o conjunto de arestas de  $G$  em si mesmo, tal que  $\{f_1\phi, f_2\phi, \dots, f_k\phi\} = \{h_1, h_2, \dots, h_k\}$ . Dizemos que  $f_i\phi$  é o conjunto das arestas  $(v_i\phi, w_i\phi)$  em que  $(v_i, w_i)$  é uma aresta de  $F$ . Temos uma 1-fatoração perfeita, também chamada de *Perfect 1-factorization* (P1F) ou coloração perfeita, quando o grafo induzido pelas arestas em  $f_i \cup f_j$  ( $G(V, f_i \cup f_j)$ ) forma um circuito Hamiltoniano para cada par  $f_i$  e  $f_j$  de 1-fatores distintos. Observe que uma 1-fatoração e uma coloração própria das arestas de um grafo  $G$  com o número mínimo de cores são conceitos equivalentes para grafos completos com um número par de vértices, ou seja, um 1-fator  $f_c$  representa o conjunto de arestas que estão coloridas com a cor  $c$ .

Seja  $G[\alpha, \beta]$  o subgrafo de  $G$  induzido pelas arestas coloridas com as cores  $\alpha$  e  $\beta$ . Cada componente conexa de  $G[\alpha, \beta]$  é um ciclo no qual as arestas estão coloridas alternadamente pelas cores  $\alpha$  e  $\beta$ . Um caminho dentro desse ciclo é chamado de *caminho*  $-\alpha\beta$ . Seja  $P(v, w, \alpha, \beta)$  um *caminho*  $-\alpha\beta$  partindo do vértice  $v$  com a cor  $\alpha$  e terminando no vértice  $w$  com a cor  $\beta$ . O procedimento de troca das cores  $\alpha$  e  $\beta$  em  $P(v, w, \alpha, \beta)$  é chamado de alternar  $P(v, w, \alpha, \beta)$ .

A estrutura *fan* em  $v$  (detalhada em (Misra and Gries, 1992)), é uma sequência de arestas distintas  $fan(v) = \{(v, w_1), \dots, (v, w_s)\}$ , na qual  $s = |fan(v)|$ . Nós dizemos que  $v$  é o centro de  $fan(v)$  e que  $w_i$  são suas folhas.

Um deslocamento da estrutura  $fan(v)$  é um deslocamento circular das cores de suas arestas. Em um deslocamento para frente, em que  $1 \leq i < s$  e  $j = i + 1$ , a aresta  $(v, w_j)$  recebe a cor da aresta  $(v, w_i)$  e a aresta  $(v, w_1)$  recebe a cor da aresta  $(v, w_s)$ . Em um deslocamento para trás, em que  $1 \leq i < s$  e  $j = i + 1$ , a aresta  $(v, w_i)$  recebe a cor da aresta  $(v, w_j)$  e a aresta  $(v, w_s)$ , por sua vez, recebe a cor da aresta  $(v, w_1)$ . Um deslocamento de um  $fan(v)$  constrói uma nova coloração para  $G$  com o mesmo conjunto de cores.

Todos as vizinhanças serão modeladas em termos de coloração de arestas de grafos completos com um número par de vértices. Portanto, a partir de agora, as equipes serão representadas por vértices em grafos  $K_n$ , partidas serão representadas por arestas e rodadas serão representadas pelas cores das arestas. Por exemplo, se uma aresta  $e(v, w)$  é colorido por uma cor  $c$ , significa que a partida entre as equipes  $t_v$  e  $t_w$  será disputada na  $r_c$ -ésima rodada.

O restante deste artigo está organizado da seguinte forma. A Seção 2 introduz as vizinhanças existentes para os Problema de Programação de Tabelas para Competições Esportivas. A descrição detalhada da vizinhança Rotação de Rodadas, seguida da prova de sua corretude é dada na Seção 3. Em seguida, uma análise experimental da vizinhança Rotação de Rodadas é apresentada na Seção 4. Na última seção nós definimos as nossas conclusões.

## 2. Espaço de Busca

Partindo de um grafo  $K_n$ , as colorações iniciais são geradas de duas formas, como descritas em (Costa et al., 2012):

- **Método do Polígono.** Também conhecido como método Canônico, ou 1-fatoração Canônica (de Werra, 1981). Nesta coloração, cada fator  $E^i$ , para  $i = \{1, \dots, n - 1\}$ , é definido da seguinte forma:

$$E^i = \{(n, i)\} \cup \{(f_1(i, k), f_2(i, k)) : k = 1, \dots, n/2 - 1\},$$

com

$$f_1(i, k) = \begin{cases} i + k & \text{para } i + k < n \\ i + k - n + 1 & \text{para } i + k \geq n \end{cases}$$

e

$$f_2(i, k) = \begin{cases} i - k & \text{para } i - k > 0 \\ i - k + n - 1 & \text{para } i - k \leq 0 \end{cases}$$

- **Algoritmo de Vizing.** Do teorema de Vizing (Vizing, 1964) sabemos que o número mínimo de cores necessárias para a construção de uma coloração própria de um

grafo ou é igual a  $\Delta$  ou a  $\Delta + 1$ , no qual  $\Delta$  é o grau máximo do grafo. Utilizando o algoritmo construtivo obtido diretamente da prova do seu teorema, nós podemos colorir as arestas de um grafo  $K_{n-1}$  utilizando  $n - 1$  cores. Uma vez que  $n - 1$  é ímpar, existe uma única cor que não incide em cada um dos vértices. Desta forma, nós adicionamos um novo vértice ao grafo  $K_{n-1}$  e uma nova aresta conectando cada vértice  $v$  de  $K_{n-1}$  ao novo vértice. A nova aresta incidente em cada vértice  $v$  de  $K_{n-1}$  é colorida com a cor que não estava sendo usada para colorir as outras arestas incidentes em  $v$ . Este processo gera uma coloração de  $K_n$  com exatamente  $n - 1$  cores.

No contexto do PTVEP, ambos os procedimentos de construção de colorações iniciais podem violar os limites sobre o número máximo de jogos consecutivos em casa ou fora de casa. As inviabilidades serão reparadas durante o procedimento de busca local.

## 2.1. Vizinhanças Existentes

Diversas estruturas de vizinhança já foram usados em procedimentos de busca local para Problemas de Programação de Tabelas para Competições Esportivas ((Anagnostopoulos et al., 2006), (Ribeiro and Urrutia, 2007) e (Di Gaspero and Schaefer, 2007)). Consideramos as quatro seguintes vizinhanças no contexto da PTVEP, como consideradas em (Costa et al., 2012).

Cada coloração na vizinhança Troca de Equipes ( $N_1$ ) é obtida pela troca das posições de dois vértices distintos  $v_i$  e  $v_j$ . Após o movimento,  $v_i$  terá uma atribuição de cores nas suas arestas que anteriormente pertencia a  $v_j$ , e vice-versa.

Em uma vizinhança Troca de Rodadas ( $N_2$ ) tomamos duas cores distintas,  $c_k$  e  $c_l$ , e efetuamos trocas nas atribuições de cores para todas as arestas nas quais essas cores incidem, ou seja, nós trocamos a dois fatores  $f_{c_k}$  e  $f_{c_l}$  associados às cores  $c_k$  e  $c_l$ . Os grafos resultantes das vizinhanças  $N_1$  e  $N_2$  ainda serão válidos após os movimentos porque, neste movimento, nenhuma troca nas atribuições introduz qualquer conflito de cores nos grafos.

A vizinhança Troca Parcial de Equipes ( $N_3$ ) é uma generalização da vizinhança  $N_1$ . Para qualquer cor  $c_1$  e para quaisquer dois vértices distintos  $v_1$  e  $v_2$ , com  $c_1 \neq c(v_1, v_2)$ , seja  $S$  o subconjunto de cores com cardinalidade mínima incluindo  $c_1$  onde os vértices adjacentes de  $v_1$  e  $v_2$  são os mesmos, ou seja,  $S = \{c_1, \dots, c_k\} \subseteq C$  é mínimo tal que  $c \in S$  e  $\{w \in V : \exists c_j \in S \text{ tal que } c(w, t_1) = c_j\} = \{w \in V : \exists c_j \in S \text{ tal que } c(w, t_2) = c_j\}$ . Um vizinho é obtido trocando a atribuição de cores entre as equipes  $v_1$  e  $v_2$  para o subconjunto de cores  $S$ . Nos casos em que  $S$  é igual a  $C \setminus c(v_1, v_2)$ ,  $N_3$  será equivalente a  $N_1$ .

Para a Troca Parcial de Rodadas ( $N_4$ ), selecione duas cores distintas  $c_1$  e  $c_2$  e considere o ciclo no subgrafo induzido pelas arestas coloridas com essas cores. Em seguida, troque as cores do ciclo para obter uma nova coloração. Se as arestas envolvidos incidem em todos os vértices do grafo ou seja, se as arestas coloridas com as cores  $c_1$  e  $c_2$  formam um circuito Hamiltoniano, então este movimento estará contido na vizinhança  $N_2$ . Como consequência,  $N_4$  e  $N_2$  são vizinhanças equivalentes para P1F.

## 3. Rotação de Rodadas

O movimento  $RR$  utiliza como parâmetros: dois vértices  $v_1$  e  $v_2$  e uma cor  $c_1$ . Além disso, os parâmetros devem estar de acordo com as seguintes condições iniciais:  $v_1, v_2 \in V$ ,  $c_1 \in C$ ,  $v_1 \neq v_2$  e  $c_1 \neq c(v_1, v_2)$ . A partir de um determinada coloração  $\eta$ ,  $RR$  é capaz de

gerar um conjunto  $\Phi$  de colorações, sendo que uma delas é obtida por um movimento da vizinhança  $N_3$ , e retorna a melhor coloração em termos de uma dada função objetivo.

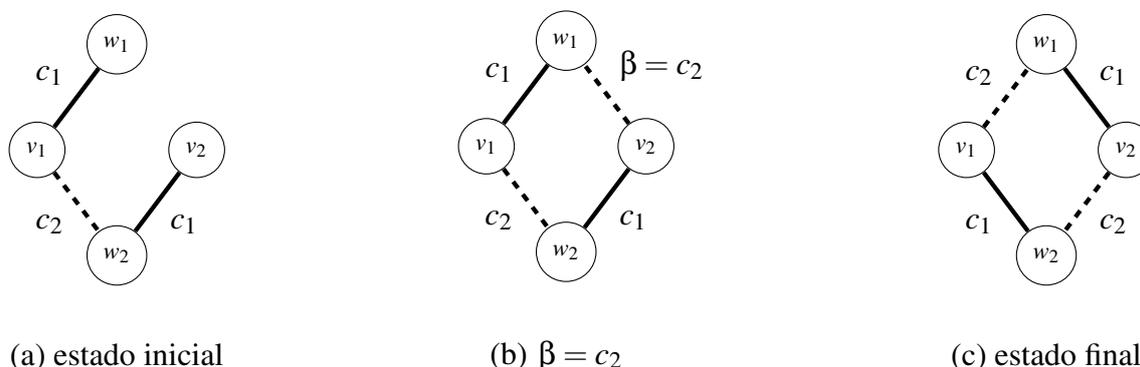


Figura 1. Rotação de Rodadas quando  $\beta = c_2$ .

Inicie com  $\Phi = \emptyset$  e considere  $w_1$  como o vértice adjacente a  $v_1$  tal que  $c_1 = c(v_1, w_1)$ ,  $w_2$  como o vértice adjacente a  $v_2$  tal que  $c_1 = c(v_2, w_2)$ ,  $c_2$  como a cor da aresta  $(v_1, w_2)$  e  $fan(v_1) = \{(v_1, w_1), (v_1, w_2)\}$  e  $fan(v_2) = \{(v_2, w_1), (v_2, w_2)\}$  (ver Figura 1a).

Seja  $\beta$  a cor da aresta  $(v_2, w_1)$ . Neste momento existem dois possíveis cenários de valores para  $\beta$ : ou  $\beta = c_2$  ou  $\beta \neq c_2$ . Se  $\beta = c_2$  (ver Figura 1b) após deslocarmos  $fan(v_1)$  para frente e deslocarmos  $fan(v_2)$  para trás o grafo permanecerá com uma coloração própria e o movimento terminará com a adição da nova coloração obtida (ver Figura 1c) no conjunto  $\Phi$ . Da mesma forma, esta é também uma coloração que pode ser obtida com um movimento da vizinhança  $N_3$ .

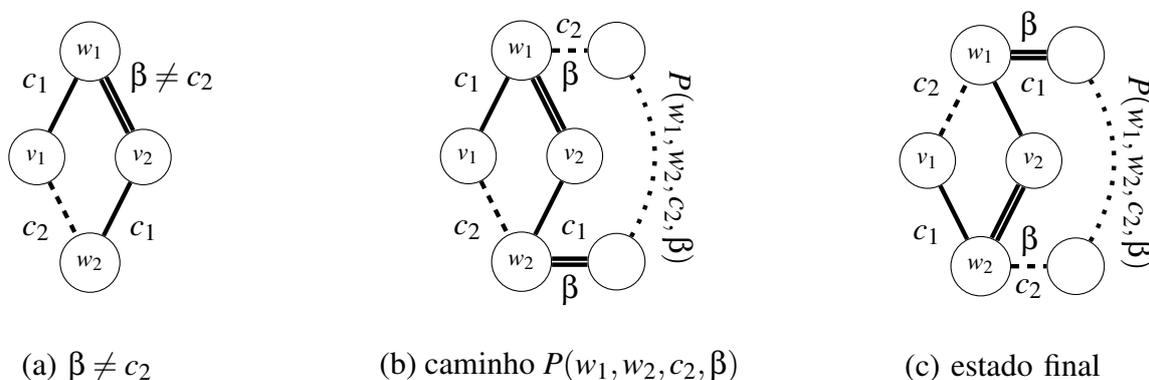
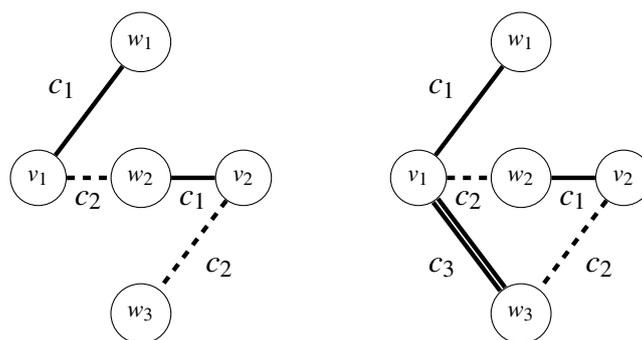


Figura 2. O movimento Rotação de Rodadas quando o caminho  $P(w_1, w_2, c_2, \beta)$  chega no vértice  $w_2$  não passando por  $v_1$  ou por  $v_2$ .

