

Edge Coloring: A Natural Model for Sports Scheduling

Tiago Januario, Sebastián Urrutia*

*Department of Computer Science, Universidade Federal de Minas Gerais
Av. Antônio Carlos, 6627, Belo Horizonte
31270-901, Minas Gerais, Brazil
Phone: +55 31 3409 5860*

Celso C. Ribeiro

*Department of Computer Science, Universidade Federal Fluminense
R. Passo da Pátria 156, Niterói
24210-240, Rio de Janeiro, Brazil
Phone: +55 21 2629 5646*

Dominique de Werra

*Département de Mathématiques, Ecole Polytechnique Fédérale de Lausanne (EPFL)
CH-1015 Lausanne, Switzerland
Phone: +41 21 69 32562*

Abstract

In this work, we consider some basic sports scheduling problems and introduce the notions of graph theory which are needed to build adequate models. We show, in particular, how edge coloring can be used to construct schedules for sports leagues. Due to the emergence of various practical requirements, one cannot be restricted to classical schedules given by standard constructions, such as the circle method, to color the edges of complete graphs. The need of exploring the set of all possible colorings inspires the design of adequate coloring procedures. In order to explore the solution space, local search procedures are applied. The standard definitions of neighborhoods that are used in such procedures need to be extended. Graph theory provides efficient

Email addresses: januario@dcc.ufba.br (Tiago Januario), surrutia@dcc.ufmg.br (Sebastián Urrutia*), celso@ic.uff.br (Celso C. Ribeiro), dominique.dewerra@epfl.ch (Dominique de Werra)

*Corresponding author

tools for describing various move types in the solution space. We show how formulations in graph theoretical terms give some insights to conceive more general move types. This leads to a series of open questions which are also presented throughout the text.

Keywords: OR in sports, Scheduling, Graph Theory, Edge Coloring, Local Search.

1. Introduction

Multiple agents of sports events, such as organizers, players, referees, fans, journalists, medical doctors, and airlines play an important role in professional sports leagues and tournaments. Major events such as the Olympic Games and the Football World Cup create thousands of jobs and economic opportunities to their hosts [19, 29].

Many discrete problems in several areas can be formulated in graph theoretical terms and then solved using graph algorithms (see, e.g., [12]). In this work, we show how basic concepts of graph theory provide a natural and very adequate tool for formulating and solving some of the problems which tournament organizers face. We will start by taking into account only some of the most frequent requirements.

We assume throughout the text that we have a round robin tournament involving an even number n of teams. Every game involves two teams, say i and j . Therefore, it is natural to associate each team with a vertex of a graph and every game involving i and j with an edge (i, j) of this graph. Figure 1 shows an edge-colored complete graph corresponding to a tournament of $n = 4$ teams.

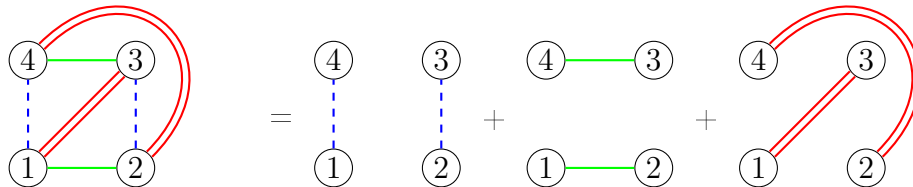


Figure 1: Example of a tournament represented by an edge-colored complete graph with $n = 4$ vertices.

The primary goal of tournament scheduling is to find an assignment of

each game to some time slot (or round) such that each team plays at most one game in each round and the total number of rounds is as small as possible. If each team has p games to play, any schedule clearly needs at least p rounds. For convenience, we associate each round with a specific color. Therefore, each schedule corresponds to an assignment of a color to each edge of the corresponding graph.

Games are divided into subsets F_s , each of them formed by all games assigned to a specific round s . Figure 1 illustrates a tournament schedule with $n = 4$ teams represented by a complete graph K_4 . Edges of subset F_1 are represented by dashed lines, while those of F_2 are represented by straight solid lines and edges of F_3 are represented by double solid lines. Each of these subsets represents the set of games played in a given round. We observe that, in this case, in every subset F_s of games each team is matched with exactly one other team, forming a perfect matching.

A tournament is said to be *compact* if each team plays exactly once in each round. In a very typical tournament format among n teams, each team plays exactly once (resp. twice) against all other teams in a given number of rounds. This type of competition format is called a single round robin (SRR) (resp. double round robin (DRR)) tournament. In view of simplicity, we consider the problem of scheduling a compact SRR tournament involving an even number of teams. The ideas and results presented here for SRR tournaments can also be easily adapted to handle the case of DRR tournaments. Most of the results for tournaments with an even number of teams may be extended to tournaments with an odd number of teams by simply adding a dummy team.

A basic problem in sports scheduling consists in defining the round and the venue in which each game of a tournament will be played at. Even though it may seem simple to schedule a tournament at first glance, the addition of simple constraints transforms this task into challenging combinatorial optimization problems belonging, in most cases, to the NP-hard class [32]. A great research effort has been made to devise improved algorithms, resulting in better schedules obtained in smaller computational times. For many types of problems, instances with no more than 20 teams are considered as large-scale and approximate approaches are often used to obtain good results [29]. In the related literature, it is possible to find methods that range from integer and constraint programming to metaheuristics approaches and various hybrid algorithms [3, 10, 11, 22, 30, 31].

The text is organized as follows. Sections 2 and 3 lay out the terms and

definitions, respectively, in graph theory and sports scheduling that will be used in the remainder of the text. Section 4 tackles the problem of generating SRR schedules from scratch. Section 5 describes in edge colorings terms the neighborhoods commonly used by local search sports scheduling algorithms. Section 6 presents results on the connectivity of the solution space for the classical neighborhoods. Section 7 analyzes scheduling problems when considering the venue of each game and sketches some formulations in terms of list colorings of graphs. Concluding remarks and suggestions for further research are made in the last section.

