Interfaces

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Scheduling the German Basketball League

Stephan Westphal

Interfaces
Vol. 44, No. 5, September–October 2014, pp. 498–508
ISSN 0092-2102 (print) | ISSN 1526-551X (online)

informs®
http://dx.doi.org/10.1287/inte.2014.0764

Scheduling the German Basketball League

Stephan Westphal
Institute for Applied Stochastics and Operations Research, Clausthal University of Technology,
38678 Clausthal-Zellerfeld, Germany, stephan.westphal@tu-clausthal.de

In this paper, we discuss the problem of finding an optimal schedule for the German Basketball League (BBL) for the 2011–2012 season. A major issue that we address is that most of the games take place in multipurpose arenas that are also used for other events and are thus not always available. In addition, we must minimize the number of successive home or away games, assign the most interesting games to television broadcasting slots, minimize the distance teams must drive on a newly established derby day, and meet team requests for home or away games and to play specific teams.

We present several algorithmic approaches and show how these models fit the BBL's requirements. In this process, we prove that the classic models, which the BBL had applied previously and many other leagues still apply, are too limited to meet these requirements. We show that canonical schedules do not have the desired properties. We proved to the BBL that mirrored schedules cannot meet its needs, and thus convinced it to use nonmirrored schedules for the first time in its history. The BBL's requirements are typical of sports leagues; therefore, the approaches presented are also applicable to many other sports leagues. We implemented our approach in scheduling software that we developed for the BBL and applied to its 2011–2012 season scheduling.

Keywords: sports scheduling; basketball; timetabling; tournaments; integer programming.

The German Basketball League (BBL) consists of 18 teams that play against each other in 34 rounds. Each team plays one game per round; over the whole season, each team meets every other team once at its own home venue (home game) and once at the other team's venue (away game). The games are organized such that these two games are played in separate halves of the season, and each team has at least eight and at most nine home games in each season half.

In addition to these fundamental requirements, other considerations must be addressed. The most desirable time slots for a match are on weekends, because the average fan is more likely to attend a sports event on a weekend than in the middle of the week. Thus, more tickets can be sold for a game taking place on a Saturday or Sunday. For this reason, the rounds are distributed over the season such that each of the seasons' 30 weekends corresponds to one round. The games of the remaining four rounds are scheduled on four different Wednesdays in the middle of the season.

Most of the games are played in multipurpose arenas (i.e., arenas that also host other events); therefore, these venues are not always available. Using the previous planning approach, games that were scheduled on a weekend often had to be rescheduled, causing numerous problems. Because the new time slot was frequently in the middle of the week, fewer tickets were sold. When the new time slot resulted in scheduling two or even three home games for a team within a few days, ticket sales were even lower. In addition, by the end of the weekend on which the game had been scheduled originally, all teams had not played the same number of games; thus, comparing their performance during the season was no longer possible. At almost no point in time had all teams played the same number of games. Some teams had played more (up to two) games than others; thus, the table reporting their standings was not meaningful (i.e., the table was unbalanced). These scheduling problems are critical to the BBL; therefore, compliance with these availability restrictions is the fundamental scheduling requirement.

Furthermore, we can assume that the average sports fan does not want to go to a sporting event each weekend; however, that fan might not want to wait too long to see a favorite team playing at home. Moreover, even the most passionate fan might not want to travel to a different venue several weeks in a row. Therefore, designing each team's schedule in such a way that each home game is followed by an away game and each away game is followed by a home game would seem to be a logical rule; violating this rule, which is called a break, should be avoided whenever possible. Thus, our work also considers the classical objective of minimizing the number of breaks.

Another important consideration is utilization by the media. Broadcasting slots are available for 30 rounds. Therefore, the games that promise the highest television viewer audience should be scheduled in these broadcasting slots. In this paper, we denote these games as A-games, and we denote the second most promising games as B-games. BBL schedules include 20 A-games and 60 B-games. Its goal is to show all the A-games on television; however, because 30 broadcasting slots are available for only 20 A-games, 10 slots can be allocated for B-games.

The BBL assumes that fans are unlikely to travel the day after New Year's Day; therefore, it developed the concept of a derby—a single round in which only games between teams that are located close to each other are played. To address this requirement, we try to minimize the total distance between the teams playing against each other in the round played on this newly established derby day.

For several reasons, some specific encounters (i.e., two teams playing against each other) must be scheduled on specific days; we call these requests encounter wishes. Finally, some teams have specific wishes for scheduling home or away games on specific days; for example, a sports site might be under renovation and otherwise unavailable for play, a team might have access to a larger arena but only at specific times, or a team might want to play at home because it expects to sell more tickets to a home game; we term these home-away wishes.

In this paper, we examine algorithmic concepts and their applicability to the BBL's scheduling problem.

Previous Work and Our Results

Several papers have been published about the various techniques used to generate feasible schedules for sports leagues; examples include Della Croce and Oliveri (2006), Flatberg et al. (2009), Froncek (2001), Goossens and Spieksma (2009), Henz (2001), Nemhauser and Trick (1998), and Wright (2006). Kendall et al. (2010) provide a good overview of these techniques.

Griggs and Rosa (1996) examined schedules of the highest division of 25 European soccer leagues. They showed that 16 of these leagues used canonical plans, which we explain in the *Standard Schedule* section. These leagues all use the same schedule; they just change the name of the teams each year. Recently, Goossens and Spieksma (2012) reviewed the scheduling for these 25 competitions and found out that 13 of the leagues still apply canonical plans.