No caso em que  $\beta \neq c_2$  (ver Figura 2a), construímos o caminho  $P(w_1, w_2, c_2, \beta)$  a partir de  $w_1$  até  $w_2$  começando com a cor  $c_2$ . Mais uma vez, dois casos podem surgir. No primeiro caso,  $P(w_1, w_2, c_2, \beta)$  chega ao vértice  $w_2$  com a cor  $\beta$ , não passando por  $v_1$  nem por  $v_2$  (ver Figura 2b). Após deslocarmos  $fan(v_1)$  para a frente, deslocarmos  $fan(v_2)$  para trás e alternarmos o caminho  $P(w_1, w_2, c_2, \beta)$ , o grafo permanecerá com uma coloração própria e o movimento atingirá uma nova coloração, como a que pode ser vista na Figura 2c. Esta nova coloração é então adicionado ao conjunto  $\Phi$ .

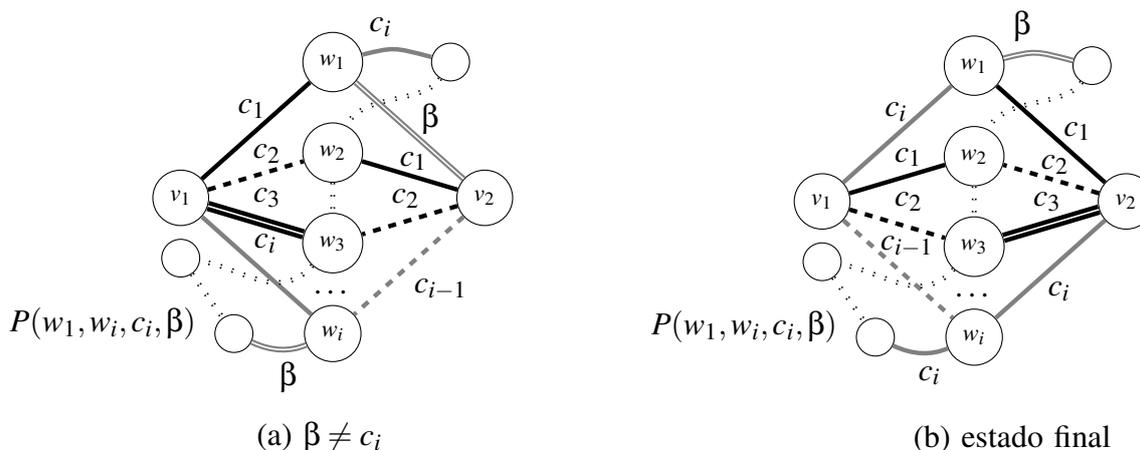
No segundo caso,  $P(w_1, w_2, c_2, \beta)$  passa por  $v_1$  ou por  $v_2$  antes de chegar a  $w_2$  e assim nenhuma coloração é adicionada a  $S$ .



(a) adição do vértice  $w_3$       (b) próximo estado inicial

**Figura 3. A Rotação de Rodadas quando o caminho  $P(w_1, w_2, c_2, \beta)$  passa por  $v_1$  ou por  $v_2$  antes de chegar a  $w_2$ .**

Em qualquer caso, o caminho  $P(w_1, w_2, c_2, \beta)$  é descartado e a coloração original é restaurada. Neste ponto, introduziremos o vértice  $w_3$  à coloração atual tal que  $c(v_2, w_3) = c_2$  (ver Figura 3a). Após definir  $c_3$  como a cor da aresta  $(v_1, w_3)$  e adicionando  $(v_1, w_3)$  à estrutura  $fan(v_1)$  e  $(v_2, w_3)$  à estrutura  $fan(v_2)$ , por fim estabelecemos o próximo estado de partida para o movimento, como mostrado na Figura 3b.



(a)  $\beta \neq c_i$       (b) estado final

**Figura 4. Caso geral do movimento Rotação de Rodadas quando  $\beta \neq c_i$ .**

A partir da nova coloração devemos refazer todos os passos do movimento a partir do teste de igualdade entre as cores  $\beta$  e  $c_3$ . Observe que em qualquer etapa o caminho  $P(w_1, w_i, c_i, \beta)$  partindo do vértice  $w_1$  até o vértice  $w_i$  pode passar através de  $w_j$  para qualquer valor  $1 < j < i$  e ainda ser usado para gerar uma nova coloração (ver Figura 4). Na verdade, o caminho somente não será usado se ele passar a por  $v_1$  ou  $v_2$ .

