A branch-and-cut algorithm for scheduling the highly-constrained Chilean soccer tournament

Thiago F. Noronha[1], Celso C. Ribeiro[2],
Guillermo Duran[3], Sebastian Souyris[3], and Andres Weintraub[3]

[1] Department of Computer Science, Catholic University of Rio de Janeiro,
Rua Marquês de São Vicente 225, Rio de Janeiro, RJ 22453-900, Brazil.
[2] Department of Computer Science, Universidade Federal Fluminense,
Rua Passo da Pátria 156, Niterói, RJ 24210-240, Brazil.
[3] Department of Industrial Engineering, University of Chile,
Republica 701, Santiago, Chile.
{tfn,celso}@inf.puc-rio.br, {gduran,ssouyris,aweintra}@dii.uchile.cl

Abstract. The Chilean soccer championship follows the structure of a compact single round robin tournament. Good schedules are of major importance for the success of the tournament, making them more balanced, profitable, and attractive. The schedules were prepared by ad hoc procedures until 2004, when a rough integer programming strategy was proposed. In this work, we improve the original integer programming formulation. We derive valid inequalities for improving the liner relaxation bound and we propose a new branch-and-cut strategy for the problem. Computational results on a real-life instance illustrate the effectiveness of the approach and the improvement in solution quality.

1 Introduction

There are 20 teams in the first division of the Chilean national soccer championship, organized by the National Association of Professional Soccer (ANFP). It is organized in two phases: qualifying and playoffs. The qualifying phase follows the structure of a compact single round robin tournament. The teams are evenly distributed over four groups with five teams each. The groups are formed according to the performance of each team in the last tournament. Each of the first four teams is placed in one of the four groups. The teams from the 5th to the 8th places are randomly distributed in different groups. The same happens with the teams from the 9th to the 12th places. This procedure is repeated until all teams are assigned to a group. At the end of the qualifying phase, the teams that end up in the two first positions of each group qualify for the playoffs.

The schedules of the Chilean soccer championship were prepared by ad hoc procedures until 2004. As for most European and South American soccer championships, the games were randomly assigned to slots in a predefined round sheet. There were several drawbacks with these schedules that made them less attractive for fans and less profitable for teams: (a) classical games at inconvenient rounds, (b) weak teams playing away all their games against strong teams, (c) teams playing too many consecutive home games or too many consecutive away games, and (d) no games between traditional teams and teams from tourist cities during summer rounds, when many people are visiting the tourist regions. Weintraub et al. [6] tackled the problem of scheduling the Chilean soccer championship by integer programming in 2005, handling the above issues. The model was solved by a standard branch-and-cut procedure of the CPLEX solver. However, this procedure would take up to two hours of computation time to find a feasible solution. The procedure would be interrupted at this time, the set of possible home-away patterns fixed, and the resulting simplified model solved to optimality using a limited set of decision variables. In consequence, the model becomes easy and solvable in a few seconds. Although the resulting schedules were better than those obtained by the ad hoc procedures, the duality gaps could be very large and solutions lacked of quality.

Good schedules have a major importance in the success of sports tournaments, making them more balanced, profitable, and attractive. Many authors tackled the problem of tournament scheduling optimization in different leagues and sports. Bean and Birge [2] focused on the scheduling problem for the National Basketball Association, in which the most limiting constraints concerned rest days and stadium availability. Costa [4] considered the scheduling of the National Hockey League, for which one of the objectives consisted in the minimization of the total distance traveled by all teams. Henz [8] used constraint programming to improve the processing times of the enumerative approach proposed in [10] to compute schedules for a college basketball conference. These results were later improved by Zhang [15], once again using constraint programming. We refer to Henz [9] for recent advances in constraint programming for scheduling problems in sports, as well as to Trick [13, 14] for the combination of integer and constraint programming. Bartsch et al. [1] developed a branch-and-bound procedure for scheduling the professional soccer leagues of Austria and Germany. Goossens and Spieksma [7] proposed an integer programming formulation for scheduling the Belgian soccer league, whose objective function consisted in the minimization of the violations of soft constraints. Ribeiro and Urrutia [12] solved the problem of scheduling the Brazilan soccer tournament by an approach combining backtracking and integer programming, which found optimal solutions very quickly. Croce and Oliveri [5] used a three phase strategy based on integer programming for scheduling the Italian major soccer league, involving round robin and television constraints and minimizing the number of violations of home-away pattern constraints.