2. Elements of Graph Theory

A graph $G = (V, E)$ is defined by two sets V and E , in which the n elements of V are the vertices of the graph G and the m elements of E are its edges. Two distinct vertices $u, v \in V$ joined by an edge $e = (u, v) \in E$ are said to be adjacent. In this case, both u and v are the endpoints of e and e is said to be incident to both u and v . Two edges are adjacent if they are incident to the same vertex. The degree $d(v)$ of a vertex v is the number of edges incident to it and the number $\Delta(G) = \max\{d(v) : v \in V\}$ is the maximum degree of a vertex in G . In addition, $\delta(v)$ stands for the set of edges incident to vertex v .

A *chain* is a sequence of edges with common vertices, where each edge in the sequence has one vertex in common with the preceding edge and the other in common with the next one. The length of a chain is the number of its edges. A graph is *connected* if for any pair of vertices there is a chain linking them. A *cycle* is a chain formed by at least three edges starting and ending in the same vertex. A *Hamiltonian cycle* is a cycle that visits every vertex of the graph exactly once.

A *proper edge coloring* of G is a mapping $c : E \rightarrow \mathcal{C}$ so that $c(e_i) \neq c(e_j)$ for any adjacent edges e_i and e_j , where the elements of \mathcal{C} are the available colors. A *k -coloring* is a proper edge coloring with $|\mathcal{C}| = k$. The *chromatic index* of a graph G is defined as $\chi'(G) = \min\{k : \text{a } k\text{-coloring of } G \text{ exists}\}$. A color is present in a vertex if any edge incident to this vertex has that color; otherwise, the color is free (or available) in that vertex. A *conflict* is characterized by the existence of two edges with the same color incident to a common vertex. In other words, a proper edge coloring is a color assignment without conflicts to the edges of a graph. A proper partial edge coloring is a color assignment without conflicts such that some edges of G

may be uncolored. If $V' \subseteq V$ and $E' \subseteq E$, then $G' = (V', E')$ is a subgraph of $G = (V, E)$ (and G a supergraph of G'), written as $G' \subseteq G$. If $G' \subseteq G$ and $G' \neq G$, then G' is a proper subgraph of G . If $G' \subseteq G$ and G' contains all the edges $(v, w) \in E$ with $v, w \in V'$, then G' is the subgraph of G induced by the subset of vertices V' . The subgraph of G spanned by a subset E' of edges contains exactly all the edges in E' and all the endpoints of edges in E' . A *connected component* of G is an inclusion-wise maximal connected induced subgraph of G .

A *one-factor* of G is a set of edges $F \subseteq E$, such that all vertices in V have degree equal to one in the subgraph $G' = (V, F)$. In this case, F is also called a *perfect matching* in G . One-factors may only exist for graphs with an even number of vertices. A *one-factorization* of G is a partition of E into one-factors, i.e., a set $\mathcal{F} = \{F_1, F_2, \dots, F_k\}$ of disjoint one-factors such that $\cup_{i=1}^k F_i = E$. A one-factorization is said to be *perfect* when $G(V, F_i \cup F_j)$ is a Hamiltonian cycle for any two distinct one-factors F_i and F_j , for $i, j = 1, \dots, k$.

Two one-factorizations $\mathcal{F} = \{F_1, F_2, \dots, F_k\}$ and $\mathcal{H} = \{H_1, H_2, \dots, H_k\}$ of a graph $G = (V, E)$ are called *isomorphic* if there is a bijection $\varphi : V \rightarrow V$ such that $H_i = \{(\varphi(x), \varphi(y)) : (x, y) \in F_i\}$ for each $i = 1, \dots, k$ [9]. The concepts of a proper edge coloring and that of a one-factorization are equivalent whenever each color is present in each vertex of the graph. In this case, each color relates to a one-factor. In the remainder of this work, the term proper edge coloring refers to one-factorizations.

3. Elements of Sports Scheduling

Due to several research groups working in parallel on sports scheduling problems, it is not uncommon to find in the literature different names corresponding to the same concept [19, 29, 30].

A round robin tournament is a competition involving n different teams indexed by $i \in \{1, \dots, n\}$ in which all teams play against each other a fixed number g of games. The games of the tournament must be scheduled in a number r of rounds in such a way that each team plays at most one game in each round. As mentioned, we assume that the number of teams is even, which is usually the case in real competitions.

An SRR tournament is a competition in which all teams face each other exactly once (i.e., $g = 1$). A DRR tournament is a competition in which all teams face each other twice (i.e., $g = 2$) and any two teams meet once in each

other's home venue. We shall assume here that all round robin tournaments are compact. This implies that $n/2$ games are held in each round.

The graph corresponding to an SRR tournament is a complete graph K_n , i.e., one edge between any two vertices. The graph corresponding to a DRR tournament has two edges between any pair of vertices since there are two games between any pair of teams.

A common way to describe the schedule of a round robin tournament is by using a *timetable* as the one depicted in Figure 2. A *timetable* is a matrix of n rows and r columns, in which n is the number of teams and r is the number of rounds of the tournament. Each row refers to a team i , each column represents a round k and each cell (i, k) represents the opponent of team i in round k , for all $1 \leq i \leq n$ and $1 \leq k \leq r$. A sign associated with the opponent denotes the location of the match. The positive sign (+) indicates that team i will play at home against its opponent, i.e., this is a *home-game* for i . A negative sign (-) indicates that team i will play away against its opponent, i.e., this is an *away-game* for i .

A mirrored double round robin tournament (MDRR) is a special type of a DRR tournament consisting of a simple SRR in the first $n - 1$ rounds (first leg), followed by the same tournament with reversed venues in the last $n - 1$ rounds (second leg).

First leg					Second leg				
-6	+2	-3	+4	+5	+6	-2	+3	-4	-5
-5	-1	+6	+3	-4	+5	+1	-6	-3	+4
-4	+5	+1	-2	-6	+4	-5	-1	+2	+6
+3	-6	+5	-1	+2	-3	+6	-5	+1	-2
+2	-3	-4	+6	-1	-2	+3	+4	-6	+1
+1	+4	-2	-5	+3	-1	-4	+2	+5	-3

Figure 2: A *timetable* representation of a DRR tournament schedule for an instance with six teams and ten rounds.

Concerning the geographical positions and the distances traveled by the teams, we assume that each team is associated with a home venue located in its city and the distances between the venues are known beforehand and given by an $n \times n$ distance matrix D . We assume that each team starts the tournament at its home venue. Whenever a team plays two consecutive

away games, it goes directly from the city of the first opponent to that of the second, without returning to its own home city. At the end of the tournament, if a team played its last match away from home, then it must return to its home venue.