Seguindo esses passos, a Rotação de Rodadas inclui uma nova coloração em  $\Phi$  sempre que o caminho  $P(w_1, w_i, C_i, \beta)$  chega ao vértice  $w_i$  sem passar por  $v_1$  ou  $v_2$ . Neste caso, deslocamos  $fan(v_1)$  para a frente, deslocamos  $fan(v_2)$  para trás e alternamos  $P(w_1, w_i, c_i, \beta)$ . Em algum momento este movimento adicionará a última coloração  $\phi$  (que também pode ser obtida com a vizinhança  $N_3$ ) quando  $c_i = \beta$ , como mostrado na Figura 5. Em seguida, ele selecionará dentro de  $\Phi$  a melhor coloração em termos de uma dada função objetivo. A Figura 6 mostra o algoritmo para computar o conjunto de

colorações  $\Phi$  geradas pelo movimento da vizinhança  $RR$ .

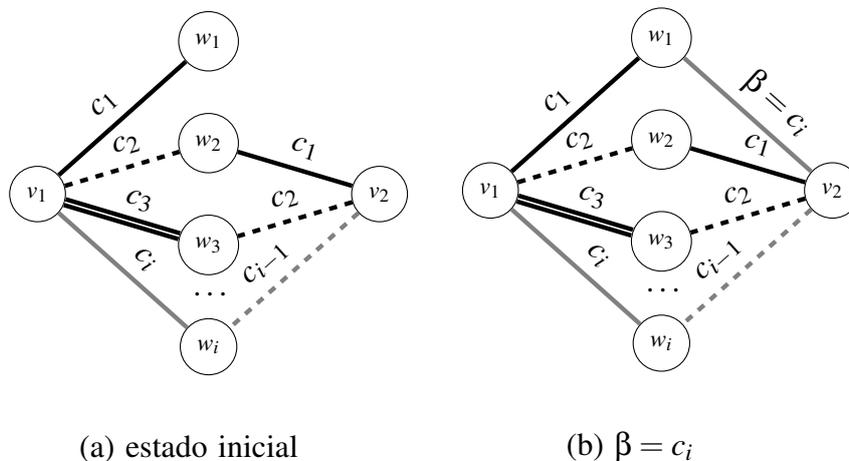


Figura 5. Caso geral do movimento Rotação de Rodadas quando  $\beta = c_i$ .

### 3.1. A Corretude da Vizinhança Rotação de Rodadas

Nesta seção nós mostraremos que a coloração de um grafo permanece própria quando aplicamos um movimento da vizinhança Rotação de Rodadas. Para isso, analisamos as trocas de cores incidentes em cada vértice envolvido no movimento:

- Vértices contidos no caminho  $P(w_1, w_i, c_i, \beta)$ :  
 Observe que as cores  $c_i$  e  $\beta$  incidem em todos esses vértices. Portanto, ao trocarmos as cores  $c_i$  e  $\beta$ , o conjunto de cores incidentes nesses vértices não se altera.
- Vértices  $v_1$  and  $v_2$ :  
 Esses vértices são afetados pelo deslocamento (para frente ou para trás) de algumas das cores incidentes. Uma vez que o deslocamento é circular, o conjunto de cores incidentes não se altera.
- Vertices  $w_j$  for  $1 < j < i$ :  
 Para qualquer vértice  $w_j$ , podemos afirmar que  $c(v_1, w_j) = c_j$  e  $c(v_2, w_j) = c_{j-1}$ . Após o deslocamento de ambas as estruturas  $fan$ , a nova coloração satisfará as igualdades  $c(v_1, w_j) = c_{j-1}$  e  $c(v_2, w_j) = c_j$ . As demais arestas incidentes a  $w_j$  não são modificadas e como resultado a coloração se mantém própria.
- Vertices  $w_1$  and  $w_i$ :  
 Podemos afirmar para o vértice  $w_1$  que  $c(v_1, w_1) = c_1$  e  $c(v_2, w_1) = \beta$ . Após o deslocamento de ambas as estruturas  $fan$ , nós teremos  $c(v_1, w_1) = c_i$  e  $c(v_2, w_1) = c_1$ . Se  $\beta = c_i$ , o movimento não afeta as cores incidentes em  $w_1$ . Caso contrário, a cor  $\beta$  estará ausente e a cor  $c_i$  estará repetida nas cores de duas arestas que incidem  $w_1$ . Ao inverter as cores do caminho  $P(w_1, w_i, c_i, \beta)$  uma aresta mudará de cor, passando de  $c_i$  para  $\beta$ , assim restaurando a corretude da coloração. Uma análise equivalente mostra que a nova coloração se mantém própria em relação ao vértice  $w_i$ .