In this work, we tackle the problem of scheduling the highly-constrained Chilean soccer tournament. The original integer programming formulation of Weintraub et al. [6] is improved and valid inequalities are derived to strengthen the linear relaxation bound. We also developed a new branch-and-cut strategy that finds much better results than the previous approach.

The integer programming formulation is presented in Section 2. The solution approach and the branching strategy are described in Section 3. Computational results are reported in Section 4. Concluding remarks are made in the last section.

2 Problem formulation

In this section, the problem of scheduling the Chilean soccer tournament is stated. We present the constants, variables, and constraints of the mathematical formulation, as well as its objective function. We first remind some widely used terminology in sports scheduling. A single round robin tournament is one in which each team plays against every other exactly once. A round robin tournament is compact if every team plays exactly once in each round. A home-away pattern (HAP) is a sequence of home and away games for a given team. A break is a subsequence of two consecutive home games or two away games.

The following subsets of teams are defined:

- POP: popular teams, which are those with more fans;
- STR: strong teams, better qualified in the last tournaments;
- TRD: traditional teams (Universidad Católica, Colo-Colo, and Universidad de Chile);
- STG: teams whose home city is Santiago; and
- TUR: teams from tourist cities, visited in summer and holidays.

Some constraints involve games and relationships between specific pairs of teams:

- CMP: pairs of teams with complementary HAPs (whenever one of them plays at home the other plays away, and vice-versa);
- EXC: pairs of excluding teams (whenever a third team plays against one of them at home, then it should play away against the other, and vice-versa); and
- GRP: pairs of teams in a same group.

Since some constraints involve some specific rounds, we also define:

- SUM: summer rounds; and
- WED: Wednesday rounds.

Chile is geographically divided into thirteen regions (numbered from 1 to 13) and three zones (North, South, and Central):

- $TRG = \{4, 5\}$: tourist regions;
- $ZNS = \{\text{North, South, Central}\}$: zones;
- $FRG(r)$: teams whose home cities are at region r; and

– $FZN(z)$: teams whose home cities are at zone z.

We first define the following decision variables:

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

$$y_{ik} = \begin{cases} 1, & \text{if team } i \text{ has an away break in round } k+1, \\ 0, & \text{otherwise.} \end{cases}$$

Round robin constraints are those defining a timetable in which (i) each team plays against every other team exactly once (constraints (1)) and (ii) every team plays exactly once in each round (constraints (2)):

$$\sum_{k=1}^{19} (x_{ijk} + x_{jik}) = 1 \quad \forall i, j = 1, \ldots, 20, \quad \text{with } i < j \tag{1}$$

$$\sum_{\substack{j=1 \\ j<i}}^{20} (x_{ijk} + x_{jik}) = 1 \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \tag{2}$$

HAP constraints restrict the home-away patterns, imposing a fair balance between home and away matches: (i) each team must play at least nine (and at most ten) games at home and the others away (constraints (3)), (ii) a team may never have two consecutive breaks (constraints (4) and (5)), (iii) a team may play at most three games at home in any five consecutive rounds (constraints (6)), (iv) there may be no breaks in rounds 2, 17, and 19 (beginning and end of the tournament, constraints (7)).

$$9 \leq \sum_{\substack{j=1 \\ j \neq i}}^{20} \sum_{k=1}^{19} x_{ijk} \leq 10 \quad \forall i = 1, \ldots, 20 \tag{3}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{20,} (x_{ij(k-1)} + x_{ijk} + x_{ij(k+1)}) \leq 2 \quad \forall i = 1, \ldots, 20, \quad \forall k = 2, \ldots, 18 \tag{4}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{20} (x_{ji(k-1)} + x_{jik} + x_{ji(k+1)}) \leq 2 \quad \forall i = 1, \ldots, 20, \quad \forall k = 2, \ldots, 18 \tag{5}$$

$$\sum_{\substack{j=1\\j\neq i}}^{20}(x_{ij(k-2)}+x_{ij(k-1)}+x_{ijk}+x_{ij(k+1)}+x_{ij(k+2)})\leq 3 \qquad \forall i=1,\ldots,20,\quad \forall k=3,\ldots,17 \tag{6}$$

$$\sum_{\substack{j=1\\j\neq i}}^{20}(x_{ij(k-1)}+x_{ijk})=1 \quad \forall i=1,\ldots,20,\quad \forall k=2,17,19 \tag{7}$$

