J Sched (2009) 12: 607–622
DOI 10.1007/s10951-008-0097-1

The traveling tournament problem with predefined venues

Rafael A. Melo · Sebastián Urrutia · Celso C. Ribeiro

Received: 21 November 2007 / Accepted: 3 November 2008 / Published online: 6 January 2009

Abstract Sports scheduling is a very attractive application area not only because of the interesting mathematical structures of the problems, but also due to their importance in practice and to the big business that sports have become. In this paper, we introduce the Traveling Tournament Problem with Predefined Venues, which consists in scheduling a compact single round robin tournament with a predefined venue assignment for each game (i.e., the venue where each game takes place is known beforehand) while the total distance traveled by the teams is minimized. Three integer programming formulations are proposed and compared. We also propose some simple enumeration strategies to generate feasible solutions to real-size instances in a reasonable amount of time. We show that two original enumeration strategies outperform an improvement heuristic embedded within a commercial solver. Comparative numerical results are presented.

Keywords Sports scheduling · Traveling tournament problem · Integer programming · Tournaments · Venues

Work of R.A. Melo was partially funded by a CAPES scholarship. Work of C.C. Ribeiro was supported by CNPq research grants 301.694/2007-9 and 485.328/2007-0, and by FAPERJ research grant E-152.522/2006.

R.A. Melo · C.C. Ribeiro
Department of Computer Science, Universidade Federal Fluminense, Rua Passo da Pátria 156, Niterói, RJ 24210-240, Brazil

R.A. Melo
e-mail: melo@ic.uff.br

C.C. Ribeiro
e-mail: celso@ic.uff.br

Present address:
R.A. Melo
Center for Operations Research and Econometrics, Université Catholique de Louvain, Rue Voie du Roman Pays, 34, 1348 Louvain-la-Neuve, Belgium
e-mail: rafael.demelo@uclouvain.be

S. Urrutia (✉)
Department of Computer Science, Universidade Federal de Minas Gerais, Av. Antônio Carlos 6627, Belo Horizonte, MG 31270-010, Brazil
e-mail: surrutia@dcc.ufmg.br

1 Introduction and problem statement

Sports have become a big business. Professional leagues involve millions of fans and significant investments in players, broadcast rights, merchandizing, and advertising. They also involve multiple other agents, such as organizers, media, players, and security.

Sports scheduling has been attracting the attention of an increasing number of researchers in multidisciplinary areas such as operations research, scheduling theory, constraint programming, graph theory, combinatorial optimization, and applied mathematics. Particular importance is given to round robin scheduling problems, in which each team is associated with a particular venue, due to their relevance in practice and to their interesting mathematical structure. The difficulty of the problems in the field leads to the use of a number of approaches, including integer programming (Nemhauser and Trick 1998; Ribeiro and Urrutia 2007b), constraint programming (Henz 1999), hybrid methods (Duarte et al. 2007; Easton et al. 2003), and heuristic techniques (Anagnostopoulos et al. 2006; Ribeiro and Urrutia 2007a). We refer to (Easton et al. 2004; Rasmussen and Trick 2008) for literature surveys.

In round robin tournaments, every team face each other a fixed number of times in a given number of rounds. Every team face each other exactly once in a single round robin (SRR) tournament and twice in a double round robin (DRR) tournament. In the case of an even number of teams, the tournament is said to be compact if the number of rounds is minimum and, consequently, every team plays exactly one game in every round. In an intermural tournament, each team has its own venue at its home city, and each game is played at the venue of one of the two teams in confrontation. The team that plays at its own venue is called the home team and is said to play a home game, while the other is called the away team and is said to play an away game. A schedule determines the round and the venue in which each game is played.

The problem of scheduling round robin tournaments is often divided into two subproblems. The construction of the timetable consists in determining the round in which each game will be played. The home–away pattern (HAP) set determines in which condition (home or away) each team plays in each round. The timetable and the home–away pattern set together determine the tournament schedule.

Some round robin scheduling problems involve the construction of both the timetable and the home–away pattern set. For example, the traveling tournament problem (TTP) (Easton et al. 2001) calls for a schedule minimizing the total distance traveled by the teams.

However, either the timetable or the HAP set may be predefined and known beforehand in some situations. In the first case, the timetable is given and the problem consists in finding a feasible HAP set optimizing a certain objective function. The break minimization problem (Régin 2001) deals with the minimization of the number of breaks (two consecutive home games or two consecutive away games of the same team) in the schedule, while the timetable constrained distance minimization problem (Rasmussen and Trick 2006) calls for the minimization of the total distance traveled.

In the second case, the HAP set is predetermined and a timetable is requested. There is a feasibility issue, since not every HAP set may be associated with a compatible timetable. A necessary condition for feasibility is given in (Miyashiro et al. 2003). The problem of constructing a timetable compatible with a given HAP set and optimizing a certain objective appears as a subproblem in several approaches to solve real-life scheduling problems, see, e.g., (Nemhauser and Trick 1998).

A Home–Away Assignment (HAA) (Knust and von Thaden 2006) associated with an SRR tournament is an assignment of a venue to each game. A HAA is balanced if the difference between the number of home games and the number of away games is at most one for every team. If the timetable is predefined, i.e., the round of every game is known beforehand, the HAP set and the HAA give the same information and determine the schedule.

We consider and formulate the problem of scheduling single round robin tournaments with predefined home–away assignments. The venue of each game is known beforehand and the problem consists in determining a timetable minimizing a certain objective function. Variants of this problem find interesting applications in real-life leagues whose DRR tournaments are divided into two SRR phases. Games in the second phase are exactly the same of the first phase, except for the inversion of their venues. Therefore, the venues of the games in the second phase are known beforehand and constrained by those of the games in the first phase. This is the case, e.g., of the Chilean soccer professional league (Durán et al. 2007) and of the German table tennis federation of Lower Saxony (Knust 2007).

We assume the tournament is played by n teams indexed by $1, \dots, n$, where n is even. Each team has its own venue at its home city. All teams are initially at their home cities, to where they return after their last away game. The distance $d_{ij} \geq 0$ from the home city of team i to that of team j is known beforehand, for $i, j = 1, \dots, n$. A road trip is a sequence of consecutive away games played by a team at the venues of its opponents. This team travels from the venue of one opponent to that of the next, without returning home.

Let G be a set of games, represented by ordered pairs of teams determined by the predefined HAA. The game between teams i and j is represented either by the ordered pair (i, j) or by the ordered pair (j, i). In the first case, the game between i and j takes place at the venue of team i; otherwise, at that of team j. For every two teams i and j, either $(i, j) \in G$ or $(j, i) \in G$.

The Traveling Tournament Problem with Predefined Venues (TTPPV) consists in finding a compact single round robin schedule compatible with G, such that the total distance traveled by the teams is minimized and no team plays more than three consecutive home games or three consecutive away games.

A necessary (but not sufficient) condition for an instance to be feasible is that

$$\min\big(h(t), a(t)\big) \geq \left\lfloor \frac{n-1}{4} \right\rfloor,$$

where $h(t)$ (resp., $a(t)$) denotes the number of home (resp., away) games of team t, for every team $t = 1, \dots, n$. This condition holds because each team must play at least one home game and one away game at every four consecutive rounds. If the number of consecutive home (and away) games is not bounded, then there is a feasible schedule for every HAA.

In Sect. 2, three integer programming formulations for this problem with, respectively, $O(n^3)$, $O(n^4)$ and $O(n^5)$ variables are described and the strength of their linear relaxations is compared. Computational results comparing the formulations are presented in Sect. 3. In Sect. 4, a simplified formulation to obtain feasible solutions for large size instances is presented. Two strategies to improve the solutions found by this simplified formulation are described and compared. The first strategy uses a heuristic embedded in the mixed integer programming solver, while the second uses the solver to enumerate solutions. Concluding remarks are drawn in the last section.

2 Integer programming formulations

In this section, three integer programming formulations for the Traveling Tournament Problem with Predefined Venues are proposed. These formulations are compared with respect to the bounds provided by their linear relaxations.

2.1 Formulation with $O(n^3)$ variables

We define the following decision variables:

$$z_{tjk} = \begin{cases} 1, & \text{if team } t \text{ plays at home against team } j \text{ in round } k, \\ 0, & \text{otherwise;} \end{cases}$$

$$y_{tij} = \begin{cases} 1, & \text{if team } t \text{ travels from the facility of team } i \text{ to the facility of team } j, \\ 0, & \text{otherwise.} \end{cases}$$

The y variables represent the journey performed by a team between two cities. Since each game occurs exactly once, a journey between the cities of two different teams is performed at most once by each team. Therefore, these variables are binary. Variables z and y are used in the formulation (1)–(13) of TTPPV with $O(n^3)$ variables:

$$\min F_1(z, y) = \sum_{t=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} \cdot y_{tij} \tag{1}$$

subject to:

$$\sum_{q=1}^{n-1} z_{tjq} = 1, \quad \forall (t, j) \in G, \tag{2}$$

$$\sum_{q=1}^{n-1} z_{tjq} = 0, \quad \forall (j, t) \in G, \tag{3}$$

$$\sum_{\substack{j=1 \\ j \neq t}}^{n} (z_{tjk} + z_{jtk}) = 1, \quad t = 1, \ldots, n, \; k = 1, \ldots, n-1, \tag{4}$$

$$y_{tij} \geq z_{it,k-1} + z_{jtk} - 1, \quad t, i, j = 1, \ldots, n, \quad \text{with } t \neq i \neq j, \; k = 2, \ldots, n-1, \tag{5}$$

$$y_{tit} \geq z_{it,k-1} + \sum_{\substack{j=1 \\ j \neq t}}^{n} z_{tjk} - 1, \quad t, i = 1, \ldots, n \quad \text{with } t \neq i, \; k = 2, \ldots, n-1, \tag{6}$$

$$y_{tti} \geq \sum_{\substack{j=1 \\ j \neq t}}^{n} z_{tj,k-1} + z_{itk} - 1, \quad t, i = 1, \ldots, n \quad \text{with } t \neq i, \; k = 2, \ldots, n-1, \tag{7}$$

$$y_{tti} \geq z_{it1}, \quad t, i = 1, \ldots, n \text{ with } t \neq i, \tag{8}$$

$$y_{tit} \geq z_{it,n-1}, \quad t, i = 1, \ldots, n \text{ with } t \neq i, \tag{9}$$

$$\sum_{q=k}^{k+3} \sum_{\substack{j=1 \\ j \neq t}}^{n} z_{jtq} \leq 3, \quad t = 1, \ldots, n, \; k = 1, \ldots, n-4, \tag{10}$$

$$\sum_{q=k}^{k+3} \sum_{\substack{j=1 \\ j \neq t}}^{n} z_{jtq} \geq 1, \quad t = 1, \ldots, n, \; k = 1, \ldots, n-4, \tag{11}$$

$$z_{tjk} \in \{0, 1\}, \quad t, j = 1, \ldots, n, \; k = 1, \ldots, n-1, \tag{12}$$

$$0 \leq y_{tij} \leq 1, \quad t, i, j = 1, \ldots, n. \tag{13}$$

The objective function (1) defines the minimization of the total distance traveled by the teams. Constraints (2) ensure that each game in G occurs exactly once, while constraints (3) set to zero the variables corresponding to games between teams t and j in the venue of team t if this game is predetermined to occur in the venue of team j. Constraints (4) guarantee that each team plays one game in each round. Constraints (5) enforce team t to perform a trip from the home city of team i to that of team j if it plays two consecutive away games against teams i and j, in this order. Constraints (6) enforce team t to perform a trip from the home city of team i to its own home city if it has an away game against the latter followed by a home game in the next round. Constraints (7) enforce team t to travel from its own home city to that of team i to play away against the later after a home game in the previous round. Constraints (8) enforce team t to travel to the home city of team i if it plays away against the latter in the first round. Constraints (9) enforce team t to return from the home city of team i if it plays away against the latter in the last round. Constraints (10) establish that team t cannot play more than three consecutive away games, while constraints (11) guarantee that team t cannot play more than three consecutive home games. Constraints (12) define the integrability requirements for the z variables. The y variables act as generalized upper bounds to constraints (5) to (9). Since their costs in the objective function to be minimized are non-negative, they will always assume the minimal possible value, which is necessarily either 0 or 1. Therefore, the integrability requirements on the y variables may be replaced by lower and upper bounds expressed as constraints (13).

This formulation has $O(n^4)$ constraints: $O(n^2)$ of types (2), (4), (8), (9), (10) and (11), $O(n^3)$ of types (6) and (7), and $O(n^4)$ of type (5).

2.2 Formulation with $O(n^4)$ variables

With this formulation, we reformulate the problem as a network flow model, in which each team may be viewed as a different commodity. We denote by $h(t) = |\{(t, j) \in G : j = 1, \ldots, n, \text{ with } j \neq t\}|$ the number of home games played by team $t = 1, \ldots, n$. We also introduce a dummy round n, to represent the fact that every team must return to its home city after its last away game, where it stays until the end of the tournament. The new decision variables are:

$$x_{tijk} = \begin{cases} 1, & \text{if team } t \text{ travels from the facility of team } i \\ & \text{to the facility of team } j \text{ in round } k, \\ 0, & \text{otherwise.} \end{cases}$$

According to the above definition, we notice that if a team t was already at its home city in round k and plays another home game in round $k+1$, then $x_{ttt,k+1} = 1$ as if there was a fictitious trip with cost zero leaving from and arriving to the home city of team t to play in round $k+1$. The new x variables are used to build the formulation (14)–(23) with $O(n^4)$ variables:

$$\min F_2(x) = \sum_{t=1}^{n}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} d_{ij} \cdot x_{tijk} \tag{14}$$

subject to:

$$\sum_{k=1}^{n-1}\sum_{i=1}^{n} x_{tijk} = 1, \quad \forall (j,t) \in G, \tag{15}$$

$$\sum_{j=1}^{n} x_{ttj1} = 1, \quad t = 1, \ldots, n, \tag{16}$$

$$\sum_{\substack{i=1\\ i\neq t}}^{n}\sum_{j=1}^{n} x_{tij1} = 0, \quad t = 1, \ldots, n, \tag{17}$$

$$\sum_{j=1}^{n} x_{tijk} = \sum_{j=1}^{n} x_{tji,k-1},$$
$$t, i = 1, \ldots, n, \; k = 2, \ldots, n, \tag{18}$$

$$\sum_{\substack{j=1\\ j\neq t}}^{n}\sum_{i=1}^{n} x_{jitk} = \sum_{i=1}^{n} x_{titk},$$
$$t = 1, \ldots, n, \; k = 1, \ldots, n-1, \tag{19}$$

$$\sum_{i=1}^{n}\sum_{k=1}^{n} x_{titk} = h(t) + 1, \quad t = 1, \ldots, n, \tag{20}$$

$$\sum_{q=k}^{k+3}\sum_{\substack{j=1\\ j\neq t}}^{n}\sum_{i=1}^{n} x_{tijq} \leq 3,$$
$$t = 1, \ldots, n, \; k = 1, \ldots, n-4, \tag{21}$$

$$\sum_{q=k}^{k+3}\sum_{\substack{j=1\\ j\neq t}}^{n}\sum_{i=1}^{n} x_{tijq} \geq 1,$$
$$t = 1, \ldots, n, \; k = 1, \ldots, n-4, \tag{22}$$

$$x_{tijk} \in \{0, 1\}, \quad t, i, j, k = 1, \ldots, n. \tag{23}$$

The objective function (14) defines the minimization of the total distance traveled by the teams. Constraints (15) ensure that each game in G is scheduled exactly once. Constraints (16) and (17) enforce every team t to perform a trip starting from its home city in the first round. If team t plays at home in the first round, then this will be a fictitious trip originating and ending at the home city of team t. Constraints (17) are redundant and not necessary in the integer programming formulation, but they contribute to strengthen the lower bounds provided by the linear relaxation. Constraints (18) require that if team t leaves facility i in round k, then it must have traveled to facility i in the previous round. Constraints (19) state that team t is at home in round k if and only if another team travels to play at its home city. Constraints (20) enforce team t to return to its home city in the end of the tournament, represented by the dummy round indexed by n. Since team t plays $h(t)$ home games during the tournament, these constraints enforce it to be at home at the end of the tournament by imposing that it has to perform $h(t)+1$ trips to its home city. Constraints (21) establish that team t cannot play more than three consecutive away games, while constraints (22) guarantee that team t cannot play more than three consecutive home games. Constraints (23) are the integrability requirements.

This formulation has $O(n^3)$ constraints: $O(n)$ of types (16), (17) and (20), $O(n^2)$ of types (15), (19), (21) and (22), and $O(n^3)$ of type (18).

2.3 Formulation with $O(n^5)$ variables

This formulation considers complete road trips. The variables represent road trips of different sizes, giving a more direct representation of the problem. Three new types of decision variables are defined and used in this third formulation:

$$w^1_{tik} = \begin{cases} 1, & \text{if team } t \text{ starts, in round } k, \text{ a road trip} \\ & \text{visiting team } i \text{ and returning home in} \\ & \text{round } k+1 \text{ (with } t \neq i), \\ 0, & \text{otherwise;} \end{cases}$$

$$w^2_{tijk} = \begin{cases} 1, & \text{if team } t \text{ starts, in round } k, \text{ a road trip} \\ & \text{visiting first team } i, \text{ then team } j, \text{ and} \\ & \text{returning home in round } k+2 \text{ (with} \\ & t \neq i \neq j), \\ 0, & \text{otherwise;} \end{cases}$$

$$w^3_{tijlk} = \begin{cases} 1, & \text{if team } t \text{ starts, in round } k\text{, a road trip visiting first team } i\text{, then team } j\text{, next team } l\text{, and returning home in round } k+3 \text{ (with } t \neq i \neq j \neq l),\\ 0, & \text{otherwise.}\end{cases}$$

Two dummy rounds (indexed by -1 and 0) are created to simplify the formulation. The variables corresponding to every road trip starting in any of these dummy rounds are set to 0. The auxiliary costs c_{ij}, c_{ijm}, and c_{ijml} represent the costs of road trips of length one, two, and three performed by team i, respectively:

$$c_{ij} = d_{ij} + d_{ji}, \tag{24}$$

$$c_{ijm} = d_{ij} + d_{jm} + d_{mi}, \tag{25}$$

$$c_{ijml} = d_{ij} + d_{jm} + d_{ml} + d_{li}. \tag{26}$$

The new variables are used to reformulate TTPPV as model (27)–(32) below, with $O(n^5)$ variables:

$$\min F_3(w^1, w^2, w^3) = \sum_{k=1}^{n-1}\sum_{i=1}^{n}\sum_{\substack{j=1\\(j,i)\in G}}^{n}\Bigg[c_{ij}\cdot w^1_{ijk} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg(c_{ijm}\cdot w^2_{ijmk} + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n} c_{ijml}\cdot w^3_{ijmlk}\Bigg)\Bigg] \tag{27}$$

subject to:

$$\sum_{i=1}^{n}\sum_{\substack{j=1\\(j,i)\in G}}^{n}\Bigg[\sum_{k\in\{-1,0\}} w^1_{ijk} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg(\sum_{k\in\{-1,0,n-1\}} w^2_{ijmk} + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n}\sum_{k\in\{-1,0,n-2,n-1\}} w^3_{ijmlk}\Bigg)\Bigg] = 0, \tag{28}$$

$$\sum_{k=1}^{n-1}\Bigg\{w^1_{ijk} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg[\big(w^2_{ijmk} + w^2_{imjk}\big) + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n}\big(w^3_{ijmlk} + w^3_{imjlk} + w^3_{imljk}\big)\Bigg]\Bigg\} = 1,$$
$$\forall (j,i)\in G, \tag{29}$$

$$\sum_{\substack{j=1\\(j,i)\in G}}^{n}\Bigg\{w^1_{ijk} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg[\big(w^2_{ijmk} + w^2_{ijm,k-1}\big) + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n}\big(w^3_{ijmlk} + w^3_{ijml,k-1} + w^3_{ijml,k-2}\big)\Bigg]\Bigg\}$$
$$+ \sum_{\substack{j=1\\(i,j)\in G}}^{n}\Bigg\{w^1_{jik} + \sum_{\substack{m=1\\(m,j)\in G\\m\neq i}}^{n}\Bigg[\big(w^2_{jimk} + w^2_{jmi,k-1}\big) + \sum_{\substack{l=1\\(l,j)\in G\\l\neq i\neq m}}^{n}\big(w^3_{jimlk} + w^3_{jmil,k-1} + w^3_{jmli,k-2}\big)\Bigg]\Bigg\} = 1,$$
$$i = 1,\dots,n,\ k = 1,\dots,n-1, \tag{30}$$

$$\sum_{\substack{j=1\\(j,i)\in G}}^{n}\Bigg\{w^1_{ijk} + w^1_{ij,k+1} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg[w^2_{ijm,k-1} + w^2_{ijmk} + w^2_{ijm,k+1} + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n}\big(w^3_{ijml,k-2} + w^3_{ijml,k-1} + w^3_{ijmlk} + w^3_{ijml,k+1}\big)\Bigg]\Bigg\} \leq 1,$$
$$i = 1,\dots,n,\ k = 1,\dots,n-2, \tag{31}$$

$$\sum_{q=k}^{k+3}\Bigg\{\sum_{\substack{j=1\\(j,i)\in G}}^{n}\Bigg[w^1_{ijq} + \sum_{\substack{m=1\\(m,i)\in G\\m\neq j}}^{n}\Bigg(\big(w^2_{ijmq} + w^2_{ijm,q-1}\big) + \sum_{\substack{l=1\\(l,i)\in G\\l\neq j\neq m}}^{n}\big(w^3_{ijmlq} + w^3_{ijml,q-1} + w^3_{ijml,q-2}\big)\Bigg)\Bigg]\Bigg\} \geq 1,$$
$$i = 1,\dots,n,\ k = 1,\dots,n-4, \tag{32}$$

$$w^1_{ijk}\in\{0,1\},\quad i,j = 1,\dots,n,\ \text{with } i\neq j,$$
$$k = -1,\dots,n-1, \tag{33}$$

$$w^2_{ijmk}\in\{0,1\},\quad i,j,m = 1,\dots,n,$$
$$\text{with } i\neq j\neq m,\ k = -1,\dots,n-1, \tag{34}$$

$$w^3_{ijmlk}\in\{0,1\},\quad i,j,m,l = 1,\dots,n,$$
$$\text{with } i\neq j\neq m\neq l,\ k = -1,\dots,n-1. \tag{35}$$

The objective function (27) minimizes the total traveled distance. Constraints (28) set to zero the variables associated to road trips starting at the dummy rounds −1 and 0, to road trips of size two and three starting at the last round and those of size three starting at round $n-2$. Constraints (29) ensure that each game occurs exactly once. They represent the fact that each game $(j,i) \in G$ should be played in a road trip of team i formed by one, two, or three away games. Constraints (30) enforce that team i is either playing an away game or another team is visiting it in each round. This is achieved by setting to one the sum of all variables associated with road trips of team i which include round k and road trips of other teams which visit i in round k. Constraints (31) forbid team i to be engaged in simultaneous or consecutive (i.e., without returning to its home city) road trips in round k. Constraints (32) state that team i must be outside its home city to play away at least once along every four consecutive rounds. Constraints (33)–(35) guarantee the integrability requirements.

Although the number of variables increases with respect to the previous formulation, we notice that the number of constraints is quite smaller: this formulation has $O(n^2)$ constraints of types (29) to (32).

2.4 Linear relaxation bounds

We refer to the formulations with $O(n^3)$, $O(n^4)$, and $O(n^5)$ variables, respectively, as models N3, N4, and N5. We denote by $\overline{\text{N3}}$, $\overline{\text{N4}}$, and $\overline{\text{N5}}$, respectively, their linear relaxations. Let $LB3$, $LB4$, and $LB5$ be the linear relaxation bounds provided by the respective formulations N3, N4, and N5.

We state two theorems which afford comparisons between the formulations with respect to their linear relaxation bounds $LB3$, $LB4$, and $LB5$. The first theorem compares formulations N3 and N4, while the second compares formulations N4 and N5.

Theorem 1 *$LB4 \geq LB3$ holds for every instance of the TTPPV.*

Theorem 2 *$LB5 \geq LB4$ holds for every instance of the TTPPV.*

The proofs of Theorems 1 and 2 are sketched in the Appendix.

3 Computational results

To compare the proposed integer programming formulations, we created 20 home–away assignments for each national league (NL) and circular (CIRC) instance of the traveling tournament problem in (Trick 2007). Ten instances in each group have their home–away assignments randomly determined. The other ten instances correspond to balanced home–away assignments: the venues of all games were randomly selected and the iterative method in Knust and von Thaden (2006) was used to balance the home-away assignments.

The models have been solved by CPLEX 10.0, using the API Concert Technology compiled with gcc 4.0.2. The computational experiments were performed on a Dell Optiplex machine, with a Pentium D 3.0 GHz processor and 2 Gbytes of RAM memory, running under version 2.6.12 of Linux.

The linear relaxations of the three formulations described in Sect. 2 were solved for all instances. The results are summarized in Tables 1 and 2 for the NL instances and in Tables 3 and 4 for the CIRC instances. For each value of n, the second column shows the number of feasible instances (since not every HAA corresponds to a feasible instance). The third, fourth, and fifth columns of these tables give, respectively, the average lower bounds provided by models N3, N4, and N5 for the feasible instances.

The four last columns give the average and maximum gaps between the bounds $LB3$ and $LB5$ and between the bounds $LB4$ and $LB5$ (the relative gaps are given by $100 \cdot (LB5 - LB3)/LB3$ and $100 \cdot (LB4 - LB3)/LB3$, respectively), i.e., by how much $LB5$ improves upon $LB3$ and upon $LB4$.

The average bounds $LB5$ provided by formulation N5 are far better than those provided by N3. Considering, e.g., the CIRC instances with $n = 20$, the average bound $LB5$ is approximately forty one times greater than the average

Table 1 Linear relaxation lower bounds for the random NL instances

n	Feasible instances	$LB3$	$LB4$	$LB5$	avg(5,3) (%)	max(5,3) (%)	avg(5,4) (%)	max(5,4) (%)
4	10	2064.92	2090.67	5227.70	153.16	191.59	150.05	188.57
6	7	3716.91	5503.67	13756.31	273.03	312.10	154.32	210.15
8	10	4558.85	7600.34	21675.06	401.57	550.88	214.08	350.24
10	6	3858.40	7175.11	32390.32	740.40	788.92	357.24	430.72
12	9	5989.87	14520.32	57930.83	877.94	984.17	333.60	514.68
14	7	6658.78	15968.46	103499.29	1455.36	1533.61	549.76	596.60
16	9	7280.00	24599.24	135829.89	1805.62	1996.64	484.61	607.84

Table 2 Linear relaxation lower bounds for the balanced NL instances

n	Feasible instances	$LB3$	$LB4$	$LB5$	avg(5,3) (%)	max(5,3) (%)	avg(5,4) (%)	max(5,4) (%)
4	10	2063.90	2090.67	5097.50	147.00	191.24	143.82	186.80
6	10	2992.80	3896.80	14134.90	372.42	396.69	262.73	274.92
8	10	3456.92	4991.71	21278.91	515.59	536.58	326.28	340.37
10	10	3704.52	6218.44	32573.49	779.29	809.42	423.82	440.13
12	10	5377.26	9482.18	58432.26	986.73	1017.92	516.23	533.14
14	10	6533.32	15325.50	103722.30	1487.59	1529.30	576.80	593.96
16	10	6440.18	19114.40	135563.90	2004.97	2040.95	609.22	621.46

Table 3 Linear relaxation lower bounds for the random CIRC instances

n	Feasible instances	$LB3$	$LB4$	$LB5$	avg(5,3) (%)	max(5,3) (%)	avg(5,4) (%)	max(5,4) (%)
4	10	4.94	5.33	12.60	154.50	180.00	136.25	162.50
6	7	8.50	13.6	35.90	329.52	410.81	167.09	224.07
8	8	14.53	29.92	73.78	431.45	702.81	156.29	282.81
10	7	12.81	33.43	132.76	971.62	1197.72	303.46	370.40
12	10	15.03	49.54	214.23	1402.89	1667.04	358.40	450.00
14	8	16.35	60.65	326.62	1907.82	2179.34	439.15	455.73
16	10	18.90	83.77	471.70	2488.63	2860.97	502.15	596.29
18	8	19.24	91.67	649.38	3294.49	3528.10	609.96	655.17
20	8	20.83	113.73	869.29	4102.11	4252.44	684.96	729.35

Table 4 Linear relaxation lower bounds for the balanced CIRC instances

n	Feasible instances	$LB3$	$LB4$	$LB5$	avg(5,3) (%)	max(5,3) (%)	avg(5,4) (%)	max(5,4) (%)
4	10	4.94	5.33	13.40	171.00	180.00	151.25	162.50
6	10	6.92	10.80	34.70	402.08	444.19	221.30	233.33
8	10	8.74	18.29	71.70	720.81	747.78	292.11	297.40
10	10	10.49	27.78	131.07	1150.98	1194.31	371.84	382.40
12	10	12.13	39.27	213.57	1659.98	1684.34	443.80	451.70
14	10	14.15	52.77	326.30	2206.79	2250.43	518.35	527.89
16	10	16.07	68.27	466.00	2800.43	2850.29	582.62	595.80
18	10	18.02	85.76	647.30	3490.46	3548.08	654.74	666.44
20	10	20.02	105.26	867.05	4230.67	4292.39	723.70	735.05

bound $LB3$ for the random instances. The average bound $LB5$ also improve upon those given by model N4. Considering the same random CIRC instances with $n = 20$, the average bound $LB5$ is almost seven times greater than the average bound $LB4$.

We also compared the formulations in terms of the solver capability to find optimal integral solutions. We run CPLEX for at most two hours to solve each instance using each formulation. The results are shown in Tables 5 to 8. For each number of teams, we give the number of feasible instances, the number of instances solved to optimality using models N3, N4, and N5, the average ratio $LB5/F^*$ (where F^* denotes the objective value of the optimal solution), and the average and maximum times in seconds to find an optimal solution using models N4 and N5.

The three formulations were able to find the optimal solutions for all small instances with $n \leq 6$. For the random NL instances with $n = 8$, CPLEX was able to solve only five instances with formulation N3 and all instances with mod-

Table 5 Integer solutions for the random NL instances

n	Feasible instances	optima N3	optima N4	optima N5	avg($\frac{LB5}{F_3^*}$) (%)	avg. time N4 (s)	max. time N4 (s)	avg. time N5 (s)	max. time N5 (s)
4	10	10	10	10	100.00	<0.1	<0.1	<0.1	<0.1
6	7	7	7	7	97.19	1.6	3.3	0.3	0.7
8	10	5	10	10	92.28	646.6	2505.4	16.8	39.2

Table 6 Integer solutions for the balanced NL instances

n	Feasible instances	optima N3	optima N4	optima N5	avg($\frac{LB5}{F_3^*}$) (%)	avg. time N4 (s)	max. time N4 (s)	avg. time N5 (s)	max. time N5 (s)
4	10	10	10	10	100.00	<0.1	<0.1	<0.1	<0.1
6	10	10	10	10	97.81	2.0	2.4	0.2	0.5
8	10	0	8	10	92.34	1685.4	3810.0	34.4	61.8

Table 7 Integer solutions for the random CIRC instances

n	Feasible instances	optima N3	optima N4	optima N5	avg($\frac{LB5}{F_3^*}$) (%)	avg. time N4 (s)	max. time N4 (s)	avg. time N5 (s)	max. time N5 (s)
4	10	10	10	10	100.00	<0.1	<0.1	<0.1	<0.1
6	7	7	7	7	94.49	1.6	2.5	0.2	0.2
8	8	4	8	8	89.16	1060.3	3874.6	26.1	122.6

Table 8 Integer solutions for the balanced CIRC instances

n	Feasible instances	optima N3	optima N4	optima N5	avg($\frac{LB5}{F_3^*}$) (%)	avg. time N4 (s)	max. time N4 (s)	avg. time N5 (s)	max. time N5 (s)
4	10	10	10	10	100.00	<0.1	<0.1	<0.1	<0.1
6	10	10	10	10	92.32	2.2	3.7	0.3	0.5
8	10	0	10	10	90.31	2399.0	4172.8	24.6	57.9

els N4 and N5. Considering the balanced NL instances with eight teams, not a single one could be solved using formulation N3, but eight were solved using formulation N4. The solver was able to find the optimal solution for all instances with formulation N5. Furthermore, we also notice that the computation times to find the optimal solutions using formulation N5 are significantly smaller than those observed with formulation N4.

For the random CIRC instances with $n = 8$, CPLEX was able to solve only four instances using formulation N3 and all eight feasible instances using models N4 and N5. Considering the balanced CIRC instances with the same size, CPLEX was not able to solve any instance using formulation N3, but solved all of them using models N4 and N5. CPLEX was not able to solve any instance with $n > 8$.

4 Approximate solutions for real-size instances

Real-life tournaments such as the Brazilian soccer championship involve a great number of teams, with $n \geq 20$. In practice, sometimes only a feasible tournament schedule is required.

We notice that the solver had a great difficulty even to find feasible solutions for real-size instances within the time limit of two hours. Considering the NL and CIRC instances with more than 16 teams, the solver found a feasible solution for only one of them using formulation N3 within this limit. No solution was found using formulations N4 and N5.

Scheduling problems in sports are usually computationally easier when the traveled distance is not part of the objective function. Therefore, we used a simplified model S3 with $O(n^3)$ variables to obtain feasible solutions to TTPPV. This model is derived from N3 by discarding (a) the y variables representing the performed trips and (b) the objective function (1). Consequently, formulation S3 consists in solving the resulting feasibility problem defined by the constraints (2)–(4), (10)–(12).

4.1 Improvements strategies

Feasible solutions could be obtained by the solver using formulation S3 for all but two instances: one with 18 teams and the other with 20 teams. However, model S3 is a pure feasibility problem that does not consider any cost measure. Two strategies to find better solutions than those obtained by simply solving model S3 are initially proposed in this section. Finally, original approaches based on extensions of model S3 are proposed in Sect. 4.2.

4.1.1 Polishing strategy

Solution polishing (Rothberg 2007) is an evolutionary heuristic to improve feasible solutions to MIP models. It keeps a pool of feasible solutions and uses a variant of RINS (Relaxation Induced Neighborhood Search) (Danna et al. 2005) for mutations and combinations. Variables with the same value in two solutions are fixed by the combination operator. A number of randomly selected variables are fixed by the mutation operator. In both cases, a new solution is determined by solving the MIP defined by the remaining variables. The polishing heuristic can be invoked even if only one single solution is available, since the mutation operator is applied first.

The polishing strategy implemented in this paper makes use of the solution polishing heuristic embedded within the CPLEX solver. It starts by finding a feasible solution $\check{z}$ to S3. The integer variables z are fixed in the complete model N3 according to the values found for the corresponding variables in solution $\check{z}$ to S3. Such fixations lead to a partial solution $\hat{z}$ to N3. This partial solution is used to determine feasible values $\hat{y}$ for the y variables, therefore defining a feasible solution $\hat{z}, \hat{y}$ to N3. Finally, and until the time limit is reached, the solution polishing heuristic embedded within CPLEX is applied to solution $\hat{z}, \hat{y}$ to improve the distance-based objective function.

4.1.2 Basic enumerative strategy

This strategy is based on the utilization of the solver to enumerate different solutions to the model, due to its ability of quickly finding a feasible solution to S3 for most instances. Whenever the solver founds a new feasible solution, the traveled distance associated with this new solution is computed. If it is smaller than the distance associated with the best known solution so far, then the latter is updated. In any case, after the possible update of the best known solution, the solver is set to reject this solution (and, consequently, the primal bound it provides) as the incumbent in the branch-and-bound. Since the primal bound is never updated, the branch-and-bound continues the enumeration even after finding an optimal solution for the simplified model. Solution enumeration is performed until the time limit is reached.

4.2 Enhanced enumerative strategies

Two improved enumerative strategies are proposed here to speedup the search for feasible solutions to N3.

4.2.1 Relaxed consecutive games enumerative strategy

We define a *violation* as any sequence of four consecutive away games or four consecutive home games performed by the same team. We consider the previously defined 0–1 variables

$$z_{tjk} = \begin{cases} 1, & \text{if team } t \text{ plays at home against team } j \text{ in round } k, \\ 0, & \text{otherwise;} \end{cases}$$

and new decision variables:

$$v_{tk} = \begin{cases} 1, & \text{if team } t \text{ completes a violated sequence of four games in round } k, \\ 0, & \text{otherwise.} \end{cases}$$

Solutions violating the constraints imposed to the maximum number of consecutive home games or consecutive away games are allowed in the relaxed consecutive games formulation SC3 below, defined by (36)–(42):

$$\min F_C(z, v) = \sum_{t=1}^{n} \sum_{k=4}^{n-1} v_{tk} \tag{36}$$

subject to:

$$\sum_{q=1}^{n-1} z_{tjq} = 1, \qquad \forall (t, j) \in G, \tag{37}$$

$$\sum_{j=1} (t, j) \in G^n z_{tjk} + \sum_{\substack{j=1 \\ (j,t)\in G}}^{n} z_{jtk} = 1,$$
$$t = 1, \ldots, n, \; k = 1, \ldots, n-1, \tag{38}$$

$$v_{tk} \geq \sum_{q=k-3}^{k} \sum_{\substack{j=1 \\ (j,t)\in G}}^{n} z_{jtq} - 3,$$
$$t = 1, \ldots, n, \; k = 4, \ldots, n-1, \tag{39}$$

$$v_{tk} \geq \sum_{q=k-3}^{k} \sum_{\substack{j=1 \\ (t,j)\in G}}^{n} z_{tjq} - 3,$$

$$t = 1, \ldots, n, \ k = 4, \ldots, n-1, \tag{40}$$

$$z_{tjk} \in \{0, 1\}, \quad t, j = 1, \ldots, n, \ k = 1, \ldots, n-1, \tag{41}$$

$$0 \leq v_{tk} \leq 1, \quad t = 1, \ldots, n, \ k = 4, \ldots, n-1. \tag{42}$$

The objective function (36) minimizes the number of violations. Constraints (37) and (38) have the same purpose as in formulation N3. Each constraint (39) sets the v_{tk} variable to one if team t performs in round k a fourth consecutive away game. Similarly, each constraint (40) sets the v_{tk} variable to one if team t performs in round k a fourth consecutive home game. Constraints (41) and (42) ensure the integrability requirements.

Only solutions to SC3 whose cost is equal to zero are feasible to TTPPV. Solutions of this formulation are enumerated in a similar way to what was done in Sect. 4.1.2, i.e., all optimal solutions are discarded after their traveled distances are computed and the best known solution is updated.

4.2.2 *Relaxed HAA enumerative strategy*

Variables z are defined and used again as in model N3. Solutions of formulation RH3 below, defined by (43)–(48), may violate the HAA constraints, i.e., games may be played in a venue different than that which was predefined. The objective function maximizes the number of games played in the appropriate venue according to the predefined HAA:

$$\max F_D(z) = \sum_{t=1}^{n} \sum_{\substack{j=1 \\ (t,j)\in G}}^{n} \sum_{q=1}^{n-1} z_{tjq} \tag{43}$$

subject to:

$$\sum_{q=1}^{n-1} (z_{tjq} + z_{jtq}) = 1,$$

$$\forall\, t = 1, \ldots, n, \ j = 1, \ldots, n, \ \text{with } t \neq j, \tag{44}$$

$$\sum_{\substack{j=1 \\ j\neq t}}^{n} (z_{tjk} + z_{jtk}) = 1,$$

$$t = 1, \ldots, n, \ k = 1, \ldots, n-1, \tag{45}$$

$$\sum_{q=k}^{k+3} \sum_{\substack{j=1 \\ j\neq t}}^{n} z_{jtq} \leq 3,$$

$$t = 1, \ldots, n, \ k = 1, \ldots, n-4, \tag{46}$$

$$\sum_{q=k}^{k+3} \sum_{\substack{j=1 \\ j\neq t}}^{n} z_{jtq} \geq 1, \quad t = 1, \ldots, n, \ k = 1, \ldots, n-4, \tag{47}$$

$$z_{tjk} \in \{0, 1\}, \quad t, j = 1, \ldots, n, \ k = 1, \ldots, n-1. \tag{48}$$

The objective function maximizes the number of games played in the predefined venues. Constraints (44) guarantee that each team plays every other exactly once. Constraints (45) ensure each team plays exactly one game in each round. Constraints (46) and (47) play the same role as in formulation N3. Constraints (48) enforce the integrability requirements.

A solution to RH3 is a feasible solution to the TTPPV if and only if its objective value is equal to the number of games $|G| = n \cdot (n-1)/2$. Therefore, solutions are enumerated in a similar way as done in the previous strategies. However, in this case the objective function is maximized and the stored solutions are those whose cost is equal to the number of games in G.

4.3 Computational results for the enumerative strategies

All the four enumeration strategies were coded in C++ using CPLEX 10.0 and the API Concert Technology. The code was compiled with gcc 4.0.2. We applied each strategy to each instance with $n = 18$ and $n = 20$ created from the CIRC instances. We also created a new set (BR) of instances based on the TTP instance derived from the Brazilian soccer tournament (Trick 2007) with 24 teams. Ten of the instances in this set have a random HAA and the remaining ten instances have a balanced HAA. The same limit of two hours was imposed to the computation times.

The results obtained with the four strategies for the different instances are summarized in Tables 9 to 14. The first column in these tables identify each instance. The objective value of the solution obtained with formulation S3 and the processing time in seconds to find this solution are reported next. The objective value of the solution obtained by the polishing heuristic running until the time limit is reached is also reported. Finally, for each enumerative strategy (basic, relaxed consecutive games, and relaxed HAA), we display the number of solutions enumerated and the best solution found within the two-hour time limit. In the case of cells marked with a "–", the two-hour time limit was reached before either a feasible solution was found or infeasibility was proved. Underlined values are those for which the polishing strategy improved upon the solution obtained by S3. Values in bold are the best found for each instance. Table 9 displays the results for the random CIRC instances with 18 teams. A feasible solution was found with the simplified model S3 for five out of the eight feasible instances in less than one minute and in less than two minutes for the three others. The polishing heuristic improved the best

Table 9 Results of the strategies for the random CIRC instances with $n = 18$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	1170	30.1	<u>1162</u>	91	1138	5947	1136	3458	**1124**
B	*infeasible*								
C	*infeasible*								
D	1208	21.5	<u>1204</u>	14	1160	3227	1092	2761	**1060**
E	1160	90.7	1160	25	1138	2579	1096	3355	**1092**
F	1226	111.3	1226	22	1140	3322	**1098**	2620	1136
G	1152	22.5	1152	144	1130	3001	1136	2695	**1098**
H	1216	27.4	<u>1198</u>	26	1132	2285	**1110**	3373	1122
I	1188	109.3	1188	59	1132	3726	1118	2731	**1104**
J	1198	29.3	1198	63	1116	4166	1106	3576	**1102**

Table 10 Results of the strategies for the balanced CIRC instances with $n = 18$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	1172	25.7	1172	34	1142	4206	1140	4342	**1106**
B	1194	22.6	1194	77	1168	5356	**1100**	3956	1106
C	1180	23.1	<u>1162</u>	17	1170	2825	**1038**	3536	1094
D	1172	31.1	<u>1162</u>	145	1120	5192	1100	4327	**1096**
E	1188	31.8	<u>1174</u>	46	1154	4473	**1074**	4217	1094
F	1174	25.0	1174	61	1104	4711	1082	3936	**1060**
G	1162	25.9	<u>1140</u>	50	1148	3148	1102	4278	**1100**
H	1198	26.2	<u>1188</u>	120	1142	3187	1124	5182	**1094**
I	1246	24.0	1246	78	1172	6141	**1102**	4252	1122
J	1182	27.2	<u>1162</u>	96	1122	4282	1110	3655	**1078**

Table 11 Results of the strategies for the random CIRC instances with $n = 20$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	1552	129.4	1552	12	1552	1590	**1502**	1012	1538
B	1634	50.7	<u>1616</u>	2	1634	1372	**1522**	1144	1546
C	–	–	–	–	–	1435	**1488**	1397	1590
D	1562	50.2	1562	9	1562	1974	**1510**	1116	1572
E	1580	303.6	**<u>1574</u>**	17	1580	1793	1584	1360	1582
F	*infeasible*								
G	1614	57.3	1614	3	1560	1172	**1540**	1673	1564
H	*infeasible*								
I	1590	372.4	<u>1574</u>	8	1580	1337	1564	852	**1516**
J	1624	51.3	<u>1588</u>	6	**1516**	1203	1538	1218	1520

result for three instances. The basic enumerative strategy obtained better solutions than the polishing strategy for all feasible instances. However, it was overcome in all cases by the enhanced enumerative strategies. The relaxed HAA enumerative strategy achieved the best values for six out of the eight feasible instances, while the relaxed consecutive games enumerative strategy found the best solutions for the other two.

Table 10 shows the results for the balanced CIRC instances with 18 teams. A feasible solution was found by formulation S3 in less than 32 seconds for all test instances. The polishing strategy improved the solution obtained with S3 for six out of the ten instances. The basic enumerative

Table 12 Results of the strategies for the balanced CIRC instances with $n = 20$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	1622	47.6	1622	10	1614	2136	**1520**	1942	1570
B	1626	53.2	1610	50	1610	1017	**1530**	1264	1552
C	1580	53.0	1580	17	1542	2057	1572	1150	**1470**
D	1544	62.9	1544	5	1518	1922	**1464**	1300	1512
E	1658	60.3	1658	17	1598	2124	1558	1574	**1526**
F	1650	54.2	1650	4	1568	1799	1566	1428	**1546**
G	1582	52.2	1582	4	1582	1782	1550	1354	**1536**
H	1654	56.9	1614	18	1590	1806	1560	1487	**1516**
I	1618	68.0	1616	18	1560	874	1584	1529	**1544**
J	1590	56.6	1582	3	1590	2233	1588	1187	**1484**

Table 13 Results of the strategies for the random BR instances with $n = 24$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	–	–	–	–	–	473	441401	241	**433896**
B	455240	3142.0	455240	3	445080	103	**429414**	210	460500
C	–	–	–	–	–	321	433269	256	**428237**
D	–	–	–	–	–	172	**428574**	152	428701
E	–	–	–	–	–	270	**423670**	153	426615
F	–	–	–	–	–	457	**437383**	168	438961
G	–	–	–	–	–	352	425108	113	**413653**
H	–	–	–	–	–	279	**411380**	146	417969
I	–	–	–	–	–	279	**403318**	226	434724
J	427740	2207.7	427740	1	427740	206	**416248**	254	421167

Table 14 Results of the strategies for the balanced BR instances with $n = 24$

Instance	S3	time (s)	Polishing	# sol.	Enumerated	# sol.	Rel. CG	# sol.	Rel. HAA
A	456879	203.9	456879	2	456879	288	435869	220	**433630**
B	454840	228.2	454840	2	442810	425	438033	207	**433916**
C	446725	176.5	446725	3	446725	215	**433615**	281	442174
D	448931	210.5	448922	2	444703	377	**443077**	228	445335
E	458283	215.1	458283	2	450679	358	**444249**	121	444375
F	–	–	–	–	–	400	443759	259	**438710**
G	451544	185.6	451544	2	451544	259	434790	170	**421479**
H	**438713**	211.2	438713	1	438713	608	441080	204	445332
I	461105	178.8	461105	2	456797	349	437572	109	**430386**
J	–	–	–	–	–	580	**444962**	224	447746

strategy found better solutions than the polishing strategy for eight out of the ten instances. As before, the enhanced enumerative strategies found the best solutions for all test instances.

Table 11 gives the results for the random CIRC instances with 20 teams. For one instance, neither a feasible solution was found nor infeasibility was proved using formulation S3, polishing and the basic enumerative strategy. However, the enhanced enumerative strategies make it possible to find feasible solutions for this instance. A solution was obtained in less than one minute of processing time using model S3 for four out of the eight feasible instances. The polishing strategy improved upon the best solution values for four out of the seven instances for which formulation S3 provided a feasible solution. The relaxed HAA enumerative strategy found the best solution for one out of the eight feasible instances, while the relaxed consecutive games enumerative strategy found the best solutions for five out of them.

Table 12 depicts the results for the balanced CIRC instances with 20 teams. Formulation S3 led to feasible solutions in less than one minute of processing time for seven instances. The polishing strategy improved upon the solution values found with S3 for four out of the ten instances. The basic enumerative strategy found better results than the polishing strategy for seven instances in this group. The relaxed HAA enumerative strategy found the best solutions for seven out of the ten instances, while the relaxed consecutive games enumerative strategy found them for the remaining three instances.

Table 13 exhibits the results for the random BR instances with 24 teams. A feasible solution was found in only two cases using formulation S3. The polishing strategy did not improve the solution for any instance. The basic enumerative strategy found more than one feasible solution for only one instance. The relaxed consecutive games enumerative strategy found the best solutions for seven out of the ten instances, while the relaxed HAA enumerative strategy found them for the remaining three instances.

Table 14 presents the results for the balanced BR instances with 24 teams. The solver found a feasible solution for eight out of the ten instances using model S3 and for one instance this solution was the best found. The polishing strategy improved the solution in only one case. The basic enumerative strategy did not find more than three feasible solutions for any of the instances. The relaxed HAA enumerative strategy found the best solutions for five out of the ten instances, while the relaxed consecutive games enumerative strategy found them for four instances.

We notice from the above results that the processing times needed to find feasible initial solutions using S3 do not vary too much from one balanced CIRC instance to another, although they are very elastic for the random instances. We also observe that the number of solutions investigated by the enhanced enumerative strategies is far greater than the number of those examined by the basic enumerative strategy, for all sets of instances. This shows that the enhanced strategies were very effective to improve the pruning strategy and, consequently, to speedup the enumeration.

Finally, we remark that each among the polishing and the basic enumerative strategies found the best solution for only one out of the 56 feasible instances in the preceding tables within the two-hour time limit, while the relaxed consecutive games enumerative strategy succeeded for 25 instances and the relaxed HAA enumerative for 28 of them.

5 Concluding remarks

In this paper, we introduced the Traveling Tournament Problem with Predefined Venues (TTPPV). Three original integer programming formulations for this problem were proposed and compared. The linear relaxation of the formulation with $O(n^5)$ variables provides the strongest lower bounds and improves the ability of integer programming solvers to find optimal solutions.

The quality of the lower bound $LB5$ provided by the formulation with $O(n^5)$ variables helped the solver to find optimal solutions in less than one minute of computation time for the great majority of the instances with $n = 8$. All instances with eight teams were solved in small computation times, but none with 10 teams could be solved in two hours of processing time. This result is similar to what is observed for the TTP, which can be easily solved for $n = 6$, but is very hard to solve for $n \geq 8$.

Since the mixed integer programming solver was not able to find feasible solutions for large instances using any of the models N3, N4, and N5, we used a simplified integer programming model conjugated with a heuristic embedded in the solver to find good feasible solutions for instances with 18, 20, and 24 teams. In some cases, the results obtained with formulation S3 have been improved by the polishing heuristic embedded in the solver. We also used CPLEX together with simplified or relaxed formulations to perform enumerative strategies. A very basic enumeration strategy based on the simplified model S3 already provided better results than those obtained by the polishing, which does not seem to perform well when applied to the TTPPV.

Finally, we proposed two enhanced enumeration strategies. They are based on two different relaxations of the TTPPV formulation with $O(n^3)$ variables, optimizing the number of constraint violations. The results produced by the enhanced enumeration strategies clearly outperform those obtained with the polishing or the basic enumeration strategies. Each among the polishing and the basic enumeration strategies found the best solution for only one out of the 56 feasible instances considered in the computational experiments, while the relaxed consecutive games enumeration strategy found the best solutions for 25 instances and the relaxed HAA enumerative for 28 instances. These strategies are easy to implement, since it is only necessary to interact with the solver by giving directives on how to solve the simplified formulations. This simple mechanism allowed us to find various feasible solutions that could not be found by the solver alone using formulations N3, N4, or N5.

Appendix

Sketch of the proof of Theorem 1 Let x^* be an optimal solution to $\overline{\text{N4}}$, satisfying constraints (15)–(22). We build a solution $\hat{z}$, $\hat{y}$ to $\overline{\text{N3}}$ by defining

$$\hat{z}_{jtk} = \sum_{i=1}^{n} x^*_{tijk},$$

$$j, t = 1, \dots, n \text{ with } j \neq t,\ k = 1, \dots, n-1,$$

$$\hat{z}_{ttk} = 0, \quad t = 1, \dots, n,\ k = 1, \dots, n-1,$$

$$\hat{y}_{tij} = \sum_{k=1}^{n} x^*_{tijk}, \quad t, i, j = 1, \dots, n \text{ with } t \neq i \neq j,$$

$$\hat{y}_{tit} = \sum_{k=1}^{n} x^*_{titk}, \quad t, i = 1, \dots, n \text{ with } t \neq i,$$

$$\hat{y}_{tti} = \sum_{k=1}^{n} x^*_{ttik}, \quad t, i = 1, \dots, n \text{ with } t \neq i.$$

Since $d_{ii} = 0$ for every $i = 1, \dots, n$, we set

$$\hat{y}_{ttt} = 0, \quad t = 1, \dots, n.$$

It can be proved by variable substitution that $\hat{z}$, $\hat{y}$ satisfies constraints (2)–(11), i.e., it is feasible to $\overline{\text{N3}}$. We now show that $F_2(x^*) = F_1(\hat{z}, \hat{y})$:

$$\begin{aligned} F_2(x^*) &= \sum_{t=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} d_{ij} \cdot x^*_{tijk} \\ &= \sum_{t=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} \cdot \left(\sum_{k=1}^{n} x^*_{tijk} \right) \\ &= \sum_{t=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} \cdot \hat{y}_{tij} \\ &= F_1(\hat{z}, \hat{y}). \end{aligned}$$

The result follows, since $LB4 = F_2(x^*) = F_1(\hat{z}, \hat{y}) \geq F_1(z^*, y^*) = LB3$, where z^*, y^* is an optimal solution to $\overline{\text{N3}}$.

Sketch of the proof of Theorem 2 Let $w^{1^*}, w^{2^*}, w^{3^*}$ be an optimal solution to $\overline{\text{N5}}$ satisfying constraints (29)–(32). We build a solution $\hat{x}$ to $\overline{\text{N4}}$ by defining

$$\hat{x}_{tijk} = w^{2^*}_{tij,k-1} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \left(w^{3^*}_{tijl,k-1} + w^{3^*}_{tlij,k-2} \right),$$

$$t = 1, \dots, n,\ \forall i : (i,t) \in G,\ \forall j : (j,t) \in G,$$

$$k = 1, \dots, n-1,\ \text{with } i \neq j,$$

$$\hat{x}_{ttjk} = w^{1^*}_{tjk} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} w^{2^*}_{tjlk} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{\substack{m=1 \\ (m,t) \in G}}^{n} w^{3^*}_{tjlmk},$$

$$t = 1, \dots, n,\ \forall j : (j,t) \in G,\ k = 1, \dots, n-1,$$

$$\hat{x}_{tjtk} = w^{1^*}_{tj,k-1} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} w^{2^*}_{tlj,k-2} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{\substack{m=1 \\ (m,t) \in G}}^{n} w^{3^*}_{tlmj,k-3},$$

$$t = 1, \dots, n, \quad \forall j : (j,t) \in G,\ k = 2, \dots, n,$$

$$\hat{x}_{tttk} = 1 - \sum_{\substack{i=1 \\ (l,t) \in G}}^{n} \sum_{\substack{j=1 \\ (j,t) \in G}}^{n} \hat{x}_{tijk} - \sum_{\substack{j=1 \\ (j,t) \in G}}^{n} \hat{x}_{ttjk} - \sum_{\substack{j=1 \\ (j,t) \in G}}^{n} \hat{x}_{tjtk},$$

$$t = 1, \dots, n,\ k = 1, \dots, n$$

and $\hat{x}_{tijk} = 0$ for all other t, i, j, and k.

It can be proved by variable substitution that $\hat{x}$ satisfies constraints (15)–(22), i.e., it is feasible to $\overline{\text{N4}}$. We now show that $F_3(w^{1^*}, w^{2^*}, w^{3^*}) = F_2(\hat{x})$:

$$\begin{aligned} F_2(\hat{x}) &= \sum_{t=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} d_{ij} \cdot \hat{x}_{tijk} \\ &= \sum_{t=1}^{n} \sum_{\substack{j=1 \\ j \neq t}}^{n} \sum_{k=1}^{n} d_{tj} \cdot \hat{x}_{ttjk} + \sum_{t=1}^{n} \sum_{\substack{i=1 \\ i \neq t}}^{n} \sum_{\substack{j=1 \\ j \neq i \neq t}}^{n} \sum_{k=1}^{n} d_{ij} \cdot \hat{x}_{tijk} \\ &\quad + \sum_{t=1}^{n} \sum_{\substack{i=1 \\ i \neq t}}^{n} \sum_{k=1}^{n} d_{it} \cdot \hat{x}_{titk} \\ &= \sum_{t=1}^{n} \sum_{\substack{j=1 \\ j \neq t}}^{n} \sum_{k=1}^{n-1} d_{tj} \cdot \hat{x}_{ttjk} + \sum_{t=1}^{n} \sum_{\substack{i=1 \\ i \neq t}}^{n} \sum_{\substack{j=1 \\ j \neq i \neq t}}^{n} \sum_{k=1}^{n-1} d_{ij} \cdot \hat{x}_{tijk} \\ &\quad + \sum_{t=1}^{n} \sum_{\substack{i=1 \\ i \neq t}}^{n} \sum_{k=2}^{n} d_{it} \cdot \hat{x}_{titk} \\ &= \sum_{t=1}^{n} \sum_{\substack{j=1 \\ (j,t) \in G}}^{n} d_{tj} \cdot \left(\sum_{k=1}^{n-1} w^{1^*}_{tjk} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{k=1}^{n-1} w^{2^*}_{tjlk} \right. \\ &\quad \left. + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{\substack{m=1 \\ (m,t) \in G}}^{n} \sum_{k=1}^{n-1} w^{3^*}_{tjlmk} \right) \\ &\quad + \sum_{t=1}^{n} \sum_{\substack{i=1 \\ (i,t) \in G}}^{n} \sum_{\substack{j=1 \\ (j,t) \in G}}^{n} d_{ij} \cdot \left(\sum_{k=1}^{n-1} w^{2^*}_{tij,k-1} \right. \\ &\quad \left. + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{k=1}^{n-1} w^{3^*}_{tijl,k-1} + \sum_{\substack{l=1 \\ (l,t) \in G}}^{n} \sum_{k=1}^{n-1} w^{3^*}_{tlij,k-2} \right) \end{aligned}$$

$$+\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n} d_{it}\cdot\Bigg(\sum_{k=2}^{n} w^{1^*}_{ti,k-1}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=2}^{n} w^{2^*}_{tli,k-2}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{\substack{m=1\\(m,t)\in G}}^{n}\sum_{k=2}^{n} w^{3^*}_{tlmi,k-3}\Bigg).$$

By removing the variables set to zero by (28), we obtain:

$$\begin{aligned}
F_2(\hat{x}) &= \sum_{t=1}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n} d_{tj}\cdot\Bigg(\sum_{k=1}^{n-1} w^{1^*}_{tjk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-2} w^{2^*}_{tjlk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{\substack{m=1\\(m,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tjlmk}\Bigg)\\
&\quad+\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n} d_{ij}\cdot\Bigg(\sum_{k=2}^{n-1} w^{2^*}_{tij,k-1}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=2}^{n-2} w^{3^*}_{tijl,k-1}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=3}^{n-1} w^{3^*}_{tlij,k-2}\Bigg)\\
&\quad+\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n} d_{it}\cdot\Bigg(\sum_{k=2}^{n} w^{1^*}_{ti,k-1}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=3}^{n} w^{2^*}_{tli,k-2}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{\substack{m=1\\(m,t)\in G}}^{n}\sum_{k=4}^{n} w^{3^*}_{tlmi,k-3}\Bigg)\\
&= \sum_{t=1}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n} d_{tj}\cdot\Bigg(\sum_{k=1}^{n-1} w^{1^*}_{tjk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-2} w^{2^*}_{tjlk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{\substack{m=1\\(m,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tjlmk}\Bigg)\\
&\quad+\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n} d_{ij}\cdot\Bigg(\sum_{k=1}^{n-2} w^{2^*}_{tijk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tijlk}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tlijk}\Bigg)\\
&\quad+\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n} d_{it}\cdot\Bigg(\sum_{k=1}^{n-1} w^{1^*}_{tik}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-2} w^{2^*}_{tlik}+\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{\substack{m=1\\(m,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tlmik}\Bigg)\\
&= c_{tj}\cdot\Bigg(\sum_{t=1}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n}\sum_{k=1}^{n-1} w^{1^*}_{tjk}\Bigg)+c_{tij}\cdot\Bigg(\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n}\sum_{k=1}^{n-2} w^{2^*}_{tijk}\Bigg)\\
&\quad+c_{tijl}\cdot\Bigg(\sum_{t=1}^{n}\sum_{\substack{i=1\\(i,t)\in G}}^{n}\sum_{\substack{j=1\\(j,t)\in G}}^{n}\sum_{\substack{l=1\\(l,t)\in G}}^{n}\sum_{k=1}^{n-3} w^{3^*}_{tijlk}\Bigg)\\
&= F_3\big(w^{1^*},w^{2^*},w^{3^*}\big).
\end{aligned}$$

The result follows, since $LB5=F_3(w^{1^*},w^{2^*},w^{3^*})=F_2(\hat{x})\geq F_2(x^*)=LB4$, where x^* is an optimal solution to $\overline{\text{N4}}$.

References

Anagnostopoulos, A., Michel, L., Van Hentenryck, P., & Vergados, Y. (2006). A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, *9*, 177–193.

Danna, E., Rothberg, E., & Le Pape, C. (2005). Exploring relaxation induced neighborhoods to improve mip solutions. *Mathematical Programming*, *102*, 71–90.

Duarte, A., Ribeiro, C. C., & Urrutia, S. (2007). A hybrid ILS heuristic to the referee assignment problem with an embedded MIP strategy. In Th. Bartz-Beielstein, M.J. Blesa, C. Blum, B. Naujoks, A. Roli, G. Rudolph, & M. Sampels (Eds.), *Lecture notes in computer science: Vol. 4771. Hybrid metaheuristics, 4th international workshop* (pp. 82–95). Berlin: Springer.

Durán, G., Noronha, T. F., Ribeiro, C. C., Souyris, S., & Weintraub, A. (2007). Branch-and-cut for a real-life highly constrained soccer tournament scheduling problem. In E. Burke, & Rudová H. (Eds.), *Lecture notes in computer science: Vol. 3867. Practice and theory of automated timetabling VI* (pp. 174–186). Berlin: Springer.

Easton, K., Nemhauser, G. L., & Trick, M. A. (2001). The traveling tournament problem: Description and benchmarks. In T. Walsh (Ed.), *Lecture notes in computer science: Vol. 2239. Principles and practice of constraint programming* (pp. 580–584). Berlin: Springer.

Easton, K., Nemhauser, G. L., & Trick, M. (2003). Solving the traveling tournament problem: A combined integer programming and constraint programming approach. In G. Goos, J. Hartmanis, & J. van Leeuwen (Eds.), *Lecture notes in computer science: Vol. 2740. Practice and theory of automated timetabling IV* (pp. 100–109). Berlin: Springer.

Easton, K., Nemhauser, G. L., & Trick, M. (2004). Sports scheduling. In J.T. Leung (Ed.), *Handbook of scheduling: algorithms, models and performance analysis* (pp. 52.1–52.19). Boca Raton: CRC Press.

Henz, M. (1999) Constraint-based round robin tournament planning. In D. De Schreye (Ed.) *International conference on logic programming* (pp. 545–557), New Mexico.

Knust, S. (2007). *Scheduling non-professional table-tennis leagues* (Technical Report 270). Osnabrücker Schriften zur Mathematik.

Knust, S., & von Thaden, M. (2006). Balanced home-away assignments. *Discrete Optimization*, *3*, 354–365.

Miyashiro, R., Iwasaki, H., & Matsui, T. (2003). Characterizing feasible pattern sets with a minimum number of breaks. In G. Goos, J. Hartmanis, & J. van Leeuwen (Eds.), *Lecture notes in computer science: Vol. 2740*. *Practice and theory of automated timetabling IV* (pp. 78–99). Berlin: Springer.

Nemhauser, G. L., & Trick, M. A. (1998). Scheduling a major college basketball conference. *Operations Research*, *46*, 1–8.

Rasmussen, R. V., & Trick, M. A. (2006). The timetable constrained distance minimization problem. In J.C. Beck, & B.M. Smith (Eds.), *Lecture notes in computer science: Vol. 3990*. *Integration of AI and OR techniques in constraint programming for combinatorial optimization problems* (pp. 167–181). Berlin: Springer.

Rasmussen, R. V., & Trick, M. A. (2008). Round robin scheduling—a survey. *European Journal of Operational Research*, *188*, 617–636.

Régin, J. (2001). Minimization of the number of breaks in sports scheduling problems using constraint programming. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, *57*, 115–130.

Ribeiro, C. C., & Urrutia, S. (2007a). Heuristics for the mirrored traveling tournament problem. *European Journal of Operational Research*, *179*, 775–787.

Ribeiro, C. C., & Urrutia, S. (2007b). Scheduling the Brazilian soccer championship. In E. Burke, & H. Rudová (Eds.), *Lecture notes in computer science: Vol. 3867*. *Practice and theory of automated timetabling VI* (pp. 149–159). Berlin: Springer.

Rothberg, E. (2007). An evolutionary algorithm for polishing mixed integer programming solutions. *INFORMS Journal on Computing*, *19*, 534–541.

Trick, M. (2007). *Challenge traveling tournament instances*. Online reference at http://mat.gsia.cmu.edu/TOURN/, last visited on August 10, 2007.