We say that there is a *break* when a team plays two consecutive home games or two consecutive away games. A *road trip* is a maximal sequence of consecutive away games for a team, while a *home stand* is a sequence of consecutive home games. In tournaments among teams spread over large regions, it is advantageous to schedule the games with a number of consecutive away matches to reduce the distance traveled. This creates consecutive breaks on away games.

The situation in which two teams play two consecutive matches against each other is called a *repeater* and should be avoided in some DRR competitions. Such constraint, called *no-repeater*, is common in some DRR competitions.

The traveling tournament problem (TTP), introduced in [10], is one of the most studied optimization problems in the sports scheduling literature. It can be defined as follows: given two integer numbers L and U , an even number n of teams and an $n \times n$ distance matrix, find a schedule for a DRR tournament on the n teams avoiding repeaters, in which the total distance traveled by the teams is minimized and the size of any home stand and any road trip is bounded by L and U , respectively.

A schedule T has $n(n-1) - B(T)/2$ trips, where $B(T)$ is the number of breaks. Observe that $B(T)/2$ is equal to the number of home breaks, i.e., the number of times any team plays two consecutive home games. If instead of minimizing distances one is interested in minimizing the number of trips, then the problem amounts to maximizing the number of breaks $B(T)$ over all schedules T [33].

De Werra [5] was one of the first authors to discuss the application of graph theory concepts (with emphasis in edge coloring) to sports scheduling problems. Lewis and Thompson [22] presented models based on vertex coloring to solve the traveling tournament problem [10]. For a comprehensive list of graph theoretical approaches to sport scheduling we refer to [19].

As mentioned in the introduction, there is a natural one-to-one correspondence between any schedule of an SRR tournament and a proper edge coloring of the corresponding complete graph with n vertices using $n-1$ colors. This is why we recall in the next section some concepts and procedures related to edge coloring.

4. Construction Techniques for SRR Schedules

The circle method, also called the polygon method or the canonical procedure, generates the *canonical* edge coloring for a complete graph K_n . It is often used to generate initial solutions for heuristics tackling round robin scheduling problems [3, 6, 31, 35]. The circle method has been used in some real situations, such as the schedule of the Spanish First Division (BBVA League) [14].

We denote the set of teams by $T = \{1, 2, \dots, n\}$ and the set of rounds by $R = \{1, 2, \dots, n-1\}$. To schedule a tournament following the circle method, we first place points numbered from 1 to $n-1$ at equal distances on a circle. We place an additional point, numbered by n , in the center of the circle. Then we associate a team i to each point i for $1 \leq i \leq n$.

In order to determine the $n/2$ games that will be played in round $r = 1, 2, \dots, n-1$ of the tournament, the circle method considers that a straight line is drawn between point n and point r of the circle together with all possible straight lines orthogonal to (r, n) that join two other points previously defined on the circle. All matches that can be obtained this way determine the games in the r^{th} round of the tournament. Figure 3 illustrates the application of this method for $n = 6$.

Therefore, we can represent the circle method by the following opponent schedule function [26]:

$$\Upsilon(t, r) = \begin{cases} r & \text{if } t = n, \\ n & \text{if } t = r, \\ (2r - t) \bmod '(n - 1) & \text{otherwise,} \end{cases} \quad (1)$$

where $a \bmod '(b)$ is defined as follows:

$$a \bmod '(b) = \begin{cases} a \bmod (b) & \text{if } a \bmod (b) \neq 0, \\ b & \text{if } a \bmod (b) = 0. \end{cases}$$

Function $\Upsilon(t, r)$ computes the opponent of team $t \in T$ in round $r \in R$ based on the circle method. Table 1 shows a schedule constructed with the circle method for a tournament with $n = 6$ teams.

As an attempt to obtain different initial schedules for each run, the order of the rounds and the initial placement of teams in the polygon could be randomized. However, all colorings obtained with this strategy would still be isomorphic to each other.

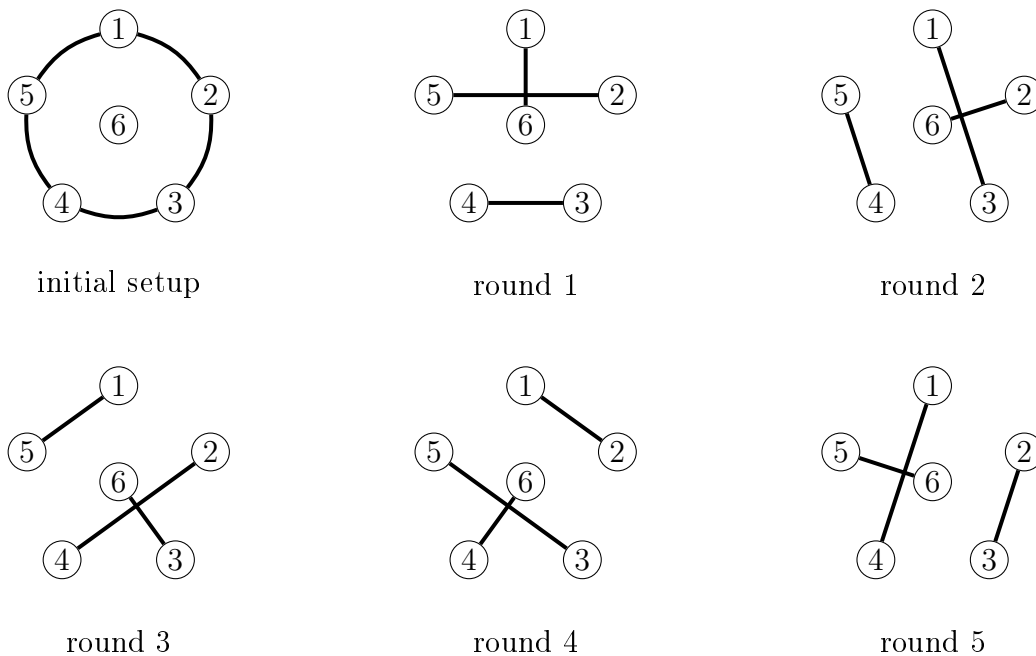


Figure 3: Circle method for K_6 .