Consecutive away games are very inconvenient and should be avoided. Constraints (8) and (9) impose that every team should have at most one away break:

$$\sum_{\substack{j=1\\j\neq i}}^{20}(x_{jik}+x_{ji(k+1)})\leq 1+y_{ik} \quad \forall i=1,\ldots,20,\quad \forall k=1,\ldots,18 \tag{8}$$

$$\sum_{k=1}^{18} y_{ik}\leq 1 \quad \forall i=1,\ldots,20 \tag{9}$$

The last two HAP constraints guarantee that (i) some pairs of teams must have complementary HAPs (constraints (10)), for security reasons to avoid more than one game in the same city at the same time, to ensure that there will ever be one game in this city or because they share the same stadium, and (ii) there may be at most four teams from Santiago playing at home in any round (constraints (11)):

$$\sum_{\substack{h=1\\h\neq i,h\neq j}}^{20}(x_{ihk}+x_{jhk})=\sum_{h\neq i,h\neq j,h=1}^{20}(x_{hik}+x_{hjk}) \qquad \forall (i,j)\in CMP,\quad \forall k=1,\ldots,19 \tag{10}$$

$$\sum_{i\in STG}\sum_{\substack{j=1\\j\neq i}}^{20}x_{ijk}\leq 4 \quad \forall k=1,\ldots,19 \tag{11}$$

Team constraints restrict the rounds in which games between special pairs of teams can be played: (i) each team should play at least one game between two consecutive games against popular teams (constraints (12)), (ii) each team may have at most two consecutive games against strong teams (constraints (13)), (iii) each traditional team plays exactly one classical game (i.e., a game against another traditional team) at home (constraints (14)), and (iv) classical games must be played between rounds 8 and 17 (constraints (15)).

$$\sum_{j \in POP \setminus \{i\}} (x_{ijk} + x_{jik} + x_{ij(k+1)} + x_{ji(k+1)}) \leq 1$$
$$\forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 18 \tag{12}$$

$$\sum_{j \in STR \setminus \{i\}} (x_{ijk} + x_{jik} + x_{ij(k+1)} + x_{ji(k+1)} + x_{ij(k+2)} + x_{ji(k+2)}) \leq 2$$
$$\forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 17 \tag{13}$$

$$\sum_{k=1}^{19} (x_{hik} + x_{jik}) = \sum_{k=1}^{19} (x_{hjk} + x_{ijk}) \quad \forall i, j, h \in TRD \tag{14}$$

$$\sum_{i \in TRD} \sum_{\substack{j \in TRD \\ j \neq i}} (\sum_{k=1}^{7} x_{ijk} + \sum_{k=18}^{19} x_{ijk}) = 0 \tag{15}$$

The strong teams are grouped into pairs to balance the hardness of home and away games. Whenever a team plays at home against one of the teams of a pair of excluding teams, then it will play away against the other (and vice-versa), as stated by constraints (16):

$$\sum_{k=1}^{19} (x_{hik} + x_{hjk}) = 1 \quad \forall (i, j) \in EXC, \quad \forall h = 1, \ldots, 20, h \neq i, h \neq j \tag{16}$$

Geographic constraints tackle Chile's particular geography of a very long and narrow country: a team from the Central zone cannot play away in the same week against a team from the South and another from the North, and vice-versa (constraints (17) and (18)). Whenever a team from the Central zone plays against a team from the North (resp. South) on a Wednesday, then it cannot play against a team from the South (resp. North) in the previous or forthcoming weekend. Furthermore, we point out that the first and last rounds are always scheduled on weekends.

