SCHEDULING THE TURKISH SUPER FOOTBALL LEAGUE

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF

ABDULLAH GUL UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Hasan Bayrak

June 2019

SCIENTIFIC ETHICS COMPLIANCE

I hereby declare that all information in this document has been obtained in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Hasan BAYRAK

REGULATORY COMPLIANCE

M.Sc. thesis titled **"SCHEDULING THE TURKISH SUPER FOOTBALL LEAGUE"** has been prepared in accordance with the Thesis Writing Guidelines of the Abdullah Gül University, Graduate School of Engineering & Science.

Prepared By

Hasan BAYRAK

Advisor

Asst. Prof. Dr. Muhammed SÜTÇÜ

Head of the Industrial Engineering Graduate Program

Prof. Dr. İbrahim AKGÜN

ACCEPTANCE AND APPROVAL

M.Sc. thesis titled **"SCHEDULING THE TURKISH SUPER FOOTBALL LEAGUE"** and prepared by Hasan BAYRAK has been accepted by the jury in the Industrial Engineering Graduate Program at Abdullah Gül University, Graduate School of Engineering & Science.

… /07/ 2019

JURY:

Asst. Prof. Dr. Muhammed SÜTÇÜ :……………………………….

Asst. Prof. Selçuk GÖREN :……………………………….

Prof. Dr. Emel KIZILKAYA AYDOĞAN :……………………………….

APPROVAL:

The acceptance of this M.Sc. thesis has been approved by the decision of the Abdullah Gül University, Graduate School of Engineering & Science, Executive Board dated .. /.. / .. and numbered …………..……. .

……….. /……….. / ………..

Prof. Dr. İrfan ALAN

ABSTRACT

SCHEDULING THE TURKISH SUPER FOOTBALL LEAGUE

Hasan BAYRAK

MSc. in Industrial Engineering

Supervisor: Asst. Prof. Dr. Muhammed SÜTÇÜ

June 2019

Sports entertainment was usually casual in the past, however, today's sport has to be organized, marketed, and administered as a business because of economic size. Teams are making big investments to transfer new players. Broadcast rights of some competitions are being sold for hundreds of millions of dollars. The schedule is important for the organization's safety, game attendance, public interest, broadcasters, and advertisers and fairness. However, creating a fair and appropriate schedule is not easy because of the different needs of the stakeholders and the needs often conflict with each other. When the number of teams in the tournament increases, it may not be possible to reach the optimal solution with traditional methods. In this thesis, we have found a number of pattern elimination methods that we call **"ladder patterns"**. Moreover, with the help of ladder patterns, we developed a 5-Step approach that can able to solve a very complicated problem easily and quickly. At the result, a better schedule created for the Turkish Super Football League.

Keywords: sports scheduling, round-robin tournament, break minimization, pattern elimination

ÖZET

TÜRKİYE SÜPER FUTBOL LİGİNİN ÇİZELGELENMESİ

Hasan BAYRAK

Endüstri Mühendisliği Bölümü Yüksek Lisans

Tez Yöneticisi: Dr. Muhammed SÜTÇÜ

Haziran 2019

Günümüzde spor etkinliklerinin ekonomik büyüklükleri nedeniyle bir işletme olarak organize edilmesi, pazarlanması ve yönetilmesi gerekiyor. Takımlar yeni oyuncuları transfer etmek için inanılmaz yatırımlar yapıyor. Populer liglerin yayın hakları yüz milyonlarca dolara satılabiliyor. İyi bir turnuva çizelgesi; adalet, organizasyonun güvenliği, maçların seyirci çekmesi, yayıncıların ve reklamcıların yatırımlarının karşılığını alabilmesi açısından oldukça önemlidir. Ancak, adil ve uygun bir program oluşturma niyeti, paydaşların farklı ihtiyaçları ve bu ihtiyaçların sürekli birbiriyle çatışıyor olması nedeniyle kolay değildir. Turnuvadaki takım sayısı arttığında, geleneksel yöntemlerle en uygun çözüme ulaşmak mümkün olmayabilir. Bu tezde, merdiven deseni dediğimiz bir dizi desen eleme yöntemi bulduk. Dahası, merdiven deseni metodu yardımıyla, çok karmaşık bir problemi kolay ve hızlı bir şekilde çözebilecek 5 adımlı bir yaklaşım geliştirdik. Sonuç olarak Türkiye Süper Futbol Ligi için daha iyi bir çizelge oluşturduk.

Anahtar kelimeler: spor çizelgeleme, yuvarlak-robin turnuvası, kırılma minimizasyonu, desen eleme

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Sports are an important part of human life. While most of us do not have the time or opportunity to do regular sports, almost everyone follows at least one sports organization and supports a team as a fan. For thousands of years, people have been greatly interested in sports organizations. During the last few decades, the interest in sports organizations has increased exponentially with the discovery and widespread use of radio, television and the internet. Therefore, the number of people who are interested in at least one of the sports organizations has reached billions. Table 1.1 shows the most popular sports in the world. Even though the past sports entertainment was usually casual, today's sport has to be organized, marketed, and administered as a business because of economic size. Teams make big investments to transfer new players. Broadcast rights of some competitions are being sold for hundreds of millions of dollars. We contribute to the growth of the sports economy by following the matches of the team we support in the stadium or by regularly monitoring their status on television, radio, newspaper and social media. Although the economic size of the sports industry varies from source to source, it is a common opinion that it is over $ 500 billion. Unsurprisingly, football, which is the most popular sports in the world, has 43% of the sports events market alone [1]. Table 1.2 shows market share of sports events in the world.

Undoubtedly, there are challenging optimization problems in professional sports leagues that have millions of fans and significant investments in players, broadcast rights, merchandising, and advertising. Even in non-professional leagues, where there is relatively less investment, coordination is essential for the success of the organization. Stakeholders expect that the schedules belonging to such large organizations are well prepared in terms of fairness. Beside fairness, the schedule is also important for the organization's safety, game attendance, public interest, broadcasters, and advertisers. Obviously, a good schedule not only meet the needs of the team, but also increase the profit in sports organizations. Moreover, schedules have a critical impact on the results of the competition itself and the success of the team we support may cause significant changes in our social life. Since the schedule is a crucial element for effective competition, organizers must consider all those aspects.

Rank	Sport	Estimated Fans	Most Popular regions
1.	Football	4 Billion	Europe, Africa, America.
2.	Cricket	2.5 Billion	Asia, Australia, UK.
3.	Field Hockey	2 Billion	Europe, Africa, Asia, Australia.
4.	Tennis	1 Billion	Europe, Asia, America.
5.	Volleyball	900 Million	Europe, Australia, Asia.
6.	Table Tennis	875 Million	Europe, Asia, America.
7.	Basketball	825 Million	America.
8.	Baseball	500 Million	America, Japan.
9.	Rugby	475 Million	UK, Commonwealth.
9.	Golf	450 Million	America, Canada.

Table 1.1 World's Most Popular Sports

Rank	Sport	Market Share Percentage
1.	Football	43
2.	American Football	13
3.	Baseball	12
4.	Formula 1	7
5.	Basketball	6
6.	Hockey	4
6.	Tennis	4
8.	Golf	3
9.	Tennis	2
10.	Cricket	2

Table 1.2 Market share of sports events in the world

Because of every sports league or organization have different objectives and constraints, sports scheduling area provides many practical applications. To create a fair and appropriate schedule is challenging due to the constraints by stakeholders and constraints often conflict with each other. For example, teams that share the same stadium may want to play their matches at home during the slots that the match will attract more spectators. However, it is not possible to satisfy both teams here, because of their conflicting needs.

Most common objectives in sports scheduling are break minimization and distance minimization, but there are few practical applications about minimizing the carry-over effect. Essentially, minimizing the carry-over effect matters when a team plays consecutive matches with very strong teams. This may cause consecutive losses for weaker teams. Moreover, consecutive losses may result in an undesirable situation for any team and their supporters due to loss of motivation and disappointment.

In the traveling tournament, teams play a couple of consecutive away matches without returning home venue. However, when this is not the case, teams usually return to the home venue after each away matches. Therefore, alternative patterns without many consecutive away match preferred because of regular income and for the supporters who want to watch their team's matches in the home venue regularly. Studies have shown that the home teams are more successful in winning matches because of headings of the crowd. Table 1.3 shows the home winning percentages for the popular team sports [2]. At the start of the league, teams with consecutive home matches might do relