Sports Timetabling Problem

IME639A (Analytics in Transport and Telecom) Course Project

Dwara Nikhesh Babu, 190323
Vyush Agarwal, 180881

Project files: https://github.com/nikheshd/SportsTimetablingProblem.git

Problem description:

The aim of this problem is to develop a time-constrained double round-robin timetables. Under this setting with an even number of teams, the total number of time slots is exactly equal to the total number of games per team, and hence each team plays exactly one game per time slot. There are two types of constraints, hard constraints represent fundamental properties of the timetable that can never be violated, while soft constraints represent preferences that should be satisfied whenever possible. The objective is to minimize the penalties from violated soft constraints. The problem instances are expressed using the standardized XML data format.

$$Let\ the\ number\ of\ teams = n\ \ (n \geq 2)$$

$$Number\ of\ total\ games\ possible = 2 \times \binom{n}{2}$$

$$Number\ of\ games\ possible\ in\ each\ time\ slot = \frac{n}{2}$$

$$Hence, total\ number\ of\ time\ slots = m = \frac{2 \times \binom{n}{2}}{\frac{n}{2}} = 2 \times (n-1)$$

$$Also, total\ number\ of\ games\ per\ team = 2 \times (n-1) = m$$

Note: Values of n and m are given in the problem instance, and they satisfy the above equation.

Linear Programming Formulation:

Decision variables:

$$x_{ijk} = 1 \; if \; (team \; i, team \; j) \; is \; in \; time \; slot \; k$$
$$= 0 \; otherwise$$
$$where \; i \neq j; \;\; i,j \in [0, n-1]; \;\; k \in [0, m-1]$$

Note: These are not the only decision variables, later we'll see decision variables for deviations in soft constraints: $y_c \in Z - Z^-$

Constraints:

Sports timetables need to satisfy a usually large set of constraints C, which is partitioned into hard constraints C_{hard} and soft constraints C_{soft}. For each constraint tag, this is denoted by the *type* attribute which can take the values 'HARD' and 'SOFT'. Hard constraints represent fundamental properties of the timetable that can never be violated. Soft constraints, in contrast, rather represent preferences that should be satisfied whenever possible. The validation of each constraint $c \in C$ results in a vector of n_c integral numbers, called the deviation vector $D_c = [d_1 \; d_2 \; . . . \; d_{n_c}]$. If a constraint is satisfied, all elements of its deviation vector are equal to zero. Contrarily, the deviation vector of a violated constraint contains one or more non-zero elements.

Common constraint 1:
A team plays exactly one match per time slot:

$$\sum_{j=0, \;\; i\neq j}^{n-1} \left(x_{ijk} + \; x_{jik} \right) = 1, \qquad \forall i, k$$

Common constraint 2:
A team plays exactly 2 games with any other team with different home-away status

$$\sum_{k=0}^{m-1} x_{ijk} = 1\,, \qquad \forall i \neq j$$

Phased game mode constraint:
Each series of games contain two phases:
Phase 1: Timeslot 0 to n-1
Phase 2: Timeslot n to m-1
A give series game mode is mentioned in the problem instance as shown below:

```
<Structure>
        <Format  leagueIds="0">
                 <numberRoundRobin>2</numberRoundRobin>
                 <compactness>C</compactness>
                 <gameMode>P</gameMode>
        </Format>
</Structure>
```

If gameMode == ”P” i.e. if the series has a phased game mode, then:
In each phase, a pair of two teams can exist only once. This pair exists again in the other phase with different home away status.

$$\sum_{k=0}^{n-1} (x_{ijk} + x_{jik}) = 1\,, \qquad \forall i \neq j$$

This combined with common constraint 2 will give a phased game mode.

Common constraint 3:
To make sure that some values in decision variable array are zeroes.

$$x_{iik} = 0, \qquad \forall\, i, k$$

The constraints above are fundamental constraints for a sports timetable so can be considered as hard constraints.

1. Capacity constraints:

Capacity constraints (CA) force a team to play home or away and regulate the total number
of games played by a team or group of teams. We consider four different capacity constraints.