$$\sum_{j \in FZN(\text{South})} (x_{ji(k-1)} + x_{ji(k+1)}) + \sum_{j \in FZN(\text{North})} 2 \cdot x_{jik} \leq 2$$
$$\forall i \in FZN(\text{Central}), \quad \forall k \in WED \tag{17}$$

$$\sum_{j \in FZN(\text{North})} (x_{ji(k-1)} + x_{ji(k+1)}) + \sum_{j \in FZN(\text{South})} 2 \cdot x_{jik} \leq 2$$
$$\forall i \in FZN(\text{Central}), \quad \forall k \in WED \quad (18)$$

There are also constraints on tourist teams and regions: (i) each tourist team should play at least once at home against a traditional team during the summer rounds (constraints (19)) and (ii) each traditional team should not play twice in the same week at the same tourist region (constraints (20)).

$$\sum_{k \in SUM} \sum_{j \in TRD \setminus \{i\}} x_{ijk} \geq 1 \quad \forall i \in TUR \quad (19)$$

$$\sum_{i \in FRG(r) \setminus \{j\}} (x_{ij(k-1)} + 2 \cdot x_{ijk} + x_{ij(k+1)}) \leq 2$$
$$\forall j \in TRD, \quad \forall r \in TRG, \quad \forall k \in WED \quad (20)$$

Since only the teams in the two first positions of each group qualify for the playoffs, games between teams in the same group are more attractive. Therefore, these games should as much as possible take place at the end of the tournament. The objective function (21) consists in maximizing the number of games between teams in the same group in the last rounds of the tournament:

$$\text{maximize} \sum_{(i,j) \in GRP} \sum_{k=1}^{19} k \cdot x_{ijk} \quad (21)$$

Weintraub et al. [6] attempted to apply a standard CPLEX branch-and-cut algorithm directly to the above formulation. However, the lower bounds provided by its linear relaxation were very poor because of the flow spread among the originally binary x variables. CPLEX heuristics were not able to find primal solutions. The computation times were very high, because the formulation is degenerated for the simplex method and can only be solved by perturbation techniques.

3 Solution approach

The original integer programming formulation described in the previous section can be significantly improved. Valid inequalities are derived in Section 3.1 to improve the lower bounds and a new branch-and-cut strategy is proposed in Section 3.2 to speedup convergence.

3.1 Improved formulation

We first the following additional binary variables:

$$z_{ik} = \begin{cases} 1, & \text{if team } i \text{ plays at home in round } k; \\ 0, & \text{otherwise.} \end{cases}$$

They can be associated with the other variables by constraints (22) and (23):

$$z_{ik} = \sum_{\substack{j=1 \\ j \neq i}}^{20} x_{ijk} \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \tag{22}$$

$$z_{ik} + z_{i(k+1)} \geq 1 - y_{ik} \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 18 \tag{23}$$

All HAP constraints may be rewritten in terms of the new variables by substituting $z_{ik} = \sum_{j \neq i, j=1}^{20} x_{ijk}$: constraints (3) are rewritten as (24), constraints (4) and (5) as (25), constraints (6) as (26), constraints (7) as (27), constraints (8) as (23), constraints (10) as (28), and constraints (11) as (29):

$$\sum_{k=1}^{19} z_{ik} \geq 9 \quad \forall i = 1, \ldots, 19 \tag{24}$$

$$1 \leq z_{i(k-1)} + z_{ik} + z_{i(k+1)} \leq 2 \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \tag{25}$$

$$\begin{aligned} z_{i(k-2)} + z_{i(k-1)} + z_{ik} + z_{i(k+1)} + z_{i(k+2)} \leq 3 \\ \forall i = 1, \ldots, 20, \quad \forall k = 2, \ldots, 19 \end{aligned} \tag{26}$$

$$z_{i(k-1)} + z_{ik} = 1 \quad \forall i = 1, \ldots, 20, \quad \forall k = 2, 17, 19 \tag{27}$$

$$z_{ik} + z_{jk} = 1 \quad \forall (i,j) \in CMP, \quad \forall k = 1, \ldots, 19 \tag{28}$$

$$\sum_{i \in STG} z_{ik} \leq 4 \quad \forall k = 1, \ldots, 19 \tag{29}$$

To strengthen the original formulation, we use an approach similar to that proposed by Trick [13], with the exception that the rounds in which the games will be played are not fixed. Additional variables are defined:

$$s_i = \begin{cases} 1, & \text{if team } i \text{ plays at home in the first round,} \\ 0, & \text{otherwise;} \end{cases}$$

$$h_{ik} = \begin{cases} 1, & \text{if team } i \text{ plays away in round } k \text{ and at home in round } k+1, \\ 0, & \text{otherwise;} \end{cases}$$

$$w_{ik} = \begin{cases} 1, & \text{if team } i \text{ plays at home in round } k \text{ and away in round } k+1, \\ 0, & \text{otherwise.} \end{cases}$$