Therefore, other techniques should be used for generating a proper edge coloring of the complete graph using exactly $n - 1$ colors, instead of the circle method, for constructing distinct SRR schedules. Different procedures have been developed to create schedules with particular structures, see, e.g., [4] where a construction technique using a division of a league into two or more sub-leagues is described.

We present a framework for the construction of an arbitrary edge coloring of K_n . Vizing [34] proved that every simple graph G has a proper coloring with at most $\Delta(G) + 1$ colors. This bound is not tight in the context of an SRR scheduling, since the circle method itself constructs a proper coloring of a complete graph K_n with $n - 1 = \Delta(K_n)$ colors.

Many of the different proofs of Vizing's theorem given in the literature [7, 8, 13, 15, 24] are constructive. They show how to obtain a proper $(\Delta(G) + 1)$ -coloring of an arbitrary graph. Some of these proofs also show how to color an additional uncolored edge of a partially colored graph (which may require changing the color of some already colored edges) never exceeding $\Delta(G) + 1$ different colors. The procedure can be repeated until all edges are colored.

Let $e_0 = (w, v_0)$ be an uncolored edge of a graph $G = (V, E)$ partially

$\Upsilon(t, r)$		r				
		1	2	3	4	5
t	1	6	3	5	2	4
	2	5	6	4	1	3
	3	4	1	6	5	2
	4	3	5	2	6	1
	5	2	4	1	3	6
	6	1	2	3	4	5

Table 1: Schedule constructed by the circle method for a tournament with $n = 6$ teams.

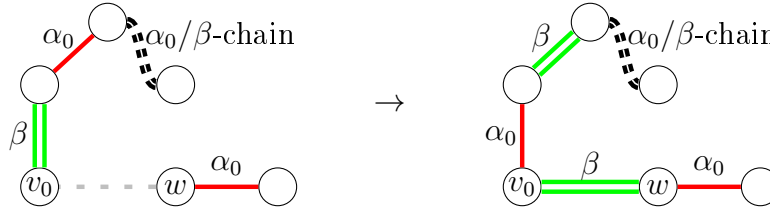


Figure 4: Color β is free at vertex v_0 after the switch.

colored with no more than $\Delta(G) + 1$ colors. Observe that since both w and v_0 have at most $\Delta(G) - 1$ colored incident edges, there are at least two available colors in both of them. We denote the set of colors available at vertex $v \in V$ by $Free(v)$. If $Free(w) \cap Free(v_0) \neq \emptyset$, then we can simply choose any color from the intersection to color edge e_0 .

Now, we assume that $Free(w) \cap Free(v_0) = \emptyset$. Let $\alpha_0 \in Free(v_0)$ and $\beta \in Free(w)$ be two free colors at vertices v_0 and w , respectively. Let P be a maximal chain that starts from v_0 so that its edges are alternately colored with colors β and α_0 . Such a chain will be called an α_0/β -chain. Two cases may arise: P ends at a vertex different from w or P ends at w . In the first case, the coloring may be augmented by exchanging the colors of the edges along the chain P and coloring edge e_0 with β that is now free in v_0 , as shown in Figure 4.

Next, we assume that the chain P ends at vertex w . If we exchange the colors of the edges of P , then $\beta \in Free(v_0)$ but $\beta \notin Free(w)$, as shown in Figure 5.

Let v_1 be the vertex adjacent to w in P and $e_1 = (w, v_1)$ be the edge which is colored with α_0 . At this point, the color α_0 is removed from edge

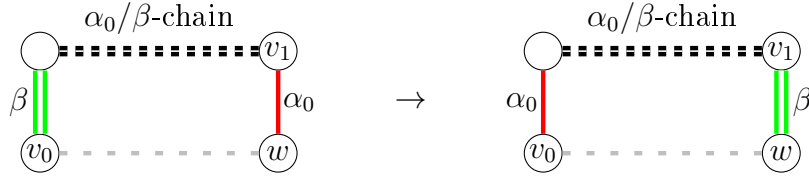


Figure 5: A situation where the path P ends at vertex w .

e_1 and edge e_0 is colored with that same color, as Figure 6 illustrates.

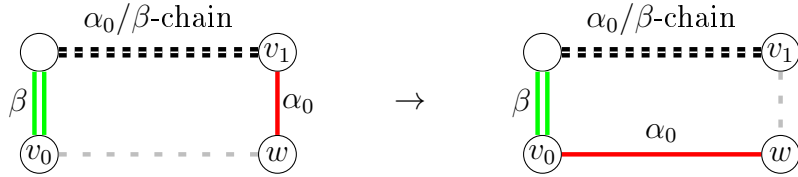


Figure 6: Color α_0 moves from edge (w, v_1) to edge (w, v_0) .

Now, the problem consists in re-coloring edge e_1 . The same procedure applied to edge e_0 can be used to assign a color to edge e_1 . If $Free(w) \cap Free(v_1)$ is also empty, then the color to be selected from $Free(v_1)$ must be different from α_0 in order to avoid cycling. Of course, this color always exists, since $|Free(v_1)| \geq 2$.

We may continue this way until $Free(w) \cap Free(v_i)$ is not empty or the chain P does not end at w . The rest of the proof of Vizing's theorem [15] shows that this will eventually be the case after at most $\Delta(G)$ iterations. This proof of Vizing's theorem immediately yields an $O(|E||V|\Delta)$ time algorithm to obtain a proper $(\Delta(G) + 1)$ -coloring of a simple graph. A more efficient algorithm is described in [13].

In this work, we are interested in finding proper edge colorings of complete graphs K_n , using at most $\Delta(K_n)$ colors to represent the $n - 1$ rounds of the tournament. To obtain an edge coloring with $\Delta(K_n)$ colors, we consider the complete graph K_{n-1} with vertex set $V = \{v_1, v_2, \dots, v_{n-1}\}$. Then, color K_{n-1} with $n - 1$ colors following the procedure derived from any proof of Vizing's theorem. Next, add vertex v_n to the graph, and connect every vertex in $V = \{v_1, v_2, \dots, v_{n-1}\}$ to v_n with an uncolored edge. Since graph

K_{n-1} was edge-colored with $n - 1$ colors and each vertex has its degree equal to $n - 2$, there is an available color at each vertex $V = \{v_1, v_2, \dots, v_{n-1}\}$. To see that the missing colors are distinct, observe that each color is used in $\lfloor (n - 1)/2 \rfloor$ edges and, as a consequence, each of the $n - 1$ colors is missing in exactly one of the $n - 1$ vertices. To complete the coloring of graph K_n , we just need to assign each edge $(v_1, v_n), (v_2, v_n), \dots, (v_{n-1}, v_n)$ to the color that is available at v_1, v_2, \dots, v_{n-1} , respectively. This brings about a proper edge coloring of K_n with exactly $\Delta(G)$ colors. We refer to this procedure as the Vizing algorithm.