- **CA1:**

<CA1 teams="0" max="0" mode="H" slots="0" type="HARD"/>

Only team in teams = i = 0;

$$\sum_{j=0}^{n-1} \sum_{k\ in\ slots} x_{ijk} \leq max + y_{ca1}$$

$if\ type = \text{"HARD"}, \qquad y_{ca1} = 0$

(Here y_{ca1} is a decision variable representing the deviation for a soft constraint)

Each team from teams plays at most max home games (mode = "H") or away games (mode = "A") during time slots in slots.

- **CA2:**

<CA2 teams1="0" min="0" max="1" mode1="HA" mode2="GLOBAL" teams2="1;2" slots ="0;1;2" type="SOFT"/>

Only team in teams1 = i = 0

$$\sum_{j\ in\ teams2} \sum_{k\ in\ slots} (x_{ijk} + x_{jik}) \leq max + y_{ca2}$$

$if\ type = \text{"HARD"}, \qquad y_{ca2} = 0$

(Here y_{ca2} is a decision variable representing the deviation for a soft constraint)

Each team in teams1 plays at most max home games (mode1 = "H"), away games (mode1 = "A"), or games (mode1 = "HA") against teams (mode2 = "GLOBAL"; the only mode we consider) in teams2 during time slots in slots.

- **CA3:**

<CA3 teams1="0" max="2" mode1="HA" teams2="1;2;3" intp="3" mode2= "SLOTS" type="SOFT"/>

Only team in teams1 = i = 0

$$\sum_{s=0}^{intp-1} \sum_{j \ in \ teams2} \left(x_{ijk+s} + x_{jik+s}\right) \leq max + y_{ca3_k} ,$$

$$\forall \ k \in [0, m - intp)$$

$$if \ type = \text{"HARD"}, \qquad y_{ca3} = 0$$

Each team in teams1 plays at most max home games (mode1 = "H"), away games (mode1 = "A"), or games (mode1 = "HA") against teams in teams2 in each sequence of intp time slots (mode2 = "SLOTS"; the only mode we consider).

- **CA4:**

<CA4 teams1="0;1" max="3" mode1="H" teams2="2,3" mode2="GLOBAL" slots ="0;1" type="HARD"/>

$$\sum_{i \ in \ teams1} \sum_{j \ in \ teams2} \sum_{k \ in \ slots} x_{ijk} \leq max + y_{ca4}$$

$$if \ type = \text{"HARD"}, \qquad y_{ca4} = 0$$

Teams in teams1 play at most max home games (mode1 = "H"), away games (mode1 = "A"), or games (mode1 = "HA") against teams in teams2 during time slots in slots (mode2 = "GLOBAL") or during each time slot in slots (mode2 = "EVERY").

2. Game constraints:

Game constraints enforce or forbid specific assignments of a game to time slots.

- **GA1:**

<GA1 min="0" max="0" meetings="0,1;1,2;" slots="3" type="HARD"/>

$$\sum_{(i,j) in \ meetings} \sum_{k \ in \ slots} x_{ijk} \leq max + y_{ga1}$$

$$\sum_{(i,j) in\ meetings} \sum_{k\ in\ slots} x_{ijk} \geq min - y_{ga1}$$

$$if\ type = \text{"HARD"}, \qquad y_{ga1} = 0$$

At least min and at most max games from G = {(i_1, j_1), (i_2, j_2), . . . } take place during time slots in slots.

3. Break constraints:

If a team plays a game with the same home-away status as its previous game, we say it has a break. Breaks usually are undesired since they have an adverse impact on game attendance, and they can be perceived as unfair due to the home advantage. Break constraints therefore regulate the frequency and timing of breaks in a competition.