Variables z, s, h, and w are related by equations (30) and (31):

$$z_{ik} = s_i + \sum_{t=1}^{k-1} (h_{it} - w_{it}) \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \qquad (30)$$

$$h_{ik} + w_{ik} \leq 1 \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \qquad (31)$$

The original formulation can now be strengthened by the following valid inequalities:

$$z_{ik} + h_{ik} \leq 1 \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \qquad (32)$$

$$z_{ik} - w_{ik} \geq 0 \quad \forall i = 1, \ldots, 20, \quad \forall k = 1, \ldots, 19 \qquad (33)$$

Constraints (32) and (33) do not improve the linear relaxation bound. However, they avoid that variables z assume fractional values, speeding up the integer programming algorithm. The new formulation (1,2,9,12–33) is still degenerated, but its coefficient matrix is more sparse and can be rapidly solved by an interior point algorithm [3]. Furthermore, the variables z play a major role in the branching strategy.

3.2 Branch-and-cut

We developed a cutting plane procedure based on odd-set cuts to improve the linear relaxation bound, as suggested by Trick [14]. Padberg and Rao [11] showed that these cuts can be separated in polinomial time. They come from the fact that each round can be seen as a perfect matching in the complete graph whose node set is formed by the teams taking part in the tournament. The odd-set constraints can be described as follows. For each particular round $k = 1, \ldots, 19$, let S be any subset of teams such that $|S|$ is odd:

$$\sum_{i \in S, j \notin S} (x_{ijk} + x_{jik}) \geq 1 \quad \forall k = 1, \ldots, 19, \quad \forall S \subseteq \{1, \ldots, 20\}, \quad |S| \text{ odd.}$$

Cuts associated with odd-set constraints violated by the solution of the linear relaxation are progressively added to the enumeration tree. The number of times the linear relaxation is solved in each node is limited, because solving the linear relaxation is computationally demanding. After the linear relaxation is solved, all odd-set cuts such that

$$\sum_{i \in S, j \notin S} (x^*_{ijk} + x^*_{jik}) \leq \Delta$$

are determined by a separation procedure and appended to the model (x^*_{ijk} denotes the value of variable x_{ijk} in the optimal solution of the linear relaxation), with Δ fixed at 0.1. This procedure is repeated while the linear relaxation bound can be improved.

The experimental results showed that this procedure strongly improved the linear relaxation bound, which was already equal to the optimal value at the root of the enumeration tree for the 2005 edition of the Chilean soccer tournament.

The branching strategy plays a major role in the success of a branch-and-cut algorithm. Branching on the x variables is not efficient, since most of them are null in integral solutions. Our branching strategy is based on the z variables. Branching on the x variables starts only after all the z variables are integral. This strategy implicitly decomposes the solution in two phases. The HAPs are computed in the first phase, while the dates of the games are established in the second. Once the z variables are fixed, the branch-and-cut algorithm needs just a few branches on the x variables to find a feasible solution or to prove infeasibility.

4 Computational experiments

The branch-and-cut strategy described in the previous section was implemented using Visual C++ 6.0 and CPLEX 8.0. The same algorithm without the odd-set cuts was also implemented to evaluate the effectiveness of the cutting plane procedure. We refer to the first algorithm as `B&C-ANFP` and to the second as `B&B-ANFP`. The computational experiments were performed on a 3 GHz Pentium IV machine with 1 Gbyte of RAM memory. We illustrate the results obtained for the 2005 edition of the Chilean soccer championship, comparing them with those reported in [6]. Table 1 shows the name of each team and its respective group, as well as the identification used to represent each team in Tables 5 and 6.

Computation times for solving the linear programming relaxation by different algorithms available with CPLEX 8.0 are given in Table 2. Since

Table 1. Teams in the 2005 edition of the Chilean soccer championship.

Group 1	Group 2	Group 3	Group 4
Colo-Colo (1)	Cobreloa (2)	U. de Concepción (3)	U. de Chile (0)
Audax Italiano (5)	Wanderers (6)	Unión Española (8)	U. Católica (4)
Huachipato (7)	Coquimbo (9)	Temuco (10)	Eeverton (11)
San Felipe (13)	Puerto Montt (12)	Palestino (16)	Cobresal (17)
Melipilla (19)	La Serena (14)	Desportes Concepción (18)	Rangers (15)

the problem is degenerated for both the primal and dual simplex methods, their computation times were very high. Therefore, the interior point algorithm presented the best computation times for solving the linear relaxation. Table 2 shows that the new formulation considerably reduced the computation time of the interior points algorithm, leading to an efficient implementation of the cutting plane strategy.

Table 2. Computation times in seconds for solving the linear relaxation.

Strategy	time (s)
Primal simplex	27
Dual simplex	21
Interior points (original formulation)	12
Interior points (with the additional z variables)	4

Detailed results obtained with algorithms `B&B-ANFP` and `B&C-ANFP` are given in Table 3. For each algorithm, we report the value of the objective function, the number of nodes in the enumeration tree, and the integrality gap after some elapsed times (ranging from ten minutes to four hours). We notice that algorithm `B&B-ANFP` finds good solutions faster than `B&C-ANFP` in the beginning. However, the former was not able to find the optimal solution within a 4-hour time limit. On the contrary, the cuts used by algorithm `B&C-ANFP` were able to improve the linear relaxation bound, which was already equal to the optimal value at the root of the enumeration tree. The number of nodes is much smaller for algorithm `B&C-ANFP`, that found the exact optimal solution in less than two hours of computation time.

Table 3. Comparison between algorithms `B&B-ANFP` and `B&C-ANFP`.

Elapsed time	`B&B-ANFP` objective	nodes	gap (%)	`B&C-ANFP` objective	nodes	gap (%)
10 minutes	617	140	3.6	474	120	25.9
30 minutes	622	600	2.9	615	330	3.9
1 hour	631	1120	1.4	633	570	1.1
2 hours	633	2190	1.1	640	1560	0.0
4 hours	639	4860	0.2	—	—	—

In Table 4, we compare the results obtained by algorithm `B&C-ANFP` with those obtained by the strategy proposed in [6]. We give the relative integrality gap from the optimal solution after 30 minutes and after two hours of computation time (on a 2.4 GHz Pentium IV computer for [6]) for both algorithms. Algorithm `B&C-ANFP` not only found a better solution quickly, but also found a much better – and optimal – solution after the same time the approach in [6] took to find a solution 9.2% away from the optimal value.

The schedules provided by [6] and `B&C-ANFP` are presented in Tables 5 and 6, respectively. The lines correspond to teams and the columns to rounds. Games between teams from the same group are underlined. Table 5 shows that the schedule obtained by [6] has games between teams from the same group spread along all rounds, while the schedule provided by the new algorithm has all games between teams from the same group in the last five rounds of the tournament.

Table 4. Comparison of algorithms `B&C-ANFP` and Weintraub et al [6].

Algorithm	30 minutes	2 hours
`B&C-ANFP`	3.9%	0.0%
Weintraub et al [6]	-	9.2 %

5 Concluding remarks

We proposed a new formulation for the highly constrained Chilean soccer tournament scheduling problem. Valid inequalities were derived and appended to the formulation to improve its liner relaxation bound. A branching strategy based on the new variables was used to speedup convergence.

Table 5. Schedule provided by [6].

T\R	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	**@11**	13	@9	16	@12	18	@2	6	@7	5	@8	@1	14	@19	**4**	10	@3	**15**	**@17**
1	12	@6	17	**@5**	3	@14	11	@10	2	@4	9	0	@16	8	@15	@18	**7**	**@13**	**19**
2	@8	16	@7	@3	10	@19	0	15	@1	18	@5	13	@11	**9**	17	@4	**14**	**@6**	**12**
3	5	@14	@11	2	@1	13	@9	7	@15	**8**	18	@17	**10**	@4	12	@6	0	**@18**	**16**
4	19	@12	@10	8	@18	6	@13	5	@14	1	@16	7	@9	3	**@0**	2	**@17**	**11**	**@15**
5	@3	11	@8	**1**	@9	16	@15	@4	10	@0	2	@18	17	@14	**19**	@12	6	**@7**	**13**
6	@13	1	@15	11	17	@4	10	@0	19	**@12**	**@14**	16	@15	7	@8	3	@5	**2**	**@9**
7	14	@18	2	@10	@16	15	17	@3	0	@11	12	@4	**19**	@6	**13**	9	**@1**	**5**	@8
8	2	@15	5	@4	11	@14	14	@19	9	**@3**	0	**@10**	12	@1	6	@13	**18**	**@16**	7
9	18	@17	0	@19	5	@10	3	@11	@8	13	@1	15	4	**@2**	16	@7	**12**	**@14**	**6**
10	17	@19	4	7	@2	9	@6	1	@5	15	@11	**8**	**@3**	@13	14	@0	**16**	@12	**18**
11	**0**	@5	3	@6	@8	12	@1	9	@13	7	10	@19	2	**@17**	18	@16	**15**	**@4**	14
12	@1	4	19	@17	0	@11	@16	13	@18	**6**	@7	**14**	@8	15	@3	5	**@9**	10	**@2**
13	6	@0	@16	15	14	@3	4	@12	11	@9	17	@2	18	10	**@7**	8	**@19**	**1**	**@5**
14	@7	3	@15	18	@13	1	@8	16	4	@17	**6**	**@12**	@0	5	@10	19	**@2**	**9**	@11
15	@16	8	14	@13	19	@7	5	@12	3	@10	18	@9	6	@12	1	**17**	**@11**	**@0**	**4**
16	15	@2	13	@0	7	@5	12	@14	17	@19	4	@16	1	**@18**	@9	11	**@10**	**8**	**@3**
17	@10	2	@1	12	@6	8	@7	18	@16	14	@13	3	@5	**11**	@2	**@15**	**4**	@19	**0**
18	@9	7	6	@14	4	@0	19	@17	12	@2	@15	5	@13	**16**	@11	1	**@8**	**3**	**@10**
19	@4	10	@12	9	@15	2	@18	8	@6	16	@3	11	**@7**	0	**@5**	@14	**13**	17	**@1**

Table 6. Schedule provided by B&C-ANFP.

T\R	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
0	16	@9	14	@3	10	@6	13	@5	@18	1	@19	12	@7	8	**@17**	**11**	**@4**	2	**@15**
1	@14	15	@2	18	@11	9	@12	17	3	@0	4	@10	6	@16	**19**	**@13**	**5**	@8	**7**
2	@5	13	1	@4	3	@15	10	16	@19	8	@18	7	@11	17	**@14**	**@12**	**6**	@0	**9**
3	@7	14	@19	0	@2	11	@17	12	@1	@13	5	@15	4	@9	6	**@8**	**10**	**18**	**@16**
4	@10	5	@16	2	@13	19	@8	18	@7	12	@1	14	@3	@6	**15**	@9	**0**	**@17**	**11**
5	2	@4	6	@10	15	16	@9	0	@12	18	@3	8	@17	14	@11	**19**	**@1**	**@7**	**13**
6	@15	10	@5	13	@16	0	@19	8	@17	11	@7	18	@1	4	@3	**14**	**@2**	**@9**	**12**
7	3	@8	17	11	@9	12	@18	@14	4	@15	6	@2	0	@10	**13**	16	**@19**	**5**	**@1**
8	@11	7	@9	19	@12	@17	4	@6	14	@2	15	@5	13	@0	**10**	**3**	**@16**	1	**@18**
9	@13	0	8	@17	7	@1	5	@10	15	@16	@11	19	@18	3	**@12**	4	**@14**	**6**	**@2**
10	4	@6	@15	5	@0	13	@2	9	@11	19	@17	1	@12	7	**@8**	**18**	**@3**	**16**	@14
11	8	@18	12	@7	1	@3	14	@13	10	@6	9	@16	2	@19	5	**@0**	**17**	**15**	**@4**
12	@17	19	@11	@15	8	@7	1	@3	5	@4	16	@0	10	@13	**9**	**2**	@18	**14**	**@6**
13	9	@2	18	@6	4	@10	@0	11	@16	3	@14	17	@8	12	**@7**	**1**	@15	**19**	**@5**
14	1	@3	@0	16	@19	18	@11	7	@8	17	13	@4	15	@5	**2**	**@6**	**9**	**@12**	10
15	6	@1	10	12	@5	2	@16	19	@9	7	@8	3	@14	18	**@4**	**@17**	13	**@11**	**0**
16	@0	17	4	@14	6	@5	15	@2	13	9	@12	11	@19	1	**@18**	@7	**8**	**@10**	**3**
17	12	@16	@7	9	@18	8	3	@1	6	@14	10	@13	5	@2	**0**	**15**	**@11**	**4**	@19
18	@19	11	@13	@1	17	@14	7	@4	0	@5	2	@6	9	@15	**16**	**@10**	12	**@3**	**8**
19	18	@12	3	@8	14	@4	6	@15	2	@10	0	@9	16	11	**@1**	**@5**	**7**	**@13**	17

The new formulation considerably reduced the computation times needed to solve the liner relaxation. The odd-set cuts improved the linear relaxation bound, which was already equal to the optimal value at the root of the enumeration tree. The new algorithm `B&C-ANFP` significantly outperformed the previous approach and found the optimal solution in less than two hours. Future work will deal with new constraints and objective functions imposed by TV sponsors, as well as with heuristics for providing integer feasible solutions for the model.

References

1. T. Bartsch, A. Drexl, and S. Kroger. Scheduling the professional soccer leagues of Austria and Germany. *Computers and Operations Research*, 33:1907–1937, 2006.
2. J.C. Bean and J.R. Birge. Reducing traveling costs and player fatigue in the National Basketball Association. *Interfaces*, 10:98–102, 1980.
3. D. Bertsimas and J.N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.
4. D. Costa. An evolutionary tabu search algorithm and the NHL scheduling problem. *Information Systems and Operational Research*, 33:161–178, 1995.
5. F. Della Croce and D. Oliveri. Scheduling the Italian football league: An ILP-based approach. *Computers and Operations Research*, 33:1963–1974, 2006.
6. G. Duran, M. Guajardo, J. Miranda, D. Sauré, S. Souyris, A. Weintraub, A. Carmash, and F. Chaineu. Programación matemática aplicada al fixture de la primera divisón del fútbol chileno. *Revista Ingeniería de Sistemas*, 5:29–46, 2005.
7. D. Goossens and F.C.R. Spieksma. Scheduling the Belgian soccer league. In E.K. Burke and H. Rudová, editors, *Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling*, pages 420–422, Brno, 2006.
8. M. Henz. Scheduling a major college basketball conference revisited. *Operations Research*, 49:163–168, 2001.
9. M. Henz, T. Müller, and S. Thiel. Global constraints for round robin tournament scheduling. *European Journal of Operational Research*, 153:92–101, 2004.
10. G.L. Nemhauser and M.A. Trick. Scheduling a major college basketball conference. *Operations Research*, 46:1–8, 1998.
11. M.W. Padberg and M.R. Rao. Odd minimum cut-sets and b-matchings. *Mathematics of Operation Research*, 7:67–80, 1982.
12. C.C. Ribeiro and S. Urrutia. Scheduling the Brazilian soccer championship. In E.K. Burke and H. Rudová, editors, *Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling*, pages 481–483, Brno, 2006.
13. M.A. Trick. A schedule-and-break approach to sports scheduling. In *Proceedings of the 3th International Conference on the Practice and Theory of Automated Timetabling*, volume 2079 of *Lecture Notes in Computer Science*, pages 242–253. Springer, 2001.

14. M.A. Trick. Integer and constraint programming approaches for round robin tournament scheduling. In *Proceedings of the 4th International Conference on the Practice and Theory of Automated Timetabling*, volume 2740 of *Lecture Notes in Computer Science*, pages 63–77. Springer, 2003.
15. H. Zhang. Generating college conference basketball schedules by a SAT solver. In *Proceedings of the Fifth International Symposium on the Theory and Applications of Satisfiability Testing*, pages 281–291, Cincinnati, 2002.