An example of how this procedure can be applied to obtain an edge coloring for K_4 is depicted in Figure 7.

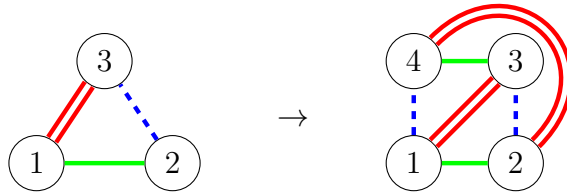


Figure 7: Finding an edge coloring of K_4 based on an edge coloring of K_3 .

The edge coloring of K_n with $n - 1$ colors constructed by Vizing's algorithm depends on the order in which the yet uncolored edges are considered. In fact, one may get any possible edge coloring, since there is always an order for which Vizing's algorithm produces a given edge coloring of K_n . This is clearly not the case for the circle method.

5. Neighborhoods in the Landscape of Edge Colorings

In this section we will exploit thoroughly the formulation in terms of graphs to describe, interpret and generalize some classical moves from one solution (an edge coloring) to a neighbor solution (a modified edge coloring). These original formulations leading to generalized moves will exhibit the power and the promises of the tools provided by graphs.

Let us first illustrate four neighborhoods commonly used in local search heuristics for SRR and DRR problems formulated in edge colorings terms. For didactic purposes, our description will be based on the SRR case. These neighborhoods have received different names, e.g., in [1, 6, 31]. Here, they are

called Team Swap (TS), Round Swap (RS), Partial Team Swap (PTS), and Partial Round Swap (PRS). Except for neighborhoods whose moves switch home-away assignments, no other neighborhood, which is not a special case or a combination of these neighborhoods, has ever been proposed in the literature.

Each solution in the TS neighborhood is obtained by exchanging the opponents of a pair of teams i and j over all rounds. This neighborhood can be seen as the exchange of the labels of two vertices in the graph that represents the tournament.

In a move within the RS neighborhood, all games assigned to a round r_i are moved to a round r_j , and vice-versa. This is equivalent to exchanging two different colors in the associated edge-colored graph.

The colorings resulting from moves in neighborhoods TS and RS remain valid after the moves, because they do not introduce color conflicts in the graph. Every move in these neighborhoods leads to edge colorings that are isomorphic to the original one. In consequence, such moves may not be very effective to find improving solutions.

In a Partial Round Swap (PRS) move, the opponents of team t in rounds r_i and r_j are exchanged and a repair chain must be applied in order to remove the resulting violations (see [6] for further details). In graph theoretical terms, a PRS move consists of two steps. Initially, any two distinct colors are selected and one considers a cycle in the subgraph spanned by the edges with these two colors (edges in two distinct factors). Next, the colors of the edges in the cycle are exchanged, leading to a new coloring. Figure 8 illustrates a move in the PRS neighborhood, in which the colors of the edges in the cycle formed by vertices 5, 6, 7, and 8 have been exchanged.



Figure 8: A move in neighborhood PRS.

If the edges involved in a move in the PRS neighborhood form a Hamiltonian cycle, then such move provides the same solution as a move in the RS neighborhood, obtaining a coloring that is isomorphic to the original one. If the cycle is not Hamiltonian, then the coloring obtained may not be

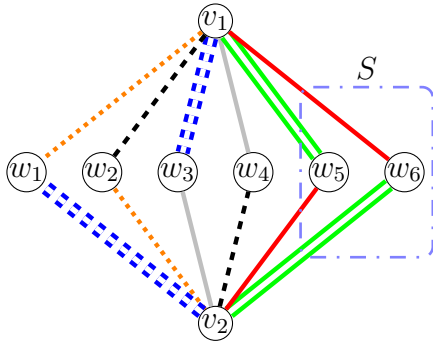


Figure 9: Subgraph $K_{2,n-2}$ with $S = \{w_5, w_6\}$

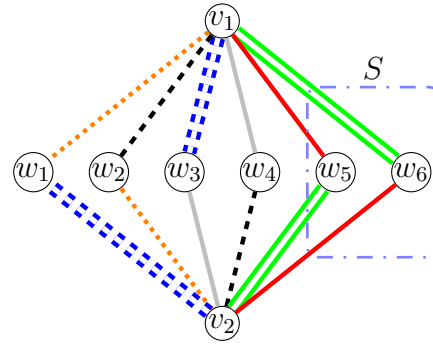


Figure 10: Subgraph of Figure 9 after the exchange of the colors in the edges incident to S .

isomorphic to the original one.

In a Partial Team Swap (PTS) move, the opponents of the teams in a given round are exchanged and a repair chain must be used in order to remove the resulting violations (see [6] for further details). In graph theoretical terms, a PTS move consists of the following steps. First, select two distinct vertices v_1 and v_2 of the complete edge colored graph. Next, consider the subgraph spanned by the edges of the symmetric difference between $\delta(v_1)$ and $\delta(v_2)$, i.e., every vertex of the graph and all edges that connect v_1 and v_2 with all other vertices. This subgraph is isomorphic to the complete bipartite graph $K_{2,n-2}$ and colored with $n-2$ colors, (all but that of the edge (v_1, v_2)). Following, compute a subset of vertices $S \subseteq V \setminus \{v_1, v_2\}$ such that the set of colors assigned to edges joining v_1 to vertices in S is equal to the set of colors assigned to edges joining v_2 to vertices in S .

Let C_1, \dots, C_p be the chains of length two linking v_1 and v_2 through S . Next, exchange the color assignment of edges in each chain C_i , for $i = 1, \dots, p$. Figure 9 shows such a subgraph $K_{2,n-2}$ with $S = \{w_5, w_6\}$ and Figure 10 illustrates the resulting colored subgraph after the exchange of the colors in the chains. If $S = V \setminus \{v_1, v_2\}$, the move is equivalent to a TS move.

