Fairness and Flexibility in Sport Scheduling

.

FAIRNESS AND FLEXIBILITY IN SPORT SCHEDULING

Roel Lambers

COLOPHON

ISBN: 978-90-386-5599-4

November 2022.

The cover page is a design by Amber Rouw. The composition of several photos together construct an image that seem to show a football field or tennis court.

This document was typeset using `classicthesis` developed by André Miede and `arsclassica` by Lorenzo Pantieri. Most of the graphics in this thesis are generated using `pgf/tikz`. The bibliography is typeset using `biblatex`.

.

Fairness and Flexibility in Sport Scheduling

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische Universiteit Eindhoven, op gezag van de rector magnificus prof.dr.ir. F.P.T. Baaijens, voor een commissie aangewezen door het College voor Promoties, in het openbaar te verdedigen op vrijdag 11 november 2022 om 13:30 uur

door

Roel Lambers

geboren te Groningen

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

Voorzitter:	Prof.dr. S. Borst
Promotor:	Prof.dr. F.C.R. Spieksma
Copromotor:	Dr. R.A. Pendavingh
Promotiecommissieleden:	Prof.dr. F.C.R. Spieksma
	Dr. R.A. Pendavingh
	Prof.dr. S. Knust (Universität Osnabrück)
	Prof.dr. M.T. de Berg
	Prof.dr. M.A. Trick (Carnegie Mellon University)
	Prof.dr. D. Goossens (Universiteit Gent)
	Prof.dr. M. Guajardo (Norwegian School of Economics)

Het onderzoek of ontwerp dat in dit thesis wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

ACKNOWLEDGEMENTS

I like to start by thanking everyone who feels like he/she contributed in a positive way to my (working) life in the years leading to this thesis.

Dan nu het meer persoonlijke werk, te beginnen met Frits. Op het moment dat ik de vacature voor een promotieplek in sportschema's zag staan vanop een scherm ergens ver buiten Eindhoven, dacht ik direct: *ja, dit is het*. Wat ik niet dacht, was dat ik dit ooit zou doen. Desalniettemin kreeg ik toch de kans, en het was nog meer hét dan ik had kunnen denken. Dat lag niet in de laatste plaats aan de geweldige begeleiding die ik de afgelopen 4 jaar heb gehad. Voor onredelijk veel geduld, wijze woorden en vooral veel open gesprekken, kon ik vrijwel altijd bij je aankloppen. Wie wil er nou niet belangrijke meetings over potentiële publicaties kunnen onderbreken om de sores op het voetbalveld of in de tenniswereld te bespreken - de *sport* in sport scheduling werd bijzonder serieus genomen.

Rudi, bedankt voor het te elfder ure aanschuiven als copromotor. Hoewel we al wel eens wat gebabbeld hadden over allerhande problemen (AYTO), werd het allemaal wat intensiever en officiëler het laatste jaar. Bedankt voor het begrijpen, bijsturen en aanscherpen van mijn slordige notaties en afgeraffelde bewijzen en teksten.

I like to thank the members of my doctoral committee for reading this dissertation and giving absurdly detailed and useful feedback on the topics. Also, Mario, thanks for being a very nice host in Bergen during these days.

Toen ik begon, waren er geen andere PhD'ers behalve één Colombiaan die net de avond ervoor op Schiphol was geland, nog geen echte verblijfplaats had, en zoals twee dagen later bleek, de bof had opgelopen (hoe dan?!) en in quarantaine moest. Quarantaine was destijds overigens nog een heel raar woord. Simon, het was leuk op en buiten het kantoor, het samen voetballen met Daan, Ale, Lorenz, de rest van de Giganten, en dan aan het einde van de dag de kleintjes bij GEWIS, en daarom des te jammerder dat het niet de hele vier jaar heeft geduurd.

Na die eerste rustige maanden, kwamen er gelukkig nog heel veel nieuwe promovendi; Celine, met wie ik kon wedijveren om de minst productieve vrijdagen en de kantoortemperatuur, Danny, Jasper, Hans, Dylan, Sjanne, Lucy, Antonina, tot uiteindelijk de halve gang helemaal vol zat. Ideaal om af en toe mee te sparren over wiskundige problemen of jezelf te spiegelen om te concluderen dat je er organisatorisch of op andere manieren weer een potje van gemaakt had, maar ook om gewoon mee te zitten na het werk.

Of course, there were also more senior members wandering around, always in for a friendly chat or discussion about whatever. It is really nice to work in a place where everyone's door seems open to everyone, which is how I perceived working on the fourth floor the last four years.

In de zomer voordat ik in Eindhoven aan de slag ging, had ik net de Tour de France-veiling van 2018 gewonnen, met een ruime 4 punten voorsprong op Pim en veel meer op de overige veilers. Sinds ik me professioneel met sportwiskunde bezig ben gaan houden, heb ik het hoogste treetje helaas niet meer gezien, hoewel we nog altijd driemaaljaars veilen; je kunt je afvragen of ik dus iets bijgeleerd heb... Wat mij betreft houden we dat veilen er in, maar ik zie jullie ook graag zonder veiling. Het moet immers wel leuk blijven.

Dan nog een andere bezigheid waar het hoogste treetje elke keer uit zicht blijft, ook al doe ik het al vele jaren waaronder een jaar of 10 in min of meer hetzelfde team: hockeyen. Ooit worden we kampioen, maar als het niet lukt, gewoon het jaar erna nog eens proberen.

Amber, bedankt voor de omslag. Erg mooi geworden!

Papa, mama, bedankt voor het eeuwige vertrouwen en steun in alle vormen denkbaar. Zonder dat was dit nooit gelukt. Evert, hoewel je in Groningen woont, lijkt het via WhatsApp altijd dichtbij. Bij elke willekeurige gebeurtenis, tv-uitzending, sportwedstrijd, kan ik sturen wat ik denk, om te lezen dat je 30 seconden ervoor al hetzelfde had gezegd. Of precies andersom.

Tot slot, liefste Catrien, bedankt voor de vele jubilea, (mini-)vakanties, tweede koffie in de ochtend, ons huis met Ollie en Domino, de blob, het samenzijn, het alles.

CONTENTS

ACRONYMS

SRR Single Round Robin

DRR Double Round Robin

COE Carry Over Effect

CM Circle Method

IP Integer Program

HAP Home/Away-Pattern

HAP-set Home/Away-Pattern Set

FBTS First-Break-Then-Schedule

PDC Professional Darts Corporation

FMA First Mover Advantage

VNL Volleyball Nations League

CPS Canonical Pattern Set

SKO Serial Knock-Out

1 INTRODUCTION

1.1 PRESENT

The world of sports is a world on its own. Everyday, thousands of athletes compete professionally against one another, millions play on a lower level for fun, to stay fit, and to be with friends, while hundreds of millions show up or tune in online or on TV to support their favorite athletes.

With great participation comes great responsibility. All around the globe, sports unions and clubs are tasked with hosting competitions and tournaments for their members and spectators. They cannot just copy paste the work done by others, since every sport, country and club has its own perks.

It matters if matches involve two sides, like football, or any number of participants, like athletics or golf. It matters if teams are expected to play at home or away, or just anywhere. It matters how many matches one can play within a short period of time, or how many matches there can be played at one specific location.

Besides these practical issues regarding the scheduling of matches, there is also the competition format itself that needs to be decided in advance. Do all the participants play each other in a big competition like the Premier League, do they play in a knock-out tournament like at Wimbledon, or do they play in a hybrid form with group stages leading to a knock-out, like the FIFA World Cup?

These are just some of the many choices that need to be considered. For every sport, every set of players and teams, every country, there is a new instance of a sport scheduling problem that is worth solving.

1.2 BACKGROUND

The mathematical world of sport scheduling is old and vast. Thomas Kirkman can be regarded as pioneer in this field, searching - in true combinatorial fashion - mid nineteenth century for nice ways to distribute persons over groups and stages. Most famous is perhaps the

Kirkman schoolgirl problem, posed in 1851:

Fifteen young ladies in a school walk out three abreast for seven days in succession: it is required to arrange them daily so that no two shall walk twice abreast. (Kirkman, 1851)

Though walking at school might be far away from what we call competitive sports nowadays, the nature of the question is what makes this an early example of competition scheduling. Think of the ladies as players and *walking abreast* as meeting to play against each other. This transforms the problem from walking hand in hand with your best friends, to having everyone meet 2 opponents every day to play a match.

Eventually, after solving the sportified schoolgirl problem, you would get a set-up where every player meets every other player exactly once; there are 7 days and a player meets 2 different players every day, meeting a total of 14 players during the week. This is exactly the number of opponents any player has within the set of 15 players.

That is a very nice property, having everyone meet everyone, and a schedule in which this occurs is called a Single Round Robin (SRR). This is a very popular way of scheduling competitions, as is the related Double Round Robin (DRR). The latter is used abundantly to schedule leagues for football, volleyball, handball, etcetera. The benefit of the DRR compared to the SRR, is that it is possible to have every team meet every other team once *at home*.

The idea of playing either at home or away is rooted deeply in the mind of the players and audience, especially in team sports. The Double Round Robin competition feels so natural and common, that for instance in Dutch, the term *hele competitie* (entire/proper competition) is reserved for a DRR rather than a SRR, which is dubbed *halve competitie* (a half competition), clearly indicating the perceived incompleteness of such a set-up.

1.3 MAKING A SCHEDULE

Although there are twice as many matches in a DRR compared to an SRR, this does not necessarily lead to more complicated problems. Phasing the competition is often desired, implying that every pair of teams meets once in the first half of the season, and once in the second half - meaning that both halves can essentially be seen as a SRR.

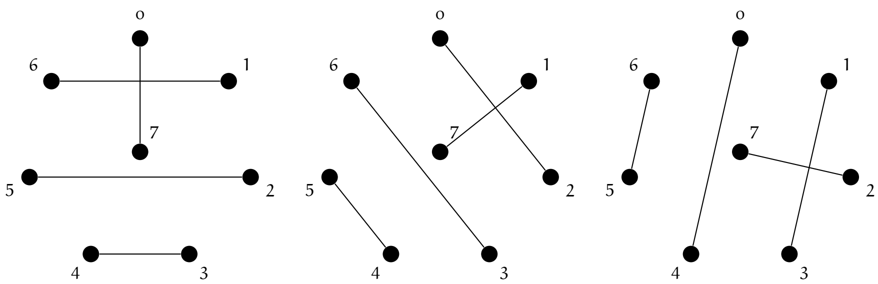


Figure 1: Visualization of the circle method on 8 teams

And that is a good thing, as there is a very natural way of scheduling a SRR, for any number of players or teams: the Circle Method (CM). Said to be invented by Kirkman - although this is disputed - the Circle Method is the basic, go-to solution to get a schedule. The attractiveness of the CM lies with its simplicity and general applicability.

According to Siemann (2020) and Lucas (1883) the CM originates from Félix Walecki.

Algorithm 1 The Circle Method

Input: A set of N players.

1: Pick a player to be the 'central' player and put a dot representing this player in the middle.
2: Place all the other players evenly distributed on a circle around the center, starting at 12 o'clock.
3: Draw a base-line between the central player and the player directly above.
4: Draw lines perpendicular to the base-line to pair all the remaining players.
5: Every player is now connected to exactly one other player. This is the first round.
6: Create round r by rotating the lines r steps clockwise.

Output: A SRR of $N-1$ rounds.

Applying algorithm 1 ensures that every player meets every other player exactly once, thus creating the desired Single Round Robin. Figure 1 illustrates the origin of the name, as well as the procedure.

Matches			
07	16	25	34
17	20	36	45
27	31	40	56
37	42	51	60
47	53	62	01
57	64	03	12
67	05	14	23

Table 2: CM Schedule on 8 teams

Besides general applicability, the Circle Method has another positive feature: the possibility to assign home/away to the matches, in an alternating and balanced fashion for the teams. Instead of lines, we draw arrows in step 4 of Algorithm 1, alternating their orientation left-to-right and right-to-left. A team at the head of the arrow, plays at home, the team at the tail plays away. We also replace the base line by an arrow, but contrary to the other arrows, it has its orientation flipped after every turn of the clock.

Figure 2 shows the extended procedure.

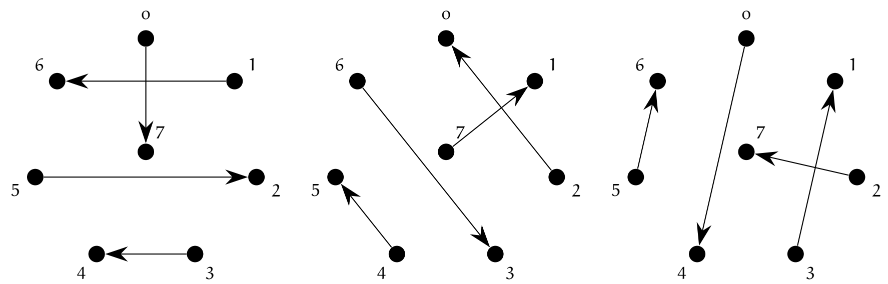


Figure 2: Home/Away-assignments per match per round

1.4 COMPLEXITY

Team	HAP
0	AHAHAHA
1	AHHAHAH
2	HAAHAHA
3	AHAHHAH
4	HAHAAHA
5	AHAHAHH
6	HAHAHAA
7	HAHAHAH

Table 3: HAP-set by CM

The Home/Away-assignments generated by the CM, are not just any assignment. The structure and choices are such that each individual team has its own Home/Away-Pattern (HAP), and in this HAP, a Home match is generally followed by an Away match, and vice versa. Only in the odd rounds there are two teams for which this rhythm is broken. In Figure 2 we see that in the third round, teams 1, 2 have the same assignment as they had in the second round.

This otherwise alternating rhythm is a property that those involved usually like in a schedule - every break in the rhythm is hence called exactly that in the literature, a *break*. The set of HAPs, called HAP-set, generated by the Circle Method has all of its breaks in the odd rounds, and every team has only one break - the pattern-set that has these properties is referred to as the Canonical Pattern Set (CPS).

So there exists a procedure that generates a SRR with a seemingly nice pattern of Home/Away's per team, what is left to study? As it turns out, a lot. For numerous reasons, we want methods that can handle more input than just the number of teams.

We distinguish two types of input that we want to see in the resulting schedule: *hard constraints* and *soft constraints*. The hard constraints are non-negotiable, for example that the schedule is a SRR, or that teams can only play at home if their venue is available at that time.

The soft constraints, on the other hand, require the organizers to point out favourable parts of a schedule, like minimized travel distance, not too many breaks per team, not playing the same opponent twice in a short period of time. These are elements in a schedule that one would like to have as good as possible.

Where the Circle Method might satisfy the hard constraints for a specific problem, it is not clear whether it scores good on the soft constraints. To get any insights on this, we need to compare it to other schedules. But, how do we make other schedules?

Besides the Circle Method, there is no other known easy method of creating a schedule for any number of players. And especially if there are additional hard constraints to take into account, a tailor-made approach is needed to find schedules *feasible* with respect to the demands - even though the number of SRR grows super exponential in terms of the number of players N, it can be extremely difficult to find even a single schedule that satisfies all the constraints, or prove that it doesn't exist.

To find good schedules that can be used to organize tournaments, a popular approach is to create an Integer Program (IP). An Integer Program is a mathematical formulation of the constraints that is suitable for solving by specialized solvers. The hard constraints can be formulated as constraints, the soft constraints can be reshaped as hard constraints or become part of an objective function, which the solver tries to optimize.

For example, minimizing breaks is a soft constraint. It can be implemented as hard constraint: no team has more than, say, 3 breaks.

A promising characteristic of solving an IP with modern solvers, is that it is guaranteed to find a feasible solution if there exists one, and even an optimal solution. The downside, however, is that this guarantee doesn't say anything about the time needed to solve, or the computational resources needed.

Solving an IP to find an optimal solution can practically take forever, thus making good choices in the implementation are crucial in finding acceptable solutions quickly. Which parameters to fix, which parts to optimize first, are all choices you can make to speed up the process of arriving at a good schedule. Any decision you make in the beginning, limits the potential solutions you get later on, hence the importance of choosing wisely.

This thesis discusses which choices to make, and when, while scheduling sports tournaments.

1.5 SET-UP

The chapters are divided into two parts, Flexibility and Fairness.

Flexibility

The first part, Flexibility, focuses on the flow of scheduling itself, and in particular, around the Home/Away-Patterns. The Home/Away-Pattern of a team, indicates in which order it plays Home and Away. As said earlier, it is often desired that this pattern has few breaks. When a team is scheduled to play at Home, it is important that it can use its home venue, so not every HAP is suitable for all the teams.

A way to ensure that both these constraints are satisfied to an acceptable degree, is to first pick a favourable Home/Away-Pattern Set (HAP-set), and after this pick, start scheduling the matches per team. By fixing the HAP-set, matches can no longer be scheduled in any round - some HAP-sets give more flexibility than others.

CHAPTER 2 THE FLEXIBILITY OF HOME AWAY PATTERN SETS dives into the scheduling approach called First-Break-Then-Schedule (FBTS). After fixing the HAP-sets and the breaks, one might still have the problem that certain matches cannot take place in specific rounds, or the objective of optimizing the schedule regarding traveled distance per team.

However, by fixing the HAP in advance, possibilities are lost, and perhaps some unwanted outcomes are enforced. Teams can only play teams with opposing Home/Away-assignment, which severely limits the scheduling possibilities of individual matches. To foresee and prevent this from happening, we introduce three measures that indicate the flexibility that is left after choosing the HAP-set, calculate the performance of some of these sets regarding these measures, as well as show some theoretical bounds.

CHAPTER 3 MAXIMUM ORTHOGONAL SCHEDULES builds upon a specific measure introduced in Chapter 2, namely the *width*.

When analyzing the CPS constructed with the Circle Method in Figure 2, we showed it had a width equal to 1. This means there are matches that are fixed to a specific round after fixing the HAP-set, which is not good. The benefit of using a HAP-set with a large width, is that every match has some flexibility, as no match fixed to certain rounds.

We construct schedules that work on the same HAP-set and are orthogonal, meaning that no match is scheduled in the same round in any of the schedules. We show that when the number of teams N equals a power of 2, a HAP-set exists for which there are $\frac{N}{2}$ orthogonal schedules - every match has $\frac{N}{2}$ rounds in which it can be scheduled.

CHAPTER 4 THE MULTI-LEAGUE SPORTS SCHEDULING PROBLEM is on the problem of scheduling a lot of leagues (hence the name) at the same time. When the organizer is not only responsible for just one competition, but a multitude of similar competitions, regarding different levels or age ranges, one cannot just schedule every competition separately. As different teams might use the same venue - because they are part of the same club, for instance - there are capacity constraints that need to be considered. On the other hand, clubs might have a preference of which teams they want to jointly play on their venue on the same day.

We show that using an FBTS-approach, we can deal with the capacity constraints optimally in linear time.

Fairness

The second part, Fairness, looks at ways to find schedules that are optimal with respect to different objectives, that are all related to a measure of fairness. Contrary to the first part, where all discussed schedules are classic round robin tournaments, different types of competitions are analyzed in this part and specific instances of real-world examples.

CHAPTER 5 MINIMIZING THE CARRY-OVER EFFECT
In many round robin competitions, the matches are scheduled per round. This means that every team or player can expect to play one match per round, and that the rounds follow each other sequentially. For every match one plays, there is a next match with a new opponent. What happens if the performance of a player is influenced by its opponent in the previous round?

The effect of a previous match influencing the upcoming match, is called the Carry Over Effect (COE). If one player tends to follow another player opponent wise, this player receives more Carry Over Effect from that player than from other players. This leads to an imbalance and possible (dis)advantage. Finding schedules with balanced COE for all pairs of players is difficult, and except for a few instances, no balanced schedules are known. We introduce a way of finding schedules that are (almost) balanced quickly, by exploiting specific features that the known balanced schedules have.

CHAPTER 6 BALANCED SERIAL KNOCK-OUT TOURNAMENTS
A potential downside to round robin competitions as a way to determine who is the best, is that some players lose interest as they are out of contention for the main prizes. This may lead to underperforming in the latter stages of the competition and skewed results. To tackle this, the Professional Darts Corporation (PDC) introduced a new type of competition that we call a Serial Knock-Out (SKO) where every round consisted of a knock-out tournament among the 8 players. The additional incentive of winning the knock-out tournament and earning money, should motivate the players near the end of the season.

The draws of the knock-out tournaments are scheduled in advance. Just as in the original DRR, the players are still paired to every other player exactly twice in the first stage of the knock-out tournaments. After that stage, in the semi-final and final, the distribution of who can meet who, is less evenly spread out. Some players can only meet in the finals, while others are set to meet in the majority of their semi-finals, given that they reach that stage.

This is unfair to some as it is unbalanced. We show how to construct a perfectly balanced SKO on 8 players, and how to construct them on any set of 2^k players.

CHAPTER 7 HOW TO SCHEDULE THE VOLLEYBALL NATIONS LEAGUE

In competitions with teams that are geographically far apart, travel times become an important issue to take into account. International competitions may require the participating nations to have their teams fly all around the world to play, and this has been shown to impact their peak performance.

Hence, a competition with a high travel load, should try to evenly distribute this among the competitors. An example of such a competition is the Volleyball Nations League, which has teams travel from continent to continent for 5 weeks in a row.

Usually, traveling problems tend to be hard to solve. The format of the Volleyball Nations League (VNL) however, allows us to use a shortcut to calculate the traveled distance and the fairness of the schedule, without scheduling the entire competition. This allows us to optimally solve the instances, by merely deciding where teams need to play before deciding who plays where.

CHAPTER 8 FAIRNESS IN PENALTY SHOOTOUTS

Sport scheduling does not stop with scheduling matches and competitions. Even on the pitch, choices need to be made in advance. Who kicks off at half time, who shoots first in a penalty series, for example. Usually, these decisions are made by a coin toss, and a fair coin should make for a fair choice.

Empirical evidence suggests that the team shooting first in the penalty shootout, has an advantage caused by the psychological pressure for the other team that has to catch up in the score. With this First Mover Advantage (FMA) as a starting point, we look for fairer sequences, by mixing the order in which the teams take their penalties.

We show that within our model for the psychological pressure, every finite repeated order will have an advantage for one of the teams, and give an algorithm that produces a fair sequence for any set of parameters that describe the FMA.

Part I

FLEXIBILITY

2 THE FLEXIBILITY OF HOME AWAY PATTERN SETS

The highest Dutch football league, the Eredivisie, publishes its entire playing schedule in July, before the start of the season a few weeks later. The clubs then have the option to push for small corrections, before the schedule is finalised.
At the start of the 2021 – 2022 season, the club FC Twente particularly complained after the corrections. The first half of the season was altered in such a way that they would only play one of their eight home matches at their preferred time, Saturday evening. Due to conflicts other teams had with international duties, and subsequent changes across the schedule, most of their home matches were moved to Sunday.
It is natural to wonder why the KNVB did not just move some other matches, not involving FC Twente, to fix the problem they faced. However, scheduling a competition is a tough task, and not all matches can just be moved to any round. As they try to make all teams have an alternating Home-Away Pattern as much as possible, together with a pretty strict notion of rounds in which each team is supposed to play, after releasing a draft schedule, there is little room to play around.
Deciding upon a Home/Away Pattern early on is a popular scheduling approach, as it is regarded one of the most important features of any schedule. Building a schedule step-by-step based on the input received by the stakeholders, it is important to choose a HAP that allows for flexibility in the latter stages of the scheduling.
This chapter discusses some ways how to indicate which HAP-sets give more flexibility and which less, when choosing them in advance.'
This chapter is based on Lambers, Goossens, and Spieksma (2022).

2.1 INTRODUCTION

Round robin tournaments are abundantly used in all kinds of sport competitions, both in professional leagues, as well as in amateur leagues. The setting where each pair of teams meets once (Single Round Robin (SRR)), or twice (Double Round Robin (DRR)) has proven to be a very popular format to arrive at a ranking of the participating teams (Goossens and Spieksma, 2012).

Deciding which matches are played in which round of a single or double round robin tournament is known to be of great practical importance in scheduling professional leagues such as soccer competitions. Hence, scheduling a round robin tournament has attracted a lot of attention in the scientific literature (Kendall, Knust, et al., 2010; Van Bulck et al., 2020).

It is very common for a round robin competition to have teams that all have their own venue, and a match between two teams takes place at the venue of one of the two teams, meaning that one team plays home (H), while the other team plays away (A). In competitions where this is the case, the difference between playing matches at home or away is often regarded significant (Pollard, 2008; Schwartz and Barsky, 1977). To have all teams play evenly at home - and consequently away - throughout the season, is generally preferred by those involved.

A popular practice when scheduling such competitions, is known as a First-Break-Then-Schedule approach (Nemhauser and Trick, 1998). This hierarchical approach consists of two phases, that we now informally describe (see Section 2.2 for more precise terminology).

In the first phase of the First-Break-Then-Schedule, each team receives a Home/Away-Pattern (HAP), i.e. it is specified for each round whether the team plays at its home venue, or not. In the second phase, the matches are scheduled: given a match between two teams, a round is chosen where it is scheduled. Of course, for such an assignment of matches to be feasible it must hold that each team plays at most one match in each round, and that the assignment is compatible with the patterns obtained in the first phase.

In such a hierarchical approach, it is clear that the scheduling decision (which matches to play in which round) crucially depends on the HAP-set that is chosen in the first phase. For instance, it is conceivable that in the second phase, a set of constraints is revealed that are incompatible with the given HAP-set. This would need to be solved by either changing the HAP-set, or by putting energy into mitigating the effects of violating that specific set of constraints.

The theme of this chapter is that not all HAP-sets have the same risk of leading to incompatible constraints. Indeed, some HAP-sets are more flexible than others. In Section 2.2, we give a number of definitions, and in Section 2.4 we introduce our measures for the flexibility of a HAP-set. We also show how single-break HAP-sets, in particular the Canonical Pattern Set (CPS), behave with respect to these measures. In Section 2.5, we describe how to compute these measures using integer programming; we conclude in Section 2.6.

2.2 PRELIMINARIES

We consider the scheduling of an SRR with an even number of teams $N = 2n, n \in \mathbb{N}$. To avoid trivialities, we assume $2n \geqslant 4$.

When $n = 1$, an SRR consists of only 1 match.

A *match* between two distinct teams i and j is denoted by an unordered pair $\{i, j\}$, $1 \leqslant i \neq j \leqslant 2n$. Every team plays every other team exactly once. We assume that the SRR consists of $2n-1$ rounds, where in each round every team plays exactly one match - this is called *time-constrained*. A *schedule* is a specification of all $\binom{2n}{2}$ matches spread out over the $2n-1$ rounds, including a team playing at home for each match. Thus, in the schedules we consider, every match $\{i, j\}$ is assigned to a specific round, the venue of the match is specified, and every team plays exactly one match in every round. For a survey of round robin scheduling, we refer to Rasmussen and Trick (2008) and Drexl and Knust (2007).

Although we restrict our analysis to SRRs, we claim that many of the ideas presented can be generalized to DRRs as well (or k-round robin settings with $k \geqslant 2$). In particular, the definitions of the measures (see Section 2.4) can be generalized to DRRs.

However, we want to point out that a match in a DRR tournament is generally seen as an ordered pair (i, j), rather than an unorderd pair $\{i, j\}$. Contrary to a single round robin tournament, where the home advantage is to be decided in the schedule, a double round robin tournament typically assumes that each team plays at home against each other team exactly once. Moreover, very often it is required that one encounter between a pair of teams occurs in the first half, whereas the other encounter occurs in the second half of the schedule; and these two encounters should be separated by a given number of rounds. Taking such issues into account would impact the corresponding definitions of the measures for DRR tournaments.

We now proceed with defining our terminology.

Definition 1 (based on Rasmussen and Trick (2008))**.** *A Home/Away-Pattern (HAP) is a vector* $h = (h_1, h_2, \dots, h_{2n-1})$*, where* $h_r \in \{H, A\}$ *specifies whether a team that plays according to pattern* h *plays Home or Away in round* r*, with* $r = 1, 2, \dots, 2n-1$.

An important property of a HAP is given by the occurrence of so-called *breaks*: the presence of two consecutive symbols that are identical (see De Werra (1981) and Goossens and Spieksma (2011)). To simplify notation, we take a circular view of a HAP, i.e., we define $h_0 \equiv h_{2n-1}$.

From a practical perspective, the circular HAP seems useless. However, mathematically this definition is aesthetically pleasing.

Definition 2 (based on (Rasmussen and Trick, 2008))**.** *A Home/Away-Pattern* h *has a* break *in round* r *if* $h_{r-1} = h_r$, *with* $r = 1, 2, \dots, 2n-1$. *In case* $h_{r-1} = h_r = A$ *we call the break an* away-break, *when* $h_{r-1} = h_r = H$ *we call it a* home-break.

Notice that, since the number of rounds $2n-1$ is odd, and given our circular view of a HAP, any HAP has at least one break. Even the HAP h that purely consists of alternating H's and A's has a single break in round 1, as $h_1 = h_{2n-1} = H$. In fact, the number of breaks of a HAP on an odd number of rounds, is bound to be odd as well.

Motivated by practice, we are interested in breaks present in a HAP. It is important to see that an entire HAP is in fact defined by the rounds in which its breaks occur, and by its value in the final round (or any other round). Thus, we introduce a break-representation that defines a HAP by exactly those two properties.

Definition 3. *HAP* h *is denoted as* $h = P^H(r_1, \dots, r_b)$ *with* $1 \leqslant r_1 < \dots < r_k \leqslant 2n-1$ *if:*

1. $h_{2n-1} = H$.
2. *For all* $i \leqslant k$, h *has a break in* r_i, *i.e.* $h_{r_i - 1} = h_{r_i}$.

When $h_{2n-1} = A$ *instead of* H, h *is denoted as* $h = P^A(r_1, \dots, r_b)$.

The break-number *of* h *is given by* $bn(h) = b$. *The pattern* h *is called* single-break *if* $bn(h) = 1$.

Two patterns h, h' with $h = P^H(r_1, \dots, r_b)$ and $h' = P^A(r_1, \dots, r_b)$, are called complementary. We formalize this definition:

Definition 4 (based on (Rasmussen and Trick, 2008))**.** *Two Home-Away patterns* h, h^c *are called* complementary *if and only if* $h_r \neq h^c_r$ *for each* $r = 1, 2, \dots, 2n-1$.

We sometimes omit the distinction between P^H, P^A and simply use P. When discussing complementary pairs of patterns, we use P^c to denote the complement of HAP P.

In a chess tournament, playing with white (black) can be regarded as home (away). See (LNS, 2020) for an analysis of the Wijk aan Zee schedule.

To illustrate this terminology, consider the HAPs used by the professional Dutch tennis competition for teams in 2019, organized by the Royal Dutch Tennis Association. This is a competition between 8 teams who play a Single Round Robin, according to the HAPs given in Table 5 (for the precise schedule, we refer to the (KNLTB, 2019)). The first column shows the team's names, and the next 7 columns

	Round 1	2	3	4	5	6	7	
Team 1 Lewabo	A	H	A	H	A	H	H	$P^H(7)$
Team 2 Spijkenisse	A	H	A	H	H	A	H	$P^H(5)$
Team 3 Suthwalda	H	A	H	A	H	A	H	$P^H(1)$
Team 4 Nieuwekerk	H	A	H	A	A	H	A	$P^A(5)$
Team 5 Arnolduspark	H	A	H	A	H	A	A	$P^A(7)$
Team 6 Leimonias	A	H	A	H	H	A	A	$P^A(5,7)$
Team 7 Naaldwijk	A	H	A	H	A	H	A	$P^A(1)$
Team 8 Kimbria	H	A	H	A	A	H	H	$P^H(5,7)$

Table 5: HAPs for the 2019-2020 top Dutch male tennis league

describe the Home-Away Patterns. The final column gives the corresponding description of the particular pattern.

We now turn our attention to sets of HAPs and their schedules; we define a HAP-set as follows:

Definition 5. *A* HAP-set $\mathcal{H} = \{h_i : i \leqslant 2n\}$ *for* $2n$ *teams is a set of* $2n$ *HAPs* h_i. *We say* $2n$ *is* the order *of* $\mathcal{H}$.

Next to the HAP-sets, we define a schedule S:

Definition 6. *A schedule* $S = (S_r)_{r \leqslant 2n-1}$ *for an SRR on* $2n$ *teams, consists of a set of* $2n-1$ *rounds* S_i, *where each round is a partition of the teams into* n *pairs. To make the schedule satisfy that it is an SRR, for each pair of teams* $\{i, j\}$, *there should be a round* r *such that* $\{i, j\} \in S_r$.

We define the following notions on HAP-sets, schedules, and their relations. As we have enumerated $2n$ teams in a schedule, and we have enumerated $2n$ HAPs in a HAP-set, for simplicity we assume that team i in schedule S, is linked to the i-th HAP-set, unless stated otherwise.

Definition 7. *Given a HAP-set* $\mathcal{H}$ *on* $2n$ *teams, we say:*

- *Schedule* S *is* compatible *with HAP-set* $\mathcal{H}$ *if for all scheduled matches* $\{i, j\} \in S_r$, *the corresponding HAPs obey* $h^i_r \neq h^j_r$.
- *For HAP-set* $\mathcal{H}$, *we define* $\mathcal{S}(\mathcal{H})$ *to be the set of all schedules that are compatible with* $\mathcal{H}$.
- *A HAP-set* $\mathcal{H}$ *is* feasible *if there exists a schedule compatible with* $\mathcal{H}$, *i.e.* $\mathcal{S}(\mathcal{H}) \neq \emptyset$, *and* infeasible *otherwise.*
- *A HAP-set* $\mathcal{H}$ *is* complementary *if for every HAP* $h \in \mathcal{H}$ *with* $h = P(r_1, \ldots, r_b) \in \mathcal{H}$, *there is a* $h' \in \mathcal{H}$ *with* $h' = P^c(r_1, \ldots, r_b)$.

- *A HAP-set $\mathcal{H}$ is called* single-break *if each pattern $h \in \mathcal{H}$ is a single-break pattern, i.e. $bn(h) = 1$ for all $h \in \mathcal{H}$.*

The HAP-set given in Table 5 is an example of a complementary HAP-set; it is, however, not single-break, as only 6 of the 8 HAPs are single-break and the patterns of team 6 and 8 both have three breaks (for both, one of these breaks is the break between round 7 and 'consecutive' round 1).

2.3 FEASIBLE SINGLE-BREAK HAP-SETS

Identifying whether a given HAP-set is feasible is known as the pattern set feasibility problem. As far as we are aware, the complexity status of this problem is not settled; Miyashiro, Iwasaki, and Matsui (2002) describe a necessary but not sufficient condition for feasibility. See also Briskorn (2008) and Horbach (2010) for a Linear Programming-formulation that they project to be sufficient.

It is an open question whether the condition set in Miyashiro, Iwasaki, and Matsui (2002) is sufficient for feasibility of a single break HAP-set.

We do not go into detail regarding the feasibility question, but we do note a few simple truths. When given a schedule compatible with a certain HAP-set for an SRR, one can (i) interchange any two rounds, and (ii) change in a single round each H to an A, and vice versa, and arrive at another HAP-set that must be feasible.

Definition 8. *On $2n$ teams, the space of feasible HAP-set is denoted with $\mathbb{H}_{2n}$. Subspaces $\mathbb{H}_{2n,k}$ for $k \leqslant 2n-1$ consist of the feasible HAP-sets with break-number less than k, for any $k \leqslant 2n-1$:*

$$\mathbb{H}_{2n,k} = \{\mathcal{H}_{2n} : bn(h) \leqslant k \textit{ for each } h \in \mathcal{H}_{2n},\ \mathcal{H}_{2n} \in \mathbb{H}_{2n}\}$$
$$\mathbb{H}_{2n} = \{\mathcal{H}_{2n} : \mathcal{H}_{2n} \textit{ feasible}\}.$$

We are primarily interested in feasible HAP-sets that consist of patterns with a limited number of breaks. In particular HAP-sets that are elements of $\mathbb{H}_{2n,1}$, the collection of single-break HAP-sets of order $2n$.

As any HAP on $2n-1$ rounds has an odd number of breaks, in fact $\mathbb{H}_{2n,2k} = \mathbb{H}_{2n,2k-1}$.

By De Werra (1981), we know that $\mathbb{H}_{2n,1} \neq \emptyset$. For example, the HAP-set generated by the CM shown in Figure 2 is single break and this method is applicable for all sets of $2n$ teams.

Besides existence, more can be said about elements in $\mathbb{H}_{2n,1}$ and how they are characterized. We state this characterisation in the following theorem.

Theorem 1 (De Werra (1981))**.** *For any element $\mathcal{H} \in \mathbb{H}_{2n,1}$, the following is true:*

- *$\mathcal{H}$ is complementary.*

- *$\mathcal{H}$ is defined by $1 \leqslant r_1 < \cdots < r_n \leqslant 2n-1$, indicating rounds in which breaks occur.*

Thus a feasible single break hap-set $\mathcal{H} \in \mathbb{H}_{2n,1}$ can be written as $\mathcal{H} = \{P(r_i), P^c(r_i) : i \leqslant n\}$

Proof. The HAP-set $\mathcal{H}$ can only be feasible if in every round, n teams are assigned to play at home, and n away. Thus, for every home-break in round r, there must be an away-break in the same round to maintain the balance. Since all HAPs $h \in \mathcal{H}$ have at most one break, $\mathcal{H}$ must therefore be complementary to be feasible.

As two teams with identical HAPs can never play against each other in a compatible schedule, in any feasible $\mathcal{H}$ there cannot be two complementary pairs with a break in the same round. This means there must be exactly n distinct rounds where a complementary pair of HAPs have a break. □

By Theorem 1, we know that any feasible HAP-set $\mathcal{H} \in \mathbb{H}_{2n,1}$ is defined by $(r_1, \dots, r_n)$, indicating the rounds r_i for which there exist HAP $h \in \mathcal{H}$ with a break in r_i. With $r_1 < \dots r_n = r_0$, we define the *break-gaps* of this HAP-set $\mathcal{H}$ to be $d_i = r_i - r_{i-1}$. Value d_i equals the number of rounds between the two breaks in round r_{i-1} and round r_i.

We define the *break-gap representation* or *D-notation* of a single-break HAP-set h, to be $D = (d_1, \dots, d_n)$ - see also De Werra (1981) and Knust and Lücking (2009). Notice that $\mathcal{H}$ can only be feasible, if $d_i \geqslant 1$ for all $i \leqslant n$. When there is no ambiguity, we may refer to $D = (d_1, \dots, d_n)$ as simply $D = d_1, \dots, d_n$.

Any single-break HAP-set $\mathcal{H}$ has a unique D-notation, but several HAP-sets can have the same D-notation. Consider for example two HAP-sets on four teams, so $n = 2$: $\{HAA, HHA, AHH, AAH\}$ and $\{HAH, HHA, AHA, AAH\}$. The former has $r_1 = 2$ and $r_2 = 3$ leading to $D = (1, 2)$ while the latter has $r_1 = 1$ and $r_2 = 2$, also leading to $D = (1, 2)$.

Not only can a feasible, single-break HAP-set $\mathcal{H}_{2n}$ be represented by a sequence $(d_1, d_2, \dots, d_n)$, it is also true that any set of n positive integers that sum to $2n-1$, corresponds to a (set of) single-break HAP-sets (which is not necessarily feasible). Although a HAP-set, when specified in terms of r_i values, uniquely determines a corresponding D sequence, a D sequence can correspond to multiple HAP-sets with different r_i's.

As permuting the rounds in a cyclic manner has no effect on feasibility, for any sequence $D = (d_1, \dots, d_n)$ there exists a HAP-set $\mathcal{H} \in \mathbb{H}_{2n,1}$ with break-gap representation D, if the same is true for all cyclic permutations of $(d_1, \dots, d_n)$. Also, reversing the order of the rounds does not impact feasibility. We can thus always assume to deal with a HAP-set for which the D-notation is lexicographic largest among its cyclic permutations and reversions.

It is not trivial which D-sequences result in a feasible HAP-set. It is known however that the CM generates a feasible, single break HAP-set, and this HAP-set, referred to as the CPS, has a particular D-notation,

Definition 9. *The* Canonical Pattern Set *of order* $2n$ *- denoted* CPS_{2n} *- is the following single-break HAP-set:*

$$CPS_{2n} = \{P^A(2i-1), P^H(2i-1) : i = 1, \dots, n\}$$

Its break-gap representation is $D = (2, 2, \dots, 2, 1)$ *or just* $D = 22\dots1$.

For small values of n, there are only a limited number of feasible single-break HAP-sets.

Team	*HAP*
0	*A*HAHAH*A*
1	AH*AA*HAH
2	HA*HH*AHA
3	AHA*HH*AH
4	HAH*AA*HA
5	AHAHA*HH*
6	HAHAH*AA*
7	*H*AHAHA*H*

Table 6: HAP-set with $D = 3121$.

Example 1. *For* $2n = 4,6,\dots,14$*, all the feasible HAP-sets are given in Table 8.*

$2n$	4	6	8	10	12	14
CPS	21	221	2221	22221	222221	2222221
			3121	31221	312221	3122221
					312212	3122212
					313121	3131221
						3213121

Table 8: Feasible HAP-sets on 4,6,8,10,12,14 teams.

Finally, we make the following observations regarding schedules compatible with single break HAP-sets.

Observation 1. *If two patterns* h, h' *in a single-break HAP-set* $\mathcal{H}$ *start with the same symbol (i.e.,* $h_1^1 = h_1$*) and have their break in rounds* $r, r+1$ *respectively, then in any compatible schedule* S *where teams* t, s *follow HAP* h, h' *respectively, the match* $\{t, s\}$ *is played in round* r.

To see why this observation holds, consider the following: As h, h' start with the same symbol, until round $r-1$, we have $h_j = h'_j$, $j = 1, \dots, r-1$. Then, since h has a break in round r, we get $h_r \neq h'_r$. However, as h' has a break in round $r+1$, we see that $h_{r+1} = h'_{r+1}$ and as both patterns are single break, $h_j = h'_j$ for $r+1 \leqslant j \leqslant 2n-1$. Ergo, h, h' can only play each other in round r.

Observation 2. *If three patterns* h, h', h'' *in a single-break HAP-set* $\mathcal{H}$ *start with the same symbol (i.e.,* $h_1 = h'_1 = h''_1$*) and have their break in rounds* $r, r+1, r+2$ *respectively (so their breaks are consecutive), there is no schedule* S *compatible with* $\mathcal{H}$ *and* $\mathcal{H}$ *is thus infeasible.*

Suppose a compatible schedule S exists, assigning t, s, u to h, h', h'' respectively. Then, by Observation 1, within S we must have matches $\{t, s\} \in S_r$ and $\{s, u\} \in S_{r+1}$. However, match $\{t, u\}$ can only be scheduled in either round r or $r+1$, which conflicts either match $\{t, s\}$ or $\{s, u\}$. Ergo, such a schedule S cannot exist.

Observation 2 can be deduced from the upcoming Condition 1 on feasibility as well.

Observation 3. *If there is no team with a break in round $r+1$, then round $r+1$ is* opposite *to round r, i.e., for each $h \in \mathcal{H}$, we have $h_r \neq h_{r+1}$. Thus, for every compatible schedule $S = (S_i)_{i \leqslant 2n-1}$, we can create a new schedule S' by setting $S'_r = S'_{r+1}$, $S'_{r+1} = S_r$ and $S'_i = S_i$ otherwise.*

2.4 MEASURING THE FLEXIBILITY OF A HAP-SET

Given a HAP-set $\mathcal{H}$ on $2n$ teams, we are interested in the diversity of the schedules in $\mathcal{S}(\mathcal{H})$, the schedules compatible with $\mathcal{H}$. The problem is that there is no clear way to determine $\mathcal{S}(\mathcal{H})$ or even $|\mathcal{S}(\mathcal{H})|$. To be able to say something about the diversity within this $\mathcal{S}(\mathcal{H})$, we introduce three distinct measures called the *width* (Section 2.4.1), the *fixed part* (Section 2.4.2), and the *spread* (Definition 16). We analyze how the CPS and other single break HAP-sets fare on these measures.

2.4.1 Measure 1: the width

First, we introduce two notions to distinguish two schedules from one another, *distinct* and *match-distinct*.

Definition 10. *Two schedules S, S' are* distinct *when there exists a match $\{t, s\}$ played in round r in S and another round r' in S'.*

Two schedules S, S' are match-distinct *or* orthogonal *when* for each *round r and* for each *match $\{t, s\} \in S_r$, there is an $r' \neq r$ such that $\{t, s\} \in S'_{r'}$. This is denoted with $S \perp S'$.*

Clearly, two schedules that are match-distinct are also distinct. Using Definition 10, we define the width of a HAP-set:

Definition 11. *The* width *of a HAP-set $\mathcal{H}$, denoted by width$(\mathcal{H})$, is the number of pairwise match-distinct schedules compatible with $\mathcal{H}$. In a formal notation:*

$$width(\mathcal{H}) = \max_{\mathcal{P} \subset \mathcal{S}(\mathcal{H})} \#\{s \in \mathcal{P} : \forall S, S' \in \mathcal{P}, S \perp S'\}| \qquad (1)$$

Chapter 3 deals with constructing HAP-sets with a very high width.

When a HAP-set $\mathcal{H}$ is infeasible, we know that $\text{width}(\mathcal{H}) = 0$. If a HAP-set has width 2 (or higher), then there exist two schedules $S, S' \in \mathcal{S}(\mathcal{H})$ such that each match occurs in different rounds in the two schedules.

The width of the HAP-set given in Table 5 equals 1, implying that no pair of match-distinct schedules compatible with that HAP-set exists.

Further, we refer to Table 9 (right) for an example of a HAP-set on 4 teams with width 2.

One can generalize the definition of width from SRR to DRR by making a distinction between (i, j) *and* (j, i).

Single-break HAP-sets do not allow large width; in fact, for any single break HAP-set, there are matches that must be played in a particular round, as witnessed by the following theorem.

Theorem 2. *For each single break HAP-set* $\mathcal{H} \in \mathbb{H}_{2n,1}$ *we have* $width(\mathcal{H}) = 1$.

Proof. Consider the break-gap representation $D = (d_1, \ldots, d_n)$ of any feasible single-break HAP-set $\mathcal{H} \in \mathbb{H}_{2n,1}$; as discussed in Section 2.3, it follows that the corresponding entries must have $\sum_i d_i = 2n - 1$. Clearly, when choosing n positive integers that sum up to $2n - 1$, there must be at least one '1' among them. Thus, there must be two pairs of complementary HAPs $(h, h^c), (h', h'^c) \in \mathcal{H}$ such that they have their break in consecutive rounds $r, r+1$. Then, by Observation 1, we see that in any schedule S compatible with $\mathcal{H}$, the match between h, h' must be scheduled in round r. Thus $\text{width}(\mathcal{H}) = 1$. □

The argument above can even be extended to say something about the width of HAP-sets that are not single break. It is not difficult to see that any feasible HAP-set $\mathcal{H}$ which contains two patterns that differ in only one round, have a width equal to 1 as well.

Theorem 2 is tight in the following sense: even when all patterns except one are single-break patterns, HAP-sets of width 2 exist. In fact, even for a very small SRR on just $2n = 4$ teams, it is possible to find a HAP-set $\mathcal{H}^*$ with width equal to 2 containing only one pattern that is not single-break. This HAP-set $\mathcal{H}^*$ is given in the following Table 9, together with the CPS on 4 teams.