## 4. Resultados

Nesta Seção nós apresentamos uma série de análises experimentais para avaliar as vizinhanças descritas neste artigo. Como discutido em (Costa et al., 2012), as vizinhanças

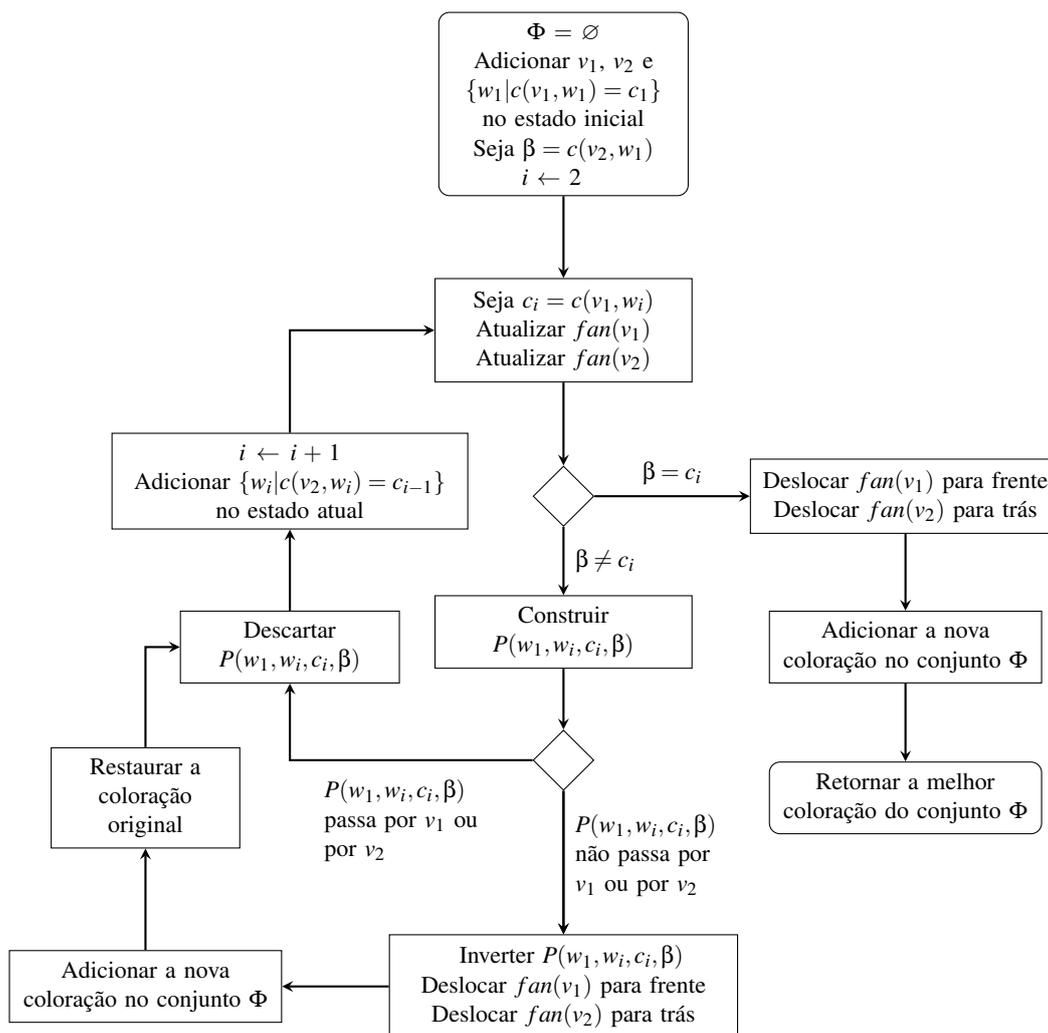


Figura 6. Fluxograma do movimento Rotação de Rodadas.

$N_3$  e  $N_4$  são generalizações das vizinhanças  $N_1$  e  $N_2$ , respectivamente. Portanto, nossa análise considerará apenas as vizinhanças  $N_3$ ,  $N_4$  e  $RR$ .

A fim de testar o desempenho das vizinhanças, usaremos a heurística ILS para o PTVEP proposta por (Costa et al., 2012). A heurística ILS constrói uma coloração inicial, aplicando ou o método do Polígono ou o algoritmo de Vizing, com a qual se inicia o procedimento de busca local. A coloração atual passa por uma função de perturbação que gera uma nova solução. Em seguida, uma outra etapa de busca local é aplicada na nova coloração resultante da perturbação. Por fim, a coloração resultante é comparada com a coloração atual, sendo aceita como a nova coloração se o critério de aceitação pré-definido for cumprido. Caso contrário, uma nova iteração será realizada até que a condição de parada do algoritmo seja alcançada. O Algoritmo 1 mostra o pseudo-código com os principais passos da heurística ILS. Nas linhas 5 e 12,  $d(\eta)$  representa a distância total percorrida pelas equipes durante o torneio e  $v(\eta)$  representa o total de violações devido a equipes que jogam mais de três jogos consecutivos em casa ou fora de casa. Recomenda-se a leitura do trabalho de (Costa et al., 2012) para maiores detalhes.

