Branching Based on Home-Away-Pattern Sets

Dirk Briskorn and Andreas Drexl

Lehrstuhl für Produktion & Logistik, Christian-Albrechts-Universität zu Kiel, Germany
briskorn@bwl.uni-kiel.de
andreas.drexl@bwl.uni-kiel.de

1 Introduction

Scheduling a sports league requires to solve a hard combinatorial optimization problem. We consider a league of a set T of n teams supposed to play in a single round robin tournament (SRRT). Accordingly, each team $i \in T$ has to play against each other team $j \in T, j \neq i$, exactly one match. The tournament is partitioned into matchdays (MD) being periods where matches can be carried out. Each team $i \in T$ shall play exactly once per MD. Hence, we have a compact structure resulting in an ordered set P of $n-1$ MDs.

Since each match has to be carried out at one of both opponents' venues breaks come into play. A break for team i occurs at MD p if i has two consecutive matches at home or away, respectively, at MDs $p-1$ and p. We distinguish between home-breaks and away-breaks depending on the breaks' venue. A popular goal concerning breaks is to minimize the number of their occurence. Apparently, the number of breaks can not be less than $n-2$; see [3].

There have been several efforts in order to find good or even optimal SRRTs, see [1] and [4] for example. However, all available algorithms search only a rather small part of the solution space which is a lack we aim to overcome. In section 2 the problem is described in detail and a corresponding mathematical model is given. Section 3 focuses on the branching scheme to tackle the problem. Some computational results are outlined in section 4 and, finally, section 5 gives some conclusions and an outlook to future research.

2 Model

Assume that we have cost $c_{i,j,p}$ if team i plays at home against team j at MD p. The goal of the SRRT introduced in [2] is to find a SRRT having the minimum sum of arranged matches' cost. In this paper, we construct cost-minimal SRRTs while assuring the minimum number of breaks (MBSRRT). The corresponding integer program can be given as follows:

We employ binary match variables $x_{i,j,p}$ being equal to 1 if and only if team i plays at home against team j at MD p. Then, the objective function (1) represents

$$\min \sum_{i\in T}\sum_{j\in T\setminus\{i\}}\sum_{p\in P} c_{i,j,p}x_{i,j,p} \tag{1}$$

$$\text{s.t.} \sum_{p\in P}(x_{i,j,p}+x_{j,i,p}) = 1 \quad \forall\, i,j\in T, i<j \tag{2}$$

$$\sum_{j\in T\setminus\{i\}}(x_{i,j,p}+x_{j,i,p}) = 1 \quad \forall\, i\in T, p\in P \tag{3}$$

$$\sum_{j\in T\setminus\{i\}}(x_{i,j,p-1}+x_{i,j,p}) - br_{i,p} \leq 1 \quad \forall\, i\in T, p\in P^{\geq 2} \tag{4}$$

$$\sum_{j\in T\setminus\{i\}}(x_{j,i,p-1}+x_{j,i,p}) - br_{i,p} \leq 1 \quad \forall\, i\in T, p\in P^{\geq 2} \tag{5}$$

$$\sum_{i\in T}\sum_{p\in P^{\geq 2}} br_{i,p} \leq n-2 \tag{6}$$

$$x_{i,j,p} \in \{0,1\} \quad \forall\, i,j\in T, i\neq j, p\in P \tag{7}$$

$$br_{i,p} \in \{0,1\} \quad \forall\, i\in T, p\in P^{\geq 2} \tag{8}$$

the goal of cost minimization. Restrictions (2) and (3) form a SRRT by letting each pair of teams meet exactly once and letting each team play exactly once per MD. Restrictions (4) and (5) set the binary break variable $br_{i,p}$ to 1 if team i plays twice at home or twice away at MDs $p-1$ and p. Inequality (6) assures that no more than $n-2$ breaks are arranged.

3 Branching Scheme

Branching must take care of the special structure of the problem at hand, because only few branches are able to assure that the corresponding subtree will have at least one single feasible solution. However, detecting infeasibility in advance is pretty complicated but worth to be done because otherwise a huge computational effort is spent on infeasible subtrees. Consider for example three branches setting variables $x_{i,j,p} = x_{i,j',p+3} = x_{i,j'',p+6} = 1$ for arbitrary pairwise disjoint teams $i,j,j',j'' \in T$ and MD $p \in P^{\leq |P|-6}$. These branches imply that team i has a break at MD $p', p+1 \leq p' \leq p+3$, as well as at MD $p'', p+4 \leq p'' \leq p+6$. However, because each team must not have more than one break in a MBSRRT (see [5]), the corresponding subtree does not permit to construct any feasible solution to the problem at hand.

[5] study the structure of MBSRRTs and proof several characteristics which can be employed in order to avoid searching subtrees having no feasible solution. The basic idea is to branch on break-variables according to rules derived from the four characteristics i to iv provided in [5]. In MBRRTs

i each team can have no more than one break,
ii at each MD either two breaks occur or none,
iii the two teams having no break can be said to have breaks at MD 1, and