Team	R1	R2	R3
1	H	A	H
2	H	A	A
3	A	H	A
4	A	H	H

Team	R1	R2	R3
1	H	H	H
2	H	A	A
3	A	H	A
4	A	A	H

Table 9: Left: CPS_4. Right: HAP-set $\mathcal{H}^*$.

The HAP-set on the right differs from CPS_4 in only two entries, and is given by $\mathcal{H}^* = \{P^H(1,2,3), P^A(3), P^A(1), P^H(2)\}$.

Both HAP-sets are compatible with two distinct schedules, $|\mathcal{S}(\mathcal{H}^*)| = |\mathcal{S}(CPS_4)| = 2$. For both HAP-sets, the schedules in $\mathcal{S}(\cdot)$ are shown in Table 10. We see that the two schedules compatible with HAP-set $\mathcal{H}^*$ are orthogonal, while this is not the case for CPS_4.

Round	S_1	S_2	Round	S_3	S_4
1	(1,3),(2,4)	(1,4),(2,3)	1	(1,3), (2,4)	(1,4), (2,3)
2	(4,1),(3,2)	(3,1),(4,2)	2	(1,4), (3,2)	(1,2), (3,4)
3	(1,2),(4,3)	(1,2),(4,3)	3	(1,2), (4,3)	(1,3), (4,2)

Table 10: $S1$, $S2$ are compatible with CPS_4; S_3, S_4 are compatible with $\mathcal{H}^*$.

2.4.2 Measure 2: the fixed part

As can be seen in Theorem 2, the width is a measure that does not differentiate between feasible single-break HAP-sets - all have a width of 1. However, that does not mean that all single-break HAP-sets are equal. Although all $\mathcal{H} \in \mathbb{H}_{2n,1}$ have at least one match that is fixed to a specific round, they don't necessarily have the same number of matches fixed among all compatible schedules. Therefore, we introduce a more refined measure that captures how many matches are fixed for a given HAP-set.

Definition 12. *The* fixed part *of a feasible HAP-set $\mathcal{H}$, denoted by $FP(\mathcal{H})$, consists of the matches that are scheduled in the same round for every schedule compatible with $\mathcal{H}$.*

$$FP(\mathcal{H}) = \{\{i,j\} \colon \exists r \text{ s.t. } \forall S \in \mathcal{S}(\mathcal{H})\{i,j\} \in S_r\} \tag{2}$$

The measure $fp(\mathcal{H})$ equals the order of the fixed part $fp(\mathcal{H}) = |FP(\mathcal{H})|$.

We often refer to $fp(\mathcal{H})$ as simply the fixed part of $\mathcal{H}$.

When there is just a single schedule compatible with $\mathcal{H}$, we see that $FP(\mathcal{H}) = \{\{i,j\} \colon \{i,j\} \subset [n]\}$ and $fp(\mathcal{H}) = \binom{2n}{2}$. Notice also that if the width of a HAP-set $\mathcal{H}$ equals 2, it follows that $FP(\mathcal{H}) = \emptyset$ and thus $fp(\mathcal{H}) = 0$. It turns out that the fixed part of the HAP-set given in Table 5 is of order 4. In particular, the fixed part FP of that HAP-set $\mathcal{H}$ consists of

$$FP(\mathcal{H}) = \{(1,7),(2,6),(3,5),(4,8)\}$$

All these matches need to be scheduled in round 7.

The following lemma provides an easy to obtain lower and upper bound on $fp(\mathcal{H})$ when $\mathcal{H}$ is single break.

Lemma 1. *For any HAP-set $\mathcal{H} \in \mathbb{H}_{2n,1}$, where $\mathcal{H}$ consists of patterns h_i, h_{i+n} that have breaks in r_i, and with D-notation $(d_1, d_2, \ldots, d_n)$, let I be the set of indices where $d_i = r_i - r_{i-1} = 1$, thus $I = \{i : d_i = 1\}$. Then the following two statements must hold:*

1. *The fixed part consists of at least the following subset:*

$$\{\{i, i-1\}, \{n+i, n+i-1\} \colon i \in I\} \subset FP(\mathcal{H})$$

2. *The order of the fixed part is bounded by:*

$$2|I| \leqslant fp(\mathcal{H}) \leqslant n|I|$$

Proof. The first statement is true, since we know from Observation 1 that $\{i, i-1\}$ must be scheduled in round r_{i-1} when $d_i = 1$, thus $\{i, i-1\} \in FP(\mathcal{H})$, and that $\mathcal{H}$ is complementary.

For the second statement, the lower bound follows directly from the first part, as $2|I|$ elements are guaranteed to be part of $FP(\mathcal{H})$.

To see why the upper bound is correct for any $\mathcal{H} \in \mathbb{H}_{2n,1}$, consider a round r in which no team has a break. Then round r is opposite to round $r-1$, thus for any schedule $S = (S_r) \in \mathcal{S}(\mathcal{H})$, we find a new schedule $S' = (S'_r) \in \mathcal{S}(\mathcal{H})$, with $S'_{r-1} = S_r$ and $S'_r = S_{r-1}$, as observed in Observation 3. Thus, none of the matches in S_r, S_{r-1} are fixed - $S_{r-1}, S_r \cap FP(\mathcal{H}) = \emptyset$.

The only rounds that are not opposite to any of the neighboring rounds, are the rounds r_{i-1} with $i \in I$, as both r_{i-1} as the next round r_i are rounds with a break. In any schedule, we have $|S_{r_{i-1}}| = n$, thus for every $i \in I$, at most n matches are fixed. □

We know that the CPS on $2n$ teams has D-notation $D(2, 2, \ldots, 2, 1)$, thus $|I| = 1$, hence $2 \leqslant fp(CPS) \leqslant n$. We now proceed to investigate the FP of the CPS into detail. We start by recalling the following definition and insights that come from Miyashiro, Iwasaki, and Matsui (2002).

Definition 13. *Let $U \subseteq \mathcal{H}$ be a subset of patterns of order $2n$. We define, for each $r \in \{1, \ldots, 2n-1\}$:*

$$\begin{aligned} H_r(U) &= |\{h \in U \colon h_r = H\}| & A_r(U) &= |\{h \in U \colon h_r = A\}|, \\ m_r^-(U) &= \min\{|H_r(U)|, |A_r(U)|\} & m_r^+(U) &= \max\{H_r(U), A_r(U)\}, \\ M_r^-(U) &= \arg\min\{H_r(U), A_r(U)\} \end{aligned}$$

In any schedule S, the teams of a set U all have to play each other once, thus resulting in a total amount of $\binom{|U|}{2} = \frac{|U|(|U|-1)}{2}$ matches between the teams of U. We introduce functions $a(U)$ that upper bounds how many matches between teams in U can be scheduled, and $\alpha(U)$, the difference between this upper bound and the required $\binom{|U|}{2}$ scheduled matches.

$$a(U) = \sum_r m_r^-(U) \qquad\qquad \alpha(U) = a(U) - \frac{|U|(|U|-1)}{2}$$

When the upper bound $a(U)$ falls short of the required $\binom{|U|}{2}$ matches that need to be scheduled, no schedule compatible with $\mathcal{H}$ can exist.

This is captured in the following *necessary condition* on feasibility of HAP-set $\mathcal{H}$.

Condition 1 (Miyashiro, Iwasaki, and Matsui (2002))**.** *Let* $\mathcal{H} \in \mathbb{H}_{2n}$ *be a HAP-set. Then:*

$$\mathcal{H} \textit{ is feasible} \implies \alpha(U) \geqslant 0 \quad \forall U \subset [2n] \tag{3}$$

If there exists a $U \subset [2n]$*, with* $\alpha(U) < 0$*,* $\mathcal{H}$ *is infeasible.*

If $\alpha(U) = 0$, we say that U is *tight*. When U is tight, this implies that in any compatible schedule, in every round the maximum possible number of matches within U need to be scheduled; that is, in every round r, a total of $m_r^-(U)$ matches $\{t, t'\}$ with $t \in M_r^-(U)$, $t' \in U \setminus M_r^-(U)$ need to be scheduled.

In particular for the CPS, we can identify a lot of tight subsets. We identify these subsets first and then prove their tightness in Lemma 2.

Definition 14. *For* $i = 1, \ldots, \lfloor \frac{n}{2} \rfloor$*, index sets* I_i^+, I_i^- *are defined as:*

$$I_i^+ = \{1, 3, \ldots, 2i-1\} \quad I_i^- = \{2n-1, 2n-3, \ldots, 2n-(2i-1)\}$$

Subsets $U_i, U_i^c \subset$ *CPS are then defined as:*

$$U_i = \{P^A(j) \colon j \in I_i^-\} \cup \{P^H(j) \colon j \in I_i^+\}$$
$$U_i^c = \{P^H(j) \colon j \in I_i^-\} \cup \{P^A(j) \colon j \in I_i^+\}$$

And subsets $R_i \subset R$ *of the* $2n-1$ *rounds* R *are defined as:*

$$R_i = \{j \colon 2n-2i \leqslant j < 2n-1\} \cup \{2n-1\} \cup \{j \colon 0 < j < i\}$$

Notice that subsets $U_1, \ldots, U_{\lfloor \frac{n}{2} \rfloor}$ are nested, i.e. $\{P^A(2n-1), P^H(1)\} = U_1 \subset U_2 \subset \ldots \subset U_{\lfloor \frac{n}{2} \rfloor}$, and that $|U_i| = |U_i^c| = 2i$. As notation suggests, U_i and U_i^c are complementary in the sense that for every complementary pair $h, h^c \in$ CPS, $h \in U_i$ if and only if $h^c \in U_i^c$.

The sets of rounds $R_1, \ldots, R_{\lfloor \frac{n}{2} \rfloor}$ are nested, as $\{2n-1\} = R_1 \subset R_2 \subset \ldots \subset R_{\lfloor \frac{n}{2} \rfloor}$, and $|R_i| = 4i-3$. Round $r = 2n-1$ is contained in all R_i, and with each unit increase of index i four rounds are added, two to the "left" of round $2n-1$ and two to the "right" of round $2n-1$.

Now that we defined U_i, U_i^c, we can formulate Lemma 2, stating that all U_i, U_i^c are tight.

Lemma 2. U_i, U_i^c *is tight for each* $i = 1, \ldots, \lfloor \frac{n}{2} \rfloor$.

Proof. As U_i, U_i^c contain complementary HAPs, it is sufficient to show that U_i is tight to see that U_i^c is tight. We do this by determining the values $a(U)$ and $\alpha(U)$.

Notice that U_i is connected to set of rounds R_i in the sense that all patterns in U_i, have their breaks in rounds contained in R_i. Moreover, for all $r \notin R_i$, and all $h, h' \in U_i$, we have $h_r = h'_r$ - all HAPs in U_i are indistinguishable outside the rounds in R_i.

We enumerate the rounds in R_i from left to right as in Definition 14, starting with round $r = 2n-2i+1$. In this round, there is only one HAP $h \in U_i$ with a different {H,A} allocation compared to the rest of U_i, namely $h = P^A(2n-2i+1)$; this is the pattern with a break in round $2n-2i+1$. Thus, $m^-_{2n-2i+1}(U_i) = 1$. Since CPS has D-notation $D(2, 2, \ldots, 1)$, when $i > 1$, there will be no break in round $2n-2i+2$, thus $m^-_{2n-2i+1}(U_i) = m^-_{2n-2i+2}(U_i) = 1$.

In the next two rounds, $2n-2i+3, 2n-2i+4$, we see that $m^-_r(U_i) = 2$, where $P^A(2n-2i+3)$ is the additional HAP in U_i.

Continuing this way, we see $m^-_{2n-2i+j} = \lceil \frac{j}{2} \rceil$, which eventually results in $m_{2n-1}(U_i) = \frac{|U_i|}{2}$. Using symmetry, we see that $m^-_{2n-1-j} = m^-_j$ for $j = 1, 2, \ldots, 2i-2$. Thus:

$$\begin{aligned} a(U_i) &= \sum_{r=2n-2i+1}^{2i-2} m^-_r(U_i) \\ &= 2\left(1+1+2+2+\ldots+\frac{|U_i|-2}{2}+\frac{|U_i|-2}{2}\right)+\frac{|U_i|}{2} \\ &= \frac{|U_i|(|U_i|-1)}{2} \end{aligned}$$

Clearly, $\alpha(U_i) = a(U_i) - \frac{|U_i|(|U_i|-1)}{2} = 0$, thus U_i is tight. □

Example
In Tables 11 and 12, the resp. two and four HAPs of U_1 and U_2 are shown around round $2n-1$.

Round:	$2n-2$	$2n-1$	1
h_n	A	A	H
h_{n+1}	A	H	H

Table 11: Partial HAP-set of U_1

The line drawn in both tables gives the division between M^+_r and M^-_r

$2n-3$	$2n-2$	$2n-1$	1	2
A	H	A	H	A
H	A	A	H	A
H	A	H	H	A
H	A	H	A	H

Table 12: Partial HAP-set of U_2

We are now in a position to prove that, for all schedules S compatible with CPS_{2n}, exactly n matches have precisely one round in which they can be played.

Theorem 3. *Let CPS_{2n} be the CPS on $2n$ teams, with $CPS = \{h_i, h_{i+n} : 1 \leqslant i \leqslant n\}$, where $h_i = P^A(2i-1)$ and $h_{i+n} = P^H(2i-1)$ form complementary pairs.*

The fixed part $FP(CPS_{2n})$ is given by:

$$FP(CPS_{2n}) = \{\{h_i, h_{2n+1-i}\} : 1 \leqslant i \leqslant n\} \tag{4}$$

Moreover, for any feasible schedule $S = (S_r)_{1 \leqslant r \leqslant 2n-1}$ compatible with CPS_{2n}, $S \in \mathcal{S}(CPS_{2n})$, we have $FP(CPS_{2n}) = S_{2n-1}$. The order of the fixed part equals $fp(CPS_{2n}) = n$.

Proof. To prove this theorem, we first show by induction that (4) is correct. To do so, we first prove the following:

Claim 1. *In any schedule S compatible with CPS_{2n}, for $i = 1, \ldots, \lfloor \frac{n}{2} \rfloor$, the match $\{P^A(2n-(2i-1)), P^H(2i-1)\}$ is played in round $2n-1$.*

Proof: We prove this claim by induction.

INDUCTION BASE Clearly, the statement is true for $i = 1$ as the set $U_1 = \{P^A(2n-1), P^H(1)\}$ is known to be tight by Lemma 2. The only round in which these two teams can meet, is round $2n-1$, hence, they must meet in that particular round.

INDUCTION STEP Suppose the claim is true for $i = 1, 2, \ldots, k-1$ ($k-1 < \lfloor \frac{n}{2} \rfloor$). As U_k is tight by Lemma 2, we see that $h = P^A(2n-(2k-1))$ has to be scheduled against a team in U_k in every round where $h \in M_r^-(U_k)$. This is the case in all rounds $2(k-1)-1, \ldots, 2n-1$.

Notice that in round $2n-1$, by the induction hypothesis, all teams in U_{k-1} are already scheduled against teams from U_{k-1}; thus t has to be scheduled against a team in $U_k \setminus U_{k-1} = \{t, P^H(2k-1)\}$. Therefore, the match $\{P^A(2n-(2k-1)), P^H(2k-1)\}$ has to be scheduled in round $2n-1$. ■

By symmetry, it follows that a similar analysis for the teams from the sets U_i^c, $1 \leqslant i \leqslant \lfloor \frac{n}{2} \rfloor$ implies that all teams from these sets play a match that can only be played in round $2n-1$. Together with the claim this means that we have shown that $\lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{2} \rfloor$ matches can only be played in round $2n-1$.

If n is odd, for two teams we have not yet shown that there is a fixed match between them: $P^A(n)$ and $P^H(n)$ are not contained in any set U_i. However, with these two teams having to play *some* match in round $2n-1$, and as all other teams are already paired, they have to play each other - fixing their match to round $2n-1$.

It is remarkable that a match between teams with complementary HAPs, can still be fixed to a single round for any compatible schedule.

Thus, in any feasible schedule compatible with the CPS, the entire round $2n-1$ consists of matches that can only be scheduled in this round.

Finally, we point out that, due to Observation 3, in any feasible schedule compatible with CPS_{2n}, the matches in round r can be interchanged with the matches in round $r+1$ ($r-1$) if r is odd (even), for $r = 1, \ldots, 2n-2$. This proves the theorem. □

The analysis of the fixed part of the CPS is not only valid for the CPS. We can use the knowledge of fixed matches on tight, nested subsets like U_i, in any HAP-set that contains such sets. We can find these sets in a single break HAP-set, by using its D-notation. As stated earlier, every 1 in the D-notation indicates 2 fixed matches. But we can say more, as every sub-sequence 212 indicates 4 fixed matches, 22122 indicates 6 fixed matches, and so forth. To formalize this knowledge, we introduce the following concept of nests:

Definition 15. *Let $\mathcal{H} \in \mathbb{H}_{2n,1}$ be a feasible HAP-set, with $\mathcal{H} \neq$ CPS, and D-notation $D = d_1 \ldots d_n$. An index i indicates a* center *at i if $d_i = 1$. The* nest *$\mathcal{N}(i)$ around i is defined as the maximum set of indices $\mathcal{N}(i) =$*

$\{i-k, \ldots, i+k\}$ *for which* $d_j = 2$ *for all* $j \in \mathcal{N}(i) \setminus i$. *A nest* $\mathcal{N}(i) = \{i-k, \ldots, i+k\}$ *is said to be* nest of order k - *the order of the nest around* i *is given by* $N_i = k$.

The fixed part $FP(\mathcal{N}(i))$ *of a nest is defined as:*

$$FP(\mathcal{N}(i)) \colon = \{\{h_j, h_{j'}\} \colon \{h_j, h_{j'}\} \in FP(\mathcal{H}), \quad j, j' \in \mathcal{N}(i)\} \tag{5}$$

$$= FP(\mathcal{H}) \cap (\mathcal{N}(i) \times \mathcal{N}(i)) \tag{6}$$

And the order of the fixed part of $\mathcal{N}(i)$ *is subsequently given by* $fp(\mathcal{N}(i)) = |FP(\mathcal{N}(i))|$.

The definition of nests is not ambiguous as $d_1 \geqslant 3$ *when* $\mathcal{H}$ *not CPS.*

Any $\mathcal{H} \in \mathbb{H}_{2n,1}$ that is not the CPS will have at least two nests. It is also important to remark that for two different centers i, j with nests $\mathcal{N}(i), \mathcal{N}(j)$, we have $\mathcal{N}(i) \cap \mathcal{N}(j) = \emptyset$. Any single break HAP-set that violates this property, is infeasible as it violates Condition 1.

Recall that the center of the nest i, where $d_i = 1$, indicates a round r_i where the match $\{h_i, h_{i+1}\}$ has to be scheduled, as HAP h_{i+1} has a break in round $r_{i+1} = r_i + d_i = r_i + 1$ - we showed this earlier to get the lower bound in Lemma 1. Using the nests around such center i, we can improve on this lower bound for $fp(\mathcal{H})$ for any $\mathcal{H} \in \mathbb{H}_{2n,1}$.

Theorem 4. *Let* $\mathcal{H} \in \mathbb{H}_{2n,1} \setminus CPS_{2n}$ *with D-notation* $D(\mathcal{H}) = d_1 \ldots d_n$. *Let* $I = \{i : d_i = 1\}$ *the set of indices where the break-gap equals* 1, *where each* i *hence indicates the center of a nest* $\mathcal{N}(i)$ *of order* N_i.

Then the following statements are true:

1. $\{\{h_{i-j}, h_{i+1+j}\} \colon 0 \leqslant j \leqslant N_i\} \subset FP(\mathcal{N}(i))$.
2. $fp(\mathcal{N}(i)) \geqslant 1 + N_i$.
3. $fp(\mathcal{H}) \geqslant 2 \sum_{i \in I} (1 + N_i)$.

Proof. Clearly, 1 implies Item 2, as the subset that is part of of $FP(\mathcal{N}(i))$ is of size $1 + N_i$.

Also, 2 implies 3, as the order of the fixed part of HAP-set $\mathcal{H}$ is at least as big as the sum of the order of disjoint subsets of the fixed part. Moreover, for each fixed match between teams in $\mathcal{N}(i)$, the match between its complementary pairs must be fixed as well, which explains the factor 2 in front of the summation term.

In Section 2.5, HAP-sets are observed for which the lower bound in Theorem 4 is not tight.

The only thing left to prove, is that 1 is correct. Let $\mathcal{N}(i)$ be a nest of order $N_i = k$ centered around i. As $d_{i-k} = \cdots = d_{i-1} = 2$ and $d_{i+1} = \cdots = d_{i+k} = 2$, the set $\mathcal{N}(i)$ is tight. This implies that, by the same reasoning used in the proof of Theorem 3, that matches $\{h_{i-j}, h_{i+1+j}\}$ must be scheduled in round r_i - and are thus fixed. This finishes the proof. □

Using Theorem 4, we can refine some lower bounds. For instance, the HAP-set $\mathcal{H}$ with $D = 312\ldots21$ still has $FP(\mathcal{H}) \geqslant 4$, as both nests are of order 0. On the other hand, HAP-set $\mathcal{H}'$ with $D = 312212$ has two nests, one of order 0 and one of order 1. Thus, $FP(\mathcal{H}') \geqslant 6$ - an improvement on the original lower bound given by Lemma 1, which was 4.

2.4.3 Measure 3: the spread

The fp of HAP-set $\mathcal{H}$ indicates how many matches are fixed to specific rounds, in any compatible schedule S. However, as we have seen in Observation 3, the matches in many if not most of the rounds, can be changed with at least one other round, making them not fixed. Hence, for most of the matches in a schedule, $fp(\mathcal{H})$ does not indicate anything, except that they have two rounds in which they can at least be scheduled.

We want a measure that says something about the flexibility of all matches in a schedule, instead of the lack of flexibility of some. The *spread* of a match $\{i, j\}$ in HAP-set $\mathcal{H}$, is the number of different rounds it can be scheduled in. The spread of HAP-set $\mathcal{H}$, is given as the sum of the spread of all matches. A formal definition is given below.

Definition 16. *Given a HAP-set $\mathcal{H}$ and a match $\{h, h'\}$, the* spread *of this match is given by:*

$$spread(h, h') = \#\{r : \exists S \in \mathcal{S}(\mathcal{H}) \text{ } s.t. \text{ } \{h, h'\} \in S_r\} \quad \forall \{h, h'\} \subset \mathcal{H}$$

The spread of HAP-set $\mathcal{H}$ is given by:

$$spread(\mathcal{H}) = \sum_{\{h,h'\} \subset \mathcal{H}} spread(h, h') \tag{7}$$

A higher value of the spread, indicates more flexibility in scheduling individual matches. Any match $\{h, h'\}$ that is fixed, has a spread equal to 1. It is important to remark here that $\text{spread}(h, h')$ is not necessarily equal to the rounds where they have oppositie HAPs, $\#\{r : h_r \neq h'_r\}$. As we've seen with the fixed part of the CPS, even complementary HAPs can have a spread equal to 1.

The HAP-set from Table 5 has a spread of 84 (16 matches can be played in 4 rounds, 8 matches have 2 possible rounds, and the fixed part consists of 4 matches, all with spread 1).

It is not immediately intuitive which values of the spread indicate particularly high flexibility and which indicate low flexibility, higher spreads naturally points to a higher flexibility. Some rather trivial bounds can be obtained. For any feasible HAP-set $\mathcal{H}$ on $2n$ teams, we have:

In Chapter 3 schedules are constructed that attain the upper bound.

1. A lower bound $\text{spread}(\mathcal{H}) \geqslant \frac{2n(2n-1)}{2} \approx 2n^2$.
2. An upper bound $\text{spread}(\mathcal{H}) \leqslant n^2(2n-1) \approx 2n^3$.

The lower bound is correct, as in any schedule S, $\frac{2n(2n-1)}{2}$ matches are scheduled, all having a spread of at least 1. The upper bound is a result from the fact that in every round, n teams have a home assignment, and n teams away. Thus, in any round, at most n^2 different matches can be scheduled. As there are $2n-1$ rounds, this leads to the upper bound.

It is possible to refine the bounds of HAP-sets by studying their structure. In particular, we tighten the upper bound for the CPS_{2n} in the following theorem.

Theorem 5. *The spread of the CPS on $2n$ teams is bounded from above by:*

$$\textit{spread}\left(\text{CPS}_{2n}\right) \leqslant \frac{n}{6}\left(10n^2 - 9n + 11\right) - \lceil \frac{n}{2} \rceil \tag{8}$$

Proof. To prove this, we first derive an upper bound for the spread of each individual match; then, we sum these upper bounds to obtain the result.

As we identify each team with a pattern $P^H(2i+1)$ or $P^A(2i+1)$ for $0 \leqslant i, j \leqslant 2n-1$, we can distinguish three types of matches:

1. Type 1-H are matches of the form

$$\{P^H(2i+1), P^H(2j+1)\} \;\; 0 \leqslant i < j \leqslant n-1. \tag{9}$$

2. Type 1-A are matches of the form

$$\{P^A(2i+1), P^A(2j+1)\} \;\; 0 \leqslant i < j \leqslant n-1. \tag{10}$$

3. Type 2 are matches of the form:

$$\{P^H(2i+1), P^A(2j+1)\} \;\; i, j \in \{0, \ldots, n-1\}. \tag{11}$$

Since the CPS is complementary, any result obtained regarding the spread of a match of Type 1-H, is valid for Type 1-A as well, and vice versa. We can partition the spread of the CPS as a sum of spreads of each of the three type of matches:

$$\begin{aligned}\text{spread}\left(\text{CPS}_{2n}\right) = &\sum_{i<j} \text{spread}(P^H(2i+1), P^H(2j+1)) + \\ &\sum_{i<j} \text{spread}(P^A(2i+1), P^A(2j+1)) + \\ &\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \text{spread}(P^H(2i+1), P^A(2j+1)).\end{aligned}$$

We proceed to separately upper bound the spread of the matches per type. We start by bounding the 'easy' Types, 1-H and 1-A.

Consider a match of Type 1-H. Given two teams $P^H(2i+1)$, $P^H(2j+1)$ with $0 \leqslant i < j \leqslant n-1$, their HAPs differ in the rounds $R_{i,j} := [2i+1, 2j]$. As two teams can only be scheduled in rounds where they have different HAPs, it follows immediately that, for each $0 \leqslant i < j \leqslant n-1$

$$\text{spread}(P^H(2i+1), P^H(2j+1)) \leqslant |R_{i,j}| = 2(j-i) \tag{12}$$

Using (12), we are able to bound the sum of the spreads of the corresponding matches:

$$\begin{aligned}
&\sum_{i<j} \text{spread}(P^H(2i+1), P^H(2j+1)) \leqslant \\
&\sum_{j=1}^{n} \sum_{i=0}^{j-1} 2(j-i) = \sum_{j=1}^{n-1} j(j+1) \\
&= \frac{1}{6}(n-1)n(2n-1) + \frac{n(n-1)}{2} \\
&= \frac{1}{6}((n-1)n(2n-1) + 3n(n-1)) =: Z_{1,2n}.
\end{aligned} \tag{13}$$

By symmetry, the spread of the matches of Type 1-A are also upper bounded by $Z_{1,2n}$.

So far we did nothing new compared to the trivial upper bound given earlier. For matches of Type 2, we will see that we can find more restrictions on the possible rounds in which matches can be scheduled.

For any match of Type 2, the two teams $P^H(2i+1), P^A(2j+1)$ have different HAPs in rounds $R^c_{i,j} := [1, 2\min(i,j)] \cup [2\max(i,j)+1, 2n-1]$. Hence, for each $i, j = 0, \ldots, n-1$:

Combining (12) and (14) results in the trivial upper bound.

$$\text{spread}(P^H(2i+1), P^A(2j+1)) \leqslant |R^c_{i,j}| \tag{14}$$

To improve on this bound, we analyse the matches op Type 2 $\{h, h'\} = \{P^H(2i+1), P^A(2j+1)\}$ by distinguishing three cases. We assume $i \leqslant j$ - the analysis is similar when $i \geqslant j$.

CASE 1 $\quad i = n-j-1$.
From the proof of Theorem 3, we see that any match $\{h, h'\} = \{P^H(2i+1), P^A(2j+1)\}$ can be scheduled in round $2n-1$ if and only if $i = n-j-1$; thus, $\{h, h'\}$ is fixed to round $2n-1$ and the the spread of this match is equal to 1.

$$\begin{aligned}
\text{spread}(P^H(2i+1), P^A(2(n-i)+1)) &= 1 \\
\forall 0 \leqslant i &\leqslant n-1
\end{aligned} \tag{15}$$

CASE 2 $i < n - j - 1$.
Recall sets U_i from Definition 14, with $h = P^H(2i+1) \in U_i$ and U_i tight by Lemma 2. This means that in rounds $1, \ldots, 2i$, h has to be scheduled against $2i$ teams from U_i. As $h' = P^A(2j+1) \in U_i$ only if $n - j - 1 \leqslant i$, and we assumed the negation, we have $h' \notin U_i$. Therefore, $\{h, h'\}$ cannot be scheduled in rounds $1, \ldots, 2i$. As we also know that this match cannot be scheduled in round $2n-1$, the only rounds left where $\{h, h'\}$ can be scheduled are $2j+1, \ldots, 2n-2$. Hence:

$$\begin{aligned} \text{spread}(P^H(2i+1), P^A(2j+1)) \leqslant 2(n-j-1) \\ i, j \in \{0, \ldots, n-1\} \text{ with } i < n-j-1. \end{aligned} \tag{16}$$

CASE 3 $i > n - j - 1$.
We apply a similar argument as in Case 2. Only this time, we observe that U_{n-j-1} is tight, which means that $h' = P^A(2j+1)$ can only play teams from U_{n-j-1} in rounds $2j+1, \ldots, 2n-2$, and $h = P^H(2i+1) \notin U_{n-j-1}$ as we assumed $i > n-j-1$. Therefore, $\{h, h'\}$ cannot be scheduled in $2j+1, \ldots, 2n-1$, and must be scheduled in one of the rounds $1, \ldots, 2i$. Hence:

$$\begin{aligned} \text{spread}(P^H(2i+1), P^A(2j+1)) \leqslant 2i \\ i, j \in \{0, \ldots, n-1\} \text{ with } i > n-j-1. \end{aligned} \tag{17}$$

We combine all these bounds to get to one upper bound for the sum of the spreads of matches of Type 2. Let $\lfloor \frac{n}{2} \rfloor = k$, so $n = 2k$ if n is even, and $n = 2k+1$ if n is odd. Implementing the right-hand sides of the expressions (15), (16), and (17) we find:

$$
\begin{aligned}
&\sum_{i=0}^{k-1}\sum_{j=0}^{n-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) = \\
&= \sum_{i=0}^{k-1}\sum_{j=i}^{n-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) + \\
&\sum_{i=0}^{k-1}\sum_{j=0}^{i-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) = \\
&= \sum_{i=0}^{k-1} \mathrm{spread}(P^H(2i+1), P^A(2(n-i-1)+1)) + \\
&\sum_{i=0}^{k-1}\sum_{j:j<n-i-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) + \\
&\sum_{i=0}^{k-1}\sum_{j:j>n-i-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) + \\
&\sum_{i=0}^{k-1}\sum_{j=0}^{i-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) \leqslant \\
&\leqslant \sum_{i=0}^{k-1} [1 + \sum_{j=i}^{n-i-2} 2(n-j-1) + \left(\sum_{j=n-i}^{n-1} 2i\right) + \sum_{j=0}^{i-1} 2(n-i-1)] = \\
&= \sum_{i=0}^{k-1} [1 + n(n-2i-1) + 2i \cdot i + 2i(n-i-1)] = \\
&= \sum_{i=0}^{k-1} [1 + n^2 - n - 2i] \\
&= k(n^2 - n + 1) - k(k-1) =: Z_{2,2n}.
\end{aligned}
$$

The value $Z_{2,2n}$ is the result of the sum of the upper bounds of the spreads corresponding to the matches $(P^H(2i+1), P^A(2j+1))$, where $i < k$. By symmetry, $Z_{2,2n}$ is also equal to the sum of the spreads corresponding to matches $(P^H(2i+1), P^A(2j+1))$ where $i \geqslant n-k$ (recall that $k = \lfloor \frac{n}{2} \rfloor$). If n is even, this would mean $k = n-k$ and the sum over all upper bounds is equal to $Z_{2,2n}$, leading to:

$$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \mathrm{spread}(P^H(2i+1), P^A(2j+1)) \leqslant 2Z_{2,2n}. \tag{18}$$

If n is odd, we still need to bound the spread of the matches $\{P^H(2k+1), P^A(2j+1)\}$ where $j \geqslant k$, and $\{P^A(2k+1), P^H(2j+1)\}$ where $j > k$. It is not difficult to see that the sum of the spread of these matches is bounded from above by $Z_{odd} = 2k \cdot k + 1 + 2k \cdot k = 4k^2 + 1$. Thus we get:

$$\sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \text{spread}(P^H(2i+1), P^A(2j+1)) \leqslant 2Z_{2,2n} + Z_{odd}. \tag{19}$$

For even n, combining the spreads of matches of all types results in an upper bound of:

$$\begin{aligned}
\text{spread}\,(\text{CPS}_{2n}) &\leqslant 2Z_{1,2n} + 2Z_{2,2n} \\
&= \frac{1}{6}\left(2(n-1)n(2n-1) + 6n(n-1)\right) + 2k(n^2 - n + 1) - 2k(k-1) \\
&= \frac{1}{6}(4n^3 - 6n^2 + 2n + 6n^2 - 6n + 6n^3 - 6n^2 + 6n - 6n(k-1)) \\
&= \frac{1}{6}(10n^3 - 6n^2 + 8n - 6nk) \\
&= \frac{1}{6}(10n^3 - 9n^2 + 8n).
\end{aligned}$$

For odd n, combining the spreads of matches of all types results in an upper bound of:

$$\begin{aligned}
&\text{spread}\,(\text{CPS}_{2n}) \leqslant 2Z_{1,2n} + 2Z_{2,2n} + 4k^2 + 1 = \\
&= \frac{1}{6}(2(n-1)n(2n-1) + 6n(n-1)) + 2k(n^2 - n + 1) \\
&- 2k(k-1) + 4k^2 + 1 \\
&= \frac{1}{6}(2(n-1)n(2n-1) + 6n(n-1) + 6(n-1)(n^2 - n + 1) \\
&+ 12k(k+1) + 6) \\
&= \frac{1}{6}(10n^3 - 12n^2 + 8n + 6(n-1)(k+1) + 6) \\
&= \frac{1}{6}(10n^3 - 6n^2 + 8n - 6(nk - k + n)) \\
&= \frac{1}{6}(10n^3 - 9n^2 + 8n + 3).
\end{aligned}$$

Some rewriting can be done to combine the two outcomes into one. When n is even, $3n = 6\lceil \frac{n}{2} \rceil$, and when n is odd, $3n = 6\lceil \frac{n}{2} \rceil - 3$. Thus, for any n, we see that:

$$\text{spread}\,(\text{CPS}_{2n}) \leqslant \frac{1}{6}(10n^3 - 9n^2 + 11n) - \lceil \frac{n}{2} \rceil.$$

This finishes the proof. □

The improved upper bound we get from Theorem 5 tells us that the $\text{spread}(\text{CPS}_{2n}) \approx \frac{5}{3}n^3$. The old bound was $\text{spread}(\text{CPS}_{2n}) \approx 2n^3$. This implies that approximately for one sixth of the rounds where

h, h' have an opposite Home/Away-assignment, the match $\{h, h'\}$ cannot be scheduled.

An interesting question would be how 'bad' a spread can get, i.e., how small can q be in $spread(\mathcal{H}) \approx q \cdot n^3$.

Although we only have an upper bound for the value of the spread of the CPS, we can calculate the actual values using the integer programming approach as described in the upcoming Section 2.5. In Table 15, we explicitly list the values of the spread of CPS_{2n} as calculated and its upper bound for values of $2n$ up to 22, and note that this bound is tight. In fact, we conjecture that this bound is tight for any number of teams $2n$.

$2n$	4	6	8	10	12	14	16	18	20	22
Bound	10	35	88	177	314	507	768	1105	1530	2051
spread(CPS)	10	35	88	177	314	507	768	1105	1530	2051

Table 15: Values for the spread and the upper bound for CPS with $2n$ teams

Conjecture 1.

$$\text{spread}\left(CPS_{2n}\right) = \frac{n}{6}\left(10n^2 - 9n + 11\right) - \lceil \frac{n}{2} \rceil.$$

2.5 COMPUTING THE MEASURES

Indicators like the *width*, *fixed part* and *spread* are only of value, if we can compare the measures for different HAP-sets. To do this, we introduce integer programs that, when solved for any particular HAP-set, give the value of the wanted measure.

In Section 2.5.1, we introduce these IPs and in Section 2.5.2 we present computational results for single-break HAP-sets.

2.5.1 Exact formulations

We build the IP in a modular way; each measure has its own perks and constraints, but they all come down to solving a SRR, either once or multiple times.

We want to find schedules on $\mathcal{T} = \{1, \ldots, 2n\}$ teams, that each have a HAP from HAP-set $\mathcal{H} = \{h^i : i = 1, \ldots, 2n\}$. The elements of the HAP-set $\mathcal{H}$ are $h^i \in \{0,1\}^{2n-1}$, thus $h^i = (h^i_r)_{1 \leqslant r \leqslant 2n-1}$, where $h^i_r = 1$ indicates a Home game in round r, and 0 otherwise. We assume that i plays according to HAP h^i. We use R to denote the set of rounds $R = \{1, 2, ..., 2n-1\}$, indexed by r.

We let W denote the number of schedules we try to create simultaneous, and $w \in \{1, \ldots, W\}$ be the index of the schedule. The program has binary decision variables $x^w_{i,j,r}$, indicating whether i plays $j \neq i$ in round r in schedule w.

The following constraints are all that are needed to find a SRR schedule S compatible with $\mathcal{H}$.

$$x^w_{i,j,r} = x^w_{j,i,r} \qquad \forall\{i,j\} \subset \mathcal{T}\ \forall r \in R\ \forall w \in \{1,\dots,W\} \tag{20}$$

$$\sum_{r \in R} x^w_{i,j,r} = 1 \qquad \forall\{i,j\} \subset \mathcal{T}\ w \in \{1,\dots,W\} \tag{21}$$

$$\sum_{j \in \mathcal{T}\setminus i} x^w_{i,j,r} = 1 \qquad \forall i \in \mathcal{T}\ r \in R\ w \in \{1,\dots,W\} \tag{22}$$

$$x^w_{i,j,r} \leqslant |h^i_r - h^j_r| \qquad \forall\{i,j\} \in \mathcal{T}, r \in R, w \in \{1,\dots,W\} \tag{23}$$

$$x^w_{i,j,r} \in \{0,1\} \qquad \forall\{i,j\} \in \mathcal{T}, r \in R, w \in \{1,\dots,W\} \tag{24}$$

With (20), we fix the notational issue that both $x_{i,j,r}$ and $x_{j,i,r}$ refer to the match $\{i,j\}$ in round r. This is more convenient later on. By (21), we ensure that every match between i and $j \neq i$ is scheduled once. By (22) we get that every team has a match scheduled in every round. To make the schedule compatible with $\mathcal{H}$, we have (23), that upper bounds the value of $x^w_{i,j,r}$ to 0 when $h^i_r = h^j_r$ - thus $x^w_{i,j,r} = 1$ is only possible when $h^i_r \neq h^j_r$.

We refer to (20)-(24) as the *basic IP*.

WIDTH To determine whether a HAP-set $\mathcal{H}$ has width at least W, we need to add the following constraint to the basic IP, namely:

$$\sum_{w} x^w_{i,j,r} \leqslant 1 \qquad \forall\{i,j\} \in \mathcal{T}\ \forall r \in R \tag{25}$$

For any of the W schedules to be orthogonal, none of the matches can be scheduled in the same round for different w, w', which is captured by (25).

Notice that solving this IP for a given value of W, can only have two possible outcomes: either $\text{width}(\mathcal{H}) \geqslant W$ or $\text{width}(\mathcal{H}) < W$, depending whether the IP has a solution or is infeasible. In order to determine the width of $\mathcal{H}$, one has to search for a W where the IP is feasible, whereas it is unfeasible for $W+1$.

FIXED PART To determine the fixed part, we use the basic IP, with $W = 2$. Then, for each match $\{i,j\}$ we run this IP with the following constraint added:

$$\sum_{w} x^w_{i,j,r} \leqslant 1 \qquad \forall r \in R \tag{26}$$

This constraint makes sure that the two different schedules $w = 1,2$, have a different round in which $\{i,j\}$ is scheduled.

This IP has to be solved for all pairs of teams $\{i,j\}$ - for each pair, if the IP is feasible, this implies that $\{i,j\} \notin \text{FP}(\mathcal{H})$ is not an element of the fixed part, whereas infeasibility proves that $\{i,j\}$ is an an element of the fixed part.

This IP has to be run $\mathcal{O}(n^2)$ times. Although in practice, the solving time per IP is limited to milliseconds, one could choose to take a somewhat smarter approach. When solving the IP for a particular match $\{i,j\}$, one could take the feasible solution - if found - containing two schedules, and check which matches are scheduled in different rounds. All these matches are excluded from being elements of the fixed part, and it is of no use to run the algorithm again for these particular matches.

In fact, a more direct approach can be executed. Fix W and instead of (26), add the following constraint:

$$\sum_{w=1}^{W} x_{i,j,r}^{w} \leqslant 1 + (W-1)\cdot y_{i,j,r} \qquad \forall \{i,j\} \in \mathcal{T}\ \forall r \in R \tag{27}$$

$$y_{i,j,r} \in \{0,1\} \qquad \forall \{i,j\} \in \mathcal{T}\ \forall r \in R$$

Contrary to just trying to find a feasible solution to the IP, add the following objective:

$$\min \sum_{\{i,j\}\in\mathcal{T}} \sum_{r\in R} y_{i,j,r} \tag{28}$$

The newly introduced binary variable $y_{i,j,r}$ serves as a penalty - by constraint (27), in any feasible solution it equals 1 if in all W schedules, the match $\{i,j\}$ is scheduled in round r. We denote the outcome of this IP by $fp_W(\mathcal{H})$. As we try to minimize the sum of the penalties, we see that:

$$fp_2(\mathcal{H}) \geqslant fp_3(\mathcal{H}) \geqslant fp_4(\mathcal{H}) \geqslant \cdots fp_{n(2n-1)}(\mathcal{H}) \qquad \forall \mathcal{H}$$

and

$$\lim_{W\to\binom{|\mathcal{T}|}{2}} fp_W(\mathcal{H}) = fp(\mathcal{H}).$$

This might not look like much of a win compared to the original formulation. Instead of solving several IP's that creates 2 schedules $\mathcal{O}(n^2)$ times, we need to solve an IP that creates $W = \mathcal{O}(n^2)$ schedules simultaneously. However, the convergence of the $fp_W(\mathcal{H}) \to fp(\mathcal{H})$ goes quick - all HAP-sets for which we calculated the order of the fixed part, had $fp_2(\mathcal{H}) = fp(\mathcal{H})$. Besides that, we also remark the following:

That $fp_2(\mathcal{H}) = fp(\mathcal{H})$ does not need to be true in general.

Observation 4. *Let $\mathcal{H}$ be a HAP-set on $2n$ teams and let $fp_W(\mathcal{H})$ denote the solution of the IP that is the basic IP together with (27) and objective (28), for a particular value W. Then:*

$$fp_W(\mathcal{H}) = fp_{W+1}(\mathcal{H}) \iff fp_W(\mathcal{H}) = fp(\mathcal{H}) \tag{29}$$

Indeed, suppose (29) is not true. That implies that given a solution S^* to the IP on W schedules that attains $\text{fp}_W(\mathcal{H})$, there is a $y_{i,j,r} = 1$ for which $\{i,j\} \notin \text{FP}(\mathcal{H})$. This means there must be a schedule $S = (S_r)$ such that $\{i,j\} \notin S_r$. As solution to the IP on $W+1$ schedules, one could simply take S^* and add S as $W+1$-th schedule, to construct a feasible solution that has an objective value strictly less than $\text{fp}_W(\mathcal{H})$.

SPREAD To compute the value of the spread of HAP-set $\mathcal{H}$, we again start with the basic IP. Set $W = 1$. For each match $\{i,j\}$ and round r, run the IP with the following added constraint:

$$x^1_{i,j,r} = 1 \tag{30}$$

The spread equals the total number of IPs that are feasible - roughly $\mathcal{O}(n^3)$ of such IPs need to be solved, one for each combination of matches and rounds.

The IP to calculate the spread can also be applied to calculate the FP($\mathcal{H}$). If for a match $\{i,j\}$ the IP is infeasible with every round except for one, $\{i,j\} \in FP(\mathcal{H})$.

Obviously, this procedure can be sped up in the same way as done for the fixed part. Namely, after finding a feasible solution to the IP for a specific match $\{i,j\}$ in round r, we have a schedule in which $n(2n-1)$ matches are scheduled in rounds. For all these match-round combinations, we know there will be a feasible schedule, so we do not need to solve the IP anymore.

An alternative approach can also be taken, where only a single IP needs to be solved to calculate the spread, and this is similar to the alternative IP to calculate the order of the fixed part. Take the basic IP for a value W and add constraints:

$$z_{i,j,r} \leqslant \sum_{w=1}^{W} x^w_{i,j,r} \qquad \forall\{i,j\} \in \mathcal{T}\ \forall r \in R \tag{31}$$

$$z_{i,j,r} \in \{0,1\} \qquad \forall\{i,j\} \in \mathcal{T}\ \forall r \in R$$

Together with objective:

$$\max \sum_{\{i,j\}\in\mathcal{T}} \sum_{r \in R} z_{i,j,r} \tag{32}$$

Variables $z_{i,j,r}$ can only attain 1 if there is a $w \in \{1,\dots,W\}$, for which $x^w_{i,j,r} = 1$, implying that match $\{i,j\}$ can be scheduled in round r.

When $\text{spread}_W(\mathcal{H})$ is the value of this IP for a particular W on HAP-set $\mathcal{H}$ - similar to $\text{fp}_W(\mathcal{H})$ - we get that:

$$\text{spread}_2(\mathcal{H}) \geqslant \text{spread}_3(\mathcal{H}) \geqslant \cdots \text{spread}_{n^2(2n-1)}(\mathcal{H}) \qquad \forall \mathcal{H}$$

And

$$\lim_{W \to n^2(2n-1)} \text{spread}_W(\mathcal{H}) = \text{spread}(\mathcal{H}) \qquad \forall \mathcal{H}$$

Contrary to the convergence of the the fixed part $\mathrm{fp}_W \to \mathrm{fp}$, there seems to be no clear value W^* for which $\mathrm{spread}_{W^*}(\mathcal{H}) = \mathrm{spread}(\mathcal{H})$.

2.5.2 Results

We have used the formulations in Section 2.5.1 to compute the spread and the fixed part for all D-notations corresponding to feasible single-break HAP-sets of $2n \leqslant 16$. Recall that each such D-notation represents a family of HAP-sets with the same value for the flexibility measures. All computations were done using IBM Ilog Cplex 12.8 on a Dell Latitude 7490 with Intel Core i7-8650 @ 1.9 GHz and 16 GB RAM. The basic model together with the relevant simple constraints, were all solved in less than one second; all flexibility measures were computed in less than 1 minute for up to 16 teams. Recall that we know that the width equals one for all single-break HAP-sets from Section 2.4.1.