Both partial swap neighborhoods, PRS and PTS, may be applied to any sport round robin scheduling problem but they are currently used almost only for the TTP and its variants.

A more general move would be to consider edge disjoint alternating chains C_1, \dots, C_p of even length, possibly larger than two between vertices v_1 and v_2 . The set of colors on edges of C_1, \dots, C_p incident to vertex v_1 is, as in

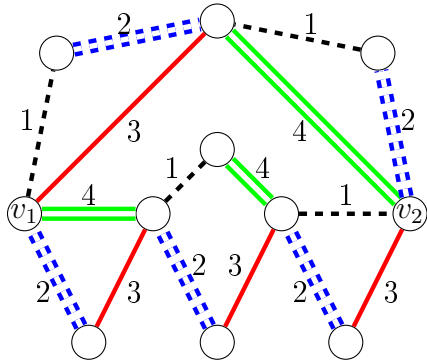


Figure 11: Initial coloring.

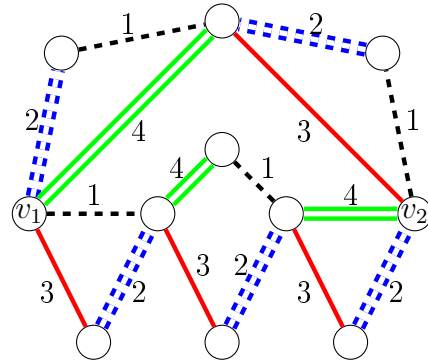


Figure 12: Final coloring.

PTS, the same as the set of those incident to v_2 . Exchanging colors in each one of the alternating chains gives a new proper coloring that could not be directly obtained with PTS.

Figure 11 shows the initial coloring as an example of this type of move. C_1 is a 1/2-chain of length 4, C_2 a 2/3-chain of length 6, C_3 a 3/4-chain of length 2 and C_4 a 4/1-chain of length 4. Note that some of these chains have common vertices. The resulting coloring is given in Figure 12.

Januario and Urrutia have shown in [17] that a generalization of the PTS neighborhood increases the solution space connectivity of the problem in hand and allows a local search heuristic to obtain better results.

The generalized moves described above have shown how the formalism and the language of graph theory are helpful to describe and to understand the local modifications of schedules generated when moves are performed. They are also useful for designing procedures to create schedules from scratch. In the related literature, some neighborhoods are described as intricate structures, as ejection chains, that do not generally provide a deeper insight of what is really happening.

6. Connectivity of the Solution Space

The size of the solution space of SRR scheduling problems is large even when compared with the size of the solution space of classical combinatorial optimization problems as the traveling salesman problem. There is no known formula for the number of different proper edge colorings of K_n . The number of non-isomorphic one-factorizations of K_n increases very fast

with n . Dinitz et al. [9] computed the complete set of 526,915,620 non-isomorphic one-factorizations of K_{12} . Kaski and Ostergard [18] reported a total of 1,132,835,421,602,062,347 non-isomorphic one-factorizations of K_{14} .

In a perfect one-factorization, or perfect proper edge coloring (PPEC), the sub-graph containing all vertices and edges belonging to the union of any two one-factors is a Hamiltonian cycle. Kotzig [21] conjectured that a PPEC of K_n exists for every even value of $n > 2$. Since 2009, when Wolfe [36] found a PPEC of K_{52} after an exhaustive search that took more than 400 days of computing time, K_{56} is the smallest complete graph for which the existence of a PPEC remains an open problem.

A PPEC may not be desirable as an initial solution for local search algorithms of SRR scheduling problems. If a given proper edge coloring is perfect, then any move in the PRS neighborhood has the same effect as a move in the RS neighborhood, which generates colorings that are isomorphic to the original one.

The circle method, introduced in Section 4, is a simple strategy to obtain initial solutions for any local search heuristic for SRR scheduling problems. However, it is able to generate only one single type of edge coloring (the canonical edge coloring) for a complete graph. In fact, the circle method generates PPECs for every complete graph K_n for which $n = p + 1$, where p is a prime number. Therefore, whenever an initial solution is generated by the circle method, when $n = p + 1$, a move in the PRS neighborhood is equivalent to a move in the RS neighborhood. In consequence, these neighborhoods always lead to colorings which are isomorphic to the canonical edge coloring.

One may argue that the PTS neighborhood may be used to escape from canonical colorings. However, in some cases, this neighborhood suffers from a similar property. Partial team swaps are equivalent to team swaps, whenever the set S of vertices computed as described in Section 5 is equal to $V \setminus \{v_1, v_2\}$. One could ask if all possible partial team swap moves are equivalent to team swap moves, for some value of n for which the canonical coloring is perfect. In order to provide an answer to this conjecture, we applied an exhaustive search procedure to check if that was the case for each even value of $n = p + 1 \leq 100$, where p is a prime number. We found that the partial team swap neighborhood is, in fact, equivalent to the team swap neighborhood for canonical edge colorings of K_n for $n = 4, 6, 12, 14, 20, 30, 38, 54, 60, 62, 68$, and 84.

The above result shows that for all above mentioned values of n , any local search heuristic using an initial solution constructed by the circle method will

be trapped in a small portion of the solution space formed by all colorings isomorphic to the canonical one. In such cases, applying Vizing’s algorithm instead of the circle method, or using the ideas sketched in Section 5 for extending the neighborhoods, may improve the effectiveness of the local search by exploring a broader portion of the search space.

An experiment using Vizing’s algorithm to build initial solutions for the traveling tournament problem with predefined venues [23] was reported by Costa et al. [3]. The authors evaluated instances with 18 and 20 teams. For the instances with 20 teams, an improvement of almost 19% on average in the cost function was obtained only because Vizing’s algorithm was used instead of the circle method for constructing the initial solution. Their experiment revealed that the choice of the coloring method based on Vizing’s theorem can lead to better results for the considered problem.

7. Home-Away Assignments

Let us now examine the situation of a league in which each team has its own venue (a stadium) and each game (i, j) is played either at the venue of the team i or at that of the team j . For simplicity, we deal with the case of SRR tournaments. The extension to DRR tournaments can be easily carried out.