- **BR1:**

<BR1 teams="0" intp="0" mode2="HA" slots="1" type="HARD"/>

Only team in teams = i =0;

$$\sum_{k\ in\ slots,\ \ k\neq 0} \left(\left\lfloor \frac{\sum_{j=0}^{n-1} (x_{ijk-1} + x_{ijk})}{2} \right\rfloor + \left\lfloor \frac{\sum_{j=0}^{n-1} (x_{jik-1} + x_{jik})}{2} \right\rfloor \right) \leq intp + y_{br1}$$

$$if\ type = \text{"HARD"}, \qquad y_{br1} = 0$$

Each team in teams has at most intp home breaks (mode2 = "H"), away breaks (mode2 = "A"), or breaks (mode2 = "HA") during time slots in slots.

- **BR2:**

<BR2 homeMode="HA" teams="0;1" mode2="LEQ" intp="2" slots="0;1;2;3" type="HARD "/>

$$\sum_{i\ in\ teams} \left(\sum_{k\ in\ slots,\ \ k\neq 0} \left(\left\lfloor \frac{\sum_{j=0}^{n-1} (x_{ijk-1} + x_{ijk})}{2} \right\rfloor + \left\lfloor \frac{\sum_{j=0}^{n-1} (x_{jik-1} + x_{jik})}{2} \right\rfloor \right) \right) \leq intp + y_{br1}$$

$$if\ type = \text{"HARD"}, \qquad y_{br2} = 0$$

The sum over all breaks (homeMode = "HA", the only mode we consider) in teams is no more than (mode2 = "LEQ", the only mode we consider) intp during time slots in slots.

3. Fairness constraints:

To increase the fairness and attractiveness of competitions, the following constraint can be used.

- **FA2:**

<FA2 teams="0;1;2" mode="H" intp="1" slots="0;1;2;3" type="HARD"/>

$$\left| \sum_{k\ in\ slots} \sum_{j=0}^{n-1} \left(x_{i_1 jk} - x_{i_2 jk} \right) \right| \leq intp + y_{fa2}, \qquad \forall (i_1, i_2) \in teams$$

$$if\ type = \text{"HARD"}, \qquad y_{fa2} = 0$$

Each pair of teams in teams has a difference in played home games (mode = "H", the only mode we consider) that is not larger than intp after each time slot in slots.

4. Separation constraints

Separation constraints regulate the number of rounds between consecutive games involving the same teams.

- **SE1:**

<SE1 teams="0;1" min="5" mode1="SLOTS" type="HARD"/>

$$\left| \sum_{k=0}^{m-1} \left(k \times x_{ijk} - k \times x_{jik} \right) \right| \geq \min - y_{se1}\,, \qquad \forall (i,j) \in teams$$

$$if\ type = \text{"HARD"}, \qquad y_{se1} = 0$$

Each pair of teams in teams has at least min time slots (mode1 = "SLOTS", the only mode we consider) between two consecutive mutual games.

Objective Function:

$$\sum_{c\ in\ constraints} penalty_c \times y_c$$

Each constraint $c \in C$ triggers a penalty $p_c = w_c \sum^{n_c}_{i=1} d_i = w_c\, y_c$ that is equal to the sum of the elements of the deviation vector multiplied with weight w_c (denoted by the attribute penalty in the soft constraint tags). The objective we use for the ITC2021 problem instances sums over all violated soft constraint penalties, that is $\sum_c p_c$, $c \in C_{soft}$.

Results:

```
Time taken for ITC2021_Test1: 73163537 microseconds
Time taken for ITC2021_Test2: 119532 microseconds
Time taken for ITC2021_Test3: 49472 microseconds
Time taken for ITC2021_Test4: 29213 microseconds
Time taken for ITC2021_Test5: 136038532 microseconds
```

Heuristic:

STEP 1: Consider only the hard constraints and solve the LP.
STEP 2: Now from the solution, calculate penalties for all soft constraints and sort them in decreasing order. Convert first x soft constraints to hard constraints.
STEP 3: Repeat step 1. If we get a solution, then repeat step 2,3. Else, go to step 4.
STEP 4: Reconvert the last x converted constraints to again soft hard constraints. Now only select the top x-1 constraints and repeat all the steps until the number of constraints to be converted reduces to zero (x=0).

References:

1. ITC2021 – Sports Timetabling Problem Description and File Format by David Van Bulck , Dries Goossens, Jeroen Beliën and Morteza Davari