The results are summarized in Table 16. For 4 and 6 teams, only 1 single-break HAP-set exists, the Canonical Pattern Set. This is no longer the case when the number of teams is 8 or more.

On 8 teams, there are two elements in $\mathbb{H}_{8,1}$. The popular canonical pattern set is the best choice with respect to the spread and scores as good as the other single-break HAP-set when we focus on the fixed part.

For 10 to 16 teams, however, the canonical pattern set is clearly dominated by another type of schedule, all indicated by the D-notation 312...21. Moreover, for all values of $2n$ that we considered, this single-break HAP-set shows the highest spread as well as the lowest fixed part.

We also point out the following: if some match $\{P(r), P(s)\}$ in a single break HAP-set is in the fixed part, then its complementary match, given by $\{P^c(r), P^c(s)\}$, is also in the fixed part. Ergo, the set of matches in the FP can be seen as pairs. Interestingly, for $2n \in \{10, 14\}$, there exist multiple feasible single-break HAP-sets with an odd number of matches in the FP. This can only happen when the complement of a match is the match itself, as in the match is of the type $\{P^H(r), P^A(r)\}$. Thus, this implies that for such HAP-sets there is a match between two teams with complementary HAPs (and hence could, seemingly, play in each round), that can only be played in one round.

2n	D-notation	spread	fp
4	21 (CPS)	10	2
6	221 (CPS)	35	3
8	2221 (CPS)	88	4
	3121	76	4
10	22221 (CPS)	177	5
	31221	161	4
12	222221 (CPS)	314	6
	312221	332	4
	321221	266	6
	313121	254	6
14	2222221 (CPS)	507	7
	3122221	557	4
	3212221	471	6
	3131221	439	6
	3213121	423	7
16	22222221 (CPS)	768	8
	31222122	686	8
	31222212	796	6
	31222221	864	4
	32122212	632	8
	31223121	672	8
	31231221	690	6
	31312212	614	8
	31312221	838	6
	32123121	684	8
	31313121	640	8
	41213121	552	8

Table 16: Flexibility measures for single-break HAP-sets, up to 16 teams

2.6 CONCLUSION AND OUTLOOK

In a first-break-then-schedule approach, it is important to have freedom to schedule the individual matches after the Home-Away Patterns have been chosen. We have proposed three measures that indicate the amount of freedom associated with a particular HAP-set, namely width, FP and spread. We have given some theoretic insights into how the most popular HAP-set, the CPS, fares on these measures. Using an integer programming approach, we also determined the values of the FP and the spread of all single-break HAP-sets up to order 16.

From a practical point of view, it is interesting to see that when the number of teams exceeds 10, the CPS is not the most flexible HAP-set by any measure. When intending on scheduling competitions with a single-break HAP-set, we would advise to go for the $32 \ldots 21$ HAP-set instead - it has higher spread and a much smaller fixed part.

From a theoretical point of view, many questions can still be asked and perhaps even answered. With a similar approach as given in the proof of Theorem 5, it is doable to find tighter upper bounds for different generic pattern sets. One can wonder whether upper bounds constructed in such a way, are all tight. This would however partly coincide with finding a general way of scheduling feasible single break HAP-sets, for which no generic method is known to date.

Also, some other quirks are nice to figure out. In Chapter 3, we search for HAP-sets with very large width. One could also try to construct HAP-sets with a relatively large Fixed Part. So far, all single break HAPs we have encountered have FP $\leqslant n$ - it might be that this bound is tight, which would be nice to prove.

3 MAXIMUM ORTHOGONAL SCHEDULES

When scheduling a competition, choices have to be made. As seen in the previous chapter, the choices made in the beginning, can result in unwanted limitations and even infeasibility later on. What if you schedule a tournament among some clubs, have all clubs reserve their stadiums, and after publishing the full schedule, it turns out that one match is wrongly scheduled?

It would be nice, if the individual matches can be rearranged in a way, such that the Home/Away-Pattern can be maintained for all clubs, and this 'poisoned' match is moved to another round. However, as a scheduler, one can only guarantee this in advance if it is possible for every possible match. And to be able to guarantee this, the HAP-set that is used must have a *width* of at least two. But can we even guarantee such a thing, for any number of teams?

And if we can guarantee that two orthogonal schedules exist on the same HAP-set, why stop there? Can we do more, say, construct HAP-sets with 3 orthogonal schedules that are compatible? Or 4? And how should these HAP-sets and schedules look like?

In this chapter we look at HAP-sets and schedules in a more theoretical way. We examine orthogonal schedules, where all matches are played in different rounds compared to each other, which relates it to the popular research topic of finding orthogonal Latin squares. We construct HAP-sets for which we know that two such schedules exist, and even better ones when the number of teams is a power of 2.

This chapter is based on joint work with Mehmet Akif Yıldız, Jop Briët, Viresh Patel and Frits Spieksma (Lambers, Yıldız, et al., 2022).

3.1 INTRODUCTION

A basic and popular format for a competition is the well-known Single Round Robin (SRR). Given a set of teams $\mathcal{T}$, with $|\mathcal{T}| = 2n$ even, an SRR has every pair of teams play each other once, and when the SRR is tight, this is done in $|\mathcal{T}| - 1$ rounds, such that each team plays once in each round. Typically, in each match, one team plays at home, while the other team plays away - abbreviated with H, A respectively.

When faced with the task of deciding upon the fixtures, i.e., to come up with a *schedule* that specifies which match is played in which round, and which team plays home and away in each match, various strategies have been described in literature; we simply refer to survey papers describing these, see Rasmussen and Trick (2008) and Kendall, Knust, et al. (2010). We also mention Knust and Lücking (2009) as an important source of references.

In this chapter we focus on a question that is relevant for a set of strategies that are known as First-Break-Then-Schedule (FBTS). These are strategies that follow a 2-phased approach: in the first phase, decide upon the home/away designation of each team in each round (thereby specifying a Home/Away-Pattern (HAP) for each team). In the second phase, schedule all the matches in a way that is compatible with these HAPs; see Schreuder (1992) and Russell and Leung (1994) for early references. A key question is to what extent specifying the HAP-set in Step 1 impacts the set of possible schedules in Step 2. Or in other words, what is the diversity of schedules compatible with a given HAP-set?

This issue has been investigated in the previous chapter (see also Lambers, Goossens, and Spieksma (2022)) where various measures for the flexibility of a HAP-set are proposed and analyzed. One such measure is called the *width*. Informally, the width of a HAP-set equals the number of schedules such that each pair of these schedules has no match in the same round (see Section 3.2 for precise definitions). Clearly, a HAP-set with a larger width, has more flexibility. As already deduced in the previous chapter, the width of the popular HAP-set known as the Canonical Pattern Set equals 1. This means that there exists at least one match that, in every schedule compatible with this canonical HAP-set, is always scheduled in the same round.

In this chapter we will focus on the following questions: Do there exist HAP-sets that have large width? And how large can the width of a HAP-set be?

Section 3.2 gives the preliminaries and precise definitions, in Section 3.3 we give upper and lower bounds for the width. In Section 3.4, we prove that the upper bound on the width from Section 3.3 can be achieved for a particular HAP-set when the number of teams is a power of 2. Section 3.5 details a construction that allows one to combine HAP-sets and their corresponding schedules on a small number of teams, in a way that preserves the width. We close with describing an extension in Section 3.6.

3.2 PRELIMINARIES AND NOTATION

We consider a set of teams $\mathcal{T}$, with $2n := |\mathcal{T}|$. To avoid trivialities, we assume $n \geqslant 2$. On these teams, we want to schedule a SRR competition in $2n-1$ rounds denoted by the set R. For each team $t \in \mathcal{T}$, its *Home-Away Pattern* (HAP) is given by $H(t) = (H_r(t))_{r\in R}$, with $H_r(t) \in \{0,1\}$, where $H_r(t) = 0$ indicates team t playing Home and $H_r(t) = 1$ indicates team t playing Away in round $r \in R$. We define a *HAP-set* $\mathcal{H}$ to be a $\mathcal{H} = \{(H_r(t))_{r\in R} : t \in \mathcal{T}\}$, containing a HAP for every team $t \in \mathcal{T}$. Given two teams t, t', we define $\Delta(t,t') = \#\{r : H_r(t) \neq H_r(t')\}$ to be the number of rounds where t, t' differ in Home/Away-allocation.

A schedule S is a partition of the set of all matches $\binom{\mathcal{T}}{2}$ on $2n$ teams into $2n-1$ rounds, such that every round is a matching. We denote this with $S = \cup_{r\in R} S_r$, where each S_r is a round. As we partition all matches, for every match between t, t', there exists a round $r \in R$ such that $\{t, t'\} \in S_r$. A *schedule* S can be generated from such a partition by assigning a Home-team and Away-team in any match in the partition. As notation, we write $S(t,t') = r$ to indicate that in schedule S, the match between teams t, t' is scheduled in round $r \in R$.

Thus, to any schedule S we can associate a corresponding HAP-set $\mathcal{H}(S)$. Two distinct schedules can have equal HAP-sets. We say a HAP-set $\mathcal{H}$ is *feasible* if there exists a schedule S such that $\mathcal{H} = \mathcal{H}(S)$.

We define $\mathbb{H}_n$ to be the set of all feasible HAP-sets on $2n$ teams. As stated earlier, we define $\mathcal{H}(S)$ to be the HAP-set generated by schedule S. We also define $\mathcal{S}(\mathcal{H}) = \{S : \mathcal{H}(S) = \mathcal{H}\}$ to be the set of all schedules S that have HAP-set $\mathcal{H}$. We say that schedule S is *compatible* with HAP-set $\mathcal{H}$ if $S \in \mathcal{S}(\mathcal{H})$.

Definition 17. *Two schedules* S, S' *are* orthogonal - $S \perp S'$ - *if for every pair of distinct teams* $t, t' \in \mathcal{T}$, *the round* $S(t,t') \neq S'(t,t')$.

In words, schedules S, S' being orthogonal means that no match is scheduled in the same round for S, S'.

Definition 18. *Two schedules* S, S' *are* rotational orthogonal - $S \perp_{rot} S'$ - *if there is a permutation of the rounds* $\sigma : R \to R$ *without fixed elements, such that* $S_r = S'_{\sigma(r)}$ $\forall r \in R$.

Clearly, any two schedules S, S' that are rotational orthogonal are also orthogonal. Using these definitions of orthogonality, we define the following properties of a given HAP-set $\mathcal{H}$.

Definition 19. *(Lambers, Goossens, and Spieksma, 2022) Given a HAP-set $\mathcal{H}$ for a set $\mathcal{T}$ of $2n$ teams, we define:*

- $opp(\mathcal{H}) = \min_{t,t' \in \mathcal{T}} |\{r : H_r(t) \neq H_r(t')\}|$,
- $width(\mathcal{H}) = \max_{\mathcal{S} \subset \mathcal{S}(\mathcal{H})} |\{\mathcal{S} : S \perp S' \ \forall S, S' \in \mathcal{S}\}|$,
- $rotw(\mathcal{H}) = \max_{\mathcal{S} \subset \mathcal{S}(\mathcal{H})} |\{\mathcal{S} : S \perp_{rot} S' \ \forall S, S' \in \mathcal{S}\}|$.

In words, for a given $\mathcal{H} \in \mathbb{H}$, the properties $\mathrm{opp}(\mathcal{H})$, $\mathrm{width}(\mathcal{H})$, $\mathrm{rotw}(\mathcal{H})$ are defined as:

- $\mathrm{opp}(\mathcal{H})$: The minimum number of rounds over distinct pairs of teams t,t' such that teams t, t' have a different (or opposite) Home/Away-assignment in $\mathcal{H}$ - i.e., $\min_{t,t'} \Delta(t, t')$.
- $\mathrm{width}(\mathcal{H})$: The maximum number of schedules compatible with HAP-set $\mathcal{H}$ that are pairwise orthogonal, thus were every match (t, t') is played in a different round in a different schedule.
- $\mathrm{rotw}(\mathcal{H})$: The maximum number of schedules with HAP-set $\mathcal{H}$ that are pairwise rotational orthogonal.

It is not difficult to see that $\mathrm{opp}(\mathcal{H}) \geqslant \mathrm{width}(\mathcal{H}) \geqslant \mathrm{rotw}(\mathcal{H})$ for each feasible $\mathcal{H} \in \mathbb{H}$.

Remark 1. *For notational purposes, we can refer to the rotational width of a schedule $S = (S_r)_r$ compatible with $\mathcal{H}$, so $width_{\mathrm{rot}}(S)$ instead of $width_{\mathrm{rot}}(\mathcal{H})$. This should be interpreted as the largest set of pairwise orthogonal schedules that can be obtained by permuting the rounds S_r in a way that preserves the HAP-set. Obviously $width_{\mathrm{rot}}(S) \leqslant width_{\mathrm{rot}}(\mathcal{H})$.*

In Definition 19, we have defined three properties of a given HAP-set $\mathcal{H}$. Our main goal here is to find HAP-sets with extremal values for these properties. Therefore, for each of these properties, we define o_n, w_n, x_n as follows:

Definition 20. *For each $2n \geqslant 4$, we define:*

- $o_n = \max_{\mathcal{H} \in \mathbb{H}_n} opp(\mathcal{H})$,
- $w_n = \max_{\mathcal{H} \in \mathbb{H}_n} width(\mathcal{H})$,
- $x_n = \max_{\mathcal{H} \in \mathbb{H}_n} rotw(\mathcal{H})$.

Simply put, o_n, w_n, x_n equal the value of the HAP-set that scores best on the respective measure for a given n.

With these definitions, we get to the following fundamental question:

Question 1. *For a given value of n, what is w_n? And what is o_n, x_n?*

3.3 UPPER AND LOWER BOUNDS FOR THE WIDTH

We establish the following lower and upper bounds on the width:

Theorem 6. *For each* $n \geqslant 2$*:*

$$2 \leqslant x_n \leqslant w_n \leqslant o_n \leqslant n.$$

Proof. The inequalities $x_n \leqslant w_n \leqslant o_n$ are imminent, as for each $\mathcal{H} \in \mathbb{H}_n$, we have $\text{opp}(\mathcal{H}) \geqslant \text{width}(\mathcal{H}) \geqslant \text{rotw}(\mathcal{H})$.

We now argue that $o_n \leqslant n$. Consider any HAP-set $\mathcal{H} \in \mathbb{H}_n$. As $\mathcal{H}$ is feasible, it follows that in every round, there are n^2 pairs of teams with a different Home/Away-allocation, leading to a total sum of different Home/Away-allocations equal to $\sum_{(t,t') \in \binom{\mathcal{T}}{2}} \Delta(t,t') = (2n-1)n^2$. As there are $\binom{2n}{2} = n(2n-1)$ pairs of teams, there must be a pair of teams $t, t' \in \mathcal{T}$ with $\Delta(t,t') \leqslant n$. Thus, for any HAP-set $\mathcal{H} \in \mathbb{H}_n$, we have $\text{opp}(\mathcal{H}) \leqslant n$, which implies $o_n \leqslant n$.

Next, we show that $2 \leqslant x_n$. We prove this inequality by first considering a partition of the set of all matches into $2n-1$ rounds; next, we construct a HAP-set such that there exist two schedules compatible with it, such that all matches in round r in one schedule, are scheduled in round $r+1$ in the other schedule, $r \mod 2n$.

There are many ways to partition the set of $\binom{2n}{2}$ matches into $2n-1$ rounds $S_1, \dots, S_{2n-1}$ such that each round consists of n matches featuring each team exactly once. One possibility is the well-known Circle Method, see eg Lambrechts et al. (2018), Siemann (2020). Thus, we can assume we are given a set of rounds $S_1, \dots, S_{2n-1}$; notice that the Home/Away assignment for the matches in these rounds has not been specified. We will give a procedure that constructs a HAP-set $\mathcal{H}$ in a round-by-round fashion, in such a way that there exist two schedules compatible with $\mathcal{H}$ that are rotational orthogonal.

Fix a round $r \in R$, and construct a simple undirected graph $G_{r,r+1} = (V = \mathcal{T}, E_{r,r+1})$, where $(t,t') \in E_{r,r+1}$ iff match (t,t') is played in round r or $r+1$ (indices are read modulo $2n-1$, thus $2n = 1$). Clearly, $G_{r,r+1}$ is a regular graph of degree 2, where every node (team) is incident to an edge corresponding to its match-up in round r and to an edge corresponding to its match-up in round $r+1$. As $G_{r,r+1}$ is of degree 2, it can be seen as a collection of cycles. For every cycle, define an orientation (clockwise or counter-clockwise), loop through every cycle, and for every edge (match) that is scheduled in round $r+1$, assign the first node (with respect to the orientation) to be the Home playing team and the second node the team that plays Away in round $r+1$.

Notice that this Home/Away assignment accommodates the matches in S_{r+1} as the nodes corresponding to each pair of teams that are matched in S_{r+1} are connected in $G_{r,r+1}$, and the construction ensures that these nodes receive a different Home/Away assignment. It is also true that this Home/Away assignment simultaneously accommodates the matches in S_r. Indeed, again the nodes corresponding to each pair of teams that are matched in S_r are connected in $G_{r,r+1}$, and hence receive a different Home/Away assignment.

When we perform this procedure sequentially for $r = 1, 2, \ldots, 2n-1$, we thereby specify a HAP-set $\mathcal{H}$. Obviously, by construction, this HAP-set accommodates the matches in the rounds S_r for each $r \in R$, thereby specifying a schedule S that is compatible with this HAP-set $\mathcal{H}$.

Consider now a set of rounds that is specified by $S'_r = S_{r-1}$, $r \in R$. Thus the matches in round $r-1$ in S are played in round r in S'_r, perhaps with a different Home/Away assignment. We claim that there is a schedule S' consisting of the rounds $S'_r, r \in R$ as follows. Consider a pair of teams that meet in S'_r for some $r \in R$. By construction of the graph $G_{r-1,r}$ the nodes corresponding to this pair of teams are connected. But that means that their Home/Away assignment is different, which implies the existence of a schedule S' that is compatible with $\mathcal{H}$. Further, it is evident that $S \perp_{rot} S'$.

Concluding, we have constructed a HAP-set $\mathcal{H}$, and we have shown that there exist two schedules S and S' that are compatible with it while $S \perp_{rot} S'$. This concludes the proof.

□

Remark 2. *Notice that, if n is odd, the upperbound from Theorem 6 can be improved. Indeed, any feasible HAP-set $\mathcal{H}$ on $2n$ teams has exactly $n(2n-1)$ Home assignments, which is an odd number when n is odd. When two teams t, t' both have an even number of Home's assigned, they have a different Home/Away-allocation in an even number of rounds, $\Delta(t,t')$ is even. Also, when two teams t, t' both have an odd number of Home's assigned, $\Delta(t,t')$ is even. When there are more than 2 teams, there must be a pair of teams t, t' that have the same parity number of Home games, so they will have $\Delta(t,t')$ even. As n is presumed odd, and the only way to make a HAP-set with opp$(\mathcal{H}) = n$ is if every pair of teams t, t' has $\Delta(t,t') = n$, we see that opp$(\mathcal{H}) < n$.*

Notice also that the procedure sketched in the second part of the proof of Theorem 6 starts from any feasible partition of the matches into rounds. In case one would start from the partition that is generated by the well-known circle method, we can, using this procedure identify an explicit HAP-set. In fact, we claim that the HAP-set from Table 17 arises, and it follows that this HAP-set has rotational width at least 2. One might comment that this HAP-set is unbalanced in the sense that Team 0 only has Away matches, whereas each other team

Team	Round 1	2	3	4	...	Round $2n-1$
0	A	A	A	A	...	A
1	H	H	A	H	...	A
2	A	H	H	A	...	H
3	H	A	H	H	...	A
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\vdots$
$2n-1$	H	A	H	A	...	H

Table 17: A HAP-set with rotational width at least 2

plays only $n-1$ away matches. However, as the operation of inverting all Home/Away assignments in a single round does not impact the (rotational) width, one can improve this balance by inverting the Home/Away assignments in $\frac{n}{2}$ rounds, leading to a HAP-set such that no team plays more than $\frac{3}{2}n-1$ matches away.

3.4 HAP-SETS WITH MAXIMUM WIDTH

In this section, we identify a family of HAP-set whose width equals the upper bound established in Section 3.3 in case the number of teams is a power of 2.

Theorem 7. *When $n = 2^\ell$ ($\ell \in \mathbb{N}$), there is a HAP-set $\mathcal{H}^* \in \mathbb{H}_n$ with width$(\mathcal{H}^*) = n$.*

Proof. We prove this by constructing the HAP-set $\mathcal{H}^*$ and providing n orthogonal schedules that are compatible with $\mathcal{H}^*$.

CONSTRUCTING $\mathcal{H}^*$

We have a set of $2n$ teams $\mathcal{T}$ indexed by $0, \dots, 2n-1$ and let the rounds be indexed by $1, \dots, 2n-1$. Let $b_i(j)$ be the i-th bit of the binary representation of j, thus $j = \sum_{i=1} b_i(j) 2^{i-1}$. Then, we choose for each $r = 1, \dots, 2n-1$ and for each $t = 0, \dots, 2n-1$:

$$H_r(t) := \sum_{i=1}^{\ell} b_i(r) b_i(t) \mod 2. \tag{33}$$

In case $H_r(t) = 0$, this implies a Home match for team $t \in \mathcal{T}$ in round $r \in R$, otherwise $H_r(t) = 1$ implying an Away match. In other words, if the number of bits in the same position for r and t that are equal to 1, is even, then $H_r(t) = 0$, otherwise $H_r(t) = 1$. We claim that the HAP-set $\mathcal{H}^* = \{(H_r(t))_{r \in R} : t \in \mathcal{T}\}$ generated this way, is a HAP-set with width n, which we will prove later on.

This HAP-set has the following properties which we state without proving them (it is not a difficult exercise to do so):

- $H_r(0) = 0 \;\; \forall r$ - thus, Team 0 plays Home in every round.
- $\sum_r H_r(t) = n \;\; \forall t \in \mathcal{T} \setminus \{0\}$ - thus, Team $1, \ldots, 2n-1$ play at Home in $n-1$ rounds, and Away in n rounds.
- $\sum_r |H_r(t) - H_r(t')| = n \;\; \forall \{t, t'\} \subset \mathcal{T}$ - for every pair of teams, there are n rounds in which their assignment differs.

As an example, the HAP-set for $N = 2 \cdot 2^2 = 8$ teams is given in Table 18.

	Round	1	2	3	4	5	6	7
Teams		001	010	011	100	101	110	111
0	000	0	0	0	0	0	0	0
1	001	1	0	1	0	1	0	1
2	010	0	1	1	0	0	1	1
3	011	1	1	0	0	1	1	0
4	100	0	0	0	1	1	1	1
5	101	1	0	1	1	0	1	0
6	110	0	1	1	1	1	0	0
7	111	1	1	0	1	0	0	1

Table 18: HAP-set on 8 teams

GENERATING THE SCHEDULES Our next task is to show that $\mathcal{H}^*$ allows for n orthogonal schedules. We do this by first constructing $2n-1$ rounds of n matches each, that partition all the $\binom{\mathcal{T}}{2}$ matches that need to be scheduled. Next, we show that all these rounds can be rotated in n different ways, thus leading to n orthogonal schedules.

Recall the construction of the HAP-set $\mathcal{H}^*$. Given two teams s, t, we define $R(s,t) = \{r : H_r(s) \neq H_r(t)\}$, the set of rounds in which teams s, t can potentially meet each other - note that $|R(s,t)| = \Delta(s,t) = n$ as stated earlier. Furthermore, we define the equivalence class $[\cdot, \cdot]$, where $[s,t] = \{(s', t') \in \binom{\mathcal{T}}{2} : R(s,t) = R(s', t')\}$ to be the set of matches that can be scheduled in the same rounds as (s,t), namely $R(s,t)$. We also define the index-set $I(s,t)$ for teams s, t, where $I(s,t) = \{i \leqslant k : b_i(s) \neq b_i(t)\}$. By construction of the HAP-set, we see that $R(s,t) = \{r : \sum_{i \in I(s,t)} b_i(r) = 1 \mod 2\}$. We denote binary addition with the operator $\cdot$, implying $s \cdot t = \sum_i (b_i(s) + b_i(t) \mod 2) 2^{k-i}$ - as example, $6 \cdot 5 = 4(1+1 \mod 2) + 2(1+0 \mod 2) + (0+1 \mod 2) = 3$. Remark that by construction, $s \cdot t \cdot t = s$.

Claim 2. *The classes* $[0,t]$ *have* $[0,t] = \{(s, s \cdot t) : s \leqslant 2n-1\}$ *and* $\cup_t [0,t] = \binom{\mathcal{T}}{2}$ *- that is, all matches are part of one of the classes* $[0,t]$.

Proof. Clearly $\cup_t\{(s, s\cdot t) : s \in [1, 2n-1]\}$ is a partition of $\binom{\mathcal{T}}{2}$, so we only need to prove that $[0, t] = \{(s, s\cdot t) : s \in [1, 2n-1]\}$ for all t.

To do this, we need to show that $R(s, s\cdot t) = R(0, t)$ for all s. Recall that $R(0, t) = \{r : \sum_{i\in I(0,t)} b_i(r)\}$ with $I(0, t) = \{i : b_i(0) \neq b_i(t)\}$. However,

$$I(s, s\cdot t) = I(0\cdot s, t\cdot s) = \{i : b_i(0\cdot s) \neq b_i(t\cdot s) = I(0, t)$$

Thus, $R(s, s\cdot t) = R(0, t)$ and $P(0, t) = \{(s, s\cdot t) : s \in [0, 2n-1]\}$

□

To illustrate how this partition looks, we show $[0, 1], \ldots, [0, 7]$ for 8 teams in the Example 2

Example 2. *When* $N = 2n = 8$, *we have sets* $P(0, 1), \ldots, P(0, 7)$, *where:*

$$\begin{aligned}
[0,1] &= \{(0,1),(2,3),(4,5),(6,7)\} & R(0,1) &= \{1,3,5,7\}\\
[0,2] &= \{(0,2),(1,3),(4,6),(5,7)\} & R(0,2) &= \{2,3,6,7\}\\
[0,3] &= \{(0,3),(1,2),(4,7),(5,6)\} & R(0,3) &= \{1,2,5,6\}\\
[0,4] &= \{(0,4),(1,5),(2,6),(3,7)\} & R(0,4) &= \{4,5,6,7\}\\
[0,5] &= \{(0,5),(1,4),(2,7),(3,6)\} & R(0,5) &= \{1,3,4,6\}\\
[0,6] &= \{(0,6),(1,7),(2,4),(3,5)\} & R(0,6) &= \{2,3,4,5\}\\
[0,7] &= \{(0,7),(1,6),(2,5),(3,4)\} & R(0,7) &= \{1,2,4,7\}
\end{aligned}$$

Now that we've established that the elements of the class $[0, t]$ form a matching of all teams, for all different t which can be scheduled in n different rounds, it remains to be shown that we can create n different schedules. It is not difficult to see that this is possible.

Construct the bipartite graph $G = (V_1 \times V_2, E)$, with $V_1 = [0, t]$ for $t \in \{1, \ldots, 2n-1\}$ representing the classes and $V_2 = \{1, \ldots, 2n-1\}$ representing the rounds, and $(v_t^1, v_r^2) \in E$ with $v_t^1 \in V_1$ and $v_r^2 \in V_2$ if $v_r^2 \in R(0, t)$. This is a regular bipartite graph of order n, for which we know that a 1-factorization exists. For any 1-factorization into n color classes, every color class implies a schedule. As there are n color classes there are n different schedules which are all orthogonal by construction. □

Example 3. *Using the above construction, we find the following four schedules when the number of teams* $N = 8$ *and the HAP-set is as in Table 18.*

Schedule	Round 1	2	3	4	5	6	7
S_1	0 − 1	0 − 2	0 − 5	0 − 6	0 − 3	0 − 4	0 − 7
	2 − 3	1 − 3	3 − 6	1 − 7	2 − 1	1 − 5	3 − 4
	4 − 5	4 − 6	4 − 1	2 − 4	5 − 6	6 − 2	5 − 2
	6 − 7	5 − 7	7 − 2	3 − 5	7 − 4	7 − 3	6 − 1
S_2	0 − 3	0 − 6	0 − 2	0 − 7	0 − 1	0 − 5	0 − 4
	2 − 1	1 − 7	3 − 1	1 − 6	2 − 3	1 − 4	3 − 7
	4 − 7	4 − 2	4 − 6	2 − 5	5 − 4	6 − 3	5 − 1
	6 − 5	5 − 3	7 − 5	3 − 4	7 − 6	7 − 2	6 − 2
S_3	0 − 5	0 − 7	0 − 1	0 − 4	0 − 6	0 − 3	0 − 2
	2 − 7	1 − 6	3 − 2	1 − 5	2 − 4	1 − 2	3 − 1
	4 − 1	4 − 3	4 − 5	2 − 6	5 − 3	6 − 5	5 − 7
	6 − 3	5 − 2	7 − 6	3 − 7	7 − 1	7 − 4	6 − 4
S_4	0 − 7	0 − 3	0 − 6	0 − 5	0 − 4	0 − 2	0 − 1
	2 − 5	1 − 2	3 − 5	1 − 4	2 − 6	1 − 3	3 − 2
	4 − 3	4 − 7	4 − 2	2 − 7	5 − 1	6 − 4	5 − 4
	6 − 1	5 − 6	7 − 1	3 − 6	7 − 3	7 − 5	6 − 7

Table 19: Four orthogonal schedules for 8 teams

3.5 GENERAL SCHEDULES

Maximum opposing HAP-sets

In this section we introduce a procedure to create HAP-sets $\mathcal{H}$ that have maximum opp$(\mathcal{H}) = n$, provided that we have two other HAP-sets, $\mathcal{H}_1, \mathcal{H}_\in$ on $2n_1, 2n_2$ teams with opp$(\mathcal{H}_i) = n_i$, and $n = 2n_1n_2$.

Suppose we have HAP-sets $\mathcal{H}_1, \mathcal{H}_2$ given by:

$$\mathcal{H}_1 = \left\{(H^1_r(t))_{r \leqslant 2n_1 - 1} : t \in [1, \dots, 2n_1]\right\} \tag{34}$$

$$\mathcal{H}_2 = \left\{(H^2_r(t))_{r \leqslant 2n_2 - 1} : t \in [1, \dots, 2n_2]\right\} \tag{35}$$

We will create a HAP-set on $2n = 4n_1n_2$ teams and $4n_1n_2 - 1$ rounds by gluing the smaller HAP-sets together, which is described in the following Algorithm 2. Recall that a Home-allocation is denoted with a 0, and Away with a 1. The algorithm uses $+_2$, addition modulo 2.

In Table 20, a block-structured overview of the newly created HAP-set is given.

Theorem 8. *Let $\mathcal{H}_1, \mathcal{H}_2$ be HAP-sets on $2n_1, 2n_2$ teams and opp$(\mathcal{H}_i) = n_i$ for $i = 1, 2$ is maximum. Then, opp$(G(\mathcal{H}_1, \mathcal{H}_2)) = 2n_1n_2 = 2n$ is maximum as well.*

Algorithm 2 Gluing

Input: Two HAP-sets $\mathcal{H}_1, \mathcal{H}_2$ on $2n_1, 2n_2$ teams with $\text{opp}(\mathcal{H}_i) = n_i$.

1: Let $\mathcal{T}$ be a set of $4n_1n_2$ teams, and R a set of $4n_1n_2 - 1$ rounds.

2: Partition $\mathcal{T}$ in subsets T_i, with:

$$\mathcal{T} = \cup_i T_i \qquad T_i = \{2n_2(i-1)+1, \ldots, 2n_2 i\} \;\; 1 \leqslant i \leqslant 2n_1$$

We say $[j] \in T_i$ is the team $j \in T_i$ with index $j \mod 2n_2$.

3: Partition the rounds R in subsets R_j with:

$$R = R_\delta \cup \left(\cup_j R_j\right) \quad R_j = \{(2n_2-1)j+1, \ldots, (2n_2-1)(j+1)\}$$
$$R_\delta = \{(2n_2-1)2n_1+1, \ldots, 4n_2n_1 - 1\}$$

We say $[r] \in R_j$ is the round $r' \in R_j$ such that $r = r' \mod 2n_2 - 1$

4: In rounds R_0, assign to teams $[j] \in T_i$ the HAP $(H^2_r(j))_{r \in R_0}$.

5: For rounds R_s, $1 \leqslant s \leqslant 2n_1 - 1$, assign to team $[j] \in T_i$ the HAP $(H^1_s(i) +_2 H^2_r(j))_{r \in R_s}$.

6: For the rounds $s \in R_\delta$, assign $H^1_s(i)$ to all teams in T_i.

Output: A HAP-set $G(\mathcal{H}_1, \mathcal{H}_2)$ on $4n_1n_2$ teams with $\text{opp}(G(\mathcal{H}_1, \mathcal{H}_2)) = 2n_1n_2$.

Teams/Rounds	R_0	R_j	R_δ
T_1	$\mathcal{H}_2$	$\mathcal{H}_2 +_2 H^1_j(1)$	$H^1(1)$
T_i	$\mathcal{H}_2$	$\mathcal{H}_2 +_2 H^1_j(i)$	$H^1(i)$
T_{2n_1}	$\mathcal{H}_2$	$\mathcal{H}_2 +_2 H^1_j(2n_1)$	$H^1(2n_1)$

Table 20: Structure of $G(\mathcal{H}_1, \mathcal{H}_2)$

Proof. We prove this by simply counting how many times a pair of teams have a different allocation. It is important to recall that two teams with HAPs from $\mathcal{H}_1$, have a different allocation in exactly n_1 rounds, and an equal allocation in the other $n_1 - 1$ rounds. Similarly, this is true for two teams with HAPs in h_2, they have opposite assignment in n_2 rounds and equal assignment in $n_2 - 1$ rounds.

- Suppose $t, t' \in T_i$ for some i. Then they have a different allocation exactly $\text{opp}(\mathcal{H}_2) = n_2$ times in each of the blocks R_j, $j = 0, \ldots, 2n_1 - 1$. There are $2n_1$ such blocks, thus they have opposite assignment $\Delta(t, t') = 2n_1n_2$ times.

- Suppose $t \in T_i, t' \in T'_i$ for different i, i', and $[t] \neq [t']$. They differ n_2 times in a block R_j when $H^1_i(j) = H^1_{i'}(j)$ with $j = 0, \ldots, 2n_1 - 1$, and they differ $n_2 - 1$ times in the other blocks. The number of blocks R_j where $H_i(j) \neq H_{i'}(j)$ is n_1, and in $n_1 - 1$ blocks, $H_i(j) = H_{i'}(j)$. In block R_δ, the teams differ in n_1 rounds. Together, this means they differ in:

$$\Delta(t,t') = n_2 + (n_1-1)n_2 + (n_2-1)n_1 + n_1 = 2n_1n_2$$

- Suppose $t \in T_i, t' \in T_{i'}$ with $i \neq i'$ and $[t] = [t']$. Then they have a different allocation in all the $2n_2 - 1$ rounds in block R_j whenever $H_i^1(j) \neq H_{i'}^1(j)$ - this occurs in opp$(\mathcal{H}_1) = n_1$ blocks. They also have different allocation in n_1 rounds in block R_δ. Combined, we see $\Delta(t,t') = (2n_2-1)n_1 + n_1 = 2n_1n_2$.

As in all three cases, teams t, t' have $\Delta(t,t') = 2n_1n_2$, we can conclude that $G(\mathcal{H}_1, \mathcal{H}_2) = 2n_1n_2$ □

Corollary 1. *When $o_n = n$, $o_{2n} = 2n$.*

Proof. If $o_n = n$, there must exist a $\mathcal{H}$ on $2n$ teams, such that opp$(\mathcal{H}) = n$. We know the trivial HAP-set on $2n = 2$ teams, $\mathcal{H}^*$, has opp$(\mathcal{H}^*) = 1 = n_1$. So both $\mathcal{H}, \mathcal{H}^*$ are maximum opposing HAP-sets, thus $G(\mathcal{H}, \mathcal{H}^*) = 2n$ is maximum opposing as well. As $G(\mathcal{H}, \mathcal{H}^*)$ is a HAP on $4n$ teams, $o_{2n} = 2n$. □

Remark 3. *Algorithm 2 is not applicable solely to HAP-sets $\mathcal{H}$ with maximum opp$(\mathcal{H}) = n$. It is not difficult to find lower bounds on $\Delta(t,t')$ in $G(\mathcal{H}_1, \mathcal{H}_2)$, when opp$(\mathcal{H}_i) \leqslant n_i$ for $i = 1, 2$. Doing so, and again using that the HAP-set on two teams h^* has opp$(\mathcal{H}^*) = 1 = n$, one can even prove a stronger version of Corollary 1, namely that $o_{2n} \geqslant 2o_n$ for all n.*

Remark 4. *In a similar way as gluing HAP-sets $\mathcal{H}_1, \mathcal{H}_2$ to form $G(\mathcal{H}_1, \mathcal{H}_2)$ on $4n_1n_2$ teams via Algorithm 2, it is possible to glue schedules S^1, S^2 to form $G'(S_1, S_2)$ on $4n_1n_2$ teams, where $G'(S^1, S^2)$ is compatible with $G(\mathcal{H}_1, \mathcal{H}_2)$. We do not show the exact algorithm, as it is very repetitive. However, we like to point out that if rotw$(S^1) = n_1$ and rotw$(S^2) = n_2$ on $\mathcal{H}_1, \mathcal{H}_2$ respectively, then rotw$(G'(S^1, S^2)) = 2n_1n_2$. This means that if we have two schedules with perfect rotational width, we can create a new schedule on $4n_1n_2$ teams that also has perfect rotational width.*

The algorithm that glues two HAP-sets together, preserves the property of having maximum opposing rounds of two HAP-sets. However, so far we only showed for HAP-sets where the number of teams equaled a power of two that they have maximum opposing rounds. And in those cases, we can even explicitly construct the HAP-set, without using smaller sized HAP-sets, as shown in Section 3.4.

However, having maximum opposing rounds, is not reserved for HAP-sets on powers of two, as can be seen by the HAP-set $\mathcal{H}$ on 12 teams in Table 21. It has opp$(\mathcal{H}) = 6$, which is maximum. It is also the smallest possible number of teams n for which $n \neq 2^\ell$ and the HAP-set is maximum opposing.

Round	1	2	3	4	5	6	7	8	9	10	11
	0	0	0	0	0	0	0	0	0	0	0
	1	1	1	1	1	1	0	0	0	0	0
	1	1	1	0	0	0	1	1	1	0	0
	1	1	0	1	0	0	1	0	0	1	1
	1	0	1	0	1	0	0	1	0	1	1
	0	1	1	0	0	1	0	0	1	1	1
	0	0	1	1	1	0	1	0	1	0	1
	0	0	1	1	0	1	1	1	0	1	0
	0	1	0	1	1	0	0	1	1	1	0
	0	1	0	0	1	1	1	1	0	0	1
	1	0	0	1	0	1	0	1	1	0	1
	1	0	0	0	1	1	1	0	1	1	0

Table 21: HAP-set on 12 teams with opp = 6

The largest known set of orthogonal schedules compatible with the HAP-set in Table 21, has size 4.

3.6 EXTENSIONS

It is clear that the width is a measure indicating to what extent a particular HAP-set can accommodate distinct schedules where an individual match has different rounds to be played in. This is clearly relevant in a first-break-then-schedule approach, see Section 3.1. Another property one might be interested in is the notion of *match-pair disjointness*. Given a HAP-set, this property refers to the existence of two schedules such that each pair of matches in the same round in one schedule are not in the same round in the other schedule.

Notice that this property is different from orthogonality, i.e., a pair of schedules may be match-pair disjoint or not, and they may be orthogonal or not. Although we have no theoretical results for this property of match-pair disjoint, we provide, for $2n = 8$ teams, two schedules that are both match-pair disjoint, as well as orthogonal.

Teams/Rounds	1	2	3	4	5	6	7
0	H	H	H	H	H	H	H
1	A	A	A	H	A	H	A
2	H	A	H	A	H	H	H
3	A	H	A	H	H	A	H
4	A	H	H	A	H	A	A
5	H	H	H	A	A	H	A
6	H	A	A	H	A	A	H
7	A	A	A	A	A	A	A

Table 22: The HAP-set with two feasible match disjoint schedules

Rounds	1	2	3	4	5	6	7
Schedule 1	0 − 1	0 − 2	0 − 3	0 − 4	0 − 5	0 − 6	0 − 7
	2 − 3	3 − 1	2 − 1	1 − 5	2 − 7	1 − 7	2 − 5
	5 − 4	4 − 6	4 − 7	3 − 7	6 − 2	3 − 6	3 − 4
	6 − 7	5 − 7	5 − 6	6 − 2	4 − 1	5 − 3	6 − 1
Schedule 2	0 − 4	0 − 1	0 − 7	0 − 2	0 − 6	0 − 3	0 − 5
	2 − 7	3 − 6	2 − 3	1 − 7	2 − 1	1 − 4	2 − 4
	5 − 3	4 − 7	4 − 6	3 − 4	3 − 7	2 − 6	3 − 1
	6 − 1	5 − 2	5 − 1	6 − 5	4 − 5	5 − 7	6 − 7

Table 23: Two match disjoint schedules compatible with the same HAP-set.

4 THE MULTI-LEAGUE SPORTS SCHEDULING PROBLEM

The Dutch field hockeyclub USHC, a club solely for students in Utrecht, is very popular among them. Every summer more people apply for membership than are able to join the club. This is not because the club aims to be exclusive, but merely because the capacity of their venue, with only two fields, does not allow them to grow.
Hockey, like football, is a sport played in teams, eleven versus eleven. Every member of USHC is part of such a team, and during the season every team plays matches on Sundays - this is non-negotiable. When and where they play is scheduled by the Dutch Field Hockey Association KNHB, who schedules the competitions for all hockey teams in the Netherlands.
On a Sunday, playing from roughly 9 in the morning till 7 in the evening makes that approximately $6-7$ matches can be scheduled per pitch. Disregarding some small 'ifs', on a very good day, this means that USHC can host up to 14 matches.
As on average only half of the teams is expected to play at home, USHC indeed has 26 teams registered, practically the maximum given their venue size. Still, its teams have to play home matches at neighbouring clubs from time to time.
Besides capacity issues, that are common within the sport, the clubs expect more from the KNHB. Most of the clubs specifically value their Dames 1 and Heren 1 (Womens and Mens top tier team), and try to get a supporting crowd when they play at home. Thus, some clubs even prefer to have them play at home on the same Sundays, as that brings something extra to the stands.
These capacity issues and coupling preferences, are already difficult to grasp in one DRR, let alone when scheduling several hundreds of competitions simultaneous.
This chapter is based on joint work with Morteza Davari, Dries Goossens, Jeroen Beliën and Frits Spieksma that appeared as Davari et al. (2020).

4.1 INTRODUCTION

Every sports competition needs a schedule, stating who will play whom, when, and where. Depending on which constraints need to be taken into account, scheduling a single league may already be quite a challenge - see e.g. Alarcón et al. (2017), Goossens and Spieksma (2009), Recalde, Torres, and Vaca (2013).

However, while professional sports usually have only a handful of leagues, in amateur sports or youth competitions, the number of leagues and matches can be very large. For instance, in the Dutch field hockey association, each of the approximately 325 clubs belongs to one of six regions. Most teams of a club, both in the youth and senior divisions, play their matches within the region; Within one region, thousands of teams are distributed over hundreds of leagues, yielding over tens of thousands of matches in one season.

Within these leagues, clubs typically have several teams (e.g. based on age or skill of the players); however, all teams from the same club share the same infrastructure. This creates a capacity problem at each club: a club has a bound on the number of matches it can host at each point in time (which typically follows from its number of pitches). These capacity constraints create interdependencies between the leagues, making it a challenging problem to schedule all leagues while taking these capacities into account.

With respect to scheduling multiple leagues simultaneously, the literature is sparse. Kendall (2008) considers the problem of simultaneously scheduling the matches in four different leagues of the English soccer competition. However, the focus is only on two rounds, played on Boxing day and New Year's day. During these rounds, each team must play one home match and one away match such that the two opponents of each team are different, and that some pairs of teams do not meet at all. In all leagues, the objective is to minimize the total distance traveled by the teams in those two rounds. The solution offered, however, does not generalize to scheduling the entire season.

Grabau (2012) describes the scheduling of a recreational softball competition with 74 teams, split over 8 leagues, and competing on 12 fields. The scheduler must adhere to several intertwined scheduling rules, while simultaneously ensuring that the players play their allotment of matches. Burrows and Tuffley (2015) describe a scheduling problem for a competition played in two divisions. The authors try to achieve a maximal number of so-called *common fixtures* between clubs, which occur if their teams in division one and two are scheduled to play each other in the same round.

Schönberger (2015) introduces the so-called *championship timetabling problem*, which involves several leagues that are scheduled simultaneously. Two types of inter-league constraints are considered: limited venue capacity as well as player substitution opportunities between several teams of a club. Computational experiments involving a mixed-integer linear program illustrate that even finding a feasible solution for a very small instance with only two leagues of six teams each is a time-consuming task.

In this chapter, we study the multi-league scheduling problem as faced by the league organizer. Clearly, when scheduling a single league in professional sports, the precise round in which a particular match takes place can be quite important. However, such matters are not relevant when scheduling thousands of matches for hundreds of leagues. In order to cope with this huge number of matches, typically, a league organizer uses the following approach.

First, the teams are clustered into leagues of even size. Common practice is to (i) use a geographical clustering, ensuring that teams of the same strength/age category are in a same league, and (ii) to avoid teams of the same club to be present in the same league, see (Toffolo et al., 2019) for a discussion of the problem of grouping teams into leagues. Leagues of even size make sense, as they allow each team to play on each round; and although the total number of teams may not be an exact multiple of the league size, with an even league size the vast majority of the teams will be still able to play each round.

Second, the league organizer no longer assigns individual matches to individual rounds. Instead, using a prespecified set of *Home-Away patterns* that is valid for each league, and the league organizer assigns teams to these HAPs. Next, combining this assignment with a compatible specification of each team's opponent for each round, the schedule follows.

Rounds				
r_1	r_2	r_3	r_4	r_5
6-1	*1-3*	*2-4*	*1-2*	*2-3*
5-2	*2-6*	*5-1*	*3-5*	*4-1*
3-4	*4-5*	*6-3*	*4-6*	*6-5*

Rounds				
r_6	r_7	r_8	r_9	r_{10}
1-6	*3-1*	*1-5*	*2-1*	*3-2*
2-5	*6-2*	*4-2*	*5-3*	*1-4*
4-3	*5-4*	*3-6*	*6-4*	*5-6*

Table 24: A schedule compatible with Table 26

	Rounds									
	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}
h_1	A	H	A	H	A	H	A	H	A	H
h_2	A	H	H	A	H	H	A	A	H	A
h_3	H	A	A	H	A	A	H	H	A	H
h_4	A	H	A	H	H	H	A	H	A	A
h_5	H	A	H	A	A	A	H	A	H	H
h_6	H	A	H	A	H	A	H	A	H	A

Table 26: A HAP-set for a league consisting of 6 teams

As an illustration of the latter procedure, consider the HAP-set depicted in Table 26; it reflects a particular for a league consisting of 6 teams, where each team plays against each other team twice. Although a priori, the given HAP-set may allow different schedules (or none), Table 24 gives one such schedule compatible with the HAP-set from Table 26. The issue of deciding whether a schedule exists for a given HAP-set is a well-researched topic (see Miyashiro, Iwasaki, and Matsui (2002), Briskorn (2008), Horbach (2010), Goossens and Spieksma (2011)); we do not go into details here.

This chapter focuses exclusively on assigning teams to HAPs. Since such assignment dictates when each team plays home, it specifies for each club how many matches are played at the club's venue in each round. This is important, since the capacity of a club in terms of the number of matches it can host in a round is typically bounded. In fact, a capacity is given for each club; in practice this number follows from the number of available pitches, the set of possible starting times, and the availability of material and referees. Our goal is to find, for each league, an assignment of teams to HAPs minimizing the total capacity violation over the clubs. We refer to the resulting problem as the MULTI-LEAGUE SCHEDULING PROBLEM (MSP) (see Section 4.2 for a precise problem description).

We present a polynomial-time algorithm for the MSP (Section 4.4). Further, we show that, for a league consisting of at least four teams, the problem becomes difficult when all teams of each club must play according to the same pattern, or when club capacities differ throughout the season (Section 4.5).

4.2 TERMINOLOGY AND ASSUMPTIONS

Each team belongs to a club, and each club has a venue. When a team plays at its club's venue, the team plays *home*, otherwise the team plays *away*. A Double Round Robin (DRR) is a tournament where each team meets each other team twice. This is a typical format in many team sport competitions, such as soccer, basketball, volleyball, hockey; each team meets each other team once home and once away.

When scheduling a tournament, the matches must be allocated to *rounds* in such a way that each team plays at most one match in each round (typically, a round corresponds to a weekend). Since, in our case the number of teams k is even, at least $2(k-1)$ rounds are required to schedule a DRR; if that number is attained, it is called a *compact* DRR.

The sequence of home and away matches according to which a team plays in a tournament, is referred to as a *Home-Away pattern* (in short, HAP). A HAP is represented by a vector consisting of $2(k-1)$ symbols, $k-1$ of which are an 'H', and $k-1$ of which are an 'A'; these obviously refer to the home matches and away matches. A *Home-Away pattern set* (HAP-set) corresponds to the set of HAPs, one for each team in the tournament. We say that a HAP-set is *feasible* if there exists at least one schedule that is compatible with the HAP-set (i.e. for each match i vs. j in round r, team i has an 'H' in its HAP and team j has an 'A').

Two HAPs h and h' are *complementary* if whenever the team assigned to HAP h plays home, the team assigned to HAP h' plays away and vice versa. A *complementary HAP-set* is a set that only consists of complementary pairs of HAPs. For example, the HAP-set depicted in Table 26 is a feasible, complementary HAP-set with three pairs of complementary HAPs (pair 1: h_1 and h_6; pair 2: h_2 and h_3 pair 3: h_4 and h_5).

In this work, we make a number of assumptions. We assume that each league has the same even number of teams. We also assume that the league organizer uses the same complementary HAP-set for each league. This is common practice in competitions where there are few considerations, besides capacity issues. In Section 4.4, it will become clear that the choice of a particular (complementary) HAP-set is irrelevant. Finally, we exclusively deal with compact DRRs for an even number of teams. Consequently, all leagues are played simultaneously, and each team plays either home or away in each round.

We finish this section by formally describing the instances of the Multi-league scheduling problem and its expected outcome.

Multi-league scheduling problem

Input An instance I of MSP consists of:

- Sets of teams T, leagues L and clubs C, with $n = |T|$ and $m = |L|$.
- A partition of the teams $T_L = \{\bar{T}_1, \ldots, \bar{T}_m\}$ with $|\bar{T}_\ell| = k$ for all $\ell \in L$.
- A partition of the teams $T_C = \{\hat{T}_1, \ldots, \hat{T}_{|C|}\}$.
- For each club $c \in C$, its capacity δ_c.
- A feasible complementary HAP-set $\mathcal{H}$ consisting of k HAPs of length $2(k-1)$.

Output An assignment $g : T \to \mathcal{H}$ such that for each $\bar{T}_\ell \in T_L$ and for each $h \in \mathcal{H}$, there exists a $t \in \bar{T}_\ell$ with $g(t) = h$. This assignment should minimize the penalty-function PEN:

$$\mathrm{PEN}(I) = \sum_{c \in C} \sum_{r \leqslant 2(k-1)} \max\left(\sum_{t \in \hat{T}_c} g(t)_r - \delta_c, 0\right) \tag{36}$$

As can be seen in the description above, an instance I of Problem 4.2 of n teams T, clubs C and leagues L, with specification which teams play in which leagues (T_L) and belong to which clubs (T_C). Every league is supposed to be of the same size k, so $n = km$. Clubs can be of any size, where $n_c = |\hat{T}_c|$ denotes the size of club $c \in C$ - again, it holds that $\sum_c n_c = n$. Furthermore, a feasible complementary HAP-set $\mathcal{H}$ is given, with HAPs of length $2(k-1)$, which is the number of rounds in a DRR on k teams. In this HAP, we assume $h_r = 1$ denotes playing at home in round r, and $h_r = 0$ otherwise. Also, for each club c, its venue capacity δ_c is given - this corresponds to the number of matches a club can host per round.

Our solution method does not depend on the given HAP-set $\mathcal{H}$, as long as $\mathcal{H}$ is feasible and complementary, but we do require it to exist.

As output, we expect the assignment of teams to a HAP, in such a way that the capacity violations are kept to a minimum. These violations happen whenever for any club and any round the number of teams of a club that play home exceeds the capacity of the club. The total violation per club per round, is simply the exceedance of its capacity if that number is positive. Minimizing capacity violation is equivalent to minimizing function $\text{PEN}(I)$.

In short, the MULTI-LEAGUE SPORTS SCHEDULING PROBLEM (MSP) is to find an assignment g of teams to HAPs, such that the total capacity violation (i.e. the summation of violations over all clubs and all rounds) is minimized.

4.3 AN IP-FORMULATION OF MSP

The MSP is a problem suitable to be described as an IP. An Integer Program consists of a set of variables, and constraints on these variables that need to be met, while the variables can only take integer values.

4.3.1 The IP

To write the problem as an IP, we have to introduce new *binary* variables, $x_{t,h}$. Variable $x_{t,h}$ is equal to one, if team $t \in T$ is assigned HAP $h \in \mathcal{H}$, and zero otherwise. We also need auxiliary variables $z_{c,r}$ that represent the amount of violation of club $c \in C$ in round $r \in R$. An assignment $\mathbf{x}$ is feasible if and only if the teams in each league are assigned to different HAPs. Given the set of HAPs and the set of rounds, we compute (in a pre-processing step) parameters $U_{h,r}$ which equal one if the team assigned to $h \in \mathcal{H}$ plays home in round $r \in R$, and zero otherwise. The following integer program formulates MSP.

Binary variables $x_{t,h}$ lead to an assignment g by setting $x_{t,h} = 1 \iff g(t) = h$.

$$
\begin{aligned}
\nu_{IP} = &\min \sum_{c \in C} \sum_{r \in R} z_{c,r} && && (37)\\
&\text{s.t.} \\
&\sum_{t \in \bar{T}_\ell} x_{t,h} = 1 && \forall \ell \in L, h \in \mathcal{H} && (38)\\
&\sum_{h \in \mathcal{H}} x_{t,h} = 1 && \forall t \in \bar{T}_\ell, \ell \in L && (39)\\
&z_{c,r} \geqslant \sum_{t \in \hat{T}_c} \sum_{h \in \mathcal{H}} x_{t,h} U_{h,r} - \delta_c && \forall c \in C, r \in R && (40)\\
&z_{c,r} \geqslant 0 && \forall c \in C, r \in R && (41)\\
&x_{t,h} \in \{0,1\} && \forall t \in T, h \in \mathcal{H} && (42)
\end{aligned}
$$

This formulation aims to minimize total capacity violation ν_{IP}, which models the objective function $PEN(I)$ of the problem instance.

Constraints (38)-(39) enforce an assignment of teams to HAPs, while Constraints (40)-(41) determine the number of violations of each club in each round. We point out that this integer program can be modified to accommodate situations that are slightly more general than MSP; for instance, situations where each league has its own (given) HAP-set, or where not all leagues play in all rounds can be formulated with minor modifications of (37)-(42).

4.3.2 LP-relaxation of the IP

When replacing constraints (42) by $x_{t,h} \geqslant 0$ for each t and h, the LP-relaxation of formulation (37)-(42) arises; we denote the corresponding value by ν_{LP}. Solving an LP can usually be done quicker than the IP, as there are no constraints on the integrality of the obtained solution $\hat{x}$.

However, one might wonder whether all extreme vertices of the polytope corresponding to the LP-relaxation of (37)-(42) are integral. That is not the case, as witnessed by the following example.

Example 4. *We have $n = 20$ teams, distributed over $m = 5$ leagues of size $k = 4$, and belonging to six clubs: in this instance, $T = \{t_1, ..., t_{20}\}$, $C = \{c_1, ..., c_6\}$ and $L = \{\ell_1, ..., \ell_5\}$. The partition of teams into clubs, as well as the club's capacities, are given in Figure 3a, and the partition of teams into leagues is given in Figure 3b. The HAP-set is as follows:*

$$
\mathcal{H} = \left\{ \begin{aligned} h_1 &= \{H, A, H, A, H, A\}, \\ h_2 &= \{A, H, A, H, A, H\}, \\ h_3 &= \{H, A, A, A, H, H\}, \\ h_4 &= \{A, H, H, H, A, A\} \end{aligned} \right\}.
$$

	$\hat{T}_c$	δ_c
c_1 :	$\{t_1, t_2, t_3, t_4\}$	2
c_2 :	$\{t_5, t_6, t_7\}$	1
c_3 :	$\{t_8, t_9, t_{10}\}$	1
c_4 :	$\{t_{11}, t_{12}, t_{13}, t_{14}\}$	2
c_5 :	$\{t_{15}\}$	1
c_6 :	$\{t_{16}, t_{17}, t_{18}, t_{19}, t_{20}\}$	1

(a) Clubs: teams and capacity

	$\bar{T}_l$
ℓ_1 :	$\{t_1, t_6, t_{13}, t_{16}\}$
ℓ_2 :	$\{t_2, t_9, t_{12}, t_{17}\}$
ℓ_3 :	$\{t_3, t_5, t_{14}, t_{20}\}$
ℓ_4 :	$\{t_7, t_8, t_{11}, t_{18}\}$
ℓ_5 :	$\{t_4, t_{10}, t_{15}, t_{19}\}$

(b) Leagues

Figure 3: The data associated with Example 4

For this instance, provided in Figure 3, we find an optimal basic solution to the LP-relaxation of (37)-(42). In this solution, the following variables equal 1:

$$\begin{array}{cccccccc} x^*_{1,2}, & x^*_{2,1}, & x^*_{7,2}, & x^*_{8,3}, & x^*_{9,4}, & x^*_{11,4}, & x^*_{12,2}, & x^*_{14,3}, \\ x^*_{13,1}, & x^*_{15,1}, & x^*_{17,3}, & x^*_{18,1}, & x^*_{19,2}, & x^*_{20,4} & & \end{array},$$

while the following equal 0.5 (thus are non-integral):

$$\begin{array}{cccccc} x^*_{3,1}, & x^*_{3,2}, & x^*_{4,3}, & x^*_{4,4}, & x^*_{5,1}, & x^*_{5,2}, \\ x^*_{6,3}, & x^*_{6,4}, & x^*_{10,3}, & x^*_{10,4}, & x^*_{16,3}, & x^*_{16,4} \end{array}.$$

The remaining $x^*_{t,h}$ variables are zero; the values of the $z^*_{c,r}$ variables follow easily. The objective value of this solution to the linear programming relaxation is $v_{LP} = 15$.

4.3.3 A combinatorial lower bound

Instead of raw calculations, we now look into the structure of the problem to derive a lower bound on the objective value v_{IP}.

Observe that, in any HAP, there are $k-1$ 'H's. Hence, the total number of home matches of teams belonging to a club $c \in C$ must equal $(k-1)n_c$ in any assignment g. The total capacity of a club during the season equals $(2k-2)\delta_c$, namely δ_c in every of the $2k-2$ rounds. Clearly, if $(2k-2)\delta_c < (k-1)n_c$, or equivalently, when $\delta_c < \frac{n_c}{2}$, there *will be* violations for club c. Define $C^- = \{c \in C | \delta_c < n_c/2\}$.

We claim that for each $c \in C^-$ the difference between the number of home matches to be played by teams of club c, and the total capacity of club c is a lower bound for the number of violations of club c, i.e., club $c \in C^-$ will have at least $(k-1)n_c - (2k-2)\delta_c = (2k-2)(\frac{n_c}{2} - \delta_c)$ violations. To capture this number of violations we define the following quantity:

$$Q \equiv 2\,(k-1) \sum_{c \in C^-} \Big(\frac{n_c}{2} - \delta_c\Big). \tag{43}$$

The discussion above implies the following lemma.

Lemma 3. $Q \leqslant v_{IP}$.

We will show in Section 4.4 that $Q = v_{LP} = v_{IP}$; in the next theorem, we prove the first of these equalities.

Theorem 9. $Q = v_{LP}$.

Proof. Consider a solution $(\mathbf{x}^*, \mathbf{z}^*)$ that is optimal with respect to the LP-relaxation of (37)-(42). For each club $c \in C$, we have (by summing constraints (40) over the rounds):

$$\begin{aligned}\sum_{r \in R} z^*_{c,r} &\geqslant \sum_{t \in \hat{T}_c} \sum_{h \in \mathcal{H}} x^*_{t,h} \left(\sum_{r \in R} U_{h,r} \right) - 2(k-1)\delta_c \\ &= (k-1)(n_c - 2\delta_c).\end{aligned}$$

This implies

$$\sum_{r \in R} z^*_{c,r} \geqslant \begin{cases} 0 & \text{if } c \in C \setminus C^- \\ 2\,(k-1)\,\big(\frac{n_c}{2} - \delta_c\big) & \text{if } c \in C^- \end{cases}$$

Thus, the following inequality holds:

$$v_{LP} = \sum_{c \in C} \sum_{r \in R} z^*_{c,r} \geqslant \sum_{c \in C^-} 2(k-1)(\frac{n_c}{2} - \delta_c) = Q. \tag{44}$$

Next, consider solution $(\hat{\mathbf{x}}, \hat{\mathbf{z}})$ where $\hat{x}_{t,h} = \frac{1}{k}, \forall t \in T, h \in \mathcal{H}$. Due to the fact that the HAP-set is feasible, it follows that in each round $\frac{k}{2}$ HAPs have an 'H', while the remaining $\frac{k}{2}$ HAPS have an 'A'. Thus, for each $t \in T$, we have that $\sum_{h \in \mathcal{H}} \hat{x}_{t,h} = \frac{1}{k} \cdot \frac{k}{2} = \frac{1}{2}$, and hence the (fractional) number of home matches of a club $c \in C$ in each round equals $\frac{n_c}{2}$, leading to a violation in a round equaling: $\max\left\{\frac{1}{2}n_c - \delta_c, 0\right\}$. Thus, the objective value of this solution is exactly Q, which implies $v_{LP} \leqslant Q$. Together with (44), the result follows. □

In light of Theorem 9, one may wonder whether it is possible to round an optimal, fractional LP-solution into an optimal integral solution. That, however, cannot be achieved by straightforward rounding.

To see that this is indeed not trivial, consider the LP-solution discussed in Example 4, a solution in which there are no violations for club c_1. The non-zero variables associated to teams of club c_1 are $x^*_{1,2} = x^*_{2,1} = 1$ and $x^*_{3,1} = x^*_{3,2} = x^*_{4,3} = x^*_{4,4} = 0.5$, indicating the assignment of each team to any of the four HAPs $h_1, \dots, h_4$.

A straightforward rounding of this solution would imply that teams t_1 and t_2 are assigned to complementary HAPs h_2 and h_1 respectively. Therefore teams t_3 and t_4 should also be assigned to complementary HAPs; otherwise, the club c_1 will have violations in some rounds. Unfortunately, this cannot be achieved by any simple procedure that rounds the fractional assignment of team 3 and team 4, as both h_1, h_2 are not complementary to any of h_3, h_4.

This example shows us that we cannot directly expect the LP to contain integer values. The following section on the other hand, describes an algorithm that is guaranteed to solve the MSP in polynomial time, and gives insight into what an optimal solution looks like.

4.4 A POLYNOMIAL-TIME, EXACT ALGORITHM FOR MSP

In this section, we exhibit Algorithm 3 that outputs an optimal solution to MSP in polynomial time. Interestingly, the values of the capacities δ_c do not impact the solution; in other words, the solution found by this algorithm is optimal for *any* capacities δ_c. Informally, this solution is one where the home matches of teams from the same club are as balanced over the rounds as possible. Before proving correctness of Algorithm 3, we first illustrate how it works on the instance given in Example 4.

Algorithm 3

Input: An instance I of MSP.

1: Create a new instance of MSP as follows. Partition, arbitrarily, each club c into $\lfloor n_c/2 \rfloor$ arbitrary pseudo clubs of size two, and add the remaining team (if there is one) to a new club c'.

2: Partition club c' into $n_{c'}/2$ arbitrary pseudo clubs of size two. Set the capacity of all pseudo clubs in the new instance to one. Notice that in the new instance the assignment of teams to leagues remains unchanged.

3: Construct the following graph G based on the new instance: there is a vertex for each league $\ell \in L$ and there is an edge for each pseudo club that links the two vertices/leagues where the two teams of that pseudo club play. Note that G will be a multi-graph with m vertices and $km/2$ edges.

4: Consider a 2-factorization of G. Associate each 2-factor with a pair of complementary HAPs. This leads to a feasible assignment g: in each 2-factor, each edge is associated with two teams from a pseudo club, and each vertex is associated with two teams in a league that follow the associated pair of complementary HAPs.

Output: An assignment g of teams to HAPs: g.

Example 5. *Following steps 1 and 2 of Algorithm 3, we first create a new instance.*

STEP 1 & 2 *The clubs c_1 and c_4 consist of an even numbers of teams and thus we can split them into four clubs (so-called pseudo clubs) of two teams as follows:*

$$\hat{T}_{c_{1a}} = \{t_1, t_4\} \qquad \hat{T}_{c_{4a}} = \{t_{11}, t_{12}\}$$
$$\hat{T}_{c_{1b}} = \{t_2, t_3\} \qquad \hat{T}_{c_{4b}} = \{t_{13}, t_{14}\}$$

The remaining clubs all consist of an odd numbers of teams. For instance club c_6 consists of five teams, therefore we split it into two clubs of size two and add the remaining team to the new club c'. Hence, $\hat{T}_{c_{6a}} = \{t_{16}, t_{17}\}$ and $\hat{T}_{c_{6b}} = \{t_{18}, t_{19}\}$ and team t_{20} is added to club c'.

We repeat the process for the other clubs with odd numbers of teams and we finally, we split club c' as follows: $\hat{T}_{c'_a} = \{t_7, t_9\}$ and $\hat{T}_{c'_b} = \{t_{15}, t_{20}\}$.

All in all, we have 10 'new' clubs, with two teams each:

$$\hat{T}_{c_{2a}} = \{t_5, t_6\} \qquad \hat{T}_{c_{3a}} = \{t_8, t_{10}\}\hat{T}_{c_{6a}} = \{t_{16}, t_{17}\}$$
$$T_{c'_a} = \{t_7, t_9\} \qquad \hat{T}_{c'_b} = \{t_{15}, t_{20}\}\hat{T}_{c_{6b}} = \{t_{18}, t_{19}\}$$

STEP 3 & 4 *We now construct the graph $G = (V = L, E = T)$ with as vertices the leagues and the clubs $\hat{T}$ of size two as edges. On this graph, we then identify a 2-factorization, see Figure 4b.*

A 2-factor is a collection of cycles spanning all vertices of G; a 2-factorization is a partitioning of the edges of G into 2-factors.

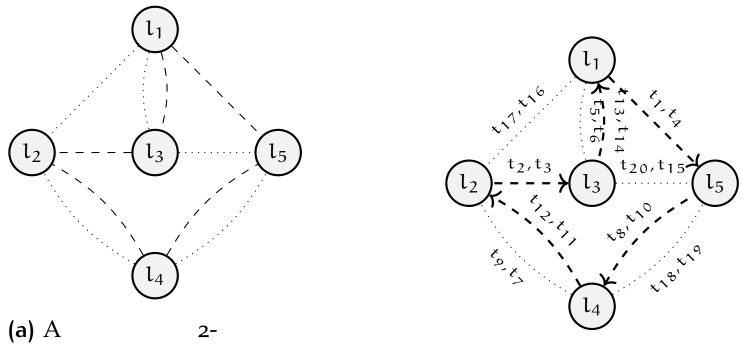
Pick one of the 2-factors and associate it with pair (h_1, h_2) and the other 2-factor with pair (h_3, h_4). To assign teams to HAPs we start with one arbitrary team that is visited in the first 2-factor, for instance team t_1, and assign it to h_1.

We traverse the 2-factor in an arbitrary direction (starting from the edge containing team t_1) and enforce the two teams associated to each edge to HAPs h_1 and h_2 (see Figure 4b). Thus, if $t_1 \to h_1$ (t_1 is assigned to h_1), $t_4 \to h_2$, $t_{10} \to h_1$, $t_8 \to h_2$, $t_{11} \to h_1$, $t_{12} \to h_2$, $t_2 \to h_1$, $t_3 \to h_2$, $t_{14} \to h_1$ and $t_{13} \to h_2$. Similarly we assign the teams in the other 2-factor to HAPs h_3 and h_4 ($t_7 \to h_3$, $t_9 \to h_4$, ...).

The capacities of clubs c_1, c_4 and c_5 are never violated. Club c_2 has one-unit violations at rounds 1, 5 and 6; club c_3 has one-unit violations at rounds 2, 3 and 4; club c_6 has one-unit violations at rounds 2, 3 and 4 and two-unit violations at rounds 1, 5 and 6. The total violation for this solution is 15.

Theorem 10. *Algorithm 3 solves* MSP *in* $O(nm)$*-time.*

Proof. We first comment on the different steps in Algorithm 3. Clearly, since the league size k is even, the construction in Steps 1 and 2 implies that each pseudo club contains exactly two teams. Further,



(a) A 2-factorization of G

(b) Labeled and directed G

Figure 4: The graphs and 2-factors associated with Example 4

the construction in Step 3 implies that G is k-regular, and thus 2-factorable (since k is even). Notice that the case where two teams of the same club play in the same league amounts to a loop in G, and will result in these two teams receiving complementary HAPs. We refer to Lovász and Plummer (2009) for details regarding finding such a 2-factorization.

We now show correctness of Algorithm 3. Consider a solution obtained by the algorithm. Each pair of teams that make up a pseudo club use complementary patterns, and hence, they jointly play one home match in each round. Thus, if $\delta_c \geqslant \frac{n_c}{2}$, i.e., if club $c \in C \setminus C^-$, there are no violations for club c. In addition, if $\delta_c < \frac{n_c}{2}$, i.e., if $c \in C^-$ then the number of violations of club c equals:

$$2\,(k-1)\,(n_c/2-\delta_c)\,.$$

Using (43), it follows that the value of the solution found by the algorithm equals Q, and is thereby necessarily optimal. Note that this implies the second equality of Theorem 9.

To establish the complexity of Algorithm 3, observe that in the first step, a new instance is generated where each club consists of exactly two teams. This is done in $O(n)$-time. In the second and third step, a graph G is constructed which is done in $O(n)$-time and then a 2-factorization of G is computed which is done in $O(nm)$-time.

Finally, the 2-factorization is mapped to a solution for the original instance, which is done in $O(n)$-time. Therefore, the algorithm runs in $O(nm)$-time. □

As mentioned earlier, in the proof of Theorem 10, the given HAP-set $\mathcal{H}$ (as long as it is complementary) has no impact, neither on the optimal solution nor on the minimum violation. Further, the proofs of Theorems 9 and 10 imply the following corollary.

Corollary 2. $Q = \nu_{LP} = \nu_{IP}$.

4.5 TWO GENERALIZATIONS OF MSP

In this section, we investigate two generalizations of MSP. In Section 4.5.1, we consider an extension of MSP where all teams from the same club must play according to the same HAP; we refer to this generalization as MSPidHAP. Next, in Section 4.5.2, we deal with an extension of MSP in which capacities are not necessarily constant over the rounds, which we call MSPwVC. We motivate both generalizations, and we show that both problems are NP-hard for $k \geqslant 4$, and give polynomial-time algorithms for the case $k = 2$, when, in case of MSPwVC, each club consists of two teams. Observe that a league size of $k = 2$ may occur in knock-out tournaments, or play-offs, where two matches decide upon the winner of a pair of teams.

4.5.1 MSP with identical HAPs (MSPidHAP)

In a setting where the capacity of clubs is not an issue, clubs may want that *all* their teams play home in the same round. There can be various reasons for this wish: for instance to create a lively atmosphere at the club's venues, or to minimize the number of times a venue is used, or, when clubs have two or more teams in one particular category (for instance a club has two amateur teams in the under 21-years-old age category), teams following the same HAP allow these teams to exchange players whenever they play home.

The input defining an instance of MSPidHAP consists of the set of teams, its two partitions (one into leagues, and one into clubs), and a feasible, complementary HAP-set. The question is: does there exist a *feasible assignment*, i.e., does there exist an assignment of teams to HAPs such that (i) all teams from a club receive the same HAP, and (ii) all teams from a league receive a different HAP? Of course, in an instance of MSPidHAP, it should not happen that two teams from a same club are in the same league, since this would clearly lead to a no-instance.

It is not difficult to see that, in case $k = 2$, this question can be answered efficiently as follows: build a simple undirected graph $G = (V, E)$ with a vertex for each club ($V = C$), and connect two vertices iff the corresponding clubs have a team in a same league ($E = L$). The existence of a feasible 2-coloring of the vertices of G decides whether the instance of MSPidHAP with $k = 2$ is a yes-instance or not. It is a fact that all teams of clubs corresponding to nodes colored with one color play according to HAP HA, and all teams of clubs corresponding to nodes colored with the other color play according to HAP AH. We record this observation formally.

Observation 5. *For $k = 2$, MSPidHAP is solvable in polynomial time.*

It is possible to extend Observation 5 to a situation where a set of pairs of teams that need the same HAP is given. However, when $k \geqslant 4$, MSPidHAP becomes more difficult.

Theorem 11. *MSPidHAP is NP-hard for each $k \geqslant 4$.*

Proof. We reduce edge coloring a 4-regular graph to MSPidHAP. In this reduction, we do not explicitly construct a feasible, complementary HAP-set. In fact, we assume that some HAP-set is specified; the proof works for any given HAP-set.

Consider now the following question: given a simple 4-regular graph $G = (V, E)$, does there exist a coloring of the edges using 4 colors such that no two adjacent edges receive the same color? This problem is known to be strongly NP-complete (Holyer, 1981; Leven and Galil, 1983).

Given a simple 4-regular graph $G = (V, E)$, we construct an instance of MSPidHAP as follows. There is a league $\ell \in L$ for each vertex in V, i.e., $L = V$. There is a club $c \in C$ for each edge $e = (v, v') \in E$, i.e., $C = E$; each club consists of two teams ($n_c = 2$), one playing in the league corresponding to node v, one playing in the league corresponding to node v'. Thus, there are $n = 2|E|$ teams. We claim that the existence of a 4-coloring of G corresponds to a feasible assignment of teams to HAPs and vice versa.

Suppose that a 4-coloring exists. Let each color correspond to a HAP. By assigning the two teams of a club to the HAP that corresponds to the color of the edge corresponding to those two teams, it becomes clear that the feasibility of the coloring implies that the four teams in each league received pairwise different HAPs, hence a feasible assignment exists.

Suppose a feasible assignment exists. Then all teams that play according to HAP i receive color i, $i = 1, \ldots, 4$; this results in a 4-coloring of G. □

4.5.2 MSP with variable capacities (MSPwVC)

Another generalization of MSP is the problem where clubs' capacities differ throughout the season. We refer to this problem as *MSP with variable capacities* (in short MSPwVC). In this generalization, instead of having a constant capacity δ_c for a club, we are given capacities $\delta_{c,r}$ that represent the number of matches that can be hosted by club c

in round r. The resulting problem can be formulated as an integer program by replacing Constraints (40) by:

$$z_{c,r} \geqslant \sum_{t \in \hat{T}_c} \sum_{h \in \mathcal{H}} x_{t,h} U_{h,r} - \delta_{c,r} \qquad \forall c \in C, r \in R. \tag{45}$$

The resulting formulation of MSPwVC becomes:

$$\left\{ \min \sum_{c \in C} \sum_{r \in R} z_{c,r} \;\middle|\; (38), (39), (41) - (42), (45) \right\}.$$

In Section 4.5.2, we provide, for the case where $k = n_c = 2$, a polynomial-time algorithm based on finding a min-cost circulation, and in Section 4.5.2 we show that the problem becomes NP-hard for $k \geqslant 4$.

MSP with variable capacities: the case $k = 2$

Consider an instance of MSPwVC consisting of clubs C, leagues L, teams T, capacities $(\delta_{c,1}, \delta_{c,2})$, that features $k = n_c = 2$ for all $c \in C$. Note that for this specific setting $m = |L| = |C|$.

First, we argue that we can restrict our attention to instances that are "connected", as explained hereunder. Indeed, we can represent such an instance by building a bipartite graph $H = (V_1 \cup V_2, E)$, where $V_1 = C$, $V_2 = L$ and $E = T$; thus, an edge $(v_1, v_2) \in E$ represents that a team from the club represented by $v_1 \in V_1$ plays in the league represented by $v_2 \in V_2$.

As $k = n_c = 2$ for all $c \in C$, the degree of each node in H equals 2, and hence the graph H consists of a collection of disjoint cycles. Clearly, we can restrict our attention to instances where H is a single cycle; we assume, without loss of generality, that by rearranging indices we have a set of clubs $C = \{c_1, c_2, \dots, c_m\}$ such that each club $c_i, i = 1, \dots, m-1$ has a team in league ℓ_i and a team in league ℓ_{i+1} and club c_m has a team in league ℓ_m and a team in league ℓ_1.

As the league size is $k = 2$, there are only two different HAPs a team can have, either HA or AH. As every club has only 2 teams, the capacity $\delta_{i,r}$ of club $c_i, 1 \leqslant i \leqslant m$, in round $r = 1, 2$ can be seen as either $0, 1$ or $\geqslant 2$. In fact, capacities whose value exceeds 2 can be set to 2 without any consequences; we thus assume $\delta_{i,r} \in \{0, 1, 2\}$ for all $i = 1, \dots, m$, $r = 1, 2$. It follows that for a particular club c_i there are nine possibilities for $(\delta_{i,1}, \delta_{i,2})$.

A solution to MSPwVC with $k = 2$ can be described as an *occupation* $(o_{i,1}, o_{i,2})$ specifying how many teams of club c_i play home in rounds 1 and 2 respectively; clearly $(o_{i,1}, o_{i,2}) \in \{(2,0), (1,1), (0,2)\}$. We say that an occupation is *ideal* for club c_i when it results in a minimum number of violations over the two rounds given its capacities

$\delta_{i,1}, \delta_{i,2}$	0	1	2
0	$\{(2,0),(1,1),(0,2)\}$	$\{(1,1),(0,2)\}$	$\{(0,2)\}$
1	$\{(2,0),(1,1)\}$	$\{(1,1)\}$	$\{(1,1),(0,2)\}$
2	$\{(2,0)\}$	$\{(2,0),(1,1)\}$	$\{(2,0),(1,1),(0,2)\}$

Table 27: Ideal occupations $(o_{i,1}, o_{i,2})$ of club i when given capacities $\delta_{i,1}, \delta_{i,2}$

$\delta_i = (\delta_{i,1}, \delta_{i,2})$. Table 27 gives, for each of the nine possibilities for $(\delta_{i,1}, \delta_{i,2})$ the set of ideal occupations.

From Table 27, we see that the occupation $(o_{i,1}, o_{i,2}) = (1,1)$ is ideal for all capacities except when $(\delta_{i,1}, \delta_{i,2}) = (2,0)$ or $(0,2)$, $1 \leqslant i \leqslant m$. This observation forms the basis of our approach which, informally said, will use the occupation $(o_{i,1}, o_{i,2}) = (1,1)$ for each club c_i as a baseline solution, and next will find a maximum number of saved violations by modifying the occupation of appropriately chosen clubs to $(2,0)$ or $(0,2)$.

We now describe the construction of a directed graph $G = (V, A)$ that is instrumental in our procedure to solve the problem. The vertex set consists of $V = L \cup \{v_0\}$, where vertex v_i corresponds to league $\ell_i \in L$, $(i = 1, \ldots, m)$. The arc set $A = A_1 \cup A_2 \cup A_3$ is defined as follows:

$$
\begin{aligned}
A_1 &= \{(v_i \to v_{i+1}) : i = 1, \ldots, m-1\} \cup \{(v_m \to v_1)\}, \\
A_2 &= \{(v_0 \to v_i) : i = 1, \ldots, m\} \text{ and} \\
A_3 &= \{(v_i \to v_0) : i = 1, \ldots, m\}.
\end{aligned}
$$

To each arc $a \in A$, we associate a capacity cap(a), and a cost-coefficient cost(a). We set cap$(a) = 1$ for each $a \in A$. The costs are defined as follows:

- for each $a \in A_1$, cost$(a) = 0$,
- for each $a_{0,i} = (v_0 \to v_i) \in A_2$ $(1 \leqslant i \leqslant m)$,

$$
\text{cost}(v_0 \to v_i) = \begin{cases} -1 & \text{if } \delta_i = (2,0), \\ 0 & \text{if } \delta_i \in \{(0,0),(1,0),(2,1),(2,2)\}, \\ 1 & \text{if } \delta_i \in \{(0,1),(0,2),(1,1),(1,2)\}. \end{cases}
$$

- for each $a_{i,0} = (v_i \to v_0) \in A_3$ $(1 \leqslant i \leqslant m)$,

$$
\text{cost}(v_i \to v_0) = \begin{cases} -1 & \text{if } \delta_i = (0,2), \\ 0 & \text{if } \delta_i \in \{(0,0),(0,1),(1,2),(2,2)\}, \\ 1 & \text{if } \delta_i \in \{(1,0),(1,1),(2,0),(2,1)\}. \end{cases}
$$

We claim that this particular definition of the cost-coefficients for arcs in A_2 (respectively, A_3) captures the number of violations saved when

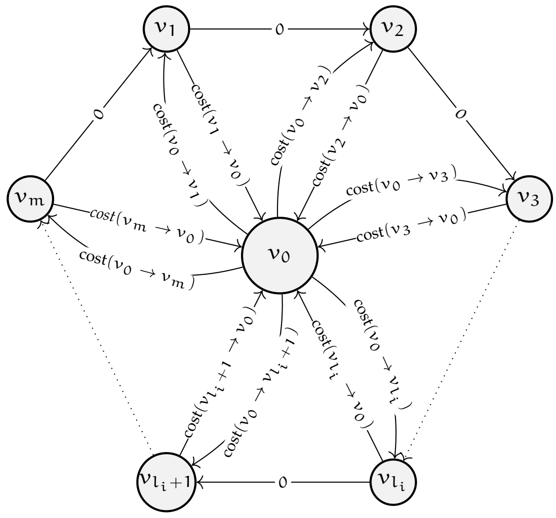


Figure 5: Directed graph G in Section 4.5.2

instead of using occupation $(1, 1)$ occupation $(2, 0)$ (respectively, $(0, 2)$) is used for club c_i with capacity $\delta_i = (\delta_{i,1}, \delta_{i,2})$ - this claim can be verified using the entries given in Table 27. Indeed, as an example, if the capacity of some club c_i equals $(2, 0)$, then the number of violations saved when using occupation $(2, 0)$ instead of occupation $(1, 1)$ equals 1; this is reflected in the -1 value of cost$(v_0 \to v_i)$ when $\delta_i = (2, 0)$. Figure 5 depicts the above-described graph G.

A circulation is a flow such that for each node, the amount of flow entering and leaving are equal.

We now state Algorithm 4 that computes a minimum cost circulation in graph G. Obtaining a minimum cost circulation can be done in polynomial time, see Ahuja, Magnanti, and Orlin (1988).

Algorithm 4

Input: Clubs C, Teams T, Leagues L, capacities $(\delta_{i,1}, \delta_{i,2})$

1: Build graph G as described above.
2: Solve a min-cost circulation problem on G, getting flow $y(a) \in \{0, 1\}$ for each arc $a \in A$.
3: Set $x_i := 1$ for $i = 1, \ldots, m$.
4: For each arc $(v_0 \to v_i) = a \in A_2$ for which $y(a) = 1$: (i) $x_i := 0, j := i$ (ii) WHILE $y(v_j \to v_{j+1}) = 1$ DO $x_j := 0, j := j + 1$.

Output: $(x_1, \ldots, x_m)$, where $x_i := 0$ (1) indicates that in league ℓ_i the team from club c_i (c_{i-1}) first plays at home.

Theorem 12. *Algorithm 4 solves* MSPwVC *in polynomial time when* $k = 2$ *and* $n_c = 2$ *for* $c \in C$.

The minimum cost circulation q is at least 0 since there is always a circulation with no flow, i.e. $y(a) = 0$ for all a.

Proof. The value of a solution to an instance of MSPwVC with $k = 2$ and $n_c = 2$ for $c \in C$, is nothing else but the total number of violations induced by the occupations of the clubs. Consider a solution where each club has occupation (1,1) - we will refer to this solution as the *baseline* solution, and we denote its value by B. Further, let the value of a minimum cost circulation in G (found in Step 2 of Algorithm 4) be denoted by q.

We prove the theorem by showing an equivalence between a minimum cost circulation in G with value q, and the existence of a solution with value $B+q$.

$\Rightarrow$ Consider an optimum solution to the circulation problem in G. Since, for each $i = 1, \dots, m$:

$$\text{cost}(v_0 \to v_i) + \text{cost}(v_i \to v_0) \geqslant 0,$$

it follows that there exists an optimum solution that does not have a unit flow using the two arcs $(v_0 \to v_i)$ and $(v_i \to v_0)$. Hence, an optimum circulation consists of cycles in G, each cycle carrying one unit of flow, such that each node of G, except v_0, occurs in at most one cycle; such a cycle can be expressed as follows: $v_0, v_i, v_{i+1}, \dots, v_j, v_0$.

The cost of an individual cycle depends solely on the costs of the two arcs (v_0, v_i) and (v_j, v_0). Notice that these costs represent, by definition, the savings in the number of violations when the occupation of club c_i (c_j) becomes (2,0) ((0,2)) instead of (1,1). Thus, a circulation in G with cost q leads to a solution of the problem with cost $B+q$.

$\Leftarrow$ In this step we show that any solution of our problem corresponds to a circulation in the graph G. As described before, a solution can be seen as the set of occupations for the clubs; associate the occupation of club c_i to node v_i in G. We claim that a solution is feasible if and only if occurrences of occupations $(2,0)$ and $(0,2)$ alternate along the cycle defined by the arcs in A_1. Given this fact, we can associate a circulation to each solution in the following way.

Let x_i denote the schedule of the league ℓ_i, where $x_i = 0$ indicates that league ℓ_i has a schedule in which the team from club c_i first plays at home, and $x_i = 1$ if the league has a schedule in which the team from club c_{i-1} will first play at home. Notice that the occupation of a club c_i can be expressed as $(1 - x_i + x_{i-1}, 1 + x_i - x_{i-1})$.

Given any solution $x = (x_1, \dots, x_n)$, we create index sets $I_j, j = 1, 2, 3$ and a flow y on the edges of the graph G in the following way:

$$\begin{array}{ll} I_1 = \{i : x_i = 0, x_{i-1} = 1\} & y(v_0 \to v_i) = 1 \quad i \in I_1 \\ I_2 = \{i : x_i = 0\} & y(v_i \to v_{i+1}) = 1 \quad i \in I_2 \\ I_3 = \{i : x_i = 1, x_{i-1} = 0\} & y(v_i \to v_0) = 1 \quad i \in I_3 \end{array}$$

The flow y created this way - where $x_0 = x_n$ - has cost:

$$\sum_{i \in I_1} \text{cost}(v_0 \to v_i) + \sum_{i \in I_3} \text{cost}(v_i \to v_0).$$

All clubs $i \in I_1$ have occupation $(2,0)$, all clubs in I_3 have occupation $(0,2)$, while all other clubs have occupation $(1,1)$. By construction of the graph, the cost of the circulation corresponds exactly with the difference in capacity violation of the solution x compared to a schedule in which all clubs have occupation $(1,1)$. Therefore, minimizing the cost of the circulation minimizes the number of violations. Hence, Algorithm 4 is an exact algorithm. □

MSP with variable capacities: the case $k \geqslant 4$

In the previous section, we saw that MSPwVC is solvable in polynomial time when league-size equals 2. We now show that MSPwVC is NP-hard when $k \geqslant 4$.

Theorem 13. MSPwVC *is NP-hard for each* $k \geqslant 4$*.*

Proof. For our reduction we use a problem known as the *restricted timetabling problem* (in short, RTT), proven to be NP-complete in Even, Itai, and Shamir (1975).

We first describe the RTT using their terminology. We are given a set of exactly three time slots (hours) $\Pi = \{\pi_1, \pi_2, \pi_3\}$, a set of teachers $\mathcal{T}$ and a set of classes V (a class refers to a group of students). Classes are always available, whereas teachers have a given availability, i.e., for each teacher $\tau \in \mathcal{T}$, there is a set of available time slots $\Pi_\tau \subseteq \Pi$.

We are also given a set S of courses, each of which must be taught by a specific teacher τ to a specific class ν during anyone of the three time slots. We denote courses by pairs (τ, ν). At most three courses are taught to each class and every teacher is either a *tight 2-teacher* or a *tight 3-teacher*. A teacher is a *tight α-teacher* if he/she teaches exactly α courses and is available for exactly α time slots, $\alpha \in \{2, 3\}$. We denote the number of courses taught to a class ν by $\rho(\nu)$, $\nu \in V$.

The question is whether there exists an assignment of time slots to each course (τ, ν) such that teachers' availabilities are satisfied and there are no overlaps (i.e., the courses taught by the same teacher are assigned to different time slots and the courses corresponding to each class are also assigned to different time slots).

Given an instance of RTT, we construct an instance of MSPwVC with clubs C, leagues L, teams T and capacities $\delta_{c,r}$ as follows.

Each class $\nu \in V$ is associated with a league $\ell \in L$ and thus our instance has $m = |V|$ leagues.

Our instance has $\sum_{\nu \in V}(k - \rho(\nu)) + |\mathcal{T}|$ clubs: we associate a club of α teams to each tight α-teacher $\tau \in \mathcal{T}$ (the resulting set of clubs is denoted by C_1); the remaining $\sum_{\nu \in V}(k - \rho(\nu))$ clubs each have exactly one team (these clubs belong to subsets C_2 and C_3 such that $|C_2| =$

$m(k-3)$ and $|C_3| = \sum_{v \in V}(3-\rho(v))$; note how $C = C_1 \cup C_2 \cup C_3$). Our instance thus has $\sum_{v \in V}(k-\rho(v)) + |S|$ teams.

Each course $(\tau, v) \in S$ represents a team $t \in T$ that belongs to a club in C_1 which is associated with teacher $\tau \in \mathcal{T}$ and plays in the league corresponding to class $v \in V$. We distribute the teams of clubs in C_2 by placing $k-3$ teams of clubs in C_2 in each of the leagues. The remaining $\sum_{v \in V}(3-\rho(v))$ teams are members of clubs $c \in C_3$; we arbitrarily add these teams to leagues such that all leagues consist of k teams.

Consider a given complementary HAP-set $\mathcal{H} = \{h_1, ..., h_k\}$ with complementary pairs $(h_{2j-1}, h_{2j}), j = 1, ..., \frac{k}{2}$. We determine the capacity of clubs $c \in C_1$ as follows: first, we associate the HAP h_κ to time slot π_κ for $\kappa = 1, 2, 3$. Then for each club $c \in C_1$, we identify the set of HAPs which correspond to the time slots during which the teacher (that gave rise to club $c \in C_1$) is available. Recall that each teacher is available either in time slots $\{\pi_1, \pi_2, \pi_3\}$, or $\{\pi_1, \pi_2\}$, or $\{\pi_1, \pi_3\}$, or $\{\pi_2, \pi_3\}$. The capacity of a club $c \in C_1$ is determined by the available time slots. We have, for each $c \in C_1, r \in R$:

$$\delta_{c,r} = \sum_{h \in \mathcal{H}_c} U_{h,r}, \tag{46}$$

where $\mathcal{H}_c$ equals either $\{h_1, h_2, h_3\}$ or $\{h_1, h_2\}$, or $\{h_1, h_3\}$, or $\{h_2, h_3\}$, depending on the availabilities of the teacher giving rise to club $c \in C_1$.

We determine the capacity of a club $c \in C_2$ as follows. We partition C_2 into $k-3$ subsets $C_2^1, ..., C_2^{k-3}$ each containing m clubs such that the teams belonging to the clubs of subset $C_2^i, i = 1, ..., k-3$ all play in different leagues. Next, we set for each club $c \in C_2^i$, $i = 1, \ldots, k-3$, and each round $r \in R$:

$$\delta_{c,r} = U_{h_{i+3},r}.$$

Finally, for each club $c \in C_3$, we set $\delta_{c,r} = 1$ for each round $r \in R$. This completes the description of an instance of MSPwVC.

We now show that a solution to MSPwVC without any capacity violations corresponds to a yes-instance of RTT and vice versa.

Suppose that the instance of MSPwVC admits a solution without any capacity violation. In such a solution it must be the case that each team from a club in C_2 has been assigned the one HAP in the HAP-set that yields no capacity violation for this club; in other words, each team from club $c \in C_2^i$ is assigned to HAP h_{i+3} for $i = 1, \ldots, k-3$.

Consider now the teams from a club $c \in C_1$. This club has a capacity given by (46) which must be fully utilized in order to have no capacity violations. Hence, the only set of patterns that satisfy this requirement are those patterns in $\{h_1, h_2, h_3\}$ that correspond with club $c \in C_1$, and we assign the teams accordingly. Teams from clubs in C_3 receive any remaining pattern. Based on this assignment of teams to HAPs, we can assign time slots to courses in RTT.

The resulting assignment is feasible since (1) each team in a league is assigned to a different HAP, thus the courses taught to each class are assigned to different time slots, (2) the teams from a single club are assigned to different HAPs, thus the courses taught by the associated teacher are assigned to different time slots.

If the instance of RTT is a yes-instance, we simply copy the existing assignment of courses to time slots to the instance of MSPwVC, where the assignment of teams of clubs in C_1 to the given HAPs h_1, h_2, h_3 follows directly from the solution to the RTT instance. Further, we give each team of a club in C_2 its corresponding pattern, and each team in C_3 any remaining pattern. This gives no capacity violations in the instance of MSPwVC. □

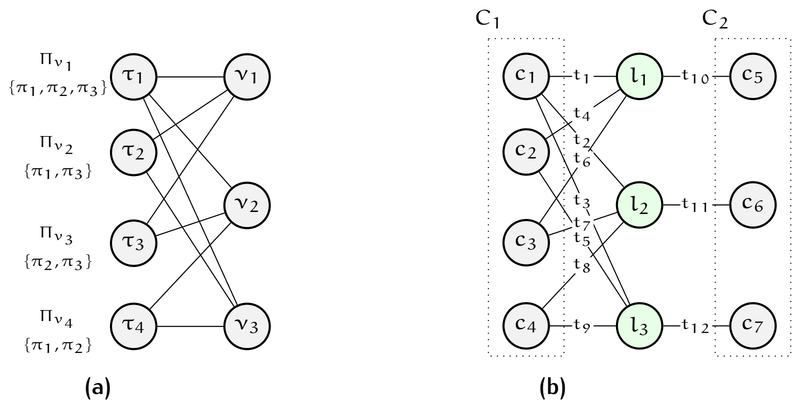


Figure 6: Graph representations for the proof of Theorem 12; (a) A graph representation for the instance of RTT. (b) a graph representation of the instance of MSPwVC.

As an illustration of the reduction in Theorem 12, consider the following instance of RTT: there are four teachers $(\tau_1, \tau_2, \tau_3, \tau_4)$, three classes (ν_1, ν_2, ν_3) and three time slots (π_1, π_2, π_3). Teacher τ_1 teaches different courses to all classes and is available on all time slots. Teacher τ_2 teaches only to classes ν_1 and ν_3 and is available on time slots π_1 and π_3. Teacher τ_3 teaches only to classes ν_1 and ν_2 and is available on time slots π_2 and π_3. Finally, teacher τ_4 teaches only to classes ν_2 and ν_3 and is available on time slots π_1 and π_2. Figure 6a shows a graph representation of this instance.

Assuming $k = 4$, we construct an instance of MSPwVC with 3 leagues, 7 clubs and 12 teams: $L = \{\ell_1, \ell_2, \ell_3\}$, $C = \{c_1, ..., c_7\}$ and $T = \{t_1, ..., t_{12}\}$ where $C_1 = \{c_1, c_2, c_3, c_4\}$ and $C_2 = \{c_5, c_6, c_7\}$. The clubs and leagues

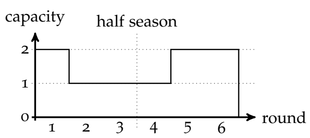
		$\hat{T}_c$	δ_c
C_1	c_1:	$\{t_1, t_2, t_3\}$	Figure 8a
	c_2:	$\{t_4, t_5\}$	Figure 8b
	c_3:	$\{t_6, t_7\}$	Figure 8c
	c_4:	$\{t_8, t_9\}$	Figure 8d
C_2	c_5:	$\{t_{10}\}$	Figure 8e
	c_6:	$\{t_{11}\}$	Figure 8e
	c_7:	$\{t_{12}\}$	Figure 8e

(a) Club in I

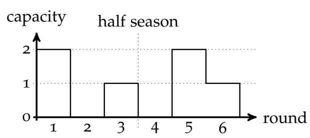
	$\bar{T}_l$
ℓ_1:	$\{t_1, t_4, t_6, t_{10}\}$
ℓ_2:	$\{t_2, t_7, t_8, t_{11}\}$
ℓ_3:	$\{t_3, t_5, t_9, t_{12}\}$

(b) Leagues in I

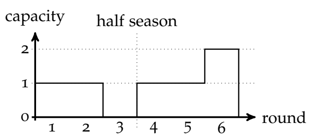
Figure 7: The instance I associated with the example in Theorem 12



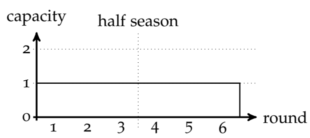
(a) Capacity of club c_1



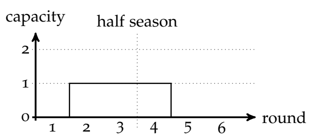
(b) Capacity of club c_2



(c) Capacity of club c_3



(d) Capacity of club c_4



(e) Capacity of clubs c_5, c_6 and c_7

Figure 8: Capacity of the associated clubs

are thus given in Figure 7a and 7b. Also, Figure 6b provide a graph representation of the clubs, leagues and teams. The capacity profiles of clubs are given in Figure 8 and are based on the HAP-set given in Example 4. As an example, the capacity profile of club c_2 (that is associated with teacher τ_2 for which $\Pi_{\tau_2} = \{\pi_1, \pi_3\}$) is $\delta_c = (U_{h_1,1} + U_{h_2,1}, ..., U_{h_1,6} + U_{h_2,6}) = (1+1, ..., 0+1) = (2,0,1,0,2,1)$ (see Figure 8b).

There is a solution with objective value of zero for this instance which is obtained by assigning $h_1 \to t_2, t_4, t_9$, $h_2 \to t_3, t_6, t_8$, $h_3 \to t_1, t_5, t_7$, and $h_4 \to t_{10}, t_{11}, t_{12}$. Hence, the given instance of RTT is a yes-instance.

Part II

FAIRNESS

5 MINIMIZING THE CARRY-OVER EFFECT

When Fabian Caruana was on the way to win his maiden crown at the Tata Steel Chess Tournament of 2020, commentators and spectators noticed something peculiar about his opponents. Of the 13 players he faced, 10 of them had played world champion Magnus Carlsen the round before. Being able to play against those who possibly had to give it all just the day before to hold off Carlsen, might have been an advantage for then world number 2 Caruana. As grand master Peter Svidler stated it, the American was able to *pick up* the opponents of Carlsen.
That Caruana was directly following Carlsen, was a coincidence. That someone was following Carlsen, was not. The scheme used by the Tata Steel Chess Tournament - one of the most prominent annual chess tournaments - is generated by the Circle Method. In the schedules created, all but one player are following another player, meeting the opponent of this other player in the subsequent round. The player Caruana turned out to follow, just happened to be Carlsen.
Following anyone might not be an advantage to everyone, but it can be to some. To prevent eventual unbalanced outcomes or advantages caused, ideally one makes a schedule such that players follow no particular player anymore than the others.
The COE is a measure for the lack of variety of opponents preceding a player. Schedules with high carry-over, have less variety and are thus more prone to unwanted effects. Finding good schedules with low carry-over, is difficult. The Circle Method generates the highest COE possible. However, modifying this method, may yield schedules with the lowest COE possible.

5.1 INTRODUCTION

One of the nice properties of a Round Robin competitions, is that every team or player, meets every competitor an equal number of times. This ensures in a way, that a player cannot have too much advantage of the draw, which could happen in for example a knock-out tournament.

However, matches are not isolated events that have no impact on those involved. When playing football for instance, facing a physical opponent might lead to injuries of your players. As rounds typically follow each other in quick succession, this can affect the squad and line-up for the next match(es).

Of course, semi-random events are that what makes sports fun, so if a team is lucky enough to gain some advantage because of the opponents previous match, nothing is lost. It does get tricky when this tends to happen structural, and to some teams more often than others in the competition.

The dangers that such a thing could occur, are particularly large when one team consistently plays the opponents of a specific other team in the next round. Indeed, if that specific team regularly injures more of the other sides players than other teams, over the season this accumulates to a significant advantage for the team that plays their opponents the next round.

Therefore, when scheduling a competition, it would be nice if it is possible to somehow evenly distribute the opponents previous opponent, the carry-over pairs, over all pairs of teams. As a measure of how even this distribution is, Russel (1980) introduced the *Carry-Over Effect*. When the evaluation of this quadratic function is low, this indicates that a lot of different carry-over pairs occur in the schedule. On the other hand, when the COE is high, this indicates a more monotonous schedule with a few pairs that are frequently occur, like it was the case in the TATA Steel Chess Tournament.

After the introduction of the concept of carry-over scheduling, it has been studied extensively. Since 1980, interesting results on finding balanced schedules were shown by for example Anderson (1999), Trick (2000) and Kidd (2010). Miyashiro and Matsui (2006) conjectured that the Circle Method would be the schedule with the highest Carry-Over Effect, and this was later proven by Lambrechts et al. (2018). It has been studied and used in practical applications, see for instance G. Durán, Guajardo, and Sauré (2017). Besides that, a lot of research is conducted in the field of the latin squares, and most notably Keedwell (2000) linked the search for particular latin squares to the search of balanced schedules in sport scheduling.

Different approaches to finding optimal - i.e. perfectly balanced schedules with lowest COE - have been proposed. Integer programming as well as constraint programming are popular methods, while Guedes and Ribeiro (2011) introduced a heuristic approach. Tabu-search was used by Kidd (2010).

The challenge of finding/creating schedules that have the lowest possible COE is still mostly unsolved, however. In fact, only for some values $2n$ - most notably when $n = 2^k$ - a proven optimal solution is known. For a schedule on as little as 12 opponents, it is still an open question what the lowest COE is - it is somewhere between 132 and 160, where 132 is the natural lower bound given by $12 \cdot 11$ and a schedule with a COE of 160 is mentioned to be found by Guedes and Ribeiro (2011).

We apply and refine a method introduced by Anderson, using *starters*, to generate schedules of a specific type. As nicely explained in Kidd (2010), for these schedules the COE can be quickly evaluated. This helps us in the search on relatively large competitions, where we show that we can find (almost) balanced schedules quickly. Our refinement of the starters, is that we don't look at any starter, but again focus our attention to a specific subset of starters, we call *mirrored*.

By exploiting their structure we are able to find schedules for a large number of teams, that are (almost) balanced.

5.2 DEFINITIONS AND TERMINOLOGY

We are dealing with SRR tournaments on N teams. The matches of the SRR need to be scheduled over $2n-1$ rounds, where each team plays one match per round. Hence, we can write any schedule as $S = (S_r)_{0 \leqslant r \leqslant 2n-2}$, with S_r the matches scheduled in round r. Notice that we do not take into account any Home/Away-patterns or any other constraints. In a feasible schedule S, every round S_r is a *perfect matching* or 1-factor of the set of teams, and every pair of teams is an element of exactly one S_r for some round r.

We are going to look at the rounds in a cyclic manner, thus following round $2n-1$ is round 1. The teams and rounds are defined as follows:

$$\mathcal{T} = \mathbb{Z}_{N-1} \cup \{\infty\} \qquad\qquad R = \mathbb{Z}_{N-1}.$$

In this chapter we extensively use elements of $\mathbb{Z}_{N-1}$, which is the additive group on $\{0, \dots, N-2\}$. In this group, we can add two elements $z, z' \in \mathbb{Z}_{N-1}$ where the result is given by:

$$\mathbb{Z}_{N-1} \ni y = z + z' \mod (N-1).$$

As we assume the number of teams to be equal to N and $\mathbb{Z}_{N-1}$ contains $N-1$ elements, we label the N-th team with ∞ to emphasize the special role it has in many of the constructions, and to avoid confusion.

We say Team t gives *carry-over* to Team s in round r, if there exists a Team u such that $\{t,u\} \in S_{r-1}$ and $\{s,u\} \in S_r$. For a schedule S, we define the carry-over from t to s, denoted with $c(t,s)$, to be:

Notice that every team gives exactly one carry-over per round, so the sum $\sum_{t,s} c(t,s) = N(N-1)$ for any schedule S.

$$c(t,s) = \#\{r \in R : \exists u \in \mathcal{T} \text{ s.t. } \{t,u\} \in S_{r-1},\ \{s,u\} \in S_r\}.$$

The total carry-over effect of schedule S, is defined as sum of the squares of all carry-overs.

$$\mathrm{COE}(S) = \sum_{t,s \in \mathcal{T}} c(t,s)^2. \tag{47}$$

The COE of any schedule on $N = 2n$ teams, has a natural lower bound $\mathrm{COE}(S) \geqslant N(N-1)$. When this lower bound is attained, it implies that $c(t,s) = 1$ for all $t \neq s \in \mathcal{T}$ if $n \geqslant 2$ - when $n = 1$, the only possible schedule, that consists of one match, has $c(t,t) = 1$ for both teams. We say a schedule is *balanced* if $\mathrm{COE}(S) = N(N-1)$.

5.3 CREATING SCHEDULES

Since the introduction of COE, it has been a field of interest to find schedules on $2n$ teams with minimal COE. When introducing the measure in 1980, Russel showed that when $2n = 2^k$ for an integer k, it is possible to construct balanced schedules. Those were perceived to be the only known schedules for a while, until Anderson (1999) came up with balanced schedules for $n = 20, 22$. It turned out, however, that in another context these schedules where already found by Tripke (1983).

Both Anderson and Tripke applied the method of *clockwise scheduling* on something the former referred to as *starters*.

Definition 21 (Anderson (1999))**.** *Define $\mathbb{Z}^*_{N-1}$ as:*

$$\mathbb{Z}^*_{N-1} := \mathbb{Z}_{N-1} \setminus \{0\} = \mathcal{T} \setminus \{0, \infty\}.$$

A starter d *is given by*

$$d : \mathbb{Z}^*_{N-1} \to \mathbb{Z}^*_{N-1}. \tag{48}$$

With $d = (d_1 \dots d_{N-2})$ it is implied that $d(i) = d_i$.

Schedule $S(d) = (S_r(d))_{r \in R}$ consists of rounds $S_r(d)$ defined as:

$$S_r(d) := \{\{i, i + d(i-r)\} : i \in \mathbb{Z}_{N-1} \setminus \{r\}\} \cup \{\{r, \infty\}\}. \tag{49}$$

Lemma 4. *For any starter d, the following are equivalent:*

1. $S_0(d)$ *is a matching.*

2. $S_r(d)$ *is a matching for all* $r \in R$.

3. *For all* $i \in \mathbb{Z}^*_{N-1}$*:*

$$d(i) + d(i + d(i)) = N - 1 \equiv 0 \tag{50}$$

Proof. Suppose $S_0(d)$ is a matching. Then for every $\{i, i'\} \in S_0(d)$ with $i, i' \neq 0$, it must hold that $\{i, i + d(i)\} = \{i, i'\}$ and $\{i', i' + d(i') = \{i', i\}$, otherwise $S_0(d)$ would not be a matching. This means that $\{i + d(i) = i'$ and $i' + d(i') = i$. Combining these expressions leads to:

$$\begin{aligned} i + d(i) + d(i') = i &\implies d(i) + d(i') = 0 \\ &\implies d(i) + d(i + d(i)) = 0 \end{aligned}$$

Similarly, when $S_r(d)$ is a matching, for any that $\{i, i'\} \in S_r(d)$ we have $i' = i + d(i - r)$ and:

$$\begin{aligned} i + d(i-r) + d(i'-r) = i &\implies d(i-r) + d(i'-r) = 0 \\ &\implies d(i-r) + d(i - r + d(i-r)) = 0 \end{aligned}$$

As this holds for all values $i - r$, we get $d(j) + d(j + d(j)) = 0$.

Now suppose $d(i) + d(i + d(i)) = 0$ for all i and let $i \neq r$. We know that $\{i, i + d(i - r)\} \in S_r(d)$. On the other hand, $i + d(i - r)$ is coupled with

$$i + d(i-r) + d(i + d(i-r) - r) = i + d(i') + d(i' + d(i')) = i$$

So $S_r(d)$ is a matching, and $S_0(d)$ is a matching as well. □

Given a starter d, we get a schedule $S(d)$ on $2n - 1$ rounds. However, we are interested in the starters d for which the schedule $S(d)$ is not just any schedule, but an SRR. The following necessary and sufficient condition on d states for which d schedule $S(d)$ is an SRR.

Condition 2. *Schedule* $S(d) = (S_r(d))_r$ *for a starter* $d : \mathbb{Z}^*_{N-1} \to \mathbb{Z}^*_{N-1}$ *is an SRR on teams* $\mathcal{T}$ *if and only if the following holds:*

1. $S_0(d)$ *is a matching* and

2. d *is one-to-one.*

Proof. Obviously, for $S(d)$ to be an SRR, $S_0(d)$ has to be a matching, as any round in an SRR has to be a matching. By Lemma 4 we know that for all $r \in \mathbb{Z}_{N-1}$, the round $S_r(d)$ is in fact a matching. Vice versa, when $S(d)$ is an SRR, $S_0(d)$ is a matching too.

Suppose that $S_r(d) \cap S_{r'}(d) \neq \emptyset$ for some $r \neq r'$. Then there is a $\{i, i'\} \in S_r(d)$ that is also in $S_{r'}(d)$ and both i, i' cannot be ∞. This implies that $i + d(i-r) = i + d(i-r') = i'$, thus $d(i-r) = d(i-r')$ and d is not one-to-one.

If $S_r(d) \cap S_{r'}(d) = \emptyset$ holds for all pairs r, r', we have $N-1$ disjoint matchings consisting of $\frac{N}{2} = n$ pairs - so in total, we have $n(N-1)$ pairs. As there are exactly $n(N-1)$ disjoint pairs in $\binom{\mathcal{T}}{2}$, $S(d)$ must be an SRR. □

Definition 22. *A* good starter *is a starter* d *for which* $S(d)$ *is an SRR.*

The definitions and lemma's presented here so far were rather formal. However, there is a very natural and intuitive way of expressing the procedure of constructing $S(d)$ when given a (good) starter d. For instance, the schedule created with the Circle Method, can be constructed with a starter, for instance.

When there are 8 teams, the circle method coincides with starter $d = (531642)$ where $S_0(d) = \{\{0,\infty\},\{1,6\},\{2,5\},\{3,4\}\}$, as shown in Figure 2. Every round $S_r(d)$ of $S(d)$, is equal to the round given in Section 1.3. We see that constructing a schedule $S(d)$ from a starter d, is merely a generalisation of the Circle Method, where $S_0(d)$ can be varied compared to the canonical $S_0(d) = \{\{0,\infty\},\{1,N-2\},\{2,N-3\},\dots\}$.

In Figure 9, the round $S_0(d)$ is shown for three different d's. The first two, with $d = (531642)$ and $d = (416235)$, satisfy the conditions of Condition 2, while the third one $d = (225516)$ does not. Recall that all additions are done modulo $N-1$.

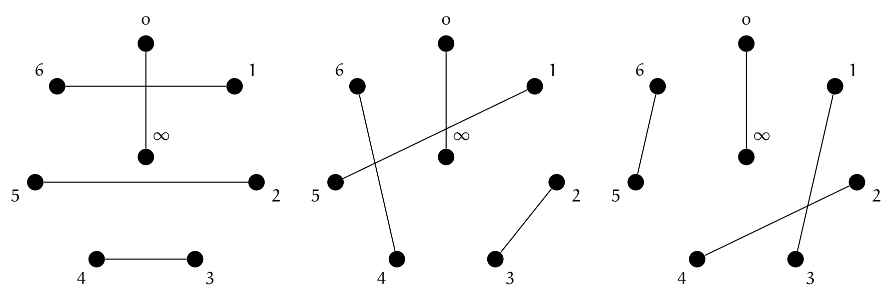


Figure 9: Three first rounds $S_0(d)$ for different d.

One of the main benefits of using starters and schedules $S(d)$, is that with very little initial information, an entire schedule can be created. All the information needed is encompassed in d. Moreover, derived metrics such as the COE can quickly be calculated solely from d. The following lemma gives an exact expression for the COE of $S(d)$.

Notice that all of i, $d(i)$ *and* $\Delta(i)$ *are elements of* $\mathbb{Z}_{N-1}$.

Lemma 5. *Let* d *be a good starter. Define* Δ_i *and* C_j *as:*

$$\Delta_i = d(i) - d(i-1) \qquad \forall i \in \mathbb{Z}_{N-1} \setminus \{0,1\}, \tag{51}$$

$$C_j = |\{i \colon \Delta_i = j\}| \qquad \forall j \in \mathbb{Z}_{N-1}. \tag{52}$$

The carry-over effect of schedule $S(d)$ *is then given by:*

$$COE(S(d)) = (N-1)\sum_{j=1}^{N-3} C_j^2 + 3(N-1). \tag{53}$$

Proof. We calculate the COE of $S(d)$ by case-distinction on the carry-overs between $\{i, i'\} \subset \mathcal{T}$.

Suppose $i \in \mathbb{Z}_{N-1}$ and $i' = \infty$, then both teams give carry-over to each other exactly once, $c(i, i') = c(i', i) = 1$.

Team i gives carry-over to ∞ in a round r if $\{i, r\} \in S_{r-1}(d)$, as $\{r, \infty\} \in S_r(d)$. This means that $r = i + d(i - r + 1)$, or $i - r + d(i - r + 1) = 0$. On the other hand, we know that $d(j) + d(j + d(j)) = 0$, and d is one-to-one. Combining this, we see that i gives carry-over to i' in round r whenever $i - r = d(-1)$, thus when $r = i + d(-1)$.

The other way around, the rounds r where ∞ gives carry-over to i, can be determined in similar fashion. This only occurs in rounds where $\{r - 1, i\} \in S_r$. For this, it must hold that $r - 1 = i + d(i - r)$ or $i - r + 1 + d(i - r + 1 - 1) = 0$. From this, we can conclude that $r = i + d(1)$.

For the next case, suppose $i \in \mathbb{Z}_{N-1}$ and $i' = i + 1$. Then, i gives carry-over to i' in round $S_{i'}$ via ∞.

The last cases are when $i, i' \in \mathbb{Z}_{N-1}$, but $i - i' \notin \{0, \pm 1\}$. Then i gives carry-over to i' in round r when there is a team j such that $\{i, j\} \in S_{r-1}$ and $\{i', j\} \in S_r$. For any team j, we know that it is matched to team $j + d(j - r)$ in round r, thus this only occurs when:

$$\begin{aligned} i = j + d(j - r + 1) \quad \& \quad i' = j + d(j - r) \implies \\ i - i' = d(j - r + 1) - d(j - r) = \delta_{j-r+1} \end{aligned}$$

So for every $r' \in C_{i-i'}$, there is a round r where i gives carry-over to i'. Ergo, for every $i \in \mathbb{Z}_{N-1}$, we know the team gives carry-over to team i' a total of $C_{i-i'}$ times, for all $i' \notin \{\infty, i + 1\}$.

Combining these three cases, we see that:

$$\begin{aligned} \mathrm{COE}(S(d)) &= N - 1 + N - 1 + N - 1 + (N-1)\sum_{j=1}^{N-3} C_j^2 \\ &= (N-1)\left(3 + \sum_{j=1}^{N-1} C_j^2\right) \end{aligned}$$

□

With Lemma 5, we can quickly calculate what the COE of $S(d)$ is - it all depends on the values $\{\Delta_i : i \in \{2, N-2\}\}$.

Definition 23. *A schedule is* balanced *when for all pairs* $\{i, i'\} \subset \mathcal{T}$, *we have* $c(i, i') = 1$. *We say a good starter is* balanced *if* $S(d)$ *is a balanced schedule. In such a balanced starter, is must hold that* $C_j = 1$ *for all* $j \in \{1, \dots, N-3\}$ *or equivalently, that for all* $j \in \{1, \dots, N-3\}$ *there is a unique* $i \in \{2, \dots, N-2\}$ *such that* $\Delta_i = j$.

Not all good starters are balanced, in fact, most are not. When d is the starter with $S(d)$ equal to the one created by the Circle Method, we see that $\Delta_i = N-3$ for all $i \in \{2, \dots, N-2\}$, which leads to $\text{COE}(S(d)) = (N-1)((N-3)^2+3)$. This is proven to be maximum by Lambrechts et al. (2018).

The following example should give some intuition on how the Δ_i can be used to calculate the COE of a schedule $S(d)$.

Example 6. *Let* d *be given by* $d = (62571834)$, *a good starter on* 10 *teams. In Table 28, we see for every* $i \in \mathbb{Z}_9^*$ *the value* $d(i)$ *and when* $i \geqslant 2$, *values* $\Delta_i = d_i - d_{i-1}$ *as well. For this starter, one value occurs twice, namely* $\Delta_i = 3$ *when* $i = 3, 5$. *All other possible values* $\{1, \dots, 7\}$ *all occur once, except* 6, *which has no occurence. Thus, we see that COE*$(S(d))$ *equals* $9(5 \cdot 1 + 1 \cdot 4 + 3) = 108$.

i	1	2	3	4	5	6	7	8
$d(i)$	6	2	5	7	1	8	3	4
Δ_i	–	5	3	2	3	7	4	1

Table 28: Δ values of starter (62571834)

5.4 SCHEDULES WITH LOW COE

It is an open question for which values of N, balanced schedules exist. The following is known:

Fact 1 ((Anderson, 1999; Russel, 1980))**.** *For SRR schedules on* $N = 2n$, *there exist balanced schedules when:*

- $N = 2^k$ *with* $k \in \mathbb{N}$.
- $N = 20, 22$

Furthermore, when $N = 6$ *or* $N = 10$, *no balanced schedule exists. For all values* N *where balanced schedules exist, there is a starter* d *on* N *teams that generates a balanced schedule.*

Especially encouraged by the last statement in Fact 1, it is a popular approach to look for schedules with low COE by trying to find a starter d that gives the best $\text{COE}(S(d))$ among all starters. This approach is also beneficial in another way, namely that the COE of a schedule $S(d)$ can be calculated quickly. It is possible to formulate the search for the best starter as an IP, and attempt an exhaustive search among all the starters.

Among others, Anderson and Kidd, tried to find starters with low COE. The best known starters for small values of N are shown in Table 29, together with the natural lower bound $N(N-1)$ and the reportedly lowest known value of a schedule on N teams.

N	$N(N-1)$	Best known	Best Starter	Value
4	12	12	$(1,2)$	12
6	30	60	$(3,1,4,2)$	60
8	56	56	$(4,1,6,2,3,5)$	56
10	90	108	$(6,2,5,7,1,8,3,4)$	108
12	132	160	$(3,4,5,8,2,7,9,6,1,10)$	176

Table 29: Best known starters and their values

In Table 29 we see some balanced starters, having a COE equal to $N(N-1)$. As they are of special interest to us, we enlist a few other balanced starters in Table 30, where $N = 16$ and the peculiar $N = 20$.

N	Starter																				
8	i	1	2	3	4	5	6														
	d(i)	4	1	6	2	3	5														
16	i	1	2	3	4	5	6	7	8	9	10	11	12	13	14						
	d(i)	3	6	11	12	5	7	2	9	13	10	1	14	8	4						
20	i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18		
	d(i)	3	7	15	16	8	5	10	6	12	2	14	17	11	13	1	18	9	4		
22	i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	d(i)	8	3	16	6	18	11	7	12	13	15	4	1	20	14	17	2	10	19	5	9

Table 30: Perfectly balanced starters

5.5 MIRRORED STARTERS

Searching for balanced schedules within starters is somewhat efficient, as there are way fewer starters than there are schedules on N teams. Using an IP-implementation, it is possible to look for the starters with best COE for up to 30 teams within reasonable time. As was already discovered by (Kidd, 2010), no new perfectly balanced starters emerged within these numbers. Still, we would like to find new schedules that are perfectly balanced. Since the number of good starters grows exponentially with N, for larger values of N, we need to look at a smaller selection of starters.

Lemma 6. *For any balanced starter* d*, it must hold that:*

$$d(N-2) - d(1) = 1 \tag{54}$$

Proof. A starter d is balanced if every $\Delta_i = d(i) - d(i-1)$ is unique, for $i \in \{2, \dots, N-2\}$, which means that for every $j \in \{1, \dots, N-3\}$, there must be a i such that $\Delta_i = j$. Hence:

$$\sum_{i=2}^{N-2} \Delta_i = \sum_{j=1}^{N-3} j = \frac{(N-2)(N-3)}{2} = 1 \mod N-1.$$

On the other hand we see:

$$\begin{aligned} \sum_{i=2}^{N-2} \Delta_i &= d(N-2) - d(N-3) + d(N-3) - \dots + d(2) - d(1) \\ &= d(N-2) - d(1). \end{aligned}$$

From which we can conclude that $d(N-2) - d(1) = 1$. $\square$

In line with property (54), we will now define a particular class of starters that we call *mirrored*.

Definition 24. *A starter* d *is* mirrored *when:*

$$d(N-1-i) - d(i) = i \qquad \forall i \in \mathbb{Z}^*_{N-1}. \tag{55}$$

The starters given in Table 30 are all mirrored. To see why mirrored starters are called mirrored, we refer to Table 31 for the opponent schedule (who plays who in which round) of the schedule generated by the balanced starter on 8 teams. The opponents of the teams $0, \dots, N-2$ are mirrored over the diagonal axis.

Round/Team	0	1	2	3	4	5	6	∞
0	∞	5	3	2	6	1	4	0
1	5	∞	6	4	3	0	2	1
2	3	6	∞	0	5	4	1	2
3	2	4	0	∞	1	6	5	3
4	6	3	5	1	∞	2	0	4
5	1	0	4	6	2	∞	3	5
6	4	2	1	5	0	3	∞	6

Table 31: Balanced schedule on 8 teams

Lemma 7. *A starter* d *is good and mirrored if and only if for all* $x \in \mathbb{Z}^*_{N-1}$, *where* $x \to y$ *implies* $d(x) = y$, *we have:*

$$\begin{tikzcd} x \arrow[rr] & & y \arrow[dl] \\ & -x-y \arrow[ul] & \end{tikzcd} \qquad \begin{tikzcd} -x \arrow[dr] & & -y \arrow[ll] \\ & x+y \arrow[ur] & \end{tikzcd}$$

Proof. Suppose d is a good starter and mirrored, and let $d(x) = y$ for $x \in \mathbb{Z}^*_{N-1}$. Then:

- As d is a good starter, $d(x + d(x)) = d(x+y) = -d(x) = -y$.
- As d is a mirrored starter, $d(-x) = y + x$.
- As d is a good starter, $d(-x + d(-x)) = d(y) = -(y + x)$
- As d is mirrored, $d(-(x+y)) = d(x+y) + x + y = x$.
- As d is a good starter, $d(-(x+y) + d(-(x+y)) = d(-y) = -x$

To see that the other way around is also true, notice that $d(x) = y$ and $d(-x) = x + y$. Hence, $d(-x) - d(x) = x$, so d is mirrored. □

When d is a mirrored starter and we know the value of $d(x)$ for some $x \in \mathbb{Z}_{N-1}$, by Lemma 7 we know the values $d(x')$ for a set of other $x' \in \mathbb{Z}_{N-1}$ as well. It is in our interest to collect these x'.

Definition 25. *Let d be a mirrored starter and $x \in \mathbb{Z}^*_{N-1}$. Then $P(x) \subset \mathbb{Z}^*_{N-1}$ is given by:*

$$P(x) := \{x, y, x+y, -x, -y, -x-y\}$$

Notice that when d is a mirrored starter and $x' \in P(x)$, we see that $d(x') \in P(x)$ and $\forall x' \in P(x)$, there is $x'' \in P(x)$ such that $(x', x'') \in S_0(d)$.

In any mirrored starter d, teams in $P(x)$ are matched among each other in $S_0(d)$:

$$(x, x+y), (-x, y), (-x-y, -y) \in P(x).$$

This is illustrated in Table 32.

i	x	y	$x+y$	$-x-y$	$-y$	$-x$
$d(i)$	y	$-x-y$	$-x$	x	$-x$	$x+y$
$i+d(i)$	$x+y$	$-x$	x	$-y$	$-x-y$	y

Table 32: Teams and opponents of $P(x)$ in round 0.

Lemma 8. *Let d be a mirrored starter and $x, x' \in \mathbb{Z}^*_{N-1}$. Then:*

1. *If $P(x) \cap P(x') \neq \emptyset$, then $P(x) = P(x')$.*
2. *For all $x \in \mathbb{Z}^*_{N-1}$ and $P(x)$ we know that one of the following must be true:*
   - $|P(x)| = 6$
   - $|P(x)| = 2$ *with* $3x = 0$ *and* $d(x) = x$.

Proof. Notice that if d is a mirrored starter, $d^3(x) = x$, and $P(x) = \{d^i(x), d^i(-x)\colon 0 \leqslant i \leqslant 2\}$. If $x' \in P(x)$, then either $x' = d^i(x)$ or $x' = d^i(-x)$ for a specific $0 \leqslant i \leqslant 2$. Thus $P(x') = P(x)$.

We know that $P(x)$ has at most 6 different elements. Suppose it has less than 6. Then at least two of the pairs $(x, x+y), (-x, y), (-x-y, -y) \in S_0(d)$ must be the same, and wlog we assume that $(x, x+y)$ coincides with one of the other pairs. As x cannot equal $-x$ and $x+y$ cannot equal $-x-y$, either $x = y$ or $x = -x-y$.

When $x = y$ we get $x + y = -x$, implying $2x = -x$ thus $3x = 0$, and all three pairs are equal to $(x, 2x)$. When $x = -x-y$ we get $x+y = -y$ implying $x = y$ and again all three pairs are equal to $(x, 2x)$.

So whenever $|P(x)| < 6$, it contains only two elements, $P(x) = \{x, 2x\}$, with $3x = 0$. This finishes the proof. $\square$

Using Lemma 8, we see that d partitions the elements in $\mathbb{Z}^*_{N-1}$ in a certain way that we will define as $\mathcal{P}(d)$.

Definition 26. *Given a mirrored starter* d*, partition* $\mathcal{P}(d)$ *of* $\mathbb{Z}^*_{N-1}$ *is given by:*

$$\mathcal{P}(d) = \{P(x)\colon x \in \mathbb{Z}_{N-1}\}$$

A partition $\mathcal{P}$ *for which there is a* d *such that* $\mathcal{P} = \mathcal{P}(d)$*, is a* mirrored partition*. The space of all mirrored partitions is denoted with* $\mathbb{P}_N$*:*

$$\mathbb{P}_N = \{\mathcal{P}\colon \exists d \textit{ mirrored starter on } N \textit{ teams with } \mathcal{P}(d) = \mathcal{P}\}$$

By Lemma 8 we see that $\mathcal{P}(d)$ can only partition $\mathbb{Z}^*_{N-1}$ if it contains $6k$ or $6k+2$ elements. We formalize this in the following lemma:

As all balanced starters thus far have been mirrored, this may suggest that when $N = 6k$, no balanced starter can exist.

Lemma 9. *Let* d *be a mirrored starter. Then* $N = 6k+2$ *or* $N = 6k+4$*. When* $N = 6k$ *no mirrored starter exists,* $\mathbb{P}_{6k} = \emptyset$*.*

Notice that a mirrored starter d has a unique $\mathcal{P}(d)$, but different starters d, d' can have $\mathcal{P}(d) = \mathcal{P}(d')$. Indeed, given $x, y \in \mathbb{Z}^*_{N-1}$, with $y \neq \pm x$, we can either set $d(x) = y$ or $d'(x) = -x-y$. Then:

$$(x, x+y), (-x, y), (-y, -x-y) \in S_0(d)$$

or

$$(x, -y), (-x, -x-y), (y, x-y) \in S_0(d')$$

And $P(x) \in \mathcal{P}(d), \mathcal{P}(d')$.

For any $\mathcal{P}(d)$, we have 2^k different d' such that $\mathcal{P}(d') = \mathcal{P}(d)$. However, we can enumerate the elements in $\mathbb{P}_N$ $\mathcal{P}$ quick and efficient, and there are significantly less mirrored starters than regular good starters.

The following algorithm, Algorithm 5, can be used to list all $\mathcal{P} \in \mathbb{P}_N$.

Algorithm 5 Partitioning

Input: Order N and an ordered set of teams $T = \{t_1, \ldots, t_{|T|}\}$.

1: Set $x = t_1$, and create empty list L.

2: For every $t_i = y \in T$ such that $t_{i'} = -y$ with $i \leqslant i'$, set:

$$P(x, y) = \{\pm x, \pm y, \pm(x + y)\}.$$

When $P(x, y) \subset T$, $P(x, y)$ is a valid set.

3: For every valid set $P(x, y)$, run the algorithm on teams $T' = T \setminus P(x, y)$ to get L', all partitions of T'. Add set $P(x, y)$ to every element of L' to get partitions of T and add these to L.

Output: List L containing all mirrored partitions of set T.

For any mirrored partition $\mathcal{P}$ of teams containing k sets of size 6 (thus $N = 6k + 2$ or $N = 6k + 4$), there are 2^k mirrored starters that correspond to that partition - $|\mathcal{D}(\mathcal{P})| = 2^k$; as noticed earlier, per element $P \in \mathcal{P}$ of size 6, there are two ways to connect the teams for the first starting round for any mirrored starter d - P is either in state 0 or 1.

When $\overline{d} \in \mathcal{D}(\mathcal{P})$ is the complement of d, $COE(S(d)) = COE(S(\overline{d}))$.

Lemma 10. *Given a mirrored partition $\mathcal{P}$ on $N = 6k + 2$ or $N = 6k + 4$ teams, for each $P_i \in \mathcal{P}$, there are two matchings $m_0(i), m_1(i)$ on the elements of P_i such that either $m_0(i) \subset S_0(d)$ or $m_1(i) \subset S_0(d)$ among all $d \in \mathcal{D}(\mathcal{P})$. Given a choice m, distinguishing $m_1(i), m_0(i)$ for all i, for each $d \in \mathcal{D}(\mathcal{P})$ there is a unique $c(d) = (c_i(d))_{i \leqslant k} \in \{0, 1\}^k$ such that:*

$$c_i(d) = j \qquad\qquad \textit{where } m_j(i) \in S_0(d)$$

And for each $c \in \{0, 1\}^k$, there is a $d \in \mathcal{D}(\mathcal{P})$ such that $c = c(d)$.

Recall that our main goal is to find d for which $COE(S(d))$ is minimal. Clearly:

$$\min_{d \text{ mirrored starter}} COE(S(d)) = \min_{\mathcal{P} \in \mathbb{P}_N} \min_{d \in \mathcal{D}(\mathcal{P})} COE(S(d))$$

With Lemma 10 we see that we can enumerate all elements $d \in \mathcal{D}(\mathcal{P})$ as bits of length k. Moreover, when we want to calculate $COE(S(d'))$ when we know $COE(S(d))$, we only need to alter for the teams in P_i where $c(d)_i \neq c(d')_i$. It is possible to sort all 2^k k-bits in such a way that two consecutive terms only differ in 1 bit - such an ordering is known as a Gray code.

If we have such an ordering and d, d' are ordered consecutive to each other, we can calculate the carry-over of $S(d')$ from the carry-over of

$S(d)$ in a constant time. We determine the COE of all $d \in \mathcal{D}(\mathcal{P})$, and find $\min_{d \in \mathcal{D}(\mathcal{P})} \mathrm{COE}(S(d))$ in $\mathcal{O}(2^k)$-time, as $|\mathcal{D}(\mathcal{P})| = 2^k$.

Checking for the existence of balanced schedules, can even be done quicker than looping through all possible starters.

Together with Algorithm 5, we ran this to find all mirrored starters on N teams and find the lowest COE value among them. The results are shown in Table 33.

N(k)	Number of partitions	N(N-1)	Best mirrored
8(1)	1	56	56
10(1)	0	90	–
14(2)	1	182	234
16(2)	1	240	240
20(3)	4	380	380
22(3)	4	462	462
26(4)	15	650	750
28(4)	9	756	864
32(5)	64	992	992*
34(5)	50	1122	1254
38(6)	445	1406	1554*
40(6)	282	1560	1716*
44(7)	3091	1892	2064
46(7)	2178	2070	2250
50(8)	25760	2450	2646
52(8)	17477	2652	2856
56(9)	236520	3080	3300
58(9)	165376	3306	3534
62(10)	2482621	3782	4026
64(10)	1741131	4032	4032

Table 33: Best COE of mirrored starters.
* indicates improvement on Kidd (2010) (up to $N = 40$).

When looking at all starters, the lowest COE a starter can have that is not balanced is $(N-1)(N+2)$.

We see in Table 33 that we found no new balanced starters. However, the alternative best starters we found for N all were imbalanced in the same way - they all had $\mathrm{COE}(s) = (N-1)(N+4)$. This indicates that every non-central team $i \in \{0, \ldots, N-2\}$ has two teams it gives carry-over to twice, and two teams it misses completely. It is not difficult to see that within mirrored starters that are not balanced, this is the best one can do.

5.6 GENERATING MIRRORED STARTERS

In the previous section, we introduced mirrored starters as subgroup of the starters, to look for schedules that are balanced. When enumerating them all, it turns out the number of mirrored starters grows exponentially in N, limiting our extensive search for balanced schedules for larger values of N. In this section, we present a more direct way to construct a partition, that looks promising when trying to find balanced schedules.

Let's recall the mirrored starter on $N = 16, 20, 22$ teams. As we know, for all these mirrored starters d there is a partition $\mathcal{P}(d)$ of the elements in $\mathbb{Z}^*_{N-1}$, and these partitions contains k sets P_i of size 6. For these N, the partitions $\mathcal{P} \setminus P^*$ consisting of sets P_i are shown in Table 34, where the elements of P_i are sorted in a specific way and we omitted the $P \in \mathcal{P}$ of size 2.

N	$\mathcal{P}$					
16	1	3	4	11	12	14
	2	6	8	7	9	13
20	1	3	4	15	16	18
	7	2	9	12	17	10
	11	14	6	13	5	8
22	1	8	9	12	13	20
	2	16	18	3	5	19
	4	11	15	6	10	17

Table 34: Partitions $\mathcal{P}$ of balanced starters

Looking at the consecutive rows per N in Table 34, each time there is a factor α by which two elements in consecutive rows differ. When $N = 16$ $\alpha = 2$, when $N = 20$ $\alpha = 7$, and when $N = 22$ $\alpha = 2$. For all these partitions, we see that if we know one of the parts $P \in \mathcal{P}$, together with α, we can generate $\mathcal{P}$ in its entirety.

Definition 27. *Let N be the number of teams. Let $P = \{\pm x, \pm y, \pm x + y\} \subset \mathbb{Z}^*_{N-1}$ be a set suitable to be in a mirrored partition of $\mathbb{Z}^*_{N-1}$. Let $\alpha \in \mathbb{Z}^*_{N-1}$ be an element of order ℓ, i.e., $\alpha^\ell = 1$. The* generator $G_\alpha(\cdot)$ *generates the following set of sets:*

$$\begin{aligned} G_\alpha(P) &= \{P^j_\alpha \colon 0 \leqslant j < \ell\} \\ &= \left\{\{\pm\alpha^j x, \pm\alpha^j y, \pm\alpha^j (x+y)\} \colon 0 \leqslant j < \ell\right\} \end{aligned}$$

We say $G_\alpha(\cdot)$ is a clean generator on P *when $G_\alpha(P)$ consists of disjoint sets:*

$$P^j_\alpha \cap P^{j'}_\alpha = \emptyset \qquad \forall P' \neq P'' \in G_\alpha(P).$$

Set $P = \{2k+1, 4k+2\}$ *when* $N = 6k+4$, *has clean generator* $G_2(P)$.

When $\mathcal{P} \in \mathbb{P}_N$ *can be written as:*

$$\mathcal{P} = \bigcup_{\{\alpha, P\}} G_\alpha(P),$$

that is, when $\mathcal{P}$ *is a union of clean generators, then* $\mathcal{P}$ *is said to be* generated.

One can check that all mirrored starters on $N = 16, 20, 22, 32, 64$ teams, have a partition that is generated, as was already partially shown in Table 34. Below, the decomposition of $\mathcal{P}$ into clean generators is shown, where we refer to P by a triple T containing the three smallest elements of P - P $= \{\pm x : x \in T\}$. For notational purposes, we've omitted the clean generator $G_2(\{2k+1\})$ if it was part of the partition:

$$\begin{aligned}
N &= 16 & G_2(\{1,3,4\}) \\
N &= 20 & G_7(\{1,3,4\}) \\
N &= 22 & G_2(\{1,8,9\}) \\
N &= 32 & G_2(\{1,11,12\}) \\
N &= 32 & G_2(\{1,12,13\}) \\
N &= 32 & G_2(\{1,13,14\}) \\
N &= 64 & G_2(\{1,5,6\}) \cup G_2(\{7,19,26\}) \cup G_2(\{9,18,27\}) \\
N &= 64 & G_2(\{3,10,13\}) \cup G_2(\{7,1,8\}) \cup G_2(\{9,18,27\}) \\
N &= 64 & G_2(\{1,24,25\}) \cup G_2(\{7,10,17\}) \cup G_2(\{9,18,27\})
\end{aligned}$$

We see that not only are all the $\mathcal{P}$ generated, but some even coincide with a single $G_\alpha(P)$ and perhaps the special $G_2(\{2k+1\})$. Any generator that generates an entire partition $\mathcal{P}$ this way is said to be a *proper generator*. Enumerating all possible proper generators, can be done quickly compared to going through all possible (mirrored) starters.

For any value in $\alpha \in \mathbb{Z}_{n-1}$, one can check the order ℓ for which $\alpha^\ell = 1 \mod N-1$. When $N = 6k+2$ or $N = 6k+4$, and $n = 3k+1$ or $3k+2$, this order must be a multiple of k. For any suitable α found, for every set $P = \{\pm 1, \pm x, \pm(x+1)\}$, with $x \in \{2, \ldots, n-1\}$, we can check if $G_\alpha(\{1, x, x+1\})$ is a clean generator. If this is the case, we found a proper generator and we set $\mathcal{P} = G_\alpha(\mathcal{P})$ together with $G_2(\{2k+1\})$ if $N = 6k+4$. All is left is to calculate the COE for all starters in $\mathcal{D}(\mathcal{P})$.

The above procedure finds all proper generators in roughly $\mathcal{O}(N^3)$, hence polynomial time, and this is by no means optimal. Finding the best starter that is created by a proper generator on N teams, can be done significantly faster than finding the best among all mirrored starters, without losing any of the balanced starters of $N = 16, 20, 22, 32$.

After enumerating all proper generators up until $N = 120$, no new balanced schedules were found unfortunately. Of course, we limited our search to these specific proper generators. It might very well be possible that within all the generated partitions, better or even optimal starters can be found for these values of N. However, enumerating all generated partitions will take more time.

5.7 CONCLUSION

A lot of work has been done on researching Carry-Over Effect and a big portion of the most recent work has been in the realm of optimizing (positive or negative) the Carry-Over Effect for given number of teams.

We examined a popular way of constructing schedules on N teams, for which the Carry-Over Effect can quickly be calculated, namely starters. These starters create schedules only using the configuration of the first round. As a lot of starters exist, in the quest to find balanced schedules, we look at a specific subset of starters, that are mirrored. These mirrored starters come with an additional structure which makes it easy to enumerate them and their resulting schedule looks nice.

For N up to 40, some improvements were found on earlier best known COE-schedules. Unfortunately, no new balanced schedules were found however among all mirrored starters for team sizes reaching 64. And even with another method that looked at a specific type of mirrored schedules, no new balanced schedules were found up until $N = 120$ teams.

Although we have not found new balanced schedules - if they exist - a likely way to do so is by exploiting and extrapolating structures found in starters with low COE. By doing this, we managed to find almost balanced starters for large N in a reasonable time consistently. Constructing mirrored starters on even bigger N, by for example combining good generators that together partition all teams, might be an even more effective way of getting to (almost) balanced schedules.

6 BALANCED SERIAL KNOCK-OUT TOURNAMENTS

Round robin competitions are used in abundance, as they are regarded a fair way to determine the best players or teams. A participant has to beat as many of the others, to get high up the ranking. However, when the end of the season nears, not all players have something left to play for.
When titles, play-offs and relegation are no longer within reach, for some motivation might be lacking for the remainder of their games. This can favor their opponents yet to come - they have the opportunity to score points with more ease, helping them to achieve whatever goals they might still have. This is not only unfavorable from a perspective of fairness, but also from a viewer's perspective - who wants to see a match involving non-motivated players? Hence, the darts association PDC came up with an alternative for their originally DRR Premier League of Darts. Instead of each round consisting of a pairing and one match per player, the 8 players would contest a knock-out tournament every week. The incentive for players to win, even though their place in the ranking is irrelevant, should come from the fact that winning the tournament on its own is rewarded with more prize money.
But who the players meet, and when, partially depends on the bracket and the results in the knock-out. Some players may never meet in the semi-finals, while others could meet each other twelve out of fourteen times in that stage of the tournament. This imbalance is unnecessary, as it is very well possible to balance the opponents of each player perfectly across the knock-out tournaments.
This chapter is based on Lambers, Pendavingh, and Spieksma (2022).

6.1 INTRODUCTION

Two popular tournament formats are the round-robin format and the knock-out format. In a round-robin format, each pair of players (or teams) meet a given number of times. In a knock-out tournament, starting from a so-called *bracket*, each stage of the knock-out tournament sees matches between all remaining players, and a player is removed from the tournament after losing a match; in this way, after $\log n$ stages a winner is determined (where n is the number of players).

Each of these formats has been studied intensely from very different viewpoints. In particular, deciding upon a bracket of the players in a single knock-out tournament has attracted a lot of attention; we refer to Vu (2010), Groh et al. (2012), Aziz et al. (2014) for more information on this subject. Most of this literature assumes that probabilities are given that denote the chance of one player beating the other.

It is not uncommon to design a tournament combining both formats: for instance, first have a number of round-robin tournaments in parallel, and then let the winners of the round robins participate in a knock-out tournament. The FIFA World Cup has such a set-up, as do many of the highest national leagues in various sports.

The UEFA Champions League starts with a knock-out for qualifiers, then goes to group stages, and ends again with a knock-out phase.

In this chapter we study a format that is an alternative combination of a knock-out tournament and a round-robin tournament. Let the number of players n be equal to 2^k for some $k \geqslant 2$, allowing us to focus exclusively on knock-out tournaments where each player has to play the same number of matches to win the tournament, which is k. Observe that such a knock-out tournament consists of k successive *stages*, where in stage i the remaining 2^{k+1-i} players compete, $i = 1, \ldots, k$. In tennis, the grand slam singles tournaments have $2^7 = 128$ players, thus the final equals stage 7.

The entire competition then consist of $2^k - 1$ rounds, where each round is a knock-out tournament on n players. We will call this format a *Serial Knock-Out* tournament, or SKO for short. The problem is to specify, for each of the individual knock-out tournaments, the bracket; the bracket specifies the leaf node of the underlying knock-out tree to which each player is assigned, see Figure 10. Once the brackets are specified, the rounds of the SKO can unfold - no other decisions in the design of the tournament need to be taken. As far as we are aware, this particular format has not been studied before. Related (but different) formats are the so-called quasi-double knockout tournament (Considine and Gallagher, 2018) and the multiple-elimination knockout tournament (Fayers, 2005).

When scheduling a knock-out tournament, it is very common to *seed* the players in advance, see for example the major tennis tournaments.

The bracket is then filled by taking this seeding into account. However, in the SKO, we treat all players equally - we make no prior assumptions on the strength of the players.

Clearly, deciding upon the bracket in a knock-out tournament also determines the possible matches that can be played in the next stages. From a fairness perspective, one may wish not to discriminate between different players, leading to the question whether one can ensure that each pair of players meets potentially equally often in each of the stages of the SKO. Thus, our interest is in obtaining the brackets such that any match between a pair of players can occur equally often throughout the stages of the tournament. We refer to this as the *stability* of an SKO tournament.

6.1.1 Motivation: The Premier League of Darts

The motivation for investigating this particular tournament design comes from the Professional Darts Corporation (PDC). We now describe this motivation in more detail.

The Premier League of Darts, organized by the PDC, and which started February 3rd 2022, is an annual competition where the best 8 darts players of the world compete over several months for the title, and the prize of £275.000. The concept of the league changed drastically compared to the previous years – in this edition, every one of the 16 rounds is a knock-out tournament on its own. Every round, there will be a winner and importantly, in every single match there is something to play for, which should cause excitement.

The 16 rounds are structured in the following way: each of the first 7 rounds have predetermined brackets, then there is a special round, again 7 rounds with predetermined brackets, and a last special round. The draws in the special rounds depend on the standings at that time. The other (regular) rounds have a fixed bracket that is determined in advance by the PDC. Our analysis focuses on the brackets in these regular rounds. The first 7 regular rounds, as well as the second 7 regular rounds each correspond to an SKO.

As a final remark, we like to point out that the SKO of the Premier League is constructed in such a way, that every player is set to meet every other player once in the first stage during the 7 rounds. Together, the first stages of the first 7 rounds thus form a SRR. Although we do not have this as a requirement for SKOs in general, we will be looking at schedules that showcase this property as well.

6.2 DEFINITIONS

In this section, we formally describe the knock-out tournaments, the Serial Knock-outs, and the concept of stability.

We start with the knock-out tournament and the bracket s that schedules the tournament.

Definition 28. *A* bracket s *is a permutation of* $n = 2^k$ *players. A* knock-out *tournament* $T(s)$ *on these* n *players, is given by placing, from left to right, the elements of* s *on the leaves on the balanced binary tree of order* 2^k*. Every node that is not a leaf, represents a match. Each match has one player that wins and one that loses - the winner is scheduled to play at the next node.*

In Example 7 it is shown how to attach the bracket $s = 01452367$ to knock-out tree on 8 players. Although the permutation itself holds all the information needed, we may place hyphens as a visual aid separating the left and right halves of the tree: $0145 - 2367$ instead of 01452367.

Example 7. *Bracket* $0145 - 2367$ *places the players in the tree as shown in Figure 10.*

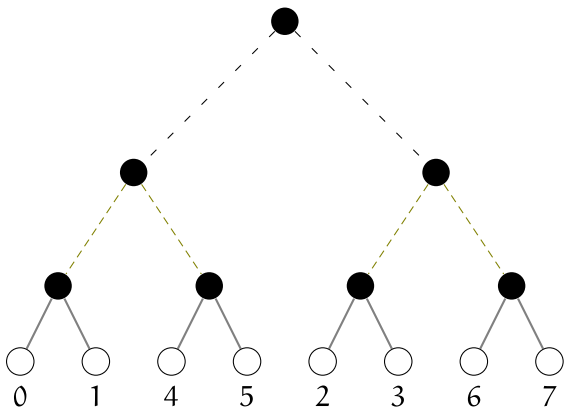


Figure 10: Knock-out $T(s)$ with bracket $s = 0145 - 2367$.

A tournament on for instance 8 players has 3 stages, stage $1, 2, 3$ - popularly referred to as quarter final, semi-final and final. Every player has one possible opponent in stage 1, two in stage 2, etcetera.

Definition 29. *Given a knock-out tournament* T *for* $n = 2^k$ *players, we say that* $\nu_T(x, x') = i$ *if players* x, x' *can meet in stage* i *of that tournament,* $i = 1, \dots, k$.

Two brackets $s \neq s'$ *can have* $\nu_{T(s)}(x, x') = \nu_{T(s')}(x, x')$ *for all pairs* x, x'.

The phrase 'can meet' in the above definition refers to the assumption that players x and x' win their matches in the stages prior to their encounter. For instance, in Example 7, players 1 and 4 can meet in Stage 2, while players 0 and 3 can meet in stage 3, the final.

Definition 30. *A set of knock-out tournaments $\mathcal{T}$ on $n = 2^k$ players is* stable in stage i *if there is a number c_i so that*

$$\#\{T \in \mathcal{T} : \nu_T(x, x') = i\} = c_i$$

for all pairs of distinct players x, x'. The set $\mathcal{T}$ is stable *if it is stable in all stages $i = 1, \ldots, k$.*

Definition 31. *A* Serial Knock-Out (SKO) *is a competition on $n = 2^k$ players over $n - 1$ rounds, where every round r consists of a knock-out tournament $T_r = T(s_r)$ among all players. The SKO is stable if $\mathcal{T} = \{T_r : r \leqslant n - 1\}$ is stable.*

From a players perspective, one might also be interested in the probability of meeting certain opponents in later stages.

Notice that in any knock-out tournament T, a player has 2^{i-1} possible opponents in advance when reaching stage i, i.e., for each player x, we have $\#\{x' : \nu_T(x, x') = i\} = 2^{i-1}$, $i = 1, \ldots, k$. As an SKO consists of $2^k - 1$ knock-out tournaments, the number of meetings that are possible in stage i for any player is given by $(2^k - 1)2^{i-1}$. With the number of opponents of any player x equal to $n - 1 = 2^k - 1$, an SKO is stable in stage i if $c_i = 2^{i-1}$, $i = 1, \ldots, k$.

We can rephrase this to the following condition:

Condition 3. *An SKO $\mathcal{T}$ on $n = 2^k$ players and $n - 1$ rounds is stable if for every pair of players x, x':*

$$\#\{T \in \mathcal{T} : \nu_T(x, x') = i\} = 2^{i-1} \qquad \forall 1 \leqslant i \leqslant k$$

We point out that the SKOs used in the Premier League of Darts are not stable. More precisely, the two SKOs that correspond to the first 7 regular knock-out tournaments and to the second 7 regular knock-out tournaments, are stable in stage 1 - each pair of players is bound to meet each other once in the first round of a knock-out tournament. However, for the other two stages, the SKOs are not stable.

6.3 CONSTRUCTING A STABLE SKO TOURNAMENT FOR 8 PLAYERS

We are going to construct a stable SKO tournament $\mathcal{T} = \{T_r : r \leqslant 7\}$ for 8 players; the result will be applicable to the PDC Premier League described in Section 6.1.1. A stable SKO on these 8 players can be any selection of 7 brackets $s_1, \ldots, s_7$ for which the set $\mathcal{T} = \{T(s_i) : i \leqslant 7\}$ is stable.

The construction uses a geometric entity called the Fano plane, see Figure 11. The plane consists of 7 points, $1, \ldots, 7$, and 7 lines. These lines all go through exactly 3 points - one of the lines is the circle going through $2, 5, 7$. Every line has a color we use to refer to a specific line - the colors are (light) blue, (light) green, red, purple and orange.

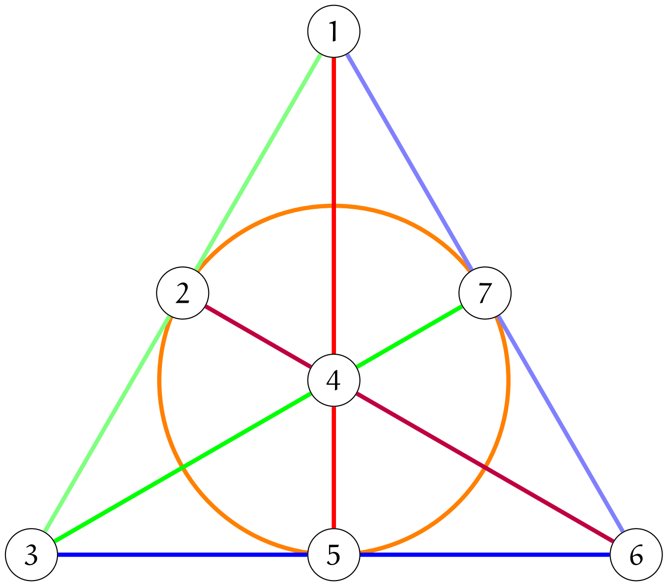


Figure 11: The Fano-plane used to construct Table 35

The following Algorithm 6, takes as input a point x and line ℓ through x on the plane, and produces a bracket s.

Constructing $s = 0145 - 2367$ *from Algorithm 6 with* $x = 1$ *and the red line.*
Step 1. $x = 1$ *is chosen, so*

$$s = 01\ldots$$

Step 2. *The red line through* x *goes through* $4, 5$.

$$s = 0145\ldots$$

Step 3. $2, 3$ *are on the light green line through* 1. *And* $6, 7$ *on the light blue line through* 1.

$$s = 0145 - 2367.$$

Algorithm 6 Line-point

Input: A set of 8 players, a point x and a line $\ell \ni x$ in Figure 11.

1: Player x meets 0 in stage 1.

$$s = 0x\ldots$$

2: Line $\ell = \{x, y, z\}$. Players y, z meet in stage 1.

$$s = 0xyz\ldots$$

3: For every line $\ell' \neq \ell$ through x, let $y', z' \in \ell'$ meet in stage 1.

$$s = 0xyz - y'z'y''z''$$

Output: Bracket s on 8 players.

Algorithm 6 gives one bracket, whereas we need a set of 7 brackets to create a stable SKO. However, the brackets constructed by the algorithm have a particular structure, stated in the following lemma.

Lemma 11. *Let* T *be the knock-out tournament that arises from the node-line pair* x, ℓ, *and let* y *be a node of the Fano plane. Then*

- $\nu_T(0, y) = 1$ *if and only if* $y = x$,

- $\nu_T(0,y) = 2$ *if and only if* $y \in \ell$ *and* $y \neq x$*, and*
- $\nu_T(0,y) = 3$ *if and only if* $y \notin \ell$*.*

Moreover, if $\ell' = \{y, x, x'\}$ *is any line of the Fano plane containing the node* y*, then* $\nu_T(x,x') = \nu_T(0,y)$*.*

We use this lemma to get to Table 35, a set of 7 point-line inputs together with their resulting brackets are shown. Together, they form a stable SKO on 8 players.

Round	Bracket	Node	Line
1	0145 – 2367	1	Red
2	0426 – 1537	4	Purple
3	0213 – 4657	2	Light green
4	0356 – 1247	3	Blue
5	0527 – 1436	5	Orange
6	0734 – 1625	7	Green
7	0617 – 2435	6	Light blue

Table 35: Brackets for a stable SKO.

Notice that in Table 35, each node and each line of the Fano plane occur exactly once, and each node is on the corresponding line. This is sufficient to obtain a stable SKO, as is stated in the next theorem.

Theorem 14. *Let* $x_1, \dots, x_7$ *be an enumeration of the nodes and* $\ell_1, \dots, \ell_7$ *be an enumeration of the lines of the Fano plane, such that* $x_r \in \ell_r$ *for* $r = 1, \dots, 7$*. Let* T_r *be the knock-out tournament that arises from the the pair* x_r, ℓ_r*. Then, the SKO defined by* $\mathcal{T} := \{T_1, \dots, T_7\}$ *is stable.*

Proof. To show that $\mathcal{T}$ is stable, we need to show that

$$\#\{T \in \mathcal{T} : \nu_T(x,x') = i\} = 2^{i-1}$$

for each pair of distinct players x, x' and each stage $i \in \{1,2,3\}$. Notice that $\mathcal{T}$ is stable in stage $i = 3$ if it is stable in both stage 1 and 2.

We first consider the case that one of x, x' is 0, say $\{x, x'\} = \{0, y\}$ for some $y \in \{1, \dots, 7\}$.

- $(0,y)$ is stable in stage $i = 1$, as $\exists! r^y$ such that $y = x_{r^y}$, implying $\#\{T \in \mathcal{T} : \nu_T(0,y) = 1\} = \#\{r : y = x_r\} = 1$.
- $(0,y)$ is stable in stage $i = 2$. This can be seen from the fact that there are exactly three lines through y, thus there exist two rounds $r, r' \neq r^y$ such that $y \in \ell_r, \ell'_r$ - meaning that $(0,y)$ can meet in stage 2 in those rounds.
  $\#\{T \in \mathcal{T} : \nu_T(0,y) = 2\} = \#\{r : y \in \ell_r, y \neq x_r\} = 2$.

Next, suppose x, x' are distinct players, both not 0. Then, the Fano plane contains a unique node y and line $\ell' = \{y, x, x'\}$ through x, x'. By Lemma 11, we have $\nu_T(x, x') = \nu_T(0, y)$ for each $T \in \mathcal{T}$. As $\#\{T \in \mathcal{T} : \nu_T(0, y) = i\} = 2^{i-1}$ for all y, this holds for any distinct pair x, x'.

The theorem follows. □

6.4 CONSTRUCTING STABLE SKO FOR $n = 2^k$

In this section we will generalize the node-line construction used on 8 players to find a stable SKO for $n = 2^k$ players. We first describe a basic idea on brackets in Section 6.4.1, which we connect to Galois fields in Section 6.4.2. We use this connection in Section 6.4.3 to prove our main result: Theorem 15.

6.4.1 The basic idea

The key idea that we will carry over from the $n = 8$ setting to the general case, is that we will construct our knock-out tournaments in a restricted way, so that for each pair of players x, x', there is a well-defined player y such that

$$\nu_T(x, x') = \nu_T(0, y)$$

in all knock-out tournaments T of this specific form. Showing that an SKO $\mathcal{T}$ is stable, where each tournament $T \in \mathcal{T}$ is of this special form, then reduces to verifying that

$$\#\{T \in \mathcal{T} : \nu_T(0, y)\} = 2^{i-1}$$

for each player y and each stage i, $i = 1, \ldots, k$.

To define the representative y of a pair of players x, x' and to create the special tournaments T, we need additional structure on the set of players. For the case $n = 8$, we identified the non-zero players with nodes of the Fano plane and used its geometry to define the tournaments. In what follows, we will identify the $n = 2^k$ players with the 2^k elements of the *Galois field* $\mathrm{GF}(2^k)$.

As $\mathrm{GF}(2^k)$ is a field, both addition and multiplication are possible operations on its elements. We construct a single tournament T such that for $x, x' \in \mathrm{GF}(2^k)$, we have

$$\nu_T(x, x') = \nu_T(0, y)$$

when $y := x - x'$.

After we constructed a base model for our knock-out tournament, we use the multiplication in $\mathrm{GF}(2^k)$ on T, to create tournaments $T(z)$ for each nonzero element z of $\mathrm{GF}(2^k)$, and argue that

$$\mathcal{T} := \{T(z) : z \neq 0\}$$

is a stable SKO.

6.4.2 The connection to Galois fields

To exploit the structure of Galois field $\mathsf{GF}(2^k)$, we first have to describe $\mathsf{GF}(2^k)$. Although we do not go into too much detail, we point out the main properties that we use. For an accessible introduction to finite fields, see Chavez and O'Neill (2022).

A polynomial $q \in \mathbb{Z}_2[X]$ is irreducible, if for any pair $r, s \in \mathbb{Z}_2[X]$ of polynomials such that $q = r \cdot s$, at least one of r, s is a constant.

Given any irreducible polynomial $q \in \mathbb{Z}_2[X]$ of degree k, it is known that $\mathsf{GF}(2^k) \cong \mathbb{Z}_2[X]/(q)$ - this is the space of polynomials where polynomial $q = q_k X^k + \ldots q_1 X + q_0$ and any multiple of q is considered equal to 0. In addition, we know that for this q, there exists an element $\alpha \in \mathsf{GF}(2^k)$, such that $q(\alpha) = 0$.

For any value of k, such polynomial q and α are guaranteed to exist. For example, when $k = 3$, the polynomial $q[X] = X^3 + X^2 + 1$ is irreducible over $\mathbb{Z}_2[X]$. Other irreducible polynomials of small degree are $X^2 + X + 1, X^4 + X + 1, X^5 + X^2 + 1$ for $k = 2, 4, 5$ respectively.

Using any fitting combination of q, α, we can express an element $x \in \mathsf{GF}(2^k)$ as linear combination of $1, \alpha, \alpha^2, \ldots, \alpha^{k-1}$ over the field $\mathbb{Z}_2$, i.e. $x = \sum_{i=0}^{\alpha} x_i \alpha^i$, where $x_i \in \mathbb{Z}_2$.

We work with coefficients x_i in $\mathbb{Z}_2$, where $1 + 1 = 0 + 0 = 0$, addition $x + x'$ of two elements $x, x' \in \mathsf{GF}(2^k)$ is given by:

$$x = \sum_{i=0}^{k-1} x_i \alpha^i \qquad x' = \sum_{i=0}^{k-1} x'_i \alpha^i \qquad x + x' = \sum_{i=0}^{k-1} (x_i + x'_i) \alpha^i.$$

For an element $x = \sum_i x_i \alpha^i \in \mathsf{GF}(2^k)$, we define the *degree* of $x \in \mathsf{GF}(2^k)$ to be $d(x) = \max\{i : x_i \neq 0\}$. The value of $d(x)$ depends on our choice of q and corresponding α we used for expressing the elemenets in $\mathsf{GF}(2^k)$, but as we assume to have chosen and fixed q, α, the degree d is well-defined.

This degree leads us to the following lemma on the existence of a tournament T with the nice property that $\nu_T(x, y) = \nu_T(0, x - y) = 1 + d(x - y)$.

Lemma 12. *There is a knock-out tournament* T *whose players are the elements of* $\mathsf{GF}(2^k)$*, so that* $\nu_T(x, y) = 1 + d(x - y)$ *for all* $x, y \in \mathsf{GF}(2^k)$.

Proof. We construct tournament T by inductively constructing T_m for incremental values $m = 1, \ldots, k$, where each T_m is a knock-out tournament on the set $P_m = \{x \in GF(2^k) : d(x) < m\}$, and all the T_m have the property that $\nu_{T_m}(x, y) = 1 + d(x - y)$ for $x, y \in P_m$. Then $T = T_k$ proves the lemma.

When $m = 1$, the set $P_0 = \{0, 1\}$ contains only two players, and the unique tournament T_1 one can construct on these two players has $\nu_{T_1}(0, 1) = 1 = 1 + d(1 - 0)$.

As induction step, assume that T_m exists such that $\nu_{T_m}(x, y) = 1 + d(x - y)$ for all $x, y \in P_m$. Let T'_m arise from a copy of T_m by adding α^m to each player. Then T'_m has players $P'_m = \{x + \alpha^m : x \in P_m\}$ and for any two players $x', y' \in P'_m$ we have

$$\nu_{T'_m}(x', y') = \nu_{T_m}(x, y) = 1 + d(x - y) = 1 + d(x' - y')$$

where $x' = x + \alpha^m$ and $y' = y + \alpha^m$ with $x, y \in P_m$.

We construct T_{m+1} for players $P_{m+1} = P_m \cup P'_m$ as the combination of tournaments T_m, T'_m, where at stage $m + 1$, the winner of T_m plays the winner of T'_m. For this T_{m+1}, we see that for $x, y \in P_{m+1}$:

$$\begin{aligned}
\nu_{T_{m+1}}(x, y) &= \nu_{T_m}(x, y) = 1 + d(x - y) && \text{if } x, y \in P_m \\
\nu_{T_{m+1}}(x, y) &= \nu_{T'_m}(x, y) = 1 + d(x - y) && \text{if } x, y \in P'_m \\
\nu_{T_{m+1}}(x, y) &= 1 + m = 1 + d(x - y) && \text{if } x \in P_m, y \in P'_m \\
& && \text{or } x \in P'_m, y \in P_m
\end{aligned}$$

This finishes the induction step. Taking $T = T_k$ gives the desired tournament. □

The construction of T with elements in $GF(2^3)$ is given in Figure 12.

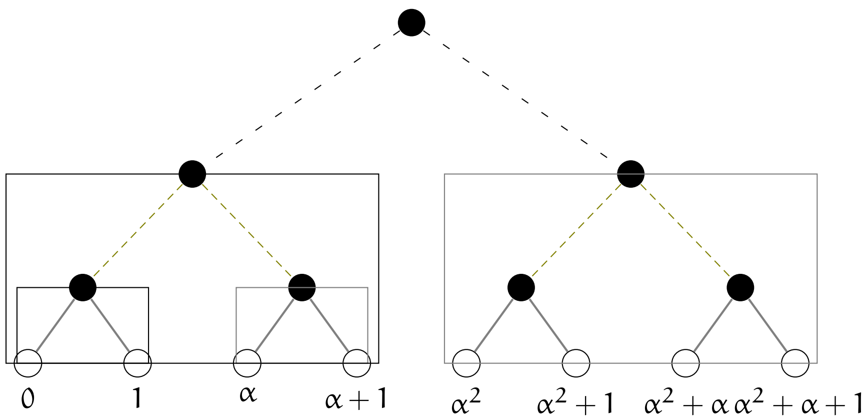


Figure 12: A knock-out tournament T so that $\nu_T(x, y) = 1 + d(x - y)$

6.4.3 The result

By Lemma 12, we know there exists a knock-out tournament T on the elements of $GF(2^k)$ such that $\nu_T(x, y) = \nu_T(0, x - y) = 1 + d(x, y)$

for all $x, y \in GF(2^k)$. In the following section, we argue that for each non-zero $z \in GF(2^k)$, the tournament $T(z)$ obtained from T by replacing each player x by zx maintains the property that $\nu_{T(z)}(x, y) = \nu_{T(z)}(0, x-y)$. Then we show that

$$\mathcal{T} = \{T(z) : z \in GF(2^k) \setminus \{0\}\}$$

is a stable SKO.

To properly define $T(z)$, we need to multiply $x, z \in GF(2^k)$ such that $xz \in GF(2^k)$. As normal multiplication would lead to polynomials of degree potentially higher than $k-1$, we use $q(\alpha) = 0$ to reduce higher order terms (degree k or higher) to a lower degree ($k-1$ or lower). To illustrate this, suppose we have $k = 3$, with $q = X^3 + X^2 + 1$ as irreducible polynomial. Then multiplying $x = \alpha^2 + 1, z = \alpha^2 + \alpha$ leads to:

$$\begin{aligned} x \cdot z &= (\alpha^2 + 1)(\alpha^2 + \alpha) = \alpha^4 + \alpha^3 + \alpha^2 + \alpha \\ &= \alpha^2 \end{aligned}$$

The degree of the polynomial was reduced from 4 to 2, using $\alpha^4 = \alpha \cdot \alpha^3$, and substituting $\alpha^3 = \alpha^2 + 1$, as $q(\alpha) = \alpha^3 + \alpha^2 + 1 = 0$.

Let T be a tournament satisfying Lemma 12, thus $\nu_T(x, y) = 1 + d(x - y)$ for all $x, y \in GF(2^k)$. Let $z \in GF(2^k)$ be non-zero and thus invertible. We construct $T(z)$ from T by replacing each player x with zx. As the map $x \mapsto zx$ is one-to-one, $T(z)$ is again a tournament whose players are the elements of $GF(2^k)$. Evidently we have $\nu_{T(z)}(x, y) = \nu_T(z^{-1}x, z^{-1}y)$ for all $x, y \in GF(2^k)$. It follows that

$$\nu_{T(z)}(x, y) = \nu_T(z^{-1}x, z^{-1}y) = \nu_T(0, z^{-1}(x - y)) = \nu_{T(z)}(0, x - y)$$

for all $x, y \in GF(2^k)$ and

$$\nu_{T(z)}(0, y) = \nu_T(0, z^{-1}y) = 1 + d(z^{-1}y)$$

for all $y \in GF(2^k)$.

Theorem 15. $\mathcal{T} := \{T(z) : z$ *a nonzero element of* $GF(2^k)\}$ *is a stable SKO.*

Proof. We need to show that $\#\{T \in \mathcal{T} : \nu_T(x, x') = i\} = 2^i$ for each pair of distinct players $x, x' \in GF(2^k)$ and each stage $i = 1, \dots, k$.

If one of x, x' is 0, say $\{x, x'\} = \{0, y\}$ with $y \neq 0$, then, for each $i = 1, \dots, k$,

$$\#\{T \in \mathcal{T} : \nu_T(0, y) = i\} = \#\{z \in GF(2^k) : z \neq 0, 1 + d(z^{-1}y) = i\}.$$

Substituting z by $r^{-1}y$ this equals

$$\begin{aligned} &\#\{r^{-1}y \in GF(2^k) : r \neq 0, 1 + d(r) = i\} = \\ &\#\{r \in GF(2^k) : r \neq 0, 1 + d(r) = i\} = 2^i \end{aligned}$$

since the map $r \mapsto r^{-1}y$ is one-to-one.

The general case reduces to the above special case, since each of the tournaments $T \in \mathcal{T}$ has $\nu_T(x, x') = \nu_T(0, x - x')$. Then

$$\#\{T \in \mathcal{T} : \nu_T(x, x') = i\} = \#\{T \in \mathcal{T} : \nu_T(0, x - x') = i\} = 2^i,$$

as required. $\square$

We close this section with an example that constructs a stable SKO on 8 players using the Galois group.

Example 8. *For the Galois group, we choose* $q(X) = X^3 + X + 1$ *as the irreducible polynomial over* $\mathbb{Z}_2$ *and set* $q(\alpha) = 0$. *The corresponding multiplication table is shown in Table 36.*

Round:	1	2	3	4	5	6	7
z:	1	α	$\alpha+1$	α^2	α^2+1	$\alpha^2+\alpha$	$\alpha^2+\alpha+1$
1	1	α	$\alpha+1$	α^2	α^2+1	$\alpha^2+\alpha$	$\alpha^2+\alpha+1$
α	α	α^2	$\alpha^2+\alpha$	$\alpha+1$	1	$\alpha^2+\alpha+1$	α^2+1
$\alpha+1$	$\alpha+1$	$\alpha^2+\alpha$	α^2+1	$\alpha^2+\alpha+1$	α^2	1	α
α^2	α^2	$\alpha+1$	$\alpha^2+\alpha+1$	$\alpha^2+\alpha$	α	α^2+1	1
α^2+1	α^2+1	1	α^2	α	$\alpha^2+\alpha+1$	$\alpha+1$	$\alpha^2+\alpha$
$\alpha^2+\alpha$	$\alpha^2+\alpha$	$\alpha^2+\alpha+1$	1	α^2+1	$\alpha+1$	α	α^2
$\alpha^2+\alpha+1$	$\alpha^2+\alpha+1$	α^2+1	α	1	$\alpha^2+\alpha$	α^2	$\alpha+1$

Table 36: Multiplication on $GF(2^3)$

Table 36 essentially gives the bracket for the SKO, since the row for multiplication by z presents the bracket for $T(z)$. *Upon replacing each polynomial with the number specified in the following table, we get the SKO of Table 37*

Row	Bracket	Row	Bracket
1	$0145 - 2367$	5	$0312 - 4756$
2	$0426 - 5173$	6	$0671 - 3542$
3	$0563 - 7214$	7	$0734 - 1625$
4	$0257 - 6431$		

Table 37: SKO constructed from Table 36

6.5 CONCLUSION AND DISCUSSION

We have analyzed a novel tournament design that is used in practice, and that can be seen as a combination of a knock-out tournament and a round robin tournament; we call it a Serial Knock-Out tournament (SKO). From the viewpoint of fairness an attractive property of an SKO is stability: whether or not pairs of players can meet equally often in the stages of the SKO. We have shown using a connection to Galois fields that this is always possible.

We like to remark here that the construction to create stable SKOs does not generate a unique tournament - for example, the order of the rounds can be changed without hurting the stableness of the SKO. And within each tournament, a tournament $T(s)$ with bracket s can be replaced by $T(s')$ as long as $\nu_{T(s)} = \nu_{T(s')}$. Not all stable SKOs are equal and from an organizers point of view, there might be additional constraints, that distinct some stable SKOs from others.

For example, in the PDC Premier League, all matches are played on one night, stage by stage, one match after the other. If a player plays the first match of first stage, he has to wait at least 3 games before his potential semi-final match is scheduled. Whether this is a benefit or not, the organizers might want to schedule the matches in the individual nights in such a way that the matches of a player are evenly spread over the timeslots - it could be that some SKOs cater better to this need than others.

7 HOW TO SCHEDULE THE VOLLEYBALL NATIONS LEAGUE

Traveling days are not rest days.

Swedish Nils van der Poel wrote these words in his unique self-study of the training regime that led him to win two speed skating olympic gold medals in Beijing. As he distinguishes only two types of days, training days and rest days, it is clear that to him, traveling is exercising.

And he is not alone in that assessment. Many of his peers made similar statements how traveling hampered their performance. Not only that, many statistical reviews investigating the impact of travel fatigue on an athletes level, found the correlation as well. Hence, in sport scheduling, the Traveling Tournament Problem is a well studied problem of scheduling competitions where traveled distance is part of the objective.

The international Volleyball Nations League in 2018/2019 can be regarded as one of the most travel-intense tournaments. In the span of five weeks, teams have to travel across the globe and back, to meet all their 15 opponents. Each week, groups were hosted *somewhere* on earth, making several teams travel from continent to continent, while others were lucky enough to stay close to their home country for most of their matches. Designing a schedule such that each team has similar and near minimum disadvantage of its travel scheme, is a hard problem in general. However, exploiting the structure of the tournament improves the solution times as well as insights in how fair schedules look like.

This chapter is based on Lambers, Rothuizen, and Spieksma (2021).

7.1 INTRODUCTION

The Volleyball Nations League is a tournament organized every year by the FIVB (Fédération Internationale de Volleyball), for both men and women (see https://www.volleyball.world/en/vnl/2021). The tournament was first organized in 2018 to replace the World League/-World Grand Prix as annual volleyball tournament. There are 16 teams participating in the tournament which consists of multiple phases. In the first phase, lasting for five weeks, all 16 teams play a single round robin, i.e., each team meets each other team once. The 6 teams that perform best qualify for the second phase, where out of two groups of three, four teams emerge to play cross finals. Our interest is exclusively on the first phase.

In the first phase of the VNL tournament, teams play in rounds. In each round, each team is in a group consisting of 4 teams, and each team in a group plays a match against its three fellow group members. After 5 rounds, each of the 16 teams has played all the other teams exactly once, and a ranking is made based on the results in this single round robin tournament. All 6 matches in a single group are held at the same venue, however, every round has its 4 groups played out in different venues. As it is a disadvantage to have traveled more than the opponent going into a match, our main interest lies in minimizing a measure that captures the imbalance in travel times between opposing teams.

The Italian team won their first 3 matches in Serbia, but failed to finish in the top 6 losing 7 out of their remaining 12 matches.

That the distribution of travel times can be skewed within a tournament, was showcased for instance in the Men's 2018 tournament. The Italians started in Serbia, then had to travel to San Juan in Argentina, Osaka in Japan, Seoul in South Korea to finish with a group stage in Rome, Italy. Contrary to traveling around the world in 30 days, the winners of that years regular competition, France, only had to move between cities in Europe.

This travel burden was also noted by Dutch captain Anne Buijs, who noted in an interview (see Volkskrant (2021)) regarding the revised COVID-proof schedule of the 2021 VNL that "It is quite an advantage to play all VNL matches at the same location. In the original schedule we would have travelled from Serbia to Canada to Korea, which makes the schedule very hard for us." This is in line with statements from the literature, for instance Samuels (2012) who concludes: "Jet lag and travel fatigue are considered by high-performance athletic support teams to be a substantial source of disturbance to athletes."

Traveling during tournaments has been extensively studied and it is well established within the scientific literature that travelling has a negative impact on sport performance. Although we do not intend to survey the literature on this subject, this finding is reported for various sports ranging from rugby (Lo et al. (2021)) to baseball (Song, Severini, and Allada (2017), and Winter et al. (2009)) and from basketball (Huyghe et al. (2018)) to triathletes (Stevens et al. (2018)); see also the references contained in these papers.

7.2 MATHEMATICAL BACKGROUND

We briefly described the set-up of the Volleyball Nations League in the previous section, with 4 groups of 4 teams for 5 weeks in a row, that jointly form an SRR over 16 teams. A priori, it is not clear how to obtain a round robin schedule that can be decomposed in such groups. In fact, finding a schedule that fits this VNL-format is related to the well known Social Golfer Problem (SGP).

In this problem we are given gp golfers and w rounds (where g, p, w are positive integers), and the SGP asks whether it is possible to let the gp golfers play in g groups of p golfers in each of the w rounds, in such a way that every pair of golfers plays in the same group in at most one round, see Triska and Musliu (2012), Liu, Löffler, and Hofstedt (2019), Dotú and Hentenryck (2005). This question is far from innocent: only for restricted sets of values for g, p, w the answer to this question is known. For instance, when $g = p = w - 1$, solutions are known to exist when g is a prime power - and no other solution to these type of instances has been found, nor has it been proven that these are the only instances for which a solution can exist (Warwick and Winterer (2005)).

Of course, in the context of the Volleyball Nations League, each golfer corresponds to a team (and groups remain groups and rounds remain rounds). Since the Volleyball Nations League has $g = p = 4$ and $w = 5$, it follows that the answer to the SGP-question is affirmative, and hence a schedule for the VNL that consists of 5 rounds, each round consisting of 4 groups, is known to exist - something that could also be concluded from the very existence of the VNL-schedule in the first place. In this paper, we introduce the *Traveling Social Golfer Problem* (TSGP), as a generalization of the SGP; the TSGP allows us to take fairness, as measured by the difference in travel times between opposing teams each round, into account. Recent other variations of the SGP are discussed in Miller et al. (2020) and Lester (2021).

A well-known problem related to the TSGP that also focusses on distances is the Travelling Tournament Problem (TTP), see Easton, Nemhauser, and Trick (2002) for a precise description. In contrast to our problem, in the TTP pairs of teams meet in the venue of one of the two opposing teams. Moreover, the objective in the TTP is to minimize total travel distance; difference in travel time between opposing teams is not considered in the TTP. We refer to Goerigk and

Westphal (2016) and G. Durán, S. Durán, et al. (2019) for an overview concerning the TTP.

A number of studies has been devoted to the scheduling of national volleyball leagues where mainly for cost reasons, the objective is to minimize total travel time. We mention Bonomo et al. (2012) who model the Argentine national volleyball league as an instance of the Traveling Tournament Problem, and Cocchi et al. (2018) who investigate the Italian volleyball league. Further, Raknes and Pettersen (2018) study the Norwegian Volleyball League; one of their models, motivated by a cost-objective, is devoted to minimizing total travel distance in that league. These leagues are organized in the format of a Double Round Robin, and as such differ from the VNL.

7.3 THE TRAVELING SOCIAL GOLFER PROBLEM (TSGP)

7.3.1 Definition of the TSGP

As described in Section 7.2, the SOCIAL GOLFER PROBLEM is a well known combinatorial problem, where the task is to schedule golfers in groups of size p over multiple rounds, such that no golfer plays with another golfer in the same group twice or more. In the TRAVELING SOCIAL GOLFER PROBLEM (TSGP), all groups have to play at (different) venues, where the objective is to create a schedule that minimizes the *unfairness* arising from golfers who play together but have different travel times between the venues.

In order to give a precise formulation of the TSGP, we use the following notation to describe the input:

- N: the number of participants,
- k: a group size,
- V: the set of venues,
- $d(v, w)$: a distance between each pair of venues $v, w \in V$, and
- $c(v)$: a multiplicity for each $v \in V$.

The multiplicity function $c(v)$ indicates the exact number of times venue $v \in V$ must host a group; in the practical situation of the VNL, it is not uncommon that a venue is host to different groups in different rounds. The multiplicities allow us to accommodate such situations.

Furthermore, we use the following notation to describe a solution:

- R: a set of rounds,
- P_i^r: the set of teams in group i in round r, $1 \leqslant i \leqslant \frac{N}{k}$, $r \in R$,

- $v_r(t)$: the venue of the group in which team $t \in \{1, \ldots, N\}$ plays in round $r \in R$.

Finally, we measure the value of a schedule S by its unfairness $u(S)$ as follows:

$$u(S) = \sum_{r \in R \setminus \{1\}} \sum_{i=1}^{\frac{N}{k}} \max_{s,t \in P_i^r} |d(v_r(s), v_{r-1}(s)) - d(v_r(t), v_{r-1}(t))|. \quad (56)$$

Let us elaborate on this expression. For every group P_i^r in every round $r \in R \setminus \{1\}$, we consider the two teams (teams s and t) whose difference in travel distance needed to arrive at the corresponding venue, is maximum over all pairs of teams in the group; this quantity is summed over all groups, and all rounds except the first round (we assume that all teams have ample time to arrive at their first venue). Thus, a lower value of $u(S)$ indicates that the difference in travel times between opposing teams was less and thus the schedule was more fair. The measure u is applicable to any schedule for N teams that has a group/round-structure.

Example 9. *A tournament with* $N = 4$*, teams* $1, 2, 3, 4$*, is organized over three rounds, and groups of size* $k = 2$*. There are four venues,* $V = \{A, B, C, D\}$*, with multiplicities* $c(A) = c(D) = 2$ *and* $c(B) = c(C) = 1$*. Distances between venues are* $d(A, B) = d(A, C) = d(B, D) = d(C, D) = 1$ *and* $d(A, D) = d(B, C) = 2$*.*

Consider the schedule S *depicted in Table 38.*

	Group	Venue	Group	Venue
Round 1	$P_1^1 = \{1, 2\}$	A	$P_2^1 = \{3, 4\}$	D
Round 2	$P_1^2 = \{1, 3\}$	A	$P_2^2 = \{2, 4\}$	B
Round 3	$P_1^3 = \{1, 4\}$	D	$P_2^3 = \{2, 3\}$	C

Table 38: A schedule S for the instance in Example 9.

Thus, according to (56), the unfairness of this schedule S *equals:*

$$u(S) = |2 - 0| + |1 - 1| + |2 - 1| + |2 - 1| = 4.$$

We state the following optimization problem that we call the (N, k)-Traveling Social Golfer Problem, or (N, k)-TSGP for short.

(N, k)-TSGP

Input. Instance $I = (N, k, V, c, d)$ with the number of teams $N \in \mathbb{N}$, a group size $k \in \mathbb{N}$, a set of venues V with multiplicity $c(v) \in \mathbb{N}_{\geqslant 1}$ for all $v \in V$, and a distance function $d : V \times V \to \mathbb{R}$.
Output. A schedule S on R rounds such that:

1. there is an equi-partitioning of N teams in groups $P^r_1, \ldots P^r_{\frac{N}{k}}$ for each round $r \in R$, with for each pair of distinct teams, a single group containing both teams, i.e, for each s, t, $(s \neq t))\exists!\ \ i, r$ with $s, t \in P^r_i$,
2. an allocation of groups to venues that results in venues $v_r(t)$ ($r \in R$, $t = 1, \ldots, N$) such that venue $v \in V$ is a host for a group exactly $c(v)$ times.

where S minimizes $u(S)$.

Clearly, depending on input I, a feasible schedule to (N, k)-TSGP need not exist; it is not difficult to find instances where there is no schedule S that satisfies all the constraints. As the schedule asks for a partitioning of the N teams in groups of size k in each round, we immediately see that N should be a multiple of k: $N \equiv_k 0$. In addition, the schedule should correspond to a single round robin tournament, and as all teams play $k-1$ matches per round, we conclude that $N-1$ should be a multiple of $k-1$: $(N-1) \equiv_{k-1} 0$.

We can combine both findings to see:

$$N \equiv_k 0 \ \&\ N-1 \equiv_{k-1} 0 \implies \exists \rho \in \mathbb{N} \text{ s.t. } N = k((k-1)\rho + 1)$$

We can conclude that a solution to (N, k)-TSGP can only exist if there is an integer ρ such that $N = k \cdot ((k-1)\rho + 1)$.

The above are necessary conditions that need to be satisfied, but are not at all sufficient. The (N, k)-TSGP can only have a solution that satisfies the single round robin format, if the corresponding instance of the SGP is solvable. In general, solutions of the SGP are known to exist when $N = k^2$ and k is a prime power. Thus, for the Volleyball Nations League, the underlying $N = 16, k = 4$-SGP problem will be solvable. In the remainder of the chapter, we assume that $N = k^2$ and $|R| = k + 1$; this ensures that the above necessary conditions are fulfilled (and observe that the VNL case arises when $k = 4$).

7.3.2 Decomposing the TSGP into Venue Assignment and Nation Assignment

As we will show next, the solving of an instance of $(N = k^2, k)-$TSGP can be decomposed into two phases:

1. Venue Assignment. This is the first phase. In this phase, we specify, for each round $r \in R$, which venues act as a host. This results in set $U_r \subset V$, the set of hosts in round r. Obviously, $|U_r| = k$ for all $r \in R$.

2. Nation Assignment. This is the second phase. In this phase, we decide upon the composition of the groups, i.e., we choose the sets P_i^r and allocate each of these groups to one of the venues in U_r.

By going through these two phases, we find a schedule S. It is crucial to observe that the unfairness of S, i.e., $u(S)$, follows directly from the venue assignment when $N = k^2$. We record this observation formally in the following theorem.

Theorem 16. *For each schedule S of a given an instance of $(N = k^2, k) -$ TSGP, $u(S)$ is determined only by the Venue Assignment, for each integer $k \geqslant 2$.*

Proof. We claim that for each schedule S:

$$\begin{aligned} u(S) &= \sum_{r \in R \setminus \{1\}} \sum_{i=1}^{k} \max_{s,t \in P_i^r} |d(v_r(s), v_{r-1}(s)) - d(v_r(t), v_{r-1}(t))| \\ &= \sum_{r \in [1,\ldots,k]} \sum_{u \in U_{r+1}} \max_{v,w \in U_r} |d(v,u) - d(w,u)|. \end{aligned}$$

The latter equality follows from the fact that, independent of the composition of the groups, the k teams that play in a group in some round, will not meet again in a next round, and hence these k teams will travel to each of the k distinct venues in the next round. $\square$

The intuitive idea behind Theorem 16 is highlighted in Figure 13. There are three groups of size 3 and two rounds. In the first round, we see groups A, B, C, in the second round there are the groups X, Y, Z. In X, Y, Z there is exactly one representative from each group A, B, C.

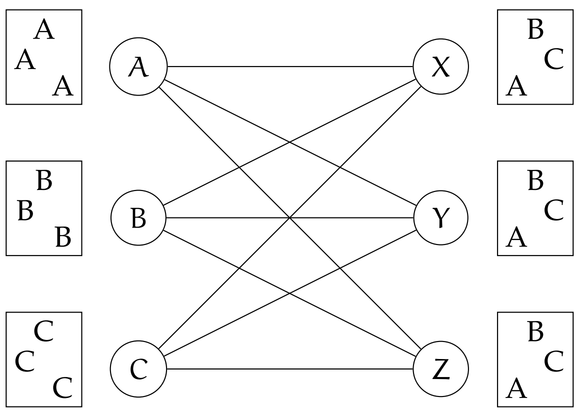


Figure 13: Distribution of teams in two different rounds

Theorem 16 allows us to compute the unfairness of a schedule S, $u(S)$, without specifying the schedule S. As a consequence, it becomes much easier in practice to find schedules for which $u(S)$ is minimum. The intuitive way to interpret Theorem 16, finding a Venue Assignment suffices to know the unfairness of any schedule compatible with the Venue Assignment. In fact, a similar statement can be made with respect to some other possible objective functions, such as the total travel time (see Section 7.6).

7.4 THE COMPLEXITY OF VENUE ASSIGNMENT

In this section, we formally establish the complexity of Venue Assignment. Given that feasible schedules to the $(N = k^2, k)$-TSGP exist, Theorem 16 implies that our task of finding an optimal solution to (N, k)-TSGP is reduced to finding an optimal venue assignment.

In an extreme case, if only a single venue v is given (with multiplicity $c(v) = k(k+1)$), then all matches in all groups in all rounds are played in the same venue, and there is no travel distance. However, in general, the set of venues V and their pairwise distances, are instrumental in finding good venue assignments. Of course, we assume that $\sum_{v \in V} c(v) = k(k+1)$. We now give a formal description.

VENUE ASSIGNMENT (VA)

Input. An instance $I = (k, V, c, d)$ value $k \in \mathbb{N}$, a set of venues V, a multiplicity $c(v) \in \mathbb{N}_{>0}$ for $v \in V$, and a distance matrix $d(v, w)$ for each $v, w \in V$.

Output. For $r \in \{1, \ldots, k+1\}$, subsets $U_r \subset V$ with $|U_r| = k$, such that $\forall v \in V$, $c(v) = |\{r : v \in U_r\}|$ that minimizes:

$$\Delta = \sum_{r \in \{1,\ldots,k\}} \sum_{u \in U_{r+1}} \max_{v,w \in U_r} |d(v,u) - d(w,u)| \tag{57}$$

There is an obvious connection between instances of (N, k)-TSGP and VENUE ASSIGNMENT. We can seemingly use the input of one to specifiy the input of the other problem. When we want to distinguish that we are dealing with an instance of either one, we write I_{TSGP} and I_{VA} respectively.

In order to establish the hardness of VENUE ASSIGNMENT, we will do a reduction from another decision problem, namely LONGEST HAMILTONIAN PATH. This problem is well known to be NP-complete and is defined as follows.

LONGEST HAMILTONIAN PATH ON A COMPLETE GRAPH (LHP)

Input. A complete graph $G = (H, E)$, $|H| = n$ with nonnegative, symmetric weights $w(h_1, h_2)$ for each $h_1, h_2 \in H$, and an integer B.
Question. Does there exist a Hamiltonian Path $(h_{i_1}, \dots, h_{i_n})$ in G such that $\sum_{j=1}^{n-1} w(h_{i_j}, h_{i_{j+1}}) \geqslant B$?

Theorem 17. VENUE-ASSIGNMENT *is NP-Hard.*

Proof. We prove this statement by a reduction from LONGEST HAMILTONIAN PATH ON A COMPLETE GRAPH.

Given an instance of LHP with vertex set $H = \{h_1, \dots, h_n\}$ and weights $w : H \times H \to \mathbb{R}$, we construct an instance of the decision problem corresponding to VA, using a parameter K, in the following way.

We choose $k := n - 1$. Furthermore, the set of venues V consists of $V = V_1 \cup V_2$, where $V_1 := H$ and $|V_2| := n - 2$. For each $v \in V_1$, $c(v) :\ = 1$ and for each $v \in V_2$, $c(v) := n$. Let $D = \max_{h_1, h_2 \in H} w(h_1, h_2)$ and define a symmetric distance function d in the following way:

$$d(u, v) := \begin{cases} w(u, v) & u, v \in V_1 \\ 2D & u \in V_1, v \in V_2 \\ 0 & u, v \in V_2. \end{cases} \tag{58}$$

Notice that the resulting distances satisfy the triangle inequality when the instance of LHP does. Finally, we set $K := k^2 \cdot 2D - B$, and ask whether there exists a venue assignment with unfairness at most K. We have now specified an instance of the decision version of VA.

Let us argue that if there exists a solution to VA with unfairness at most K, LHP is a yes-instance, and vice versa.

To find a solution to any instance of VA, we need to find $U_r \subset V$ for each $r \in \{1, \dots, k+1\}$ such that $\forall v \in V$, $c(v) = |\{r : v \in U_r\}|$. As we know that for all $v \in V_2$, $c(v) = k + 1$, we see that any feasible solution must have $V_2 \subset U_r$ for each round r. Moreover, as $c(v) = 1$ for $v \in V_1$, we get that any feasible solution must schedule every venue $v \in V_1$ exactly once. Thus, any feasible solution to VA consists of $U_r = V_2 \cup v_{i_r}$ with $v_{i_r} \in V_1$ and $v_{i_r} = v_{i'_r} \iff r = r'$. In other words, any feasible solution to VA corresponds to an ordering $(v_{i_1}, \dots, v_{i_{k+1}})$ of the venues in V_1.

Given such an ordering, we obtain the following expression for the unfairness of a schedule S that uses the ordering $(v_{i_1}, \ldots, v_{i_{k+1}})$:

$$u(S) = \sum_{r=1}^{k} \big((k-1) \cdot 2D + (2D - d(v_{i_r}, v_{i_{r+1}})).\big) \tag{59}$$

The first term in the summation results from the fact that there are $k-1$ venues from V_2 in every round and one from V_1, and since $d(v,w) - d(v,v') = 2D - 0$ for all $v, v' \in V_2$, $w \in V_1$, we get $k-1$ venues where the maximal travel difference is 2D. The second term equals the difference in travel distance between the teams traveling from any of the $v \in V_2$ to the $v_{i_{r+1}} \in V_1$, and the team traveling from $v_{i_r} \in V_1$.

Let us now assume that the instance of LHP is a yes-instance, implying the existence of a Hamiltonian Path such that $\sum_{j=1}^{n-1} d(h_{i_j}, h_{i_{j+1}}) \geqslant B$. We choose as the ordering of venues in V_1 the sequence of nodes in this Hamiltonian path. We find:

$$u(S) = \sum_{r=1}^{k} \big((k-1) \cdot 2D + (2D - d(v_{i_r}, v_{i_{r+1}}))\big) \tag{60}$$

$$= k^2 \cdot 2D - \sum_{r=1}^{k} d(v_{i_r}, v_{i_{r+1}}) \tag{61}$$

$$= k^2 \cdot 2D - \sum_{j=1}^{n-1} w(h_{i_j}, v_{i_{j+1}}) \tag{62}$$

$$\leqslant k^2 \cdot 2D - B = K. \tag{63}$$

Hence, the unfairness of this schedule S is bounded by K.

For the other way around, suppose there exists a schedule S whose unfairness is bounded by K. We obtain:

$$u(S) = k^2 \cdot 2D - \sum_{r=1}^{k} d(v_{i_r}, v_{i_{r+1}}) \leqslant K, \text{ which is equivalent to} \tag{64}$$

$$\sum_{j=1}^{n-1} w(h_{i_j}, v_{i_{j+1}}) \geqslant k^2 \cdot 2D - K = B. \tag{65}$$

Thus, solving this instance of the decision version of VA is equivalent with solving the corresponding instance of LHP, which implies that VA is NP-Hard. □

7.5 THE VNL IN PRACTICE: ABOUT THE HOME-VENUE-PROPERTY

Motivated by the current practice in the VNL, we incorporate the following issue in our problem formulation: each venue has a team that considers this venue as its home-venue; we refer to the team as the home-team.. This implies that in any feasible schedule for the VNL it must be the case that when a venue is hosting a group, the group must contain the home-team. This means that when there are multiple venues that have the same home-team, those venues can never host a group in the same round.

Each venue has a home-team, however, a team can have multiple home venues (or none). In the context of the VNL, the home-team of a venue is the national team of the country where the venue is located - literally the team that plays at home at that venue. The Chinese Women had a total of four different home venues in the season of 2019, and the team played in all four groups hosted by these venues.

Any solution that satisfies that a home-team plays in the group hosted by its venues, is said to satisfy the *home-venue property*.

Definition 32. *A solution to the TSGP satisfies the* home-venue property *when for each venue* v*, with home-team* t*, and each group* P *scheduled to be played at venue* v*, we have* $t \in P$*.*

A relevant question now becomes:

> *Do feasible schedules satisfying the home-venue property exist?*

Whenever a venue is scheduled to host a group without its home-team - and the schedule does not satisfy the home-venue property - none of the teams can be expected to feel connected to the venue, nor would it attract fans from within the country, as the national team is playing somewhere else. Therefore, the schedule used by the VNL always satisfies the home-venue property.

By no means did we ensure that solving the I_{VA} instance of the VENUE ASSIGNMENT problem, would lead to a feasible solution of the I_{TSGP} instance, satisfying the home-venue property. It is simply not true that, when given any assignment of venues to rounds, a schedule is guaranteed to exist such that every venue is a home venue. We present a counter example on 4 teams in Example 10, where a feasible solution on I_{VA} cannot be extended to a solution of I_{TSGP} satisfying the home-venue property.

Example 10. *Let* $I = (N = 4, k = 2, V, c, d)$ *be an instance of the TSGP, where* $V = \{v_1, \ldots, v_4\}$ *is the set of venues, of which all countries* t *are the home-team of venue* $v_t \in V$*, for* $t = 1, \ldots, 4$.

Let multiplicity function c *be defined as:*

- *When* $t = 1, 2$*,* v_t *has multiplicity* $c(v_t) = 2$.
- *When* $t = 3, 4$*,* v_t *has multiplicity* $c(v_t) = 1$.

Let d *be any distance function.*

A feasible and solution to instance I_{VA} *of the VA-problem is presented in Table 39.*

Group	Round 1	Round 2	Round 3
1	v_1	v_1	v_3
2	v_2	v_2	v_4

Table 39: Feasible solution to the VA-problem.

The venue assignment in Table 39 clearly satisfies the given multiplicities $c(v)$*. However, it is impossible to schedule match* $\{1, 2\}$ *in any round given that all teams must be scheduled in the groups hosted by their home-venues. Hence, there is no way to extend the solution of* I_{VA} *to a solution of* I.

We see that in practice, solving the Venue-Assignment alone is not necessarily the same as solving the VNL-problem. Moreover, even finding feasible solutions to the TSGP that satisfy the home-venue property, is of an unknown complexity.

However, we are allowed to make life a little easier. Just as every venue has a home-team, in the VNL we see that every team usually has a home venue as well. This makes sense, both from a fairness point of view (all teams want the benefit of a home crowd cheering) and a fans perspective. An instance containing a home-venue for every team, is said to be a *hosting instance*.

Definition 33. *An instance* I *of the TSGP for which every team has at least one home-venue* $v \in V$ *is called a* hosting instance.

As seen in Example 10, when $N = 4, k = 2$*, the VA cannot be extended to a schedule satisfying the home-away property, even though the instance is a hosting instance.*

It is not difficult to see that in any hosting instance, there must be at least N different venues v, one venue for each team. Perhaps surprisingly, the following theorem shows that for the particular dimensions of the VNL ($N = 16, k = 4$), if the given instance is a hosting instance, an optimal solution that satisfies the home-venue property can be found by first solving the Venue-Assignment problem.

Theorem 18. *Let* I *be a hosting instance of the* (N, k)-TSGP*, with* $N = 16$ *and* $k = 4$*. Then any solution of* Venue Assignment *instance* I_{VA}*, can be extended to a feasible solution of the TSGP satisfying the home-venue property.*

Proof. As I is a hosting instance, we know that $|V| \geqslant N = 16$, where each venue v has multiplicity $c(v)$, with $\sum_{v \in V} c(v) = k(k+1) = 20$. A solution to the Venue-Assignment instance, would distribute these venues over the 20 groups that need to be hosted.

We organize the proof as follows. First, we give a schedule specifying the groups (called the 'blueprint') in each of the 5 rounds, and we provide a partial designation of home venues. Next, we consider all distinct functions $c \colon V \to \mathbb{N}_{\geqslant 1}$ that the instance could have, which is all the input needed to solve the Venue-Assignment problem. For each of these functions c, we consider the set of distributions that could be the solution to the VA-instance. Finally, we outline how we can use the introduced blueprint to find a solution to the TSGP problem that satisfies the home-venue property.

Consider Table 40, where column "R_i" stands for Round i, $i = 1, \ldots, 5$, and where each number from $\{1, \ldots, 16\}$ stands for a team. We refer to this composition of groups as the 'blueprint', and we will argue that, for any possible set of multiplicities, and for any distribution of these multiplicities over the rounds, this blueprint can be turned into a venue assignment satisfying the home-venue property, and hence into a nation assignment.

R_1	R_2	R_3	R_4	R_5
1,5,9,13	1,6,11,16	1,4,10,15	1,3,12,14	1,2,7,8
2,6,10,14	2,5,12,15	2,3,13,16	2,4,9,11	3,4,5,6
3,7,11,15	3,8,9,10	6,7,9,12	5,7,10,16	10,11,12,13
4,8,12,16	4,7,13,14	5,8,11,14	6,8,13,15	9,14,15,16

Table 40: Blueprint that specifies the composition of the groups.

Of course, Table 40 does not constitute a feasible solution, as it has not been specified for each group which team plays at its home-venue. This will depend on the function c; in Table 41, we provide a *partial* specification of the teams that play at their home-venue.

Table 41 provides an initial assignment such that each team is the host in exactly 1 group, as each of the numbers $1, \ldots, 16$ occurs once.

We will now identify all relevant possible functions c, the multiplicity function, that may occur in a hosting instance. Without loss of

R_1	R_2	R_3	R_4	R_5
1	·	·	·	·
2	12	16	11	6
3	10	9	5	13
4	7	8	15	14

Table 41: Partial and provisional designation of teams that serve as host.

generality, we may assume that this function is non-increasing, so $c(1) \geqslant \cdots \geqslant c(|V|)$. Define V^+ as follows:

$$V^+ := \{v \in V \colon c(v) > 1\}.$$

As there are 16 venues, $|V^+| \leqslant 4$. We characterize a vector $\gamma(c)$ in the following way:

$$\gamma(c) := (c(v))_{v \in V^+}.$$

There are 5 distinct values $\gamma(c)$ can have, namely:

$$\gamma(c) \in \{(5), (4,2), (3,3), (3,2,2), (2,2,2,2)\}. \tag{66}$$

When we restrict a solution to the VA-problem to the rounds in which $v \in V^+$ are scheduled to host, several distributions can be the result with the same $\gamma(c)$ as input. All these distributions should be considered separately, as for all these solutions we want to create a solution to the TSGP.

All three distinct distributions of $\gamma(c) = (3,3)$:

v_1	v_2
$\{1,2,3\}$	$\{1,2,3\}$
$\{1,2,3\}$	$\{1,2,4\}$
$\{1,2,3\}$	$\{1,4,5\}$

Where each row gives the rounds in which v_i is scheduled to host.

As an example, suppose $\gamma(c) = (3,3)$. A feasible venue-assignment would be to let venues v_1, v_2 be a host venue in rounds $1,2,3$. Another possibility would be to have v_1 host in round $1,2,3$ and v_2 in rounds $3,4,5$. Each of these solutions can be identified by their distributions of V^+ over the rounds. We say two distributions of the venue-assignments of V^+ are distinct when no combination of permuting the rounds or relabeling the venues, maps one distribution to the other.

Given $\gamma(c)$, it is not immediately clear how many pairwise distinct distributions exist. Let $D(\gamma(c))$ denote how many of such distributions exists. It can be verified that:

$$\begin{aligned}
D(\gamma((5))) &= 1\\
D(\gamma((4,2))) &= 2\\
D(\gamma((3,3))) &= 3\\
D(\gamma((3,2,2))) &= 11\\
D(\gamma((2,2,2,2))) &= 17
\end{aligned}$$

In Table 42, all distributions are listed.

$\gamma(c)$	VA-solution	Labeled teams	Altered Venues
(5)	(A, A, A, A, A)	$A:1$	
$(4, 2)$	(AB, A, A, A, B)	$A:1, B:2$	
	$(AB, AB, A, A, -)$	$A:1, B:4$	$R_5:7$
$(3, 3)$	(AB, A, A, B, B)	$A:1, B:2$	$R_1:11, R_4:3$
	$(AB, AB, A, B, -)$	$A:1, B:2$	$R_3:11, R_4:12, R_5:8$
	$(AB, AB, AB, -, -)$	$A:1, B:2$	$R_1:16, R_2:4, R_4:12, R_5:7$
$(3, 2, 2)$	$(ABC, ABC, A, -, -)$	$A:1, B:2, C:4$	$R_4:12, R_5:7$
	$(ABC, AB, AC, -, -)$	$A:1, B:7, C:8$	$R_1:6, R_4:3, R_5:2,4$
	$(ABC, AB, A, C, -)$	$A:1, B:4, C:3$	$R_5:7$
	$(AB, AB, AC, C, -)$	$A:1, B:2, C:12$	$R_2:14, R_5:7,9$
	$(ABC, A, A, BC, -)$	$A:1, B:14, C:11$	$R_4:6, R_5:2,3,15$
	(ABC, A, A, B, C)	$A:1, B:3, C:2$	
	$(AB, AC, A, BC, -)$	$A:1, B:2, C:12$	$R_3:11, R_5:8$
	(AB, AC, A, B, C)	$A:1, B:3, C:7$	
	(AB, AB, A, C, C)	$A:1, B:4, C:14$	$R_5:7$
	(AB, A, A, BC, C)	$A:1, B:4, C:14$	$R_3:11, R_5:8$
	(A, A, A, BC, BC)	$A:1, B:7, C:14$	$R_2:5,13, R_5:12$
$(2, 2, 2, 2)$	$(ABCD, ABCD, -, -, -)$	$A:1, B:2, C:3, D:4$	$R_3:10, R_4:12, R_5:7$
	$(ABCD, ABC, D, -, -)$	$A:1, B:2, C:3, D:4$	$R_3:5, R_4:10,12, R_5:8$
	$(ABC, ABC, D, D, -)$	$A:1, B:2, C:4, D:15$	$R_4:12, R_5:7$
	$(ABCD, AB, CD, -, -)$	$A:1, B:2, C:3, D:4$	$R_3:5, R_4:16,12, R_5:8$
	$(ABCD, AB, C, D, -)$	$A:1, B:2, C:4, D:3$	$R_2:13, R_5:7,12$
	$(ABC, ABD, CD, -, -)$	$A:1, B:2, C:4, D:8$	$R_2:13, R_4:12, R_5:7,10$
	$(ABC, ABD, C, D, -)$	$A:1, B:10, C:4, D:5$	$R_4:12, R_5:2$
	$(ABC, AB, CD, D, -)$	$A:1, B:2, C:4, D:11$	$R_4:12, R_5:8$
	(ABC, AB, C, D, D)	$A:1, B:3, C:8, D:14$	$R_2:4, R_3:10, R_5:7$
	$(AB, AB, CD, CD, -)$	$A:1, B:4, C:15, D:16$	$R_2:5, R_4:12, R_5:7$
	(AB, AB, CD, C, D)	$A:1, B:2, C:15, D:8$	$R_4:12$
	$(ABCD, A, B, C, D)$	$A:1, B:4, C:3, D:2$	
	$(ABC, AD, BD, C, -)$	$A:1, B:4, C:3, D:8$	$R_1:10, R_5:2$
	(ABC, AD, B, C, D)	$A:1, B:4, C:3, D:7$	
	$(AB, AC, BD, CD, -)$	$A:1, B:4, C:12, D:11$	$R_5:8$
	(AB, AC, BC, D, D)	$A:1, B:16, C:4, D:14$	$R_5:7$
	(AB, AC, BD, C, D)	$A:1, B:4, C:12, D:8$	

Table 42: Extending the blueprint to a feasible schedule satisfying the home-venue property.

We now describe how Table 42 can be read such a solution to the I_{VA}, of which the main characteristics are given in the second column, to a solution of I_{TSGP} with equal unfairness u.

Each row contains a distinct solution to the VA expressed in labeled teams A, B, C up to D for all teams with a venue $v \in V^+$, and this solution is given as a vector of length 5. The r-th entry of the vector relates to round r, and when label X is part of the r-th entry, this implies that X (and its venue) hosts a group in round r.

Example
Initial venue assignment is:

	R_1	R_2	R_3	R_4	R_5
1	.	.	.	.	.
2	12	16	11	6	
3	10	9	5	13	
4	7	8	15	14	

Table 43: After Step 1

We look at the fifth row of Table 42. Label A refers to 1, B to 2. They host in rounds 1, 2, 3 and 1, 2, 4 respectively:

	R_1	R_2	R_3	R_4	R_5
1	1*	1*	.	.	
2	2*	16	2*	6	
3	10	9	5	13	
4	7	8	15	14	

Table 44: After Step 3

In column 'Altered Venues' we see the rounds in which we should let some other teams host.

	R_1	R_2	R_3	R_4	R_5
1	1	1	12*	8*	
2	2	16	2	6	
3	10	9	5	13	
4	7	11*	15	14	

Table 45: After Step 4

Table 45 satisfies constraints of c and is compatible with Table 40, satisfying the home-venue property.

To get from a solution of I_{VA} to a solution of I_{TSGP}, with equal objective $u_{VA} = u_{TSGP}$, Table 42 need to be read in the following way:

1. Use the group composition from the blueprint in Table 40 and the partial venue assignment from Table 41 as a start. Notice that the latter may be altered; the former will not change.
2. The second column in Table 42 denotes in which of the rounds $R_1, \ldots, R_5$ the labeled teams/venues should host a group.
3. The third column assigns a specific team to each of the used labels. As a consequence, this may induce a change to the partial venue assignment specified in Table 41 - the hosting defined in the second column overrules the partial venue assignment.
4. In the previous step, some teams/venues $v \in V \setminus V^+$ have lost the original group they hosted, and are now left empty-handed. In the fourth column, "Altered Venues", additional changes to the schedule are denoted, all additional changes to the partial venue assignment are denoted.

After executing the above steps, all 'special' teams with venues in V^+ are labeled and have been assigned rounds in which they host. All is left is to link all remaining teams of instance I_{TSGP} with venues in $V \setminus V^+$ to venues with multiplicity 1 in the solution of I_{VA}. With this, we have constructed a solution to the I_{TSGP} instance from the solution of I_{VA}.

As we can do this for all instances of I_{TSGP} that are hosting instances, we are done. □

The previous theorem is tight in the sense that, when $k = 3$ and $N = k^2$, it is not enough to have an hosting instance to be able to extend any Venue-Assignment to a solution of the TSGP satisfying the home-venue property - counter examples exist.

We like to point out that interestingly enough, in each of the schedules created in the proof of Section 7.5, the composition of the groups as specified in the blueprint from Table 40 is identical. It is also an interesting question to see for which other, non-hosting instances I, a solution to I_{VA} could be extended to a solution of the TSGP satisfying the home-venue property, while retaining the objective value.

7.6 SOLVING REAL-LIFE INSTANCES OF THE VNL

In the previous section we have layed the theoretical groundwork that help scheduling the VNL instances with respect to our objective. In this chapter, we will apply what we know to find optimal schedules for previous instances of the VNL. In Section 7.6.1, we give an integer programming formulation of the VENUE-ASSIGNMENT Problem. Section 7.6.2 presents the outcomes.

7.6.1 An Integer Programming Formulation

Let $x_{v,r}$ be the binary variables that indicate whether venue $v \in V$ hosts a group in round $r \in \{1, \ldots, 5\} = R$. We use real variables $s_{v,w,r}$ (capturing distances between venues v and w acting as host in rounds r and $r+1$), $m_{v,r}$ (capturing the largest distance traveled to venue v in round r), and $K_{v,r}$ (capturing the difference in travel distance to venue v in round r). Let $\Delta = \max_{v,w} d(v,w)$, and let $W \subset V \times V$ be the set of pairs of venues that cannot both host a group in the same round. The following IP minimizes u, the sum of the difference in travel distances per group, over the groups.

$$\min \sum_{v \in V} \sum_{r \in R} K_{v,r} \tag{67}$$

$$\text{s.t.} \sum_{v \in V} x_{v,r} = k \qquad \forall r \in R, \tag{68}$$

$$\sum_{r \in R} x_{v,r} = c(v) \qquad \forall v \in V, \tag{69}$$

$$x_{v,r} + x_{w,r} \leqslant 1 \qquad \forall r \in R,\ \forall (v,w) \in W, \tag{70}$$

$$s_{v,w,r} \geqslant d_{v,w}(x_{v,r} + x_{w,r-1} - 1) \qquad \forall v,w \in V,\ \forall r \in R \setminus 1, \tag{71}$$

$$s_{v,w,r} \leqslant \min(d_{v,w} x_{v,r}, d_{v,w} x_{w,r-1}) \qquad \forall v,w \in V,\ \forall r \in R \setminus 1, \tag{72}$$

$$m_{v,r} \geqslant s_{v,w,r} \qquad \forall v,w \in V,\ \forall r \in R \setminus 1, \tag{73}$$

$$K_{v,r} \geqslant m_{v,r} - s_{v,w,r} - D(1 - x_{w,r-1}) \qquad \forall v,w \in V,\ \forall r \in R \setminus 1, \tag{74}$$

$$x_{v,r} \in \{0,1\}, K_{v,r} \geqslant 0 \qquad \forall v \in V, r \in R. \tag{75}$$

First, observe that (67) captures the objective function, minimizing $u = \sum_{v,r} K_{v,r}$. Next, constraints (68) ensure that in every round, k venues are host; constraints (69) ensure that every venue hosts as often as required; constraints (70) ensure that two venues that should not host simultaneously, will not host simultaneously. Auxiliary variables $s_{v,w,r}$ are at least as large as $d_{v,w}$, the distance traveled between venues w, v in rounds $r-1$ and r if these venues host in the respective rounds, by constraints (71), but never larger than $d_{v,w}$ by constraints (72). The variables $m_{v,r}$ equal the maximum distances traveled to venue v in round r (can equal zero 0 if v does not host in round r), as defined by (73), and $K_{v,r}$ resembles the difference in traveled distances towards v in round r compared to the maximum travel distance, where the terms $-D \cdot (1 - x_{w,r-1})$ in (74) nullify any influence caused by distances between a venue that does not host in round $r-1$.

7.6.2 Results

As instances of VNL satisfy the conditions of Section 7.5, we can proceed applying the integer programming formulation Equation (67)-(75) for the Venue-Assignment to the known instances of the Volleyball Nations League, and compare our solution to that of the schedules used in practice. Formulation (67)-(75) is implemented in Python 3 using Gurobi 9.0. All computations have been done on a laptop with an Intel Core i7-7700HQ CPU 2.8-GHz processor and 32 GB RAM. The distances between venues are obtained via https://www.distancecalculator.net/, and are divided by 100 and rounded down. The four instances that we analyze are the Women's and Men's tournaments of 2018 and 2019, and all these tournaments can be scheduled independent of each other. All values resulting from solving (67)-(75) are mentioned to be optimal by the solver and are found within approximately 2 hours of computation time.

In Table 49 we give the unfairness corresponding to the optimal venue assignment, $u(S_{opt})$, and we give the unfairness that corresponds to the venue assignments used in practice, $u(S_{real})$. Also we give the total travel distance for the two corresponding solutions, $d(S_{opt})$ and $d(S_{real})$, where the distance is given in units of 100 km.

As is imminent from Table 49, the fairness of the schedules used in the Volleyball Nations League can be much improved in comparison to the schedules that have been used. Moreover, these improvements in fairness do not come at the expense of the total travel distance; indeed, total travel distance is similar for our schedules when compared to the real life schedules.

We end this chapter by discusing one of the obtained optimal schedules in more detail. In Table 50, the optimal schedule for the 2018

Instance	$u(S_{opt})$	$u(S_{real})$	$d(S_{opt})$	$d(S_{real})$
M2018	233	1366	4272	4806
W2018	381	1541	4956	4169
M2019	347	1239	5237	4657
W2019	491	1288	4214	3708

Table 49: Unfairness u of VNL-instances compared to the optimal schedule, and their total travel distance.

Men's tournament is shown, and the schedule used in practice is shown in Table 51. The optimal schedule with respect to fairness u is also depicted in the map of Figure 14, where each number indicates the rounds in which the venue located at that spot hosts a group.

Round 1	Round 2	Round 3	Round 4	Round 5
Melbourne (AUS)	Goiânia (BRA)	*Katowicze (POL)*	*Aix-en-Prov. (FRA)*	Hoffman Est. (USA)
Tehran (IRA	Jiangmen (CHN)	*Kraljevo (SRB)*	*Lodz (POL)*	Ningbo (CHN)
Ufa (RUS)	Osaka (JPN)	*Rouen (FRA)*	*Ludwigsb. (GER)*	Ottawa (CAN)
Varna (BUL)	Seoul (KOR)	*Sofia (BUL)*	*Modena (ITA)*	San Juan (ARG)

Table 50: Optimal nation assignment for Men's VNL 2018, with European venues in italics.

Round 1	Round 2	Round 3	Round 4	Round 5
Rouen (FRA)	Goiânia (BRA)	Ottawa (CAN)	Seoul (KOR)	Melbourne (AUS)
Ningbo (CHN)	*Sofia (BUL)*	Osaka (JPN)	*Ludwigsb. (GER)*	Jiagmen (CHN)
Katowicze (POL)	*Lodz (POL)*	*Ufa (RUS)*	Hoffman Est. (USA)	Tehran (IRA)
Kraljevo (SRB)	San Juan (ARG)	*Aix-en-Prov. (FRA)*	*Varna (BUL)*	*Modena (ITA)*

Table 51: Real-life nation assignment for Men's VNL 2018, with European venues in italics.

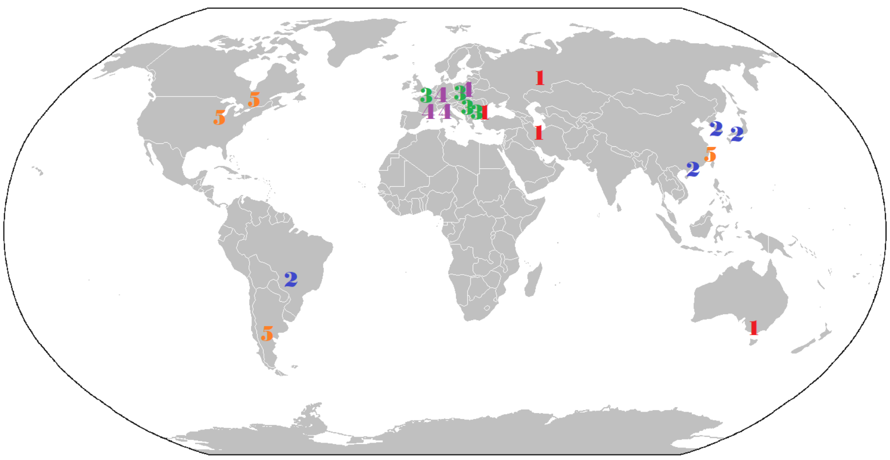


Figure 14: Optimal venues per round, VNL Men 2018

We can see that the optimal schedule creates two specific European rounds, where all groups are played within Europe, and two rounds without any group in Europe. In this way, all teams are gathered relatively close to each other, thus in the next round, they will all have roughly the same travel burden, which diminishes the unfairness of the schedule. In contrast to the optimal schedule, the schedule that was used in practice had both European and non-European venues in every round - thus leading to a high amount of unfairness.

The total distance that needs to be traveled by the teams, is more or less the same for the optimal schedule and the schedule used in practice, so the organizers could have gained a lot in making the schedule more fair without additional harm to the environment or the players

We again like to point out that, as the unfairness in travel times (as well as total traveled distance) is completely determined by the venue assignment, any nation assignment is equally good with respect to these objectives. Thus, apart from satisfying the underlying SGP and assigning teams to their designated home venues, there is complete freedom to optimize the nations assignment to whatever other objectives the organizers see fit; this can be done without compromising on the original objectives.

7.7 CONCLUSION

We have introduced the Travelling Social Golfer Problem (TSGP), generalizing the well-known Social Golfer Problem, to model the scheduling of the Volleyball Nations League. Solving the TSGP allows us to model the unfairness of a schedule that focusses on minimizing the differences in travel time between opposing teams. We show that this problem can be decomposed into two subproblems, Venue Assignment and Nation Assignment, and we argue that solving the Venue Assignment determines the amount of unfairness. We describe the home-venue property that is present in real-life solutions, and we show that, for the specific dimensions of the VNL, such solutions always exist. Finally, we model the problem as an integer program, and solve the real-life instances of 2018 and 2019.

The results show that large improvements in fairness are possible, without increasing total travel time. Moreover, it is possible to calculate and compare travel dependent metrics of a schedule, without knowing the entire schedule and just comparing venue assignment. This could help organizers scheduling new tournaments in a fairer way.

8 FAIRNESS IN PENALTY SHOOTOUTS

As the final minutes of the football European Championships final passed by, tensions grew, not only with the spectators on the couch, but also in the stadium and on the pitch. None of the teams wanted to be that team that made a fatal mistake with mere seconds left to play, and see the other lift the trophy. Thus, the game faded, with everyone waiting anxiously until the moment the referee blew the final whistle. At which point, the nerve-wrecking excitement only just started.

The penalties. In a best-of-5 series, with sudden death if that resulted in a tie, both teams had to take penalties to decide a winner. The Italians knew what they had to do, as they had won the semi-finals after penalties.

It is an odd thing that some of the most influential moments at major football tournaments have little to do with the regular game. Penalty series are a discipline on its own, or a lottery, as some less-gifted penalty takers have called it at times.

They are no lottery, you could and should practice them to increase your chances, but they are meant to give both teams an equal and fair chance of progressing to the next round or even winning the tournament. And why wouldn't they be fair? Both teams get the same task - convert as many penalties as possible - and thus have the same chance of winning. Right?

There is a catch. Both teams indeed have the same objective, but they don't have the same path. In a usual penalty series, there is one team that starts the series and the other one has to catch-up all the time. Surprisingly, statistical research has suggested that this actually gives an advantage to the team starting first! This knowledge and feeling is in fact so widely accepted that most trainers will have their team shoot first if they win the toss to determine the starting team.

This chapter is based on Lambers and Spieksma (2021).

8.1 INTRODUCTION

8.1.1 Motivation

For those still in suspense who won the penalty series in the final of the European Championships: Italy

We consider the following situation. Two teams, called A and B, play a match. To decide upon a winner of an otherwise tied game, a so-called *shootout* takes place. This shootout has two phases. Phase 1 consists of k rounds, and in every round each of the teams shoots once (what it means to shoot, depends on the particular sport). Shooting leads either to success (scoring), or to failure (missing). The team that has the most successes after Phase 1 (which we call a best-of-k) wins the match. If both teams have the same number of successes, the shootout continues with Phase 2 (which we call sudden death). This phase consists of individual rounds, and ends only when in a particular round one team scores, and the other does not.

Many popular sports use shootouts to identify a winner, though the setup can differ per sport. For example, in football, FIFA rules (FIFA, 2020) prescribe that in each round the same team shoots first both in Phase 1 as well as in Phase 2. The resulting sequence is denoted as AB/AB/AB/AB/AB||(AB)$^{\infty}$, where the symbol "||" is used to separate Phase 1 from Phase 2, and where the symbol "/" is used to separate the rounds. This sequence is called the penalty sequence. Another example of a shootout is the tiebreaker in tennis - the serving player can be seen as the team shooting a penalty in football. The rules of the tiebreak (see (ITF, 2019)) stipulate that Phase 1 is a best-of-6, with the first serving player alternating in each round, while in Phase 2 the first serving player also alternates in each round: AB/BA/AB/BA/AB/BA||(AB/BA)$^{\infty}$; this sequence is called the ABBA sequence. Other sports using shootouts are field hockey, ice hockey, rugby, waterpolo (Wikipedia, 2020).

In a seminal paper by Apesteguia and Palacios-Huerta (2010), the penalty sequence used in football is shown to give an advantage to the team that starts, Team A. This is generally explained by the pressure of lagging behind exercised on the second shooting team B, resulting in a so-called First Mover Advantage (FMA). Indeed, the consecutive nature of the two penalties in a round gives an asymmetry between the first shooting team and the second shooting team. Thus, even when the two teams are equally strong, i.e., even when their probabilities of scoring are the same in all situations, their chances of winning the match may differ. This difference in win probabilities depends on the particular sequence of the shootout. In this chapter we investigate the existence of so-called fair sequences, both for the best-of-k and for the sudden death.

8.1.2 Related Literature

Wright (2014) gives an overview of rules in various sports that affect fairness, see also Kendall and Lenten (2017), and Haigh (2009). The subject of shootouts, and the possible presence of FMA in football shootouts, is heavily debated in the literature. Since the work of Apesteguia and Palacios-Huerta (2010), the presence of a FMA has been confirmed in Palacios-Huerta (2014), Anbarci, Sun, and Ünver (2015), Vandebroek, McCann, and Vroom (2018) and Rudi, Olivares, and Shetty (2020), while it has not been found in Kocher, Lenz, and Sutter (2012), and Arrondel, Duhautois, and Laslier (2019). We quote Csató (2021): "To summarize, while the empirical evidence remains somewhat controversial, it seems probable that the team kicking the first penalty enjoys an advantage".

Even for those that doubt the existence of an FMA in football, it is well established that psychological factors have an impact on the probability of scoring. This is shown by Jordet et al. (2007), and in Arrondel, Duhautois, and Laslier (2019). In the latter study, they identify three different situations (called "survival", "catch-up", and "break point"), which are shown to have different impacts on the scoring probability.

Different models have been proposed to capture the FMA. A frequently used model explains the existence of an FMA by using a probability to score when trailing, and a probability to score when not trailing (see the appendix of (Apesteguia and Palacios-Huerta, 2010)). They derive corresponding win probabilities, and Vandebroek, McCann, and Vroom (2018) show that within this model, when using the penalty sequence, the FMA is irrespective of the length of the shootout. For a comprehensive analysis of win probabilities on various (dynamic) sequences, see Csató and Petróczy (2022).

Brams and Ismail (2018) specifically focus on fairness in shootouts. While accepting the existence of an FMA, they propose and analyze rules where the first shooting team in a round is determined by the outcomes of previous rounds. Among such dynamic sequences are the Catch Up Rule and the Behind First Alternating Order Rule (Anbarci, Sun, and Ünver, 2015),(Csató and Petróczy, 2022).

As a puzzle, FiveThirtyEight analysed a hypothetical basketball game with a peculiar Second Mover Advantage.

It is interesting to note that other sports than football, have other scoring probabilities. For instance, the probability of scoring a penalty in ice hockey equals around 33% (Kolev, Pina, and Todeschini, 2015). In such a situation, as missing the penalty is the expected outcome, the pressure moves to the goalie; one can view the goalie in ice hockey as the one taking the penalty; the goalie "scores" when the goalie stops the penalty. In line with the existence of an FMA, Kolev, Pina, and Todeschini (2015) claim there is advantage in shooting second.

A popular sequence in the scientific literature is the so-called Prohuet-Thue-Morse (PTM) sequence. This sequence has many applications in various branches of science, see Allouche and Shallit (1999) for an overview. In the field of shootouts, it has for instance been studied in Brams and Taylor (1999), Palacios-Huerta (2012), Rudi, Olivares, and Shetty (2020) and Cohen-Zada, Krumer, and Shapir (2018).

8.1.3 Our contribution

We analyze a shootout with a prescribed format, i.e., we do not allow the sequence to depend on outcomes during the shootout. Teams A and B have the same scoring probabilities in the same situation, i.e., teams A and B are *equally strong*; these scoring probabilities remain constant during the shootout. Following literature (see e.g. Brams and Ismail (2018)), we define the concept of a fair sequence in Section 8.2. We present the following results in this chapter:

- In Section 8.3.2, we show that a sudden death that is a repetition of a finite sequence, is not fair.
- In the same section, we show that the PTM sequence is unfair.
- In Section 8.3.4 we give an algorithm that outputs a fair sequence for the sudden death.
- We show how to find the least unfair sequences for best-of-k in Section 8.4.2.
- We compute least unfair sequences for relevant parameters when $k = 5$ in Section 8.4.3 and for different values of k on a specific set of parameters when Section 8.4.3.

We conclude in Section 8.5.

8.2 PRELIMINARIES

As sketched in Section 8.1, a shootout between teams A and B consists of a best-of-k, followed by a sudden death. Each of these two phases consists of rounds, and every round has one of the two teams shooting first. The problem is to specify, prior to the start of the shootout, for each of the upcoming rounds both in the best-of-k, as well as in the sudden death, which team will shoot first; a specification for the sudden death will be referred to as a *sequence*, and as a *finite sequence* if it is only a specification for a finite number of rounds. For instance, Phase 1 of the penalty sequence can be written as the finite sequence $AAAAA$, while Phase 2 of the ABBA sequence can be written as $ABAB\cdots$.

From here on, we always assume Team A and Team B to be equally strong. We say a team wins a round, if it scores in that round while the other team does not. If neither of the teams wins the round, the round is tied. The phenomenon of the first shooting team having higher probability of winning the round than the second, is called the First Mover Advantage (FMA).

To model the sudden death, we introduce the following probabilities:

- $P_+ = \mathbb{P}(\text{First shooting team wins round})$,
- $P_- = \mathbb{P}(\text{Second shooting team wins round})$,
- $P_\pm = \mathbb{P}(\text{The round is tied})$.

With $\mathcal{P} = \{P_+, P_-, P_\pm\}$ we can refer to all the relevant probabilities. The First Mover Advantage can be quantified as:

$$\lambda = \mathbb{P}_+ - \mathbb{P}_- > 0.$$

The *length* of a finite sequence is the number of rounds for which the sequence specifies the first shooting team. We denote the set of finite sequences of length n by $\mathcal{S}_n$. We say that a sequence S is *repetitive* if it consists of the concatenation of a finite sequence $S_n \in \mathcal{S}_n$, i.e., $S = S_n S_n S_n \cdots$. A sequence S may or may not be a repetitive, in any case, it should specify for all rounds $r \in \mathbb{N}$ which team shoots first. For a given sequence, we define the concept of being *fair* in the following way.

Definition 34. *Let teams A and B be equally strong. Given $\mathcal{P}$, a sequence S is called* fair, *if:*

$$\mathbb{P}(\textit{Team A wins}) = \mathbb{P}(\textit{Team B wins}). \tag{76}$$

If S is not fair, it is unfair.

We will now investigate the existence of fair sequences for sudden death in Section 8.3, and for best-of-k in Section 8.4.

8.3 SUDDEN DEATH

In this section, we consider sequences that specify the first shooting team in the sudden death phase of the shootout. We introduce the characteristic polynomial of a sequence and we show that when $P_\pm \in \mathbb{Q}$, no repetitive sequence is fair. We also show that the so-called Prohuet-Thue-Morse sequence is unfair for all $P_+ > P_-$. In Section 8.3.4, we introduce an algorithm that for a given $\mathcal{P}$, returns a fair sudden death sequence if it exists.

8.3.1 The characteristic polynomial of a sequence

In a sudden death shootout, starting each round, the score is a draw - if one team gains an advantage by winning a round, that team wins the sudden death and the game is over. Thus, for a team to win Round r, the first $r-1$ rounds resulted in a draw. We can state the probability of either team winning as:

$$\mathbb{P}(\text{Team A wins}) = \sum_{r=1}^{\infty} \mathbb{P}(r-1 \text{ rounds tied})\mathbb{P}(\text{Team A wins in } r),$$
$$\mathbb{P}(\text{Team B wins}) = \sum_{r=1}^{\infty} \mathbb{P}(r-1 \text{ rounds tied})\mathbb{P}(\text{Team B wins in } r).$$

The probability for team A to win in a certain Round r, depends on who shoots first in that round. We take I to be the index set of all rounds in which A is allowed to shoot first. In the popular $AAAA$-series, this means $I = \mathbb{N}$, while in the $ABBA$-series, this would mean $I = \{1, 3, 5, \dots\}$. Of course, in the remaining rounds $\bar{I} = \mathbb{N} \setminus I$, team B shoots first.

Recall that P_+, P_- are the probabilities that the first respectively second shooting team wins a round, with $P_\pm$ as the probability the round will be tied, and $\lambda = P_+ - P_-$ as FMA. We present the following lemma:

Lemma 13. *Let $\mathcal{P}$ be given. Teams A and B, shooting first in rounds $I, \bar{I}$ respectively, win the sudden death with equal probability if and only if:*

$$\sum_{r \in I} P_\pm^{r-1} = \sum_{r \in \bar{I}} P_\pm^{r-1}. \tag{77}$$

Proof. We prove this by a straightforward calculation.

$$\begin{aligned}
\mathbb{P}(\text{A wins}) &= \sum_{r=1}^{\infty} \mathbb{P}(r-1 \text{ rounds drawn})\mathbb{P}(\text{A wins Round } r) \\
&= \sum_{r \in I} P_\pm^{r-1} P_+ + \sum_{r \in \bar{I}} P_\pm^{r-1} P_- \\
&= \sum_{r \in I} P_\pm^{r-1} (P_- + \lambda) + \sum_{r \in \bar{I}} P_\pm^{r-1} P_- \\
&= \sum_{r=1}^{\infty} P_\pm^{r-1} P_- + \lambda \sum_{r \in I} P_\pm^{r-1} \\
&= \frac{P_-}{1 - P_\pm} + \lambda \sum_{r \in I} P_\pm^{r-1}.
\end{aligned}$$

Similarly, we find for Team B:

$$\mathbb{P}(\text{Team B wins}) = \frac{P_-}{1-P_\pm} + \lambda \sum_{r \in \bar{I}} P_\pm^{r-1}.$$

Since the first term $\frac{P_-}{1-P_\pm}$ is equal for both teams, the lemma follows. □

Definition 35. *Given an index set* I*, define* $h_I(x) = \sum_{i \in I} x^{i-1}$*. For a (finite) sequence* S*, with team* A *shooting first in rounds* I*, define the characteristic polynomial* $f_S(x) = h_I(x) - h_{\bar{I}}(x)$*.*

We state the following corollary to Lemma 13.

Corollary 3. *Let* $\mathcal{P}$ *be given. A sequence* S *is fair if and only if* $f_S(P_\pm) = 0$*, i.e., if* $P_\pm$ *is a zero of* $f_S(x)$*.*

8.3.2 Repetitive sequences are unfair

Although Lemma 13 and Corollary 3 provide a necessary and sufficient condition for a sequence to be fair, it is not clear when Equation (77) is satisfied. The following theorem clarifies the status for repetitive sequences.

Theorem 19. *Let* $P_\pm \in \mathcal{P}$ *be rational, i.e.,* $P_\pm \in \mathbb{Q}_{(0,1)}$*. Each repetitive sequence* S *is unfair.*

Proof. Consider a repetitive sequence S that is a concatenation of some finite sequence S_n. Let I_n be the index set indicating when Team A shoots first in this sequence S_n. The characteristic polynomial of S_n is then given by $f_{S_n}(x) = \sum_{i \in I_n} x^{i-1} - \sum_{i \in \bar{I}_n} x^{i-1}$, a polynomial of degree $n-1$. Since S is repetitive, it follows that:

$$f_S(x) = f_{S_n}(x) \cdot (1 + x^n + x^{2n} + \cdots) = \frac{f_{S_n}(x)}{1-x^n}.$$

Clearly:

$$f_S(P_\pm) = 0 \iff f_{S_n}(P_\pm) = 0$$

The following claim states that, for $x \in (0,1)$, there is no rational solution to $f_{S_n}(x) = 0$.

Claim 3. *Any finite degree polynomial* $u(x)$ *with coefficients in* $\{-1, 1\}$ *has no rational zero's in* $(0,1)$*.*

Proof. Consider $u(x) = \sum_{i=0}^{n} c_i x^i$ with $c_i \in \{-1, 1\}$, and let $\frac{w}{v} \in \mathbb{Q}_{(0,1)}$ with $\gcd(w, v) = 1$, be such that $u(\frac{w}{v}) = 0$. Then, it must hold that $v^n u(\frac{w}{v}) = 0 \mod v$ as well. However, $v^n u(\frac{w}{v}) = c_n w^n \mod v$. As $\gcd(w, v) = 1$, this leads to $c_n w^n = 0 \mod v$ and this is not possible unless $w = v = 1$ or $w = 0$. □

As $P_\pm \in \mathbb{Q}$, there are no $P_\pm$ that satisfy $f_{S_n}(P_\pm) = 0$ and we conclude that S is unfair. □

Remark. It follows easily from Theorem 19 that, given rational $\mathcal{P}$, no finite sequence will be fair. In addition, observe that the popular $AAAA$- and $ABAB$-sequences are unfair for all real values of $\mathcal{P}$ when $\lambda = P_= - P_- > 0$. Indeed, this follows as $f_{AAAA}(x) = \frac{1}{1-x} > 0$ and $f_{ABAB}(x) = \frac{1}{1+x} > 0$ for $x \in (0, 1)$.

The relevance of Theorem 19 lies in the guaranteed presence of unfairness; when picking/determining $\mathcal{P}$ from empirical data and deciding upon a finite sequence to be repeated, any resulting sequence is unfair.

8.3.3 The Prouhet-Thue-Morse sequence is unfair

In order to mitigate the First Mover Advantage, the Prouhet-Thue-Morse (PTM) sequence is suggested in (Brams and Taylor, 1999) *The Win-Win solution*. It was also proposed in (Palacios-Huerta, 2012). For the tennis tiebreak, it was discussed by Cohen-Zada, Krumer, and Shapir (2018). In contrast to their observation, we show that in our model, for any $P_\pm \in \mathcal{P}$, the PTM-sequence is an unfair sequence.

The first steps of constructing the PTM-sequence:

Iter 0 1

Iter 1 10

Iter 2 1001

Iter 3 10010110

Definition 36. *The* Prouhet-Thue-Morse *sequence is a sequence containing only zeroes and ones, and is obtained in the following way.*

1. *Start with sequence* $s = (1)$.
2. *Given* $s = (s_i)_i$, *construct* $\bar{s}$:

$$\bar{s} := (\bar{s}_i) \qquad \bar{s}(i) = 1 - s(i) \forall i$$

3. *Set* $s = (s\bar{s})$.
4. *Go to Step* 2.

When looking at this as a penalty sequence, one can let team A start the Round n when $PTM(n) = 1$ and team B when $PTM(n) = 0$.

Theorem 20. *The PTM-sequence is unfair for all* $\mathcal{P}$.

Proof. We will construct the characteristic polynomial of the PTM-sequence and show that is bounded away from 0 in the interval $(0, 1)$.

Let S_n be the first n rounds of the PTM-sequence, and for notational purposes, let $f_n(x)$ be its characteristic polynomial. Clearly, for $n = 2$, $f_2(x) = 1 - x$. By construction, the first 2^k terms of the characteristic polynomial of the PTM-sequence are inverted to obtain the terms $2^k + 1$ up to 2^{k+1}. For the characteristic polynomial, inversion of a coefficient is just multiplication by -1, which leads to the following expression:

$$f_{2^{k+1}}(x) = f_{2^k}(x)(1 - x^{2^k}).$$

To explain this equality, notice that the first 2^k terms of $f_{2^{k+1}}$ terms are the same as $f_{2^k}(x)$. The next 2^k terms all have degree x^{2^k} or higher, and are the inverse of $f_{2^k}(x)$ - hence the multiplication of $f_{2^k}(x)$ with the term $-x^{2^k}$.

Applying this expression iteratively for $2^1, 2^2, 2^4 \ldots$, we get (Allouche and Shallit (1999)):

$$f_{PTM}(x) = (1 - x)(1 - x^2)(1 - x^4) \cdots = \prod_{k=0}^{\infty} \left(1 - x^{2^k}\right). \tag{78}$$

We now prove that there is no $x \in (0, 1)$ for which $f_{PTM}(x) = 0$, implying there is no $P_{\pm}$ for which the PTM-sequence is fair.

Claim 4. $f_{PTM}(x) > 0$ *for* $x \in (0, 1)$.

Proof. Clearly, $f_{PTM}(x)$ is decreasing on $(0, 1)$. Notice that $f_{PTM}(x) = (1 - x) f_{PTM}(x^2)$, for all x. Suppose there is an $x \in (0, 1)$ for which $f_{PTM}(x) = 0$. Consequently, as $1 - x > 0$, this implies that $f_{PTM}(x^2) = 0$. As $x^2 < x$, this implies that if there is an $x \in (0, 1)$ for which $f_{PTM}(x) = 0$, we can pick an arbitrarily small x' for which $f_{PTM}(x') = 0$. As the function f_{PTM} is decreasing on $(0, 1)$, and attaining $f(1) = 0$, either $f(x) = 0$ for all $x \in (0, 1)$, or $f(x) > 0 \ \forall x \in (0, 1)$.

Take $x = \frac{1}{2}$. Then:

$$\begin{aligned} f_{PTM}\left(\frac{1}{2}\right) = \left(1 - \frac{1}{2}\right)\left(1 - \frac{1}{2^2}\right) \cdots \quad &> 1 - \frac{1}{2} - \frac{1}{4} - \frac{1}{16} \cdots \\ > 1 - \frac{1}{2} - \frac{1}{4} - \frac{1}{8} \quad &= \frac{1}{8} > 0. \end{aligned}$$

□

Using Claim 4, we conclude that there are no values $x \in (0, 1)$ for which the PTM-sequence is fair, and hence the proof is complete. □

8.3.4 An algorithm generating a fair sequence

We established in Section 8.3.2 that, for given values $\mathcal{P} \subset \mathbb{Q}$, there are no finite fair sequences. However, it is possible to construct infinite sequences that are fair, for any value of $P_\pm \geqslant 0.5$. Consider the following algorithm which takes $P_\pm \geqslant 0.5$ as input.

Notice that if $P_\pm < 0.5$, the team shooting first will have an insurmountable FMA and no fair sequence exists

---

Algorithm 7 Fair Shootout

Let Team A shoot first in Round 1 and set $I_1 = \{1\}$. Starting from $n = 1$, construct I_{n+1} from I_n in the following way:

STEP 1: If $h_{I_n}(P_\pm) < h_{\bar{I}_n}(P_\pm)$:

- $I_{n+1} = I_n \cup \{n+1\}$ and $\bar{I}_{n+1} = \bar{I}_n$.

Else:

- $I_{n+1} = I_n$ and $\bar{I}_{n+1} = \bar{I}_n \cup \{n+1\}$.

STEP 2: n:=n+1. Go to Step 1.

We prove the following theorem.

Theorem 21. *Let $\mathcal{P}$ be such that $P_\pm \geqslant 0.5$. Then Algorithm 7 returns a fair sequence.*

Proof. Define $d_n \equiv |f_{I_n}(P_\pm) - f_{\bar{I}_n}(P_\pm)|$ and write $d_n = |f_n - \bar{f}_n|$ By induction on n, we will prove that $d_n \leqslant P_\pm^{n-1}$ $\forall n$ if $P_\pm \geqslant 0.5$.
When $n = 1$, $d_n \leqslant P_\pm^{n-1} = 1$. Suppose that $d_k \leqslant P_\pm^{k-1}$ holds for $k = 1, \ldots, n-1$ and suppose that $f_{n-1} > \bar{f}_{n-1}$. Then $\bar{f}_n = \bar{f}_{n-1} + P_\pm^{n-1}$ and $f_n = f_{n-1}$. Either $\bar{f}_n \geqslant f_n$, or $f_n > \bar{f}_n$.
In the first case, clearly $d_n \leqslant P_\pm^{n-1}$.
In the second case, $d_n = d_{n-1} - P_\pm^{n-1} \leqslant P_\pm^{n-2} - P_\pm^{n-1} \leqslant P_\pm^{n-1}$ as $P_\pm \geqslant 0.5$. In both cases, $d_n \leqslant P_\pm^{n-1}$.

Thus, $\lim_{n\to\infty} f_{I_n}(P_\pm) - f_{\bar{I}_n}(P_\pm) = 0$, and as a consequence Algorithm 7 constructs a fair sequence. □

We point out that Algorithm 7 can be criticized from the point of view that it does not end. That property however, seems necessarily linked to its goal, namely finding a fair sequence, which by Theorem 19 cannot be achieved by a sequence of finite length. In any case, when one is prepared to specify $P_\pm$, it is certainly possible to precompute a very large number of entries, and use that as proxy for a fair sequence.

Of course, the resulting sequence depends on the chosen values for $P_+, P_-, P_\pm$. Thus, to decide which particular sequence to use in practice, one would have to decide on the values for these probabilities. When choosing $P_+ = \frac{1}{4}, P_- = \frac{3}{16}$ (see Apesteguia and Palacios-Huerta (2010), Vandebroek, McCann, and Vroom (2018)), this results in a sequence that starts with: ABBBABABBAA.... Finally, we mention that, even though the algorithm identifies a single fair sequence, the index set I is not necessarily a unique index set, i.e., multiple distinct fair sequences may exist.

8.4 BEST-OF-k SERIES

As described in Section 8.1, the first phase of a shootout consists of a best-of-k, where the winner is the team that scored the most penalties after Round k. In case of a tie, there is a Phase 2 which consists of a sudden death. Apesteguia and Palacios-Huerta (2010) model the existence of an FMA in a best-of-k by assuming that the pressure of being behind affects the scoring probability negatively, see Section 8.4.1. In Section 8.4.2 we show how to compute the degree of unfairness in this model, and we apply this to the best-of-5 in Section 8.4.3, and to the best-of-k when $p = \frac{3}{4}, q = \frac{2}{3}$ in Section 8.4.4.

8.4.1 Modeling psychological pressure

The main feature of the model is to encompass the added pressure of shooting while lagging behind in a fitting and realistic way. Recall that we assume Team A and B are equally strong. Next, we introduce parameters $p, q \in (0, 1)$ as the probability of a team scoring. A team has probability p of scoring, if it is equal or ahead, and probability q if the team is trailing. We assume $p > q$.

If the score, at the beginning of the round, is equal, and Team A shoots first, Team B second, this results in the following possible outcomes with probabilities:

$$\begin{aligned}
P_+ &= \mathbb{P}(\{\text{A scores, B does not}\}) && = p(1-q),\\
P_- &= \mathbb{P}(\{\text{B scores, A does not}\}) && = (1-p)p,\\
P_\pm &= \mathbb{P}(\{\text{A and B score equally often}\}) && = pq + (1-p)^2.
\end{aligned}$$

Notice that the probability that Team B scores after Team A scored, is q. If Team A missed, however, the probability that Team B would score is equal to p. This makes sense as in the first case, the score would have been in favor of Team A, adding pressure to Team B to catch up. As we assume that $p > q$, we immediately see that $P_+ > P_-$, indicating the existence of (FMA) λ, where $\lambda := P_+ - P_-$.

If, at the start of a round, one team (say A) leads, the possible outcomes with probabilities are slightly different:

$$\begin{aligned}
Q_+ &= \mathbb{P}(\{\text{Team A extends lead with 1}\}) && = p(1-q),\\
Q_- &= \mathbb{P}(\{\text{Team B decreases lead with 1}\}) && = (1-p)q,\\
Q_\pm &= \mathbb{P}(\{\text{A and B score equally often}\}) && = pq + (1-p)(1-q).
\end{aligned}$$

The values of p, q are, of course, not known, but they can be estimated from real world results. In football, $p = \frac{3}{4}, q = \frac{2}{3}$ is a common pick in the literature ((Brams and Ismail, 2018)) and (Csató, 2021)).

8.4.2 Degree of unfairness

Theorem 21 tells us that it is possible to create a sudden death sequence that is fair. Hence, when the shootout is tied after k rounds, we can use Theorem 21 to construct a sequence such that both teams have an equal probability of winning. That leaves us with only k choices to be made, indicating who shoots first in the k rounds of the best-of-k. To model our choice of who shoots first, we introduce $\sigma_i \in \{-1, 1\}, i = 1, \ldots, k$, where $\sigma_i = 1$ indicates Team A shooting first in round i, $\sigma_i = -1$ indicates Team B shooting first. We will derive formulas for W_i ($\overline{W}_i$), indicating the probability of team A (B) winning the shootout given that there is a tie at the start of round i ($1 \leqslant i \leqslant k$). We also define and calculate $\Delta_i = W_i - \overline{W}_i$, the difference between the probability of winning for team A and B, $i = 1, \ldots, k$. The objective is to find values for the σ_i that minimizes $|\Delta_1|$ which is the difference in winning probabilities between Team A and Team B at the start of the shoot out. Lastly, we define the auxiliary term K_j as follows: given that one team is ahead by 1 at the start of the round, K_j is the probability that the score will be leveled for the first time after j rounds.

Suppose the score is tied at the beginning of Round i ($1 \leqslant i \leqslant k$). One of the following three events occurs:

Event 1 The teams draw this round, and Round $i+1$ will start with a tied score.

Event 2 One of the teams wins the Round, and somewhere between the beginning of Round $i+2$ and Round $k+1$, the score is leveled again.

Event 3 One of the teams wins the round and stays ahead for the rest of the shootout.

The only advantage a team can have by shooting first in a specific round, is when Event 3 happens in that round. Both Event 1 and 2 let the teams return to a tie at the beginning of a round later in the shootout, thus no lasting advantage was obtained by either of the teams.

We can see Team A's winning probability W_i as the sum of the probability of winning in each of these events. For each of these events, we list the probability of occurring and the winning probability for Team A.

Event 1 Clearly:

$$\mathbb{P}(\{\text{Event } 1\}) = P_\pm$$

The probability of Team A winning in this setting is thus given by:

$$\mathbb{P}(\{\text{Event } 1, A \text{ wins}\}) = P_\pm W_{i+1}$$

Event 2 The probability that a team wins Round i equals $1 - P_\pm$. The score then levels again after exactly j rounds with probability K_j, for $j = 1, \dots, k - i$. Thus:

$$\mathbb{P}(\{\text{Event } 2, A \text{ wins}\}) = (1 - P_\pm) \sum_{j=1}^{k-1} K_j W_{i+1+j}$$

Event 3 The only way for Team A to win in the case of Event 3, is to be the team that wins the round. This happens with probability:

$$\mathbb{P}(\{\text{Team } A \text{ wins round}\}) = \frac{1}{2} \left((1 + \sigma_i)P_+ + (1 - \sigma_i)P_-\right)$$

The probability that a leading team stays ahead, is given by $1 - \sum_{j=1}^{k-i} K_j$. Thus:

$$\mathbb{P}(\{\text{Event } 2, A \text{ wins}\}) =$$

$$\frac{1}{2} \left((1 + \sigma_i)P_+ + (1 - \sigma_i)P_-\right) \cdot \left(1 - \sum_{j=1}^{k-i} K_j\right)$$

Combining all cases leads to the following winning probability for Team A:

$$\begin{aligned} W_i &= P_\pm W_{i+1} + (1 - P_\pm) \sum_{j=1}^{k-i} K_j W_{i+j+1} \\ &\quad + \left(1 - \sum_{j=1}^{k-i} K_j\right) \frac{1}{2} \left((1 + \sigma_i)P_+ + (1 - \sigma_i)P_-\right). \end{aligned}$$

From this, we derive, with $\Delta_i = W_i - \overline{W}_i$:

$$\begin{aligned} \Delta_i &= P_\pm \Delta_{i+1} + (1 - P_\pm) \sum_{j=1}^{k-i} K_j \Delta_{i+j+1} + \left(1 - \sum_{j=1}^{k-i} K_j\right) \sigma_i (P_+ - P_-) \\ &= P_\pm \Delta_{i+1} + (1 - P_\pm) \sum_{j=1}^{k-i} K_j \Delta_{i+j+1} + \left(1 - \sum_{j=1}^{k-i} K_j\right) \sigma_i \lambda. \end{aligned}$$

Notice that we can write Δ_i as a linear combination of terms $\lambda\sigma_j$, with $j \geqslant i$. Thus $\Delta_i = \sum_{j=i}^{k} D_{i,j}\lambda\sigma_j$. Applying this iteratively, we find the following expression for Δ_i:

$$\begin{aligned}\Delta_i = &\sum_{j=i+1}^{k}\left(P_{\pm}D_{i+1,j} + (1-P_{\pm})\sum_{\ell=1}^{j-i-1} K_{\ell}D_{i+1+\ell,j}\right)\lambda\sigma_j \\ &+\left(1-\sum_{j=1}^{k-i}K_j\right)\lambda\sigma_i.\end{aligned}$$

For Δ_1, which represents the unfairness that we want to minimize, we arrive at:

$$\begin{aligned}\Delta_1 = &\sum_{j=2}^{k}\left(P_{\pm}D_{2,j} + (1-P_{\pm})\sum_{\ell=1}^{j-2} K_{\ell}D_{2+\ell,j}\right)\lambda\sigma_j \\ &+\left(1-\sum_{j=1}^{k-1}K_j\right)\lambda\sigma_1.\end{aligned}$$

8.4.3 Results for the best-of-5

We can apply this general setting to the popular case of $k = 5$. This results in:

$$\begin{aligned}
i=6:&\quad \Delta_6 = 0.\\
i=5:&\quad \Delta_5 = \lambda\sigma_5.\\
i=4:&\quad \Delta_4 = P_{\pm}\lambda\sigma_5 + (1-K_1)\,\lambda\sigma_4.\\
i=3:&\quad \Delta_3 = (P_{\pm}^2 + (1-P_{\pm})K_1)\lambda\sigma_5 +\\
&\qquad P_{\pm}(1-K_1)\lambda\sigma_4 + (1-K_1-K_2)\lambda\sigma_3.\\
i=2:&\quad \Delta_2 = (P_{\pm}D_{3,5} + (1-P_{\pm})K_1D_{4,5} + (1-P_{\pm})K_2)\lambda\sigma_5\\
&\qquad + (P_{\pm}D_{3,4} + (1-P_{\pm})K_1D_{4,4})\lambda\sigma_4\\
&\qquad + P_{\pm}D_{3,3}\lambda\sigma_3 + (1-\sum_{j=1}^{3}K_j)\lambda\sigma_2.\\
i=1:&\quad \Delta_1 = \begin{pmatrix} P_{\pm}D_{2,5} + (1-P_{\pm})K_1D_{3,5} + \\ (1-P_{\pm})K_2D_{4,5} + (1-P_{\pm})K_3 \end{pmatrix}\lambda\sigma_5\\
&\qquad + (P_{\pm}D_{2,4} + (1-P_{\pm})K_1D_{3,4} + (1-P_{\pm})K_2D_{4,4})\lambda\sigma_4\\
&\qquad + (P_{\pm}D_{2,3} + (1-P_{\pm})K_1D_{3,3})\lambda\sigma_3\\
&\qquad + P_{\pm}(1-\sum_{j=1}^{3}K_j)\lambda\sigma_2 + (1-\sum_{j=1}^{4}K_j)\lambda\sigma_1.
\end{aligned}$$

All the unknowns that remain, are the terms K_j. These terms represent the probability that the leading team gives away the lead after exactly j rounds. In our model, the probabilities for these events are given by:

$$K_1 = Q_-.$$
$$K_2 = Q_\pm Q_-.$$
$$K_3 = (Q_\pm^2 + Q_+Q_-)Q_-.$$
$$K_4 = (Q_\pm^2 + 2Q_+Q_-)Q_\pm Q_-.$$

The aim is to have an assignment that minimizes $|\Delta_1|$, the unfairness at the start of Round 1. We assume that Team A shoots first in Round 1, which gives $\sigma_1 = 1$. Given values for p, q, we compute which of 16 possible assignments of $\sigma_2, \sigma_3, \sigma_4, \sigma_5$, minimizes $|\Delta_1|$. The corresponding sequences are presented in the following Table 52, with equal colors for equal sequences. We choose a grid with $0.6 \leqslant q < p \leqslant 0.8$ to reflect realistic probabilities in football.

p/q	0,6	0,61	0,62	0,63	0,64	0,65	0,66	0,67	0,68	0,69	0,7	0,71	0,72	0,73	0,74	0,75	0,76	0,77	0,78	0,79	0,8
0,6	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,61	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,62	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,63	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,64	ABABB	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,65	ABABB	ABABB	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,66	ABBAB	ABABB	ABBBA	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,67	ABBAB	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0,68	ABBAB	ABBAB	ABBBA	ABABB	ABBBA	ABBBA	ABBBA	ABBBA	x	x	x	x	x	x	x	x	x	x	x	x	x
0,69	ABBAB	ABBAB	ABBBA	ABBBA	ABABB	ABBBA	ABBBA	ABBBA	AABBB	x	x	x	x	x	x	x	x	x	x	x	x
0,7	ABBAB	ABBAB	ABBAB	ABBBA	ABBBA	ABABB	ABABB	ABBBA	ABBBA	AABBB	x	x	x	x	x	x	x	x	x	x	x
0,71	ABBBA	ABBAB	ABBAB	ABBBA	ABBBA	ABABB	ABABB	ABABB	ABBBA	ABBBA	AABBB	x	x	x	x	x	x	x	x	x	x
0,72	ABBBA	ABBBA	ABBBA	ABBAB	ABBBA	ABBBA	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	x	x	x	x	x	x	x	x	x
0,73	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBBA	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	x	x	x	x	x	x	x	x
0,74	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBBA	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	x	x	x	x	x	x	x
0,75	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	x	x	x	x	x	x
0,76	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	AABBB	x	x	x	x	x
0,77	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	AABBB	x	x	x	x
0,78	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	AABBB	x	x	x
0,79	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	AABBB	AABBB	x	x
0,8	ABBBA	ABBBA	ABBBA	ABBBA	ABBBA	ABBAB	ABBAB	ABBAB	ABBAB	ABBAB	ABABB	ABABB	ABABB	ABABB	ABABB	ABABB	AABBB	AABBB	AABBB	AABBB	x

Table 52: Least unfair sequences for given p and q in best-of-5

It is interesting to see the variety of preferred sequences, even when fixing the value of p or q, or their difference. This can be explained as the underlying calculation of the unfairness in a best of k shootout, is a polynomial of degree 2k.

8.4.4 Results for best-of-k with fixed p, q

The expressions obtained in Section 8.4.3 to calculate the unfairness of a sequence of any length, given p, q, is used to calculate the fairness of shootouts of another length than Best-of-5. The fairest sequences for $p = \frac{3}{4}, q = \frac{2}{3}$ for shootouts of length $k = 2, 3, \ldots, 10$ are shown in Table 53.

We like to point out that the values reported in the column "measure of unfairness" are the differences in winning chances between both teams, assuming that in the event of a tie after k rounds, both teams have an equal chance of winning the subsequent sudden death. It is remarkable that comparing two fairest sequences of different length, the longer penalty sequence is not automatically less unfair, even if p, q are the same for both sequences.

k	Sequence	Measure of unfairness (10^{-3})
2	AB	16.93
3	ABB	7.62
4	ABBA	1.47
5	ABABB	0.21
6	AABBBB	0.24
7	ABBABAB	0.01
8	ABAABBBB	0.04
9	AABBBBBBA	0.03
10	ABABABBABB	0.002

Table 53: Least unfair sequence and their offset for various shootout lengths with $p = \frac{3}{4}, q = \frac{2}{3}$

8.5 CONCLUSION

Many sports use shootouts to identify a winner in an otherwise tied game. Popular examples include football, field hockey, ice hockey, tennis, rugby, and waterpolo. A shootout has rounds, where in each round two teams, in an alternating fashion, have the possibility to score a point. Such a shootout has two phases. Phase 1 is a best-of-k (in soccer $k = 5$, the tie-break in tennis has $k = 6$), and if the shootout is tied after Phase 1, it continues with Phase 2: a sudden death. It is widely accepted that shooting when behind impacts the chances of scoring a point compared to shooting when not behind. We consider the problem to specify a sequence that determines which team shoots first in each round of the shootout such that identical teams have equal chance of winning the shootout; such sequences are called fair.

Using a common way to model the discrepancy between scoring chances for both teams, we show that in a sudden death, repetitive sequences are not fair, for any choice of the parameters; we also show that the PTM-sequence is not fair for any choice of parameters. There is however, an algorithm that outputs a fair sequence.

For a shootout decided over best-of-k, we show that no fair sequence exists. Using the popular choice $p = \frac{3}{4}$, $q = \frac{2}{3}$ (reflecting the probabilities of scoring when not behind, and when behind, respectively), the least unfair sequence in a best-of-5 is AB/BA/AB/BA/BA. More generally, we show that the degree of unfairness depends on the length of the shootout: longer shootouts can be significantly fairer than shorter ones.

BIBLIOGRAPHY

Ahuja, Ravindra K, Thomas L Magnanti, and James B Orlin (1988). *Network flows*. Cambridge, Mass.: Alfred P. Sloan School of Management, Massachusetts . . .

Alarcón, Fernando, Guillermo Durán, Mario Guajardo, Jaime Miranda, Hugo Muñoz, Luis Ramírez, Mario Ramírez, Denis Sauré, Matías Siebert, Sebastián Souyris, et al. (2017). "Operations research transforms the scheduling of Chilean soccer leagues and South American world cup qualifiers." In: *Interfaces* 47.1, pp. 52–69.

Allouche, Jean-Paul and Jeffrey Shallit (1999). "The ubiquitous prouhet-thue-morse sequence." In: *Sequences and their applications*. Springer, pp. 1–16.

Anbarci, Nejat, Ching-Jen Sun, and M Utku Ünver (2015). "Designing fair tiebreak mechanisms: The case of FIFA penalty shootouts." In: *Available at SSRN 2558979*.

Anderson, Ian (1999). "Balancing carry-over effects in tournaments." In: *Chapman and Hall CRC Research Notes in Mathematics*, pp. 1–16.

Apesteguia, Jose and Ignacio Palacios-Huerta (2010). "Psychological pressure in competitive environments: Evidence from a randomized natural experiment." In: *American Economic Review* 100. Appendix at https://assets.aeaweb.org/asset-server/articles-attachments/aer/data/dec2010/20081092_app.pdf, pp. 2548–2564.

Arrondel, Luc, Richard Duhautois, and Jean-François Laslier (2019). "Decision under psychological pressure: The shooter's anxiety at the penalty kick." In: *Journal of Economic Psychology* 70, pp. 22–35.

Aziz, Haris, Serge Gaspers, Simon Mackenzie, Nicholas Mattei, Paul Stursberg, and Toby Walsh (2014). "Fixing a Balanced Knockout Tournament." In: *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, pp. 552–558.

Bonomo, Flavia, Andrés Cardemil, Guillermo Durán, Javier Marenco, and Daniela Sabán (2012). "An application of the traveling tournament problem: The Argentine volleyball league." In: *Interfaces* 42.3, pp. 245–259.

Brams, Steven J and Mehmet S Ismail (2018). "Making the rules of sports fairer." In: *SIAM Review* 60.1, pp. 181–202.

Brams, Steven J and Alan D Taylor (1999). *The win-win solution - guaranteeing fair shares to everybody*. New York: WW Norton & Company.

Briskorn, Dirk (2008). "Feasibility of home–away-pattern sets for round robin tournaments." In: *Operations Research Letters* 36.3, pp. 283–284.

Burrows, Wayne and Christopher Tuffley (2015). "Maximising common fixtures in a round robin tournament with two divisions." In: *Australasian Journal of Combinatorics* 63, pp. 153–169.

Chavez, Anastasia and Christopher O'Neill (2022). "The fundamental theorem of finite fields: a proof from first principles." In: *The American Mathematical Monthly* 129.3, pp. 268–275.

Cocchi, Guido, Alessandro Galligari, Federica Picca Nicolino, Veronica Piccialli, Fabio Schoen, and Marco Sciandrone (2018). "Scheduling the Italian national volleyball tournament." In: *Interfaces* 48.3, pp. 271–284.

Cohen-Zada, Danny, Alex Krumer, and Offer Moshe Shapir (2018). "Testing the effect of serve order in tennis tiebreak." In: *Journal of Economic Behavior & Organization* 146, pp. 106–115.

Considine, John and Liam Gallagher (2018). "Competitive balance in a quasi-double knockout tournament." In: *Applied Economics* 50, pp. 2048–2055.

Csató, László (2021). "A comparison of penalty shootout designs in soccer." In: *4OR* 19.2, pp. 183–198.

Csató, László and Dóra Gréta Petróczy (2022). "Fairness in penalty shootouts: Is it worth using dynamic sequences?" In: *Journal of Sports Sciences*, pp. 1–7.

Davari, Morteza, Dries Goossens, Jeroen Beliën, Roel Lambers, and Frits Spieksma (2020). "The multi-league sports scheduling problem, or how to schedule thousands of matches." In: *Operations Research Letters* 48.2, pp. 180–187.

De Werra, Dominique (1981). "Scheduling in sports." In: *Studies on Graphs and Discrete Programming* 11, pp. 381–395.

Dotú, Iván and Pascal Van Hentenryck (2005). "Scheduling social golfers locally." In: *International Conference on Integration of Artificial Intelligence (AI) and Operations Research (OR) Techniques in Constraint Programming*. Springer, pp. 155–167.

Drexl, Andreas and Sigrid Knust (2007). "Sports league scheduling: Graph-and resource-based models." In: *Omega* 35.5, pp. 465–471.

Durán, Guillermo, Santiago Durán, Javier Marenco, Federico Mascialino, and Pablo A Rey (2019). "Scheduling Argentina's professional basketball leagues: A variation on the travelling tournament problem." In: *European Journal of Operational Research* 275.3, pp. 1126–1138.

Durán, Guillermo, Mario Guajardo, and Denis Sauré (2017). "Scheduling the South American Qualifiers to the 2018 FIFA World Cup by integer programming." In: *European Journal of Operational Research* 262.3, pp. 1109–1115.

Easton, Kelly, George Nemhauser, and Michael Trick (2002). "Solving the travelling tournament problem: A combined integer programming and constraint programming approach." In: *International conference on the practice and theory of automated timetabling*. Springer, pp. 100–109.

Even, Shimon, Alon Itai, and Adi Shamir (1975). "On the complexity of time table and multi-commodity flow problems." In: *16th annual symposium on foundations of computer science (sfcs 1975)*. IEEE, pp. 184–193.

Fayers, Matthew (2005). "Multiple-elimination knockout tournaments with the fixed-win property." In: *Discrete Mathematics* 290.1, pp. 89–97.

FIFA (2020). *FIFA Rules for Football*. https://img.fifa.com/image/upload/khhloe2xoigyna8juxw3.pdf. Accessed: September 1, 2020.

FiveThirtyEight (n.d.). *Who wins a very boring basketball game*. https://fivethirtyeight.com/features/who-wins-a-very-boring-basketball-game/.

Goerigk, Marc and Stephan Westphal (2016). "A combined local search and integer programming approach to the traveling tournament problem." In: *Annals of Operations Research* 239.1, pp. 343–354.

Goossens, Dries and Frits Spieksma (2009). "Scheduling the Belgian soccer league." In: *Interfaces* 39.2, pp. 109–118.

— (2011). "Breaks, cuts, and patterns." In: *Operations Research Letters* 39.6, pp. 428–432.

— (2012). "Soccer schedules in Europe: an overview." In: *Journal of Scheduling* 15.5, pp. 641–651.

Grabau, Mark (2012). "Softball scheduling as easy as 1-2-3 (strikes you're out)." In: *Interfaces* 42.3, pp. 310–319.

Groh, Christian, Benny Moldovanu, Aner Sela, and Uwe Sunde (2012). "Optimal seedings in elimination tournaments." In: *Economic Theory* 49.1, pp. 59–80.

Guedes, Allison CB and Celso Ribeiro (2011). "A heuristic for minimizing weighted carry-over effects in round robin tournaments." In: *Journal of Scheduling* 14.6, pp. 655–667.

Haigh, John (2009). "Uses and limitations of mathematics in sport." In: *IMA Journal of Management Mathematics* 20.2, pp. 97–108.

Holyer, Ian (1981). "The NP-completeness of edge-coloring." In: *SIAM Journal on Computing* 10, pp. 718–720.

Horbach, Andrei (2010). "A combinatorial property of the maximum round robin tournament problem." In: *Operations Research Letters* 38.2, pp. 121–122.

Huyghe, Thomas, Aaron T Scanlan, Vincent J Dalbo, and Julio Calleja-González (2018). "The negative influence of air travel on health and performance in the National Basketball Association: A narrative review." In: *Sports* 6.3, p. 89.

ITF (2019). *ITF Rules for Tennis*. https://www.itftennis.com/media/2510/2020-rules-of-tennis-english.pdf. Accessed: December 1, 2019.

Jordet, Geir, Esther Hartman, Chris Visscher, and Koen APM Lemmink (2007). "Kicks from the penalty mark in soccer: The roles of stress, skill, and fatigue for kick outcomes." In: *Journal of Sports Sciences* 25.2, pp. 121–129.

Keedwell, Anthony Donald (2000). "Designing Tournaments with the aid of Latin Squares: a presentation of old and new results." In: *Utilitas Mathematica* 58, pp. 65–85.

Kendall, Graham (2008). "Scheduling English football fixtures over holiday periods." In: *Journal of the Operational Research Society* 59.6, pp. 743–755.

Kendall, Graham, Sigrid Knust, Celso Ribeiro, and Sebastián Urrutia (2010). "Scheduling in sports: An annotated bibliography." In: *Computers & Operations Research* 37.1, pp. 1–19.

Kendall, Graham and Liam JA Lenten (2017). "When sports rules go awry." In: *European Journal of Operational Research* 257.2, pp. 377–394.

Kidd, MP (2010). "A tabu-search for minimising the carry-over e ects value of a round-robin tournament." In: *ORiON* 26.2.

Kirkman, TP (1851). "Solution to query VI." In: *Lady's and Gentleman's Diary* 48.

KNLTB (2019). www.knltb.nl/tennissers/competitie/voorjaarscompetitie/eredivisie/eredivisie-heren/. Accessed at Sept 27, 2019.

Knust, Sigrid and Daniel Lücking (2009). "Minimizing costs in round robin tournaments with place constraints." In: *Computers & Operations research* 36.11, pp. 2937–2943.

Kocher, Martin G, Marc V Lenz, and Matthias Sutter (2012). "Psychological pressure in competitive environments: New evidence from randomized natural experiments." In: *Management Science* 58.8, pp. 1585–1591.

Kolev, Gueorgui I, Gonçalo Pina, and Federico Todeschini (2015). "Decision making and underperformance in competitive environments: Evidence from the national hockey league." In: *Kyklos* 68.1, pp. 65–80.

Lambers, Roel, Dries Goossens, and Frits Spieksma (2022). "The flexibility of home away pattern sets." In: *Journal of Scheduling*, pp. 1–11.

Lambers, Roel, Jesper Nederlof, and Frits Spieksma (2020). *How the schedule in the TATA Steel Chess Championship forced Carlsen to help Caruana win*. www.networkpages.nl/how-the-schedule-in-the-tata-steel-chess-championship-forced-carlsen-to-help-caruana-win/. Accessed at Aug 26, 2021. *Referenced as LNS, 2020.*

Lambers, Roel, Rudi Pendavingh, and Frits Spieksma (2022). "Perfectly balanced serial knock-outs." In.

Lambers, Roel, Laurent Rothuizen, and Frits Spieksma (2021). "The Traveling Social Golfer Problem: The Case of the Volleyball Nations League." In: *International Conference on Integration of Constraint Programming, Artificial Intelligence, and Operations Research*. Springer, pp. 149–162.

Lambers, Roel and Frits Spieksma (2021). "A mathematical analysis of fairness in shootouts." In: *IMA Journal of Management Mathematics* 32.4, pp. 411–424.

Lambers, Roel, Mehmet Akif Yıldız, Jop Briët, Viresh Patel, and Frits Spieksma (2022). "Maximum orthogonal schedules."

Lambrechts, Erik, Annette Ficker, Dries Goossens, and Frits Spieksma (2018). "Round-robin tournaments generated by the circle method have maximum carry-over." In: *Mathematical Programming* 172.1, pp. 277–302.

Lester, Martin Mariusz (2021). "Scheduling reach mahjong tournaments using pseudoboolean constraints." In: *International Conference on Theory and Applications of Satisfiability Testing*. Springer, pp. 349–358.

Leven, Daniel and Zvi Galil (1983). "NP completeness of finding the chromatic index of regular graphs." In: *Journal of Algorithms* 4.1, pp. 35–44.

Liu, Ke, Sven Löffler, and Petra Hofstedt (2019). "Solving the Social Golfers Problems by Constraint Programming in Sequential and Parallel." In: *ICAART (2)*, pp. 29–39.

Lo, Michele, Robert J Aughey, Andrew M Stewart, Nicholas Gill, and Brent McDonald (2021). "The road goes ever on and on-a sociophysiological analysis of travel-related issues in Super Rugby." In: *Journal of Sports Sciences* 39.3, pp. 289–295.

Lovász, László and Michael D Plummer (2009). *Matching theory*. Vol. 367. American Mathematical Soc.

Lucas, Édouard (1883). *Récréations mathématiques*. Vol. 2. Gauthier-Villars et fils.

Miller, Alice, Matthew Barr, William Kavanagh, Ivaylo Valkov, and Helen C Purchase (2020). "Breakout group allocation schedules and the social golfer problem with adjacent group sizes." In: *Symmetry* 13.1, p. 13.

Miyashiro, Ryuhei, Hideya Iwasaki, and Tomomi Matsui (2002). "Characterizing feasible pattern sets with a minimum number of breaks." In: *International Conference on the Practice and Theory of Automated Timetabling*. Springer, pp. 78–99.

Miyashiro, Ryuhei and Tomomi Matsui (2006). "Minimizing the carry-over effects value in a round-robin tournament." In: *Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling*. PATAT, pp. 460–463.

Nemhauser, George and Michael Trick (1998). “Scheduling a major college basketball conference.” In: *Operations research* 46.1, pp. 1–8.

Palacios-Huerta, Ignacio (2012). “Tournaments, fairness and the Prouhet-Thue-Morse sequence.” In: *Economic inquiry* 50.3, pp. 848–849.

— (2014). *Beautiful Game Theory: How Soccer Can Help Economics*. Princeton, NJ and Oxford: Princeton University Press.

Pollard, Richard (2008). “Home advantage in football: A current review of an unsolved puzzle.” In: *The open sports sciences journal* 1.1.

Raknes, M. and K. Holm Pettersen (2018). “Optimizing Sports Scheduling: Mathematical and Constraint Programming to Minimize Traveled Distance with Benchmark From The Norwegian Professional Volleyball League.”

Rasmussen, Rasmus V and Michael Trick (2008). “Round robin scheduling – a survey.” In: *European Journal of Operational Research* 188.3, pp. 617–636.

Recalde, Diego, Ramiro Torres, and Polo Vaca (2013). “Scheduling the professional Ecuadorian football league by integer programming.” In: *Computers & Operations Research* 40.10, pp. 2478–2484.

Rudi, Nils, Marcelo Olivares, and Aditya Shetty (2020). “Ordering sequential competitions to reduce order relevance: Soccer penalty shootouts.” In: *PloS one* 15.12, e0243786.

Russel, K.G. (1980). “Balancing carry-over effects in round robin tournaments.” In: *Biometrika* 67, pp. 127–131.

Russell, Robert A and Janny MY Leung (1994). “Devising a cost effective schedule for a baseball league.” In: *Operations Research* 42.4, pp. 614–625.

Samuels, Charles H (2012). “Jet lag and travel fatigue: a comprehensive management plan for sport medicine physicians and high-performance support teams.” In: *Clinical Journal of Sport Medicine* 22.3, pp. 268–273.

Schönberger, Jörn (2015). “Scheduling of Sport League Systems with Inter-League Constraints.” In: *Proceedings of the 5th International Conference on Mathematics in Sport*, pp. 171–176.

Schreuder, Jan AM (1992). “Combinatorial aspects of construction of competition Dutch professional football leagues.” In: *Discrete Applied Mathematics* 35.3, pp. 301–312.

Schwartz, Barry and Stephen F Barsky (1977). “The home advantage.” In: *Social forces* 55.3, pp. 641–661.

Siemann, M.R. (2020). *A polyhedral study of the Travelling Tournament Problem*. http://essay.utwente.nl/80918/1/Siemann_MA_EEMCS.pdf. Thesis at University of Twente.

Song, Alex, Thomas Severini, and Ravi Allada (2017). “How jet lag impairs Major League Baseball performance.” In: *Proceedings of the National Academy of Sciences* 114.6, pp. 1407–1412.

Stevens, Christopher J, Heidi R Thornton, Peter M Fowler, Christopher Esh, and Lee Taylor (2018). “Long-haul northeast travel disrupts sleep and induces perceived fatigue in endurance athletes.” In: *Frontiers in Physiology* 9, p. 1826.

Toffolo, Túlio AM, Jan Christiaens, Frits Spieksma, and Greet Vanden Berghe (2019). “The sport teams grouping problem.” In: *Annals of Operations Research* 275.1, pp. 223–243.

Trick, Michael (2000). “A schedule-then-break approach to sports time tabling.” In: *Practice and Theory of Automated Timetabling III (PATAT 2000)*. Ed. by E. Burke and W. Erben. Vol. 2079. Konstanz, Germany: Springer-Verlag, pp. 242–253.

Tripke, A (1983). “Algebraische und Kombinatorische Structuren von Spielplänen mit Anwendung auf ausgewogen Spielpläne.” PhD thesis. Ruhr-Universität in Bochum.

Triska, Markus and Nysret Musliu (2012). “An effective greedy heuristic for the Social Golfer Problem.” In: *Annals of Operations Research* 194.1, pp. 413–425.

Van Bulck, David, Dries Goossens, Jörn Schönberger, and Mario Guajardo (2020). “RobinX: A three-field classification and unified data format for round-robin sports timetabling.” In: *European Journal of Operational Research* 280.2, pp. 568–580.

Vandebroek, Tom P., B. McCann, and G. Vroom (2018). “Modeling the Effects of Psychological Pressure on First-Mover Advantage in Competitive Interactions: The Case of Penalty Shoot-Outs.” In: *Journal of Sports Economics* 19, pp. 725–754.

Volkskrant (2021). *Aanvoerder Anne Buijs wijst jonkies de weg*. http://www.volkskrant.nl/sport/aanvoerder-anne-buijs-wijst-jonkies-de-weg-soms-zelfs-naar-het-strand-van-rimini~be4c406e/. In Dutch.

Vu, Thuc Duy (2010). *Knockout tournament design: a computational approach*. https://stacks.stanford.edu/file/druid:qk299yx6689/TV-thesis-final-augmented.pdf. Thesis at Stanford University, USA.

Warwick, Harvey and Thorsten Winterer (2005). “Solving the MOLR and Social Golfers Problems.” In: *Principles and Practice of Constraint Programming - CP 2005*. Ed. by Peter van Beek. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 286–300.

Wikipedia (2020). *Wikipediapage Penalty shootouts*. https://en.wikipedia.org/wiki/Penalty_shootout. Accessed: June 1 2020.

Winter, W Christopher, William R Hammond, Noah H Green, Zhiyong Zhang, and Donald L Bliwise (2009). “Measuring circadian advantage in Major League Baseball: a 10-year retrospective study.” In: *International Journal of Sports Physiology and Performance* 4.3, pp. 394–401.

Wright, Mike (2014). “OR analysis of sporting rules–A survey.” In: *European Journal of Operational Research* 232.1, pp. 1–8.

SUMMARY

The world of sports relies on competition, and every competition relies on participants, as well as fans, broadcasters, you name it. Each of these groups have different needs and demands regarding the competition schedule, so as an organizer, there is the challenge to satisfy everything that is desired. On the one hand, it is incredibly difficult to find schemes that match all the needs, but on the other hand, the amount of possible schedules to choose from quickly becomes massive - so finding the best schedule is a huge task.

In this thesis, several approaches to finding good/better/best schedules - in several formats (Round Robin, Knock-Out, etc.) - are discussed. The most popular format is the Round Robin competition, where every team/player plays every opponent a fixed number of times. In chapters 2,3 and 5, constructive methods are discussed to schedule these competitions, each catering to specific needs. Chapter 2 and 3 focus on having high flexibility of scheduling individual matches, without changing the Home/Away-pattern, and how to recognize patterns that have this flexibility. Chapter 5 discusses ways to schedule the competition such that two players don't meet the same two opponents in consecutive rounds.

In the other chapters, competition scheduling is looked at both in a constructive as a theoretical way. In chapter 4, the focus is on how to schedule multiple leagues such that the capacities of clubs with multiple teams are never violated. In chapter 6, a relatively new format is discussed, where rounds are complete knock-out tournaments among all players. How to fix the draws of the knock-out in such a way that each player has a priori the same opportunities.

Chapter 7 and 8 discuss the scheduling of an international competition, where travel distances are a crucial component, and the order in which penalties are taken. In both chapters, methods and algorithms are described to organize this in a fair way.

SAMENVATTING

Sport is niets zonder competitie, en competities zijn weer niets zonder deelnemers en fans. Aan sportbonden en lokale organisatoren de taak om de wensen en mogelijkheden van de sporters, toeschouwers, tv-zenders, etc, in acht te nemen tijdens het opstellen van een wedstrijdschema voor een toernooi. Wie speelt wanneer uit, hoe vaak op rij, tegen wie, in welke volgorde, alles kan worden meegenomen in de overwegingen. Het kan ontzettend moeilijk zijn om vervolgens een schema te vinden dat aan alle wensen voldoet, maar tegelijkertijd is het aantal schema's waar initieel uit gekozen kan worden dan ook gigantisch - dus waar begin je met zoeken, en hoe moet je überhaupt op zoek naar dat perfecte schema?

In dit proefschrift worden meerdere manieren besproken die, gegeven bepaalde eisen, kunnen helpen bij het genereren van goede/betere of zelfs optimale schema's, en dat bij meerdere competitievormen. De populairste competitievorm is misschien wel de Round Robin of hele/halve competitie, zoals deze in het Nederlands heet. In hoofdstuk 2,3 en 5 worden schema's hiervoor geproduceerd die verscheidene goede eigenschappen hebben. Denk dan bijvoorbeeld aan een hoge mate van flexibiliteit voor het inroosteren van individuele wedstrijden, waarbij het uit/thuis-patroon vaststaat, en hoe je deze patronen kan herkennen. Ook voor het gelijkmatig verspreiden van tegenstanders over het seizoen - in die zin dat twee verschillende spelers de overige tegenstanders in andere volgorde ontmoeten - worden constructies gegeven.

In de overige hoofdstukken wordt ook op een constructieve als theoretische manier naar het produceren van schema's gekeken. In hoofdstuk 4 wordt besproken hoe wedstrijden van ploegen over verschillende competities kunnen worden ingedeeld, zodat de capaciteit van de clubs waar de ploegen bij horen niet wordt overschreden. In hoofdstuk 6 wordt gekeken naar een competitie waarbij de speelrondes bestaan uit een knock-out toernooi over alle spelers - hoe moet je de loting van die individuele knock-out toernooien regelen, zodat elk speler bij aanvang gelijke kansen heeft?

Hoofdstuk 7 en 8 kijken vervolgens naar het inroosteren van een internationale competitie waarbij reisafstanden cruciaal zijn, en de volgorde waarin penalties genomen worden aan het einde van een voetbalwedstrijd. Voor beide methoden worden algoritmen en methodes gegeven om dit op een eerlijkere manier te organiseren.

COURSE OF LIFE

January 1993	Born in Groningen
2004 - 2010	Gymnasium, STEDELIJK GYMNASIUM NIJMEGEN
2010 - 2014	Bachelor Wiskunde & Natuurkunde, UNIVERSITEIT UTRECHT
2014 - 2016	Master Mathematical Sciences, UNIVERSITEIT UTRECHT Master Thesis: *Logical laws on caterpillars*.
2018 - 2022	PhD-student, EINDHOVEN UNIVERSITY OF TECHNOLOGY Thesis: *Fairness and Flexibility in Sport Scheduling*. Promotor: *Frits Spieksma*, Copromotor: *Rudi Pendavingh*.