O mesmo conjunto de instâncias viáveis reportados em (Melo et al., 2009) e (Costa

---

**Algoritmo 1:** Iterated Local Search

---

```
1 início
2    $d \leftarrow \infty$ ;
3    $\eta \leftarrow \text{ConstruaColoraçãoInicial}()$ ;
4    $\eta \leftarrow \text{BuscaLocal}(\eta)$ ;
5   se  $v(\eta) \leq 0$  e  $d(\eta) < d^*$  então
6      $\eta^* \leftarrow \eta$ ;
7      $d^* \leftarrow d(\eta)$ ;
8   repita
9      $\eta' \leftarrow \text{Perturbação}(\eta)$ ;
10     $\eta' \leftarrow \text{BuscaLocal}(\eta')$ ;
11     $\eta \leftarrow \text{CritérioAceitação}(\eta, \eta')$ ;
12    se  $v(\eta) \leq 0$  and  $d(\eta) < d^*$  então
13       $\eta^* \leftarrow \eta$ ;
14       $d^* \leftarrow d(\eta)$ ;
15  até condição de parada seja atingida;
Resultado:  $\eta^*$ 
```

---

et al., 2012) foi utilizado em nossos experimentos computacionais. Os valores de distâncias foram os mesmos para os conjuntos de instâncias circ18 e circ20 do PTV, ambos disponíveis em (Trick, 2014). Todos os experimentos foram executados em uma máquina Ubuntu Linux 12.04 com 24 Intel® Xeon® 2.40GHz processadores e 32 Gbytes de RAM. O critério de parada foi de 300 segundos de tempo de execução. O código da implementação da vizinhança *RR* foi inserido no código da heurística ILS, fornecido pelos autores de (Costa et al., 2012), utilizada neste artigo.

Os resultados numéricos da Tabela 1 foram obtidos de uma média de 30 execuções independentes de cada algoritmo, onde cada execução tem uma duração de 300s. Os nomes das instâncias encontram-se na primeira coluna da tabela. As segunda e quinta colunas da tabela apresentam as médias das distâncias totais obtidas pela heurística ILS em conjunto com a vizinhança  $N_3 \cup N_4$  quando a solução inicial é obtida através do método do Polígono e do algoritmo de Vizing, respectivamente. As terceira e sexta colunas apresentam as médias das distâncias totais obtidas pela heurística ILS em conjunto com a vizinhança *RR* quando a solução inicial é obtida através do método do Polígono e do algoritmo de Vizing, respectivamente. Por fim, as quarta e sétima colunas apresentam os percentuais de melhoria devido ao uso da vizinhança *RR*, para ambos os métodos de geração de colorações iniciais.

A partir dos resultados computacionais pode-se observar que, salvo em raras exceções, o desempenho da heurística é melhorado quando se utiliza a nova estrutura de vizinhança *RR*, em comparação com o uso da estrutura de vizinhança  $N_3 \cup N_4$ . A dominância da vizinhança *RR* sobre a vizinhança  $N_3 \cup N_4$  é confirmada por uma análise de observação emparelhado com nível de confiança de 99,9% (Jain, 1991), independentemente do método adotado para gerar as colorações iniciais.

## 5. Conclusões

A principal contribuição deste trabalho é a proposta de uma nova vizinhança chamada de Rotação de Rodadas. Usando o PTVEP como estudo de caso, mostramos que a vizinhança Rotação de Rodadas, combinada com a heurística ILS, obteve bons resulta-

**Tabela 1. Resultados numéricos da heurística ILS quando o método do Polígono e o algoritmo de Vizing são usados como métodos de geração de colorações iniciais.**

Instância	Método do Polígono			Algoritmo de Vizing		
	$N_3UN_4$	RR	Melhoria	$N_3UN_4$	RR	Melhoria
circ18abal	845.1	842.2	0.35%	846.3	841.3	0.58%
circ18bbal	847.1	842.7	0.52%	846.7	842.6	0.49%
circ18cbal	848.4	840.5	0.93%	842.1	836.6	0.66%
circ18dbal	842.4	840.6	0.21%	845.2	836.7	1.01%
circ18ebal	841.3	837.4	0.46%	842.3	834.1	0.98%
circ18fbal	855.8	845.2	1.24%	848.7	840.0	1.02%
circ18gbal	847.5	839.8	0.90%	840.8	837.9	0.35%
circ18hbal	843.6	833.8	1.17%	839.3	833.9	0.64%
circ18ibal	849.0	841.2	0.92%	847.7	840.1	0.90%
circ18jbal	840.9	829.9	1.31%	837.3	830.5	0.80%
circ18anonbal	866.1	860.3	0.66%	862.3	861.1	0.15%
circ18dnonbal	856.7	853.9	0.32%	855.7	853.5	0.26%
circ18enonbal	863.3	859.3	0.47%	869.7	862.1	0.87%
circ18fnonbal	867.7	863.4	0.49%	872.1	864.3	0.89%
circ18gnonbal	861.3	848.8	1.46%	861.5	855.2	0.74%
circ18hnonbal	864.7	857.0	0.90%	866.0	860.7	0.62%
circ18jnonbal	846.8	840.8	0.71%	847.5	848.1	-0.07%
circ20abal	1409.5	1157.7	17.86%	1165.9	1165.5	0.04%
circ20bbal	1405.4	1156.6	17.70%	1172.5	1157.5	1.28%
circ20cbal	1393.5	1149.6	17.50%	1162.0	1153.5	0.73%
circ20dbal	1409.0	1159.8	17.69%	1170.4	1163.5	0.59%
circ20ebal	1411.2	1160.4	17.77%	1168.7	1164.3	0.38%
circ20fbal	1394.8	1154.4	17.23%	1156.7	1151.5	0.45%
circ20gbal	1405.3	1153.0	17.96%	1164.1	1160.5	0.31%
circ20hbal	1417.3	1169.3	17.50%	1172.6	1168.3	0.37%
circ20ibal	1403.9	1167.5	16.84%	1169.7	1154.5	1.30%
circ20jbal	1405.5	1157.1	17.67%	1166.9	1162.1	0.41%
circ20anonbal	1509.6	1190.6	21.13%	1190.7	1194.7	-0.34%
circ20bnonbal	1464.6	1186.7	18.98%	1193.3	1185.2	0.68%
circ20cnonbal	1488.6	1179.7	20.75%	1187.7	1179.9	0.66%
circ20dnonbal	1495.6	1197.7	19.91%	1204.0	1194.7	0.78%
circ20enonbal	1531.0	1207.9	21.11%	1217.1	1205.5	0.96%
circ20gnonbal	1523.8	1198.5	21.35%	1207.3	1195.4	0.98%
circ20inonbal	1443.6	1173.0	18.74%	1181.8	1176.4	0.46%
circ20jnonbal	1438.3	1166.7	18.88%	1172.0	1166.9	0.44%
Média			9.99%			0.61%

dos para o problema estudado, independentemente da solução inicial ou da instância a ser resolvida.

As análises estatísticas dos resultados computacionais mostraram, com um alto grau de confiança, que a Rotação de Rodadas supera o algoritmo apresentado em (Costa et al., 2012).

O principal ponto fraco da Rotação de Rodadas aparece quando se trata do tempo gasto para avaliar todas as soluções na vizinhança. É claro que a nova vizinhança é maior que as descritas na literatura e que cada vizinho pode ser difícil de ser avaliado. No entanto, os resultados computacionais mostraram que no caso do PTVEP, este alto custo computacional foi compensada com uma melhor exploração do espaço de busca.

Em trabalhos futuros, pretendemos incluir o uso da nova vizinhança na solução de outros problemas de programação de tabelas esportivas. A nova vizinhança pode ser usada em qualquer algoritmo de busca local para quaisquer torneios SRR e ainda para a versão espelhada do PTV (Easton et al., 2001), onde estamos interessados em avaliar o seu desempenho com uma nova abordagem heurística.

## Referências

- Anagnostopoulos, A., Michel, L., Hentenryck, P. V., and Vergados, Y.** (2006). A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 9(2):177–193.
- Costa, F. N., Urrutia, S., and Ribeiro, C. C.** (2012). An ILS heuristic for the traveling tournament problem with predefined venues. *Annals of Operations Research*, 194(1):137–150.
- de Werra, D.** (1981). Scheduling in sports. In Hansen, P., editor, *Annals of Discrete Mathematics (11) Studies on Graphs and Discrete Programming*, volume 59 of *North-Holland Mathematics Studies*, pages 381 – 395. North-Holland.
- Di Gaspero, L. and Schaerf, A.** (2007). A composite-neighborhood tabu search approach to the traveling tournament problem. *Journal of Heuristics*, 13(2):189–207.
- Diestel, R.** (2006). *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Heidelberg.
- Easton, K., Nemhauser, G., and Trick, M.** (2001). The traveling tournament problem description and benchmarks. In Walsh, T., editor, *Principles and Practice of Constraint Programming - CP 2001*, volume 2239 of *Lecture Notes in Computer Science*, pages 580–584. Springer Berlin / Heidelberg.
- Gabow, H. N., Nishizeki, T., Kariv, O. Leven, D., and Terada, O.** (1985). Algorithms for edge-coloring graphs. Technical Report TRECIS-8501, Tohoku University.
- Jain, R. K.** (1991). *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley, 1 edition.
- Kendall, G., Knust, S., Ribeiro, C. C., and Urrutia, S.** (2010). Invited review: Scheduling in sports: An annotated bibliography. *Computers & Operations Research*, 37:1–19.
- Melo, R. A., Urrutia, S., and Ribeiro, C. C.** (2009). The traveling tournament problem with predefined venues. *Journal of Scheduling*, 12(6):607–622.
- Misra, J. and Gries, D.** (1992). A constructive proof of vizing’s theorem. *Information Processing Letters*, 41(3):131–133.
- Rasmussen, R. V. and Trick, M. A.** (2008). Round robin scheduling - a survey. *European Journal of Operational Research*, 188:617–636.



- Ribeiro, C. C. and Urrutia, S.** (2007). Heuristics for the mirrored traveling tournament problem. *European Journal of Operational Research*, 179:775–787.
- Trick, M.** (2014). Challenge traveling tournament problems benchmark. <http://mat.gsia.cmu.edu/TOURN/>. Accessed: 2014-03-28.
- Vizing, V. G.** (1964). On an estimate of the chromatic class of a p-graph. *Discrete Analysis*, 3:25 – 30. in Russian.