It is natural to represent a game in which i will play against j in the venue of the latter by an oriented arc from i to j . This way the schedule now corresponds, in graph theoretical terms, to an oriented edge coloring. An illustration for a league of six teams is given in Figure 13. The timetable on the left indicates in row i and column s which is the opponent of the team i in round s . The home-away pattern (HAP) set on the right indicates in row i and column s whether i plays in round s a home-game (H) or an away-game (A).

As soon as we consider the venues of the teams, difficulties arise regarding fairness. Ideally, one would like to have for each team a sequence of games which would be perfectly alternating between home and away-games. This is clearly impossible if $n > 2$. Two teams having exactly the same sequence of H’s and A’s in the HAP set could never play against each other. Therefore, in any real schedule, there must be some breaks in the perfect alternation of H’s and A’s for some teams. A break was defined in Section 3 as the occurrence of two consecutive H’s or two consecutive A’s for a team. In Figure 13, team 3 has a break in round 2 and so has team 4. In fact, the timetable and the

	<i>round</i>				
	1	2	3	4	5
1	6	3	5	2	4
2	5	6	4	1	3
3	4	1	6	5	2
4	3	5	2	6	1
5	2	4	1	3	6
6	1	2	3	4	5

	<i>round</i>				
	1	2	3	4	5
1	A	H	A	H	A
2	H	A	H	A	A
3	A	H	A	H	H
4	H	A	A	H	A
5	A	H	H	A	H
6	H	A	H	A	H

Figure 13: Timetable and home-away pattern set.

HAP set can be represented in a more compact form using the “+” and “-” signs, instead of H’s and A’s, respectively, as shown in Figure 14.

	<i>round</i>				
	1	2	3	4	5
1	-6	+3	-5	+2	-4
2	+5	-6	+4	-1	-3
3	-4	+1	-6	+5	+2
4	+3	-5	-2	+6	-1
5	-2	+4	+1	-3	+6
6	+1	-2	+3	-4	+5

Figure 14: A compact representation of a tournament schedule.

Since in any schedule there can be at most two teams with perfect alternations of H’s and A’s, $n - 2$ teams will have sequences of games with at least one break each. Therefore, the total number of breaks in any schedule will be at least $n - 2$. There is a simple construction [5] which gives a schedule with $n - 2$ breaks for any league of n teams. In any schedule for n teams with the minimum number of breaks produced by the circle method and with the HAP set constructed by the algorithm in [5], the breaks occur at rounds $2 \cdot i - 1$, for $i = 2, \dots, n/2$. However, for general schedules with $n - 2$ breaks it is an open question to characterize the rounds at which the breaks may occur. In practice, we may face the following problem: for some reason, a few entries of a HAP set have been fixed beforehand, meaning that some teams must play at home (or away) at some fixed rounds. Can one find a timetable and an extension of the partial HAP set which will give a “feasible” schedule? To solve this problem, integer programming models have been proposed [2].

However, we may also express it in list edge coloring terms [7]. In such a formulation, every edge e of a complete graph K_n has a list $L(e)$ of possible (feasible) colors. It is required to find a proper edge coloring in which each edge e of K_n has a color $c(e) \in L(e)$.

In this problem, when some entries of a HAP set have been fixed with either H or A, we proceed as follows: initially, for each edge $e = (i, j)$ we set $L(e) = \{1, 2, \dots, n - 1\}$. Then, color s is removed from $L(e)$ if and only if either both teams i and j are at home or both are away in round s .

Another frequent problem consists in finding a HAP set with a minimum number of breaks for a given timetable: after all games have been assigned to some round, it remains to determine where each game between i and j is played, i.e., in the venue of either i or of j . This problem is NP-hard and it has been shown in [28] that, for a league of n teams, the minimum number of breaks is at most $\frac{1}{4}n(n - 2)$ if $n = 4k$ or $\frac{1}{4}(n - 2)^2$ if $n = 4k + 2$.

However, it is possible to find in polynomial time for a given timetable with n teams, whether there is a HAP set with $n - 2$ breaks or not. We refer the reader to [25] for several other formulations of this type of problem.

8. Concluding Remarks

There are many more real-world problems which may be modeled by edge colorings in graphs, as in scheduling of tasks requiring the cooperation of two processors [20], file transfer operations [27], and channel assignment in wireless networks [16]. In this work, we have presented an approach of sports scheduling problems based on the systematic use of graph theory and, in particular, edge colorings. We exploited the relations between single round robin scheduling and proper edge colorings of complete graphs. After discussing the classical circle method, we introduced an edge coloring procedure based on Vizing's theorem that allows the construction of schedules which are not restricted to any particular structure.

We have also presented an edge coloring interpretation of the neighborhoods commonly used by local search algorithms for sports scheduling problems. We have shown certain properties of the neighborhoods based on this representation and we have also discussed some of their variations and extensions.

Algorithms that construct initial solutions with the circle method and use only the four classical neighborhoods in the literature may be trapped in a very small portion of the solution space. The use of Vizing's algorithm

instead of the circle method may circumvent this issue. In order to increase the connectivity of the solution space, we have suggested some neighborhood extensions.

There are many open questions to be studied in the field of sports scheduling. We have mentioned a few in the previous sections. Since it is likely that local search procedures will remain an important tool for dealing with these problems, more efforts should be devoted to study the properties of the currently known move types and to generate new ones to explore systematically the solution space of schedules. Furthermore, theoretical works on one-factorizations of complete graphs are needed to better understand the structure of schedules, thus being able to take into account the many constraints and requirements appearing in real applications. We hope to motivate further works in these research avenues.

Acknowledgements

Sebastián Urrutia was partially supported by CNPq research grant 307839/2013-3. Part of this research was carried out while Sebastián Urrutia was visiting EPFL in May and June 2015. The support of this institution is gratefully acknowledged. The work of Celso C. Ribeiro was partially supported by CNPq research grants 308687/2010-8 and 459575/2014-7 and by FAPERJ research grant E-26/201.198/2014. Tiago Januario was partially supported by CNPq and CAPES research grants.

References

- [1] Anagnostopoulos, A., Michel, L., Hentenryck, P. V., and Vergados, Y. (2006). A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 9:177–193.
- [2] Briskorn, D. (2008). Feasibility of home-away-pattern sets for round robin tournaments. *Operations Research Letters*, 36:283–284.
- [3] 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*, 194:137–150.
- [4] de Werra, D. (1980). Geography, games and graphs. *Discrete Applied Mathematics*, 2:327–337.

- [5] de Werra, D. (1981). Scheduling in sports. In Hansen, P., editor, *Studies on Graphs and Discrete Programming*, volume 11 of *Annals of Discrete Mathematics*, pages 381–395. North-Holland.
- [6] Di Gaspero, L. and Schaerf, A. (2007). A composite-neighborhood tabu search approach to the traveling tournament problem. *Journal of Heuristics*, 13:189–207.
- [7] Diestel, R. (2006). *Graph Theory*. Graduate Texts in Mathematics. Springer.
- [8] Dijkstra, E. W. and Rao, J. (1990). Constructing the proof of Vizing’s Theorem. Technical Report EWD1075, University of Texas at Austin.
- [9] Dinitz, J. H., Garnick, D. K., and McKay, B.D. (1994). There are 526,915,620 nonisomorphic one-factorizations of K_{12} . *Journal of Combinatorial Designs*, 2:273–285.
- [10] 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*, volume 2239 of *Lecture Notes in Computer Science*, pages 580–584. Springer, Berlin.
- [11] Easton, K., Nemhauser, G., and Trick, M. (2003). Solving the travelling tournament problem: A combined integer programming and constraint programming approach. In *Practice and Theory of Automated Timetabling IV*, volume 2740 of *Lecture Notes in Computer Science*, pages 100–109. Springer, Berlin.
- [12] Ebert, J. (1987). A versatile data structure for edge-oriented graph algorithms. *Communications of the ACM*, 30:513–519.
- [13] Gabow, H. N., Nishizeki, T., Kariv, O. Leven, D., and Terada, O. (1985). Algorithms for edge-coloring graphs. Technical Report TRECIS-8501, Tohoku University.
- [14] Goossens, D. R. and Spieksma, F. C. R. (2012). Soccer schedules in Europe: An overview. *Journal of Scheduling*, 15:641–651.
- [15] Gould, R. (1988). *Graph Theory*. Benjamin-Cummings.

- [16] Hsu, C.-C., Liu, P., Wang, D.-w., and Wu, J.-J. (2006). Generalized edge coloring for channel assignment in wireless networks. In *Proceedings of the 2006 International Conference on Parallel Processing*, pages 82–92, Washington. IEEE Computer Society.
- [17] Januario, T. and Urrutia, S. (2016). A new neighborhood structure for round robin scheduling problems. *Computers & Operations Research*, 70:127–139.
- [18] Kaski, P. and Ostergard, P. R. J. (2009). There are 1,132,835,421,602,062,347 nonisomorphic one-factorizations of K_{14} . *Journal of Combinatorial Designs*, 17:147–159.
- [19] Kendall, G., Knust, S., Ribeiro, C. C., and Urrutia, S. (2010). Scheduling in sports: An annotated bibliography. *Computers & Operations Research*, 37:1–19.
- [20] Kosowski, A. (2009). Approximating the maximum 2- and 3-edge-colorable subgraph problems. *Discrete Applied Mathematics*, 157:3593–3600.
- [21] Kotzig, A. (1963). Hamilton graphs and hamilton circuits. In Fiedler, M., editor, *Theory of Graphs and its Applications: Proceedings of the Symposium held in Smolenice in June 1963*, pages 62–82.
- [22] Lewis, R. and Thompson, J. (2011). On the application of graph colouring techniques in round-robin sports scheduling. *Computers & Operations Research*, 38:190–204.
- [23] Melo, R. A., Urrutia, S., and Ribeiro, C. C. (2009). The traveling tournament problem with predefined venues. *Journal of Scheduling*, 12:607–622.
- [24] Misra, J. and Gries, D. (1992). A constructive proof of Vizing’s theorem. *Information Processing Letters*, 41:131–133.
- [25] Miyashiro, R., Iwasaki, H., and Matsui, T. (2003). Characterizing feasible pattern sets with a minimum number of breaks. In Burke, E. and De Causmaecker, P., editors, *Practice and Theory of Automated Timetabling IV*, volume 2740 of *Lecture Notes in Computer Science*, pages 78–99. Springer, Berlin.

- [26] Miyashiro, R. and Matsui, T. (2006). Minimizing the carry-over effects value in a round-robin tournament. In Burke, E. and Rudova, H., editors, *Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling*, pages 216–230. PATAT.
- [27] Nakano, S., Zhou, X., and Nishizeki, T. (1995). Edge-coloring algorithms. In *Computer Science Today*, volume 1000 of *Lecture Notes in Computer Science*, pages 172–183. Springer.
- [28] Post, G. and Woeginger, G. J. (2006). Sports tournaments, home-away assignments, and the break minimization problem. *Discrete Optimization*, 3:165–173.
- [29] Rasmussen, R. V. and Trick, M. A. (2008). Round robin scheduling - A survey. *European Journal of Operational Research*, 188:617–636.
- [30] Ribeiro, C. C. (2012). Sports scheduling: Problems and applications. *International Transactions in Operational Research*, 19:201–226.
- [31] Ribeiro, C. C. and Urrutia, S. (2007). Heuristics for the mirrored traveling tournament problem. *European Journal of Operational Research*, 179:775–787.
- [32] Thielen, C. and Westphal, S. (2011). Complexity of the traveling tournament problem. *Theoretical Computer Science*, 412:345–351.
- [33] Urrutia, S. and Ribeiro, C. C. (2006). Maximizing breaks and bounding solutions to the mirrored traveling tournament problem. *Discrete Applied Mathematics*, 154:1932–1938.
- [34] Vizing, V. G. (1964). On an estimated of the chromatic class of a p-graph. *Diskrete Analiz.*, 3:25–30. in Russian.
- [35] Wei, W., Fujimura, S., Wei, X., and Ding, C. (2010). A hybrid local search approach in solving the mirrored traveling tournament problem. In *IEEE 17th International Conference on Industrial Engineering and Engineering Management*, pages 620–624, Xiamen.
- [36] Wolfe, A. J. (2009). A perfect one-factorization of K_{52} . *Journal of Combinatorial Designs*, 17:190–196.