Because canonical plans are still popular, we investigate their applicability to the BBL planning problem. We show how to find optimal canonical plans by using integer linear programming formulations. Our formulation requires only $|T|^2$ binary variables, where T denotes the set of teams, and considers all the constraints mentioned previously. This formulation can be solved optimally in only a few seconds using a noncommercial solver such as GLPK; therefore, it can also give the many minor leagues the opportunity to generate schedules that meet the requirements of more than half of Europe's national football leagues. In addition, we introduce postoptimization steps and discuss their effects. Our calculations show that the best canonical schedules still violate 31 of the 96 availability constraints, and our postoptimization steps can only lower this number to 22. Because 22 violations are too high for the BBL, we then evaluate noncanonical schedules.

Goossens and Spieksma (2012) show that 15 of 25 leagues use mirrored schedules; that is, the schedules for the second half of any season correspond exactly to the first half of that season; only the home-field advantage of the participating teams changes. Of the 10 remaining leagues, five use variants of this approach; 80 percent use symmetric schedules. Other than for historical reasons, they might use symmetric schedules because doing so reduces the planning problem to one in which they must optimize over only half of the season, thus simplifying the problem. The disadvantage of this approach is that it restricts the planner to a small subset of the possible plans. Therefore, we investigate the extent to which the mirrored plans are too restrictive for the BBL. For the BBL scheduling problem, we show that a generating a mirrored plan with less than 13 violated availability constraints is impossible. Therefore, we examine the applicability of nonmirrored schedules to our problem. We formulate our model in such a way that even the noncommercial solver GLPK can solve it in a few seconds. We explain why the most natural integer programming approach does not work, and develop a hybrid method based on all the results discussed in the preceding sections. These nonmirrored schedules meet the BBL's high demands; thus, the league used them in its 2011–2012 season. This algorithm is the core of the planning software that we developed, which the BBL used to schedule its 2012–2013, 2013–2014, and 2014–2015 seasons, and that it plans to use to schedule future seasons. Finally, we compare the performance of the various algorithms in the *Computational Results* section.

Mirrored Schedules

In this section, we look at the strategy that the BBL used previously and propose some algorithms to improve the solutions found using mirrored schedules.

Standard Schedule

To schedule the matches for the first half of the season between n teams, we apply the canonical tournament approach (de Werra 1981). Thus, we ensure that each team plays against every other team exactly once. This initial canonical schedule can be obtained by assigning the teams to the nodes of a special graph; see Figure 1 for $n = 18$.

The matches of the first time slot correspond to the pairs of vertices that are adjacent to each other. A game always takes place at the venue of the team assigned to the head of the corresponding arc. Figure 2 illustrates how to obtain matches for the second day. The schedules for the other time slots are derived analogously. The only difference is that the orientation of the arc incident to team 18 changes each time, thus ensuring that the total number of breaks attains $n-2$, which is the minimum possible value according to de Werra (1981).

We can schedule the second half of the tournament based on the first half by mirroring; that is, we repeat the matches of the first half, but change the home-field advantage. This results in a total number of breaks

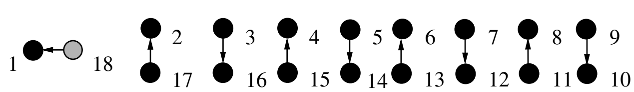

Figure 1: The graph displays the matches for the first time slot of the canonical schedule for 18 teams (i.e., $n = 18$).

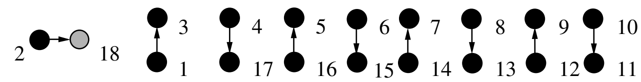

Figure 2: The matches for the second time slot of the canonical schedule can be derived by changing the assignment of the teams to the black vertices in counterclockwise direction.

of $3 \cdot (n-2)$, because each team with a break in the first half of the season will encounter a corresponding break in the second half and one break between each half of the season.

Observe that the entire schedule depends on the assignment of the teams to the nodes in the graph shown in Figure 1 in the first round. Each node has a specific home-away pattern. In the aforementioned example, team 3 must play AHHAHAHAHAHAHAHAH in the first half and HAAHAHAHAHAHAHAHA in the second half; each A and each H stands for an away and home game, respectively, for team 3. Therefore, we can easily calculate the number of violated availability restrictions and home-away wishes for this example in which team 3 is being assigned to that pattern. We can then find the optimal canonical schedule with respect to availability restrictions and home-away wishes in polynomial time by solving a minimum-cost flow problem. For this reason, consider the complete bipartite graph consisting of nodes that represent the teams on one side and nodes that represent the nodes in Figure 1 on the other side. Because each perfect matching in this graph corresponds to some feasible schedule and vice versa, we can determine the canonical schedule with the minimum number of violated availability restrictions by finding a minimum-weight perfect matching in this graph.

We can easily see that a canonical schedule that fulfills half of the availability restrictions always exists. Consider an arbitrary canonical schedule. If this schedule does not already obey at least half of the given availability restrictions, change the home-field advantage for each match. This way, we obtain another canonical schedule, which fulfills exactly those restrictions that have been violated previously. Furthermore, this new schedule cannot violate more than half of the given availability restrictions. Observe that the total number of breaks does not change. Each home break becomes an away break and vice versa.

Although the availability restrictions are the most critical, we must also consider all other constraints and wishes. Therefore, we do not apply network flow algorithms to generate optimal canonical schedules with respect to the number of violated availability restrictions; instead, we develop an integer programming model that also addresses the other constraints (with the exception of the derbies). The *Standard Schedule* section in the appendix describes this model in detail.

The BBL's 2011–2012 season has 96 availability restrictions. We know that a canonical schedule exists that fulfills at least 48 of these restrictions. Because this is only a lower bound, we would hope for a much better fulfillment. However, by applying the model discussed previously, we find that the best canonical schedule satisfies only 65 availability constraints, leaving 31 unsatisfied. Because 31 violations are too high for the BBL, we conclude that applying canonical schedules cannot meet the BBL's requirements.

Enhanced Standard Schedule

The BBL always used canonical schedules. In the previous section, we showed how to calculate the best canonical schedule; however, we also showed that this schedule does not meet the BBL's requirements. Therefore, we try to improve the optimal canonical schedules by applying postoptimization steps.

Step 1 (Postoptimization by best scheduling of encounters). First, we fix the HA patterns of the teams based on the optimal canonical schedule and then look for the best schedule that obeys these patterns. We can view this as a first-break-then-schedule approach (Rasmussen and Trick 2008); the only difference is that finding the best breaks also ensures that an underlying best canonical schedule already exists. Such an approach has this advantage: in the first step, we have already decided which teams will play at home and which will play away on each day. Thus, we do not have $n \cdot (n-1)$ possible encounters, but only $n/2 \cdot n/2 = n^2/4$ possible games each day. This reduces the number of variables to approximately one-fourth. Additionally, because the HA patterns of some teams are so similar that only a few rounds exist in which they do not play at home or away at the same time, these teams can meet only in specific rounds. This second effect leads to a further reduction of variables, such that even noncommercial integer programming solvers (e.g., GLPK) can solve the resulting integer program in a few seconds. The *Enhanced Standard Schedule* section in the appendix provides a detailed description of this model.

Step 2 (Postoptimization by best HA assignment). Whereas the preceding step mainly addresses derby day and encounter wishes, we apply another postoptimization step that finds the best HA assignment based on the schedule computed in the preceding step. The appendix provides details.

Applying Step 2 improves the results of Step 1 by violating only 22 availability constraints, compared to 31 for canonical schedules; however, the number of breaks increases to 120 breaks. In contrast, the previous results generated only 48 breaks—the minimum possible number of breaks for mirrored round robin tournaments for 18 teams. Therefore, our efforts bring us closer to the desired plan, but not close enough.

Optimal Assignment of Venues

The best canonical schedule was not good enough, and we could not improve it to the desired extent; therefore, we want to determine whether any mirrored schedule fits the BBL's requirements.

Finding a mirrored schedule that meets all BBL requirements under all possible constraints is clearly not possible. For example, suppose a concert is scheduled on the opening day at the O2 arena in Berlin; this would force the ALBA Berlin basketball team to play an away game on this day. If we restrict ourselves to mirrored schedules, this results in a home game in the first round of the second half of the season. If the O2 arena is already blocked on this day, at least one error would result. Even if such a direct conflict does not exist, we cannot prevent a certain number of violations if we restrict ourselves to mirrored schedules.

In this section, we therefore discuss whether the use of multipurpose arenas and the associated high prioritization of the availability constraints prohibit our consideration of mirrored plans in principle.

To find a plan with the minimum number of violated availability constraints, we set up another integer linear program, denoted in the appendix as the Venue-Based Model. Again, the solution of this model does not determine the venues of the games. It only determines which teams play against each other. Thus, we can only partially consider the encounter wishes. In this approach, we also do not consider the number of breaks. By performing the postoptimization step described previously (see Step 2), we then determine the best HA assignment based on the previously scheduled encounters.

Using this approach, we can find a schedule that violates only 13 of the given availability constraints. More importantly, this also means that finding a mirrored schedule that violates fewer than 13 of these constraints is impossible. Hence, a restriction to mirrored plans means that at least 13 games must be postponed. These games can then only be scheduled in the middle of the week, which usually results in significantly fewer tickets sold. Moreover, the basketball authorities must address the issue of unbalanced tables, as we mention previously. Hence, we conclude that no mirrored schedule can meet all BBL requirements.

Nonmirrored Schedules

The standard approach discussed previously leaves 31 of 93 availability constraints unsatisfied and, as we proved, no mirrored plan exists that violates fewer than 13 of these constraints; therefore, we examine the applicability of nonmirrored plans.

Standard Integer Program Approach

The most natural approach is to solve the integer program introduced for the development of the enhanced standard schedule without the mirroring constraint—Equation (39) in the appendix—and with y, which represents home or away games specified as a variable and not as a given constant. For the BBL, we attempt to solve the problem by applying the Gurobi integer programming solver; however, after 12 hours of computation, we stop the solver after seeing no further improvement in the solution after the first 93 minutes. This solution does not violate any availability constraints; but, it includes 146 breaks, which means that a team's schedule would include an average of more than eight breaks throughout the season. In comparison, in the plans of the previous few seasons, teams averaged only three breaks. This approach was therefore not able to deliver the solutions we needed either.

If the BBL was willing to accept the 146 breaks, we would still not consider this approach to solving the problem. The BBL approached us to develop a software tool with which it can schedule its league on its own; thus, our focus is not only to provide one solution for the current season, but also to develop an algorithm that it can use to determine plans for the ensuing seasons in a reasonable amount of time. Consequently, we do not want to use an integer program that takes 93 minutes to obtain a solution for a particular instance. The data for the next seasons will surely be different, and we cannot guarantee that the integer program will then find a reasonable solution as well. Therefore, our objective is to design a multistage hybrid algorithm that combines the speed of the concepts discussed in the preceding sections and the quality of the solutions one can hope for by applying integer programming.

Hybrid Approach

Using an integer programming approach seems to be essentially correct; however, the running time and solution quality must be improved significantly. We discuss our approach in this section.

Simplification of the integer program: We simplify the integer program discussed in the previous section by using the knowledge that the number of home breaks always equals the number of away breaks. Suppose this is not the case; that is, a round with more home breaks than away breaks exists. In such a situation, more teams would play home games than away games; however, this is impossible. Thus, it suffices to only consider the number of home breaks.

Providing initial solutions to the integer programming solver: By providing an integer programming solver with good initial solutions, we can significantly accelerate the branch-and-bound process, because the solver can more rapidly prune subtrees; therefore, we employ the solutions calculated in the previous sections.

Decomposition of the problem: Although the implementation of the previous steps was helpful, we achieved a decisive breakthrough with the approach discussed next.

As we have seen in the discussion of mirrored schedules, finding single round robin tournaments is relatively easy. Hence, generating the schedules for the first half of the season and then generating the schedules for the second half seems reasonable; however, in generating the schedule for the second half, a problem arises: as a result of fixing the schedules for the first half, the home-field advantage for every game in the second half is predetermined. Thus, under certain circumstances, encounter wishes and television time-slot assignments for the second half might not be able to be satisfied. Our solution is to plan the two season halves consecutively, but not independently. First, we solve the integer program; however, we allow teams to play more than one game per round in the second half of the season—or possibly no match at all. The solver will still try to achieve a good distribution of the games. It will not schedule all the A-games in the first half, and it will try to fulfill the encounter wishes as much as possible. So, although we do not completely neglect the second half of the season in this first run, we drastically simplify the combinatorics in the second half, because the problem of scheduling the second half is now just a question of distributing the games over the days. Without this simplification, scheduling the second half of the season involves finding a single round robin tournament suitable for the objective function. This task now basically becomes that of finding a matching in a bipartite graph with the games on one side and the time slots for the rounds on the other. Hence, we remove the hard combinatorics from the second half, such that the calculation of the first step is almost as easy as finding a single round robin tournament that matches the requirements of scheduling for the first half.

In the second stage of our approach, we fix all the games that have been calculated for the first half of the season and reactivate the constraints, taking care that each team plays exactly one match a day. Now, the optimal solution for this integer program can generate a feasible plan for the entire season. Finally, we again apply the postoptimization step described earlier (see Postoptimization by best HA assignment).

In less than 18 minutes, the integer program solver, Gurobi, performs all these steps and all integer programs are solved to optimality. The plan generated has 74 breaks and does not violate any availability constraints.

Computational Results

As we mention previously, we use data provided by the BBL to test all the models discussed. Our resulting schedule has 20 A-games, 60 B-games, 30 television broadcasting slots, 11 wishes for home games, no wishes for away games, and 20 encounter wishes.

To perform all computations, we use a standard 2.4 GHz desktop computer with two cores and 4 GB RAM. To solve the models presented in the *Mirrored Schedules* section, we apply the noncommercial integer programming-linear programming solver, GLPK v4.46. Because GLPK is not able to handle the approaches previously discussed to generate nonmirrored schedules, we use the commercial solver Gurobi v4.0.0 for that purpose.

Table 1 shows the results. We report the number of violated availabilities (Viol. avail.), the total number of breaks (Breaks), the number of violated wishes for home and away games (Viol. HA), the number of television broadcasting slots on which no A-game takes place (first entry of TV-comp), the number of broadcasting slots on which neither an A-game nor a B-game takes place (second entry of TV-comp), the number of days to which an A-game is assigned although the day does not have a broadcasting slot (third entry of TV-comp), and the number of wishes for specific encounters that

Approach	Viol. avail.	Breaks	Viol. HA	TV-comp	Viol. enc.	Derby	Time (s)
Mirrored:							
—Canonical	31	48	3	(13/0/1)	2	3,462 km	14.9
+Encounter optimization	31	48	3	(15/0/1)	3	2,288 km	30.3
+Break minimization	22	120	4	(15/0/1)	3	2,288 km	75.5
—Optimized venues	13	186	2	(11/0/1)	2	1,150 km	126.9
Nonmirrored:							
—Standard integer program	0	146	—	(-/-/1)	—	1,653 km	5,613
—Hybrid	0	74	1	(10/0/0)	0	1,097 km	1,055
Lower bounds	0	64	0	(10/0/0)	0	1,097 km	—

Table 1: Computational experiments show that only nonmirrored schedules can satisfy all the home-availability constraints of the 2011–2012 season.

are not satisfied (Viol. enc.). We do not measure the violated wishes for the nonmirrored schedule. Finally, the last two columns show the total distance between the teams on the derby day (Derby) and the computation time (Time).

Because the computation of a minimum-weight perfect matching shows that there is a lower bound of 1,097 km for the distance driven on derby day, we see that the solutions obtained by the hybrid approach are optimal with respect to this criterion. Furthermore, because we have 30 broadcasting slots, but only 20 games that are considered to be A-games, 10 is clearly a lower bound for the number of days with broadcasting slots for which no A-game is available. Therefore, the plan generated by the hybrid approach is also optimal with respect to this criterion. So, the solution generated by the hybrid approach is optimal with respect to the number of violated home-availability, television-compatibility, encounter-wish, and derby-day constraints. The only question relates to the optimality of the 74 breaks and the one violated home wish. Looking at the generated schedule, we see that the home-game wish could have been satisfied easily at the cost of an additional two breaks; however, because breaks have a higher value, our solution is preferable. We still must address the question of the number of breaks a schedule must have, given that it fulfills all the availability constraints. Therefore, we formulate an integer program to find a nonmirrored schedule that minimizes the number of breaks, subject to the constraint that no availability restrictions are violated. As we previously discuss, these models are not easily solvable to optimality. Hence, we calculate only the linear relaxation of this model, which has a value of 62.36 and thus provides us with a good lower bound. Because the number of breaks is always even, we can conclude that no nonmirrored schedule obeys all availability restrictions and has fewer than 64 breaks. Of course, this does not mean that a schedule with 64 breaks exists; in addition, if such a schedule exists, we cannot know if it fulfills the other objectives. Thus, although our solution might not be optimal, it is at least a very good approximation.

Impact

The publication of the new plans received wide media coverage. For example, the German Press Agency, DPA, published an article via its ticker whose title we can translate as "Thanks to new software: BBL revolutionizes its schedule" (Bossaller 2011). Several German newspapers reprinted this article, thus making it available on almost every German news website. The BBL managing director, Jan Pommer, was quoted as saying that the BBL now has the most fan-friendly schedule in its history, and that the clubs would now get what they wanted. He concluded that better schedules result in more spectators and consequently, more money earned (Bossaller 2011). ALBA Berlin team manager Mithat Demirel agreed with him, saying that this schedule will help his team, because it can now do much better planning—even for international competitions. He also pointed out that it helps enormously that very few games will have to be postponed in the future (Bossaller 2011).

The head of sports at the BBL, Jens Staudenmeyer, added that in the past many clubs perceived the postponements of games as unjust and as a threat to the equality of competition. The new schedule created an equality that the clubs responded to positively (Bossaller 2011).

One of these clubs is the LTi Giessen 46ers. According to its CEO, Christoph Syring, only a few fans can afford to attend four home games in a month. He assumes that the club has lost several thousand euros because of this issue and that the previous schedule has resulted in fewer viewers. He complimented the league on implementing the new schedule and acknowledged that it has taken a big leap forward. Considering the historic change that the schedules are no longer mirrored, Syring said that, in his opinion, whether there are six weeks or six months between two games does not matter (Gießener Anzeiger 2011).

Conclusion and Outlook

In this paper, we considered the problem of finding optimal double round robin tournaments for sports leagues. We developed several models for the BBL and showed that many classic approaches, such as applying canonical and mirrored schedules, no longer meet all the requirements of professional sports leagues. Today's sporting events are increasingly being held in multipurpose arenas; therefore, models for these events must consider this factor. The algorithm we developed computes near-optimal plans for the BBL, and we believe it can also solve other leagues' problems.

Acknowledgments
We thank Morten Tiedemann for his work on the graphical user interface for the planning software that we developed for the BBL.

Appendix

Notation

The set of teams is denoted as $T=\{1,\dots,n\}$, and the set of rounds is denoted as $D=\{1,\dots,2n-2\}$. The season consists of two halves, $D_1=\{1,\dots,n-1\}$ and $D_2=\{n,\dots,2n-2\}$. We denote the set of availability restrictions as $F\subseteq T\times D$ with $(i,d)\in F$ if and only if team i cannot play at home in round d. The rounds for which broadcasting slots are available are called $D^{\mathrm{TV}}\subseteq D$. The most interesting games are called A-games, which we denote as A; and the second most interesting games are called B-games, which we denote as B. The single round, in which only games between teams that are located close to each other (i.e., derbies) should be scheduled, is denoted as $\hat{d}$. The encounter wishes (EW) are defined as $\mathrm{EW}\subseteq T\times T\times D$ with $(i,j,d)\in\mathrm{EW}$ if and only if team i should play at home against team j in round d. The set of home wishes is called $\mathrm{HW}\subseteq T\times D$ with $(i,d)\in\mathrm{HW}$ if and only if team i wants to play at home in round d. The set of away wishes AW is defined analogously.

Standard Schedule

All canonical schedules can be generated by assigning teams to nodes in a graph (see Figures 1 and 2). To find the best such assignment and thus the best canonical schedule, we set up the following integer linear program. For each team $i\in T$, node $v\in V$, and round $d\in D$, we define the following binary variables.

$$x_{i,v}=\begin{cases}1, & \text{if team } i \text{ is assigned to node } v \text{ in the first round;}\\ 0, & \text{otherwise.}\end{cases}\tag{1}$$

$$y_{i,d}=\begin{cases}1, & \text{if team } i \text{ plays at home in round } d;\\ 0, & \text{otherwise.}\end{cases}\tag{2}$$

$$z_{i,j,d}=\begin{cases}1, & \text{if encounter wish } (i,j,d)\in\mathrm{EW} \text{ is violated;}\\ 0, & \text{otherwise.}\end{cases}\tag{3}$$

$$t^a_{i,d}=\begin{cases}1, & \text{if } i \text{ plays an A-game at home in round } d;\\ 0, & \text{otherwise.}\end{cases}\tag{4}$$

$$t^b_{i,d}=\begin{cases}1, & \text{if } i \text{ plays a B-game at home in round } d;\\ 0, & \text{otherwise.}\end{cases}\tag{5}$$

$$p^a_d=\begin{cases}1, & \text{if there is no A-game in round } d\in D^{\mathrm{TV}};\\ 0, & \text{otherwise.}\end{cases}\tag{6}$$

$$p^b_d=\begin{cases}1, & \text{if there is neither an A-game nor a B-game in round } d\in D^{\mathrm{TV}};\\ 0, & \text{otherwise.}\end{cases}\tag{7}$$

Additionally, the variable p^o_d measures the number of A-games for every round $d\in D\backslash D^{\mathrm{TV}}$. We have seen that the assignment of teams to the nodes of the underlying graph in the first round determines the entire schedule for both halves. The values of $x_{i,v}$ and $y_{i,d}$ and the algorithm for rotating teams clockwise around the graph determine the mirrored schedule. To express this relationship, we define a function $g\colon V\times D\to V$ with $g(v,d)=w$ if and only if the teams that are assigned to the nodes v and w in the first round are to play against each other in round d. Each node v has a specific home-away pattern (e.g., AHHHAHAHAHAHAHAH). To address this information, we define the matrix h with $h_{v,d}=1$ if node v has a home-away pattern with a home game in round d; otherwise, $h_{v,d}=0$.

We can then find the optimal canonical schedule by solving the following integer program:

$$\min\Bigg\{\sum_{i\in T,\,v\in V}100\cdot c_{i,v}\cdot x_{i,v}\tag{8}$$

$$+\sum_{(i,d)\in\mathrm{AW}}\alpha_{(i,d)}\cdot y_{i,d}\tag{9}$$

$$+\sum_{(i,d)\in\mathrm{HW}}\beta_{(i,d)}\cdot(1-y_{i,d})\tag{10}$$

$$+\sum_{(i,j,d)\in\mathrm{EW}}\gamma_{(i,j,d)}\cdot z_{i,j,d}\tag{11}$$

$$+\sum_{d\in D}0.5\cdot(p^o_d+p^a_d+p^b_d)\Bigg\}\tag{12}$$

$$\text{s.t.}\ \sum_{v\in V}x_{i,v}=1\quad\forall\, i\in T,\tag{13}$$

$$\sum_{i\in T}x_{i,v}=1\quad\forall\, v\in V,\tag{14}$$

$$\sum_{v\in V}h_{v,d}\cdot x_{i,v}=y_{i,d}\quad\forall\, i\in T,\ d\in D,\tag{15}$$

$$\sum_{v\in V:\,h_{v,d}=0}x_{i,v}\le z_{i,j,d}\quad\forall\,(i,j,d)\in\mathrm{EW},\tag{16}$$

$$x_{i,v}+\sum_{j\in T,\,w\in V:\,w\neq g(v,d)}x_{j,w}\le 1+z_{i,j,d}\quad\forall\,(i,j,d)\in\mathrm{EW},\ v\in V\colon h_{v,d}=1,\tag{17}$$

$$x_{i,v}+\sum_{(i,j)\in A}x_{j,g(v,d)}\le 1+t^a_{i,d}\quad\forall\, i\in T,\ d\in D,\ v\in V\colon h_{v,d}=1,\tag{18}$$

$$\sum_{i\in T,\,d\in D}t^a_{i,d}=|A|,\tag{19}$$

$$x_{i,v} + \sum_{(i,j)\in B} x_{j,g(v,d)} \le 1 + t^a_{i,d} \quad \forall i \in T,\ d \in D,\ v \in V\colon h_{v,d} = 1, \tag{20}$$

$$\sum_{i\in T,\, d\in D} t^a_{i,d} = |B|, \tag{21}$$

$$\sum_{i\in T} t^a_{i,d} \le p^o_d \quad \forall d \in D\backslash D^{\mathrm{TV}}, \tag{22}$$

$$\sum_{i\in T} t^a_{i,d} \ge 1 - p^a_d \quad \forall d \in D^{\mathrm{TV}}, \tag{23}$$

$$\sum_{i\in T}\left(t^a_{i,d} + t^b_{i,d}\right) \ge 1 - p^b_d \quad \forall d \in D^{\mathrm{TV}}, \tag{24}$$

$$x_{i,v} \in \{0,1\} \quad \forall i \in T,\ v \in V, \tag{25}$$

$$y_{i,d}, z_{i,j,d}, t^a_{i,d}, t^b_{i,d}, p^a_d, p^b_d, p^o_d \ge 0 \quad \forall i \in T,\ j \in T,\ d \in D. \tag{26}$$

Equations (13) and (14) ensure that each team is assigned to exactly one node and each node is assigned to exactly one team, whereas Equation (15) couple (i.e., synchronize) x and y.

Clearly, team i cannot play a home game against team j in round d if i is assigned to a node v implying an away game for i in that round (16). If i is assigned to a node v with a home game in round d, but j is not assigned to the node $g(v,d)$ corresponding to v's opponent in round d, i cannot be scheduled to play that home game against team j in round d either. Thus, Equations (16) and (17) together with Equation (11) ensure that the violation of encounter wishes is penalized appropriately. In a similar way, Equations (18)–(21) ensure that $t^a_{i,d}$ and $t^b_{i,d}$ attain the correct values, such that the distribution of the most attractive games over the season can be controlled in Equations (22)–(24) and Equation (12). Finally, the violation of home- and away-game requests, HW and AW, are implemented in Equations (9) and (10). Here, we introduce coefficients $\alpha_{(i,d)}, \beta_{(i,d)}, \gamma_{(i,j,d)}$ to reflect the fact that the elements of AW, HW, and EW might not be equally important.

We have chosen the coefficients in the objective function in such a way that minimizing the number of violated availability constraints is the priority. We chose 0.5 as coefficient for Equation (12) because this value worked best for the instance we are evaluating. Furthermore, it is sufficient to require $x_{i,v}$ to be integral. Hence, we relaxed the integrality constraints for all the other variables.

Enhanced Standard Schedule

Let S be the best canonical schedule obtained by applying the algorithm described in *Standard Schedule*, and let $y_{i,d}$ indicate whether team i is playing at home based on that schedule.

In the first postoptimization step, we solve the following mixed-integer linear program. The definition of the variables is basically the same as for the previous program. The only differences are that we now define x such that $x_{i,j,d} = 1$ if and only if team i plays at home against team j in round d and y is no longer a variable, but is fixed according to the previously generated plan S. Furthermore, we define a new variable $w_{i,d}$ that indicates whether team i is having a break in round d. For each team i, we also introduce a new variable z_i that helps to balance the distance single teams must drive on the derby day.

$$\min \Bigg\{ \sum_{(i,d)\in F,\, j\in T,\, d\in D} 100 \cdot x_{i,j,d} \tag{27}$$

$$+ \sum_{(i,d)\in \mathrm{AW},\, j\in T} \alpha_{(i,d)} \cdot x_{i,j,d} \tag{28}$$

$$+ \sum_{(i,d)\in \mathrm{HW},\, j\in T} \beta_{(i,d)} \cdot x_{j,i,d} \tag{29}$$

$$+ \sum_{(i,j,d)\in \mathrm{EW},\, j\in T} \gamma_{(i,j,d)} \cdot (1 - x_{i,j,d}) \tag{30}$$

$$+ \sum_{i\in T,\, d\in D} 10 \cdot w_{i,d} \tag{31}$$

$$+ \sum_{d\in D} 0.5 \cdot (p^o_d + p^a_d + p^b_d) \tag{32}$$

$$+ \sum_{i\in T} 0.05 \cdot (\mathrm{dist}(i,j) x_{i,j,\hat{d}} + z_i) \Bigg\} \tag{33}$$

$$\text{s.t.} \quad \sum_{j\in T} (x_{i,j,d} + x_{j,i,d}) = 1 \quad \forall i \in T,\ d \in D, \tag{34}$$

$$\sum_{d\in D} x_{i,i,d} = 0 \quad \forall i \in T, \tag{35}$$

$$\sum_{d\in D} x_{i,j,d} = 1 \quad \forall i, j \in T, \tag{36}$$

$$\sum_{d\in D_1} (x_{i,j,d} + x_{j,i,d}) = 1 \quad \forall i, j \in T, \tag{37}$$

$$\sum_{d\in D_2} (x_{i,j,d} + x_{j,i,d}) = 1 \quad \forall i, j \in T, \tag{38}$$

$$x_{i,j,d} = x_{j,i,d+n-1} \quad \forall i, j \in T,\ d \in D_1, \tag{39}$$

$$\sum_{j\in T} x_{i,j,d} = y_{i,d} \quad \forall i \in T,\ d \in D, \tag{40}$$

$$\sum_{j\in T} \mathrm{dist}(i,j) \cdot x_{i,j,\hat{d}} \le 200 + z_i \quad \forall i \in T, \tag{41}$$

$$\sum_{j\in T} (x_{i,j,d} + x_{i,j,d+1}) \le 1 + w_{i,d} \quad \forall i \in T,\ d \in D, \tag{42}$$

$$\sum_{j\in T} (x_{j,i,d} + x_{j,i,d+1}) \le 1 + w_{i,d} \quad \forall i \in T,\ d \in D, \tag{43}$$

$$\sum_{(i,j)\in A} x_{i,j,d} \le p^o_d \quad \forall d \in D\backslash D^{\mathrm{TV}}, \tag{44}$$

$$\sum_{(i,j)\in A} x_{i,j,d} \ge 1 - p^a_d \quad \forall d \in D^{\mathrm{TV}}, \tag{45}$$

$$\sum_{(i,j)\in A\cup B} x_{i,j,d} \ge 1 - p^b_d \quad \forall d \in D^{\mathrm{TV}}, \tag{46}$$

$$x_{i,j,d} \in \{0,1\} \quad \forall i \in T,\ j \in T,\ d \in D, \tag{47}$$

$$w_{i,d}, z_i, p_d^a, p_d^b, p_d^o \geq 0 \quad \forall i \in T,\ d \in D. \tag{48}$$

Equations (34)–(38) address the fundamental requirements of a round robin tournament by ensuring that each team plays exactly one game in each round, every match is played exactly once, and no two teams meet each other twice in the same half of the season. Equation (39) restricts the search to mirrored schedules and Equation (40) links the given vector y corresponding to the HA patterns of the already generated schedule S to the variables x, which are to be computed here.

To obtain a solution that ensures short drives on the derby day, we could put the sum of the distances driven on that day into the objective function; however, a solution in which one team drives 50 km and another team drives 450 km would be understood to be as good as a solution in which both teams play at a site 250 km away from their home towns. So, with dist(i, j) expressing the distance between the sites of the teams i and j for every $i, j \in T$, we use Equation (41) to measure whether a team has to travel more than 200 km. In this way, every kilometer beyond the 200 km threshold is penalized twice in Equation (33), thus helping to balance the total distance that the single teams drive. This did not ultimately affect the BBL solution. The encounters of the derby day formed a minimum-weight perfect matching; only one team had to drive more than 200 km (the closest possibility for this team).

Equations (42) and (43) ensure that $w_{i,d}$ indicates whether team i has a break in round d. Finally, Equations (44)–(46) and Equation (32) ensure that the attractive games are well distributed over the given broadcasting slots.

The *Enhanced Standard Schedule* describes two postoptimization steps. The model discussed previously can be used to perform the first postoptimization step that we describe in *Postoptimization by best scheduling of encounters*. Let $S' \subset T \times T \times D$ be the games that have been computed this way.

To perform the second step discussed in *Postoptimization by best HA assignment*, we reassign the home-away assignment by again solving this model; we change the model by skipping Equation (40) and adding a new equation that ensures the resulting plan S'' differs from S' only in the home-away assignment:

$$x_{i,j,d} + x_{j,i,d} = 1 \quad \forall (i,j,d) \in S'. \tag{49}$$

This gives us then the result in the second postoptimization step.

The Venue-Based Model

We set up an integer program that minimizes the number of violations of availability constraints. The other requirements will then be optimized in a subsequent postprocessing step. The notation differs only slightly from the last model. Here, $x_{i,j,d} = 1$ if and only if team i plays against team j in round d. The difference is that $x_{i,j,d}$ does not contain any information about the site at which the games will actually take place. As a consequence, $x_{i,j,d}$ equals $x_{j,i,d}$ and several equalities must be adapted to this new approach.

In a mirrored plan, a game in which teams i and j play each other in round d of the first half requires that these teams play each other in round $d+n-1$ in the second half. We denote the set of availability restrictions as $F \subseteq T \times D$ with $(i,d) \in F$ if and only if team i cannot play at home in round d. Let $c(i,j,d)$ be the minimum number of availability restrictions violated as a result of such an assignment. Table A.1 shows the values for $c(i,j,d)$ for all possible availability configurations.

$$\min \Bigg\{ \sum_{i,j \in T,\, d \in D} 50 \cdot c(i,j,d) \cdot x_{i,j,d} \tag{50}$$

$$+ \sum_{(i,j,d) \in \mathrm{EW},\, j \in T} \gamma_{(i,j,d)} \cdot (1 - x_{i,j,d}) \tag{51}$$

$$+ \sum_{d \in D} 0.5 \cdot (p_d^o + p_d^a + p_d^b) \tag{52}$$

$$+ \sum_{i \in T} 0.05 \cdot (\mathrm{dist}(i,j) x_{i,j,\hat{d}} + z_i) \Bigg\} \tag{53}$$

$$\text{s.t.} \ \sum_{j \in T} x_{i,j,d} = 1 \quad \forall i \in T,\ d \in D_1, \tag{54}$$

$$x_{i,i,d} = 0 \quad \forall i \in T,\ d \in D, \tag{55}$$

$$x_{i,j,d} - x_{j,i,d} = 0 \quad \forall i,j \in T,\ d \in D_1, \tag{56}$$

$$x_{i,j,d} - x_{j,i,d+n-1} = 0 \quad \forall i,j \in T,\ d \in D_1, \tag{57}$$

$$\sum_{d \in D_1} x_{i,j,d} = 1 \quad \forall i,j \in T, \tag{58}$$

$$\sum_{j \in T} \mathrm{dist}(i,j) \cdot x_{i,j,\hat{d}} \leq 200 + z_i \quad \forall i \in T, \tag{59}$$

	$(i,d) \in F$ $(i,d+n-1) \in F$	$(i,d) \notin F$ $(i,d+n-1) \in F$	$(i,d) \in F$ $(i,d+n-1) \notin F$	$(i,d) \notin F$ $(i,d+n-1) \notin F$
$(j,d) \in F, (j,d+n-1) \in F$	2	1	1	1
$(j,d) \notin F, (j,d+n-1) \in F$	1	1	0	0
$(j,d) \in F, (j,d+n-1) \notin F$	1	0	1	0
$(j,d) \notin F, (j,d+n-1) \notin F$	1	0	0	0

Table A.1: This table shows the minimum number $c(i,j,d)$ of violations of availability restrictions when scheduling the encounters $i-j$ in rounds d and $d+n-1$.

$$\sum_{(i,j)\in A} x_{i,j,d} \le p_d^o \quad \forall\, d \in D\backslash D^{\mathrm{TV}}, \tag{60}$$

$$\sum_{(i,j)\in A} x_{i,j,d} \ge 1 - p_d^a \quad \forall\, d \in D^{\mathrm{TV}}, \tag{61}$$

$$\sum_{(i,j)\in A\cup B} x_{i,j,d} \ge 1 - p_d^b \quad \forall\, d \in D^{\mathrm{TV}}, \tag{62}$$

$$x_{i,j,d} \in \{0,1\} \quad \forall\, i \in T,\ j \in T,\ d \in D, \tag{63}$$

$$z_i, p_d^a, p_d^b, p_d^o \ge 0 \quad \forall\, i \in T,\ d \in D\backslash D^{\mathrm{TV}}. \tag{64}$$

References

Bossaller M (2011) Dank neuer Software: BBL revolutioniert Spielplan. *German Press Agency* (May 6).

de Werra D (1981) Scheduling in sports. Hanson P, ed. *Studies on Graphs and Discrete Programming*, *Annals of Discrete Mathematics* (North-Holland Publishing Company, Amsterdam), 381–395.

Della Croce F, Oliveri D (2006) Scheduling the Italian football league: An ILP-based approach. *Comput. Oper. Res.* 33(7):1963–1974.

Flatberg T, Nilssen E, Stølevik M (2009) Scheduling the topmost football leagues of Norway. Accessed January 4, 2013, http://folk.uio.no/trulsf/pub/euro2009.pdf.

Froncek D (2001) Scheduling the Czech National Basketball League. *Congressus Numerantium* 153:5–24.

Gießener Anzeiger (2011) Das Ende des Spielplanchaos. *Gießener Anzeiger* (May 19).

Goossens D, Spieksma F (2009) Scheduling the Belgian soccer league. *Interfaces* 39(2):109–118.

Goossens D, Spieksma F (2012) Soccer schedules in Europe: An overview. *J. Scheduling* 15(5):641–651.

Griggs T, Rosa A (1996) A tour of European soccer schedules or testing the popularity of GK_{2n}. *Bull. ICA* 18:65–68.

Henz M (2001) Scheduling a major college basketball conference—Revisited. *Oper. Res.* 49(1):163–168.

Kendall G, Knust S, Ribeiro CC, Urrutia S (2010) Scheduling in sports: An annotated bibliography. *Comput. Oper. Res.* 37(1):1–19.

Nemhauser GL, Trick MA (1998) Scheduling a major college basketball conference. *Oper. Res.* 46(1):1–8.

Rasmussen R, Trick M (2008) Round robin scheduling—A survey. *Eur. J. Oper. Res.* 188(3):617–636.

Wright MB (2006) Scheduling fixtures for Basketball New Zealand. *Comput. Oper. Res.* 33(7):1875–1893.

Stephan Westphal is a professor of discrete optimization at the Clausthal University of Technology, Germany. He received his PhD in mathematics from the University of Kaiserslautern, Germany in 2007. His research interests include combinatorial optimization, mixed integer programming, online optimization, sports scheduling, vehicle routing, scheduling, and algorithmic game theory. He has published several papers in major international journals and conferences. He has served as a scientific consultant for ILOG, a leading provider of optimization software, and developed numerous mathematical models, which are used by several big German companies engaged in, for example, steel production, automotive, and energy. Furthermore, he developed the mathematical approach and the corresponding software, which is used by the German Soccer League (DFL) to compute the best possible schedule for the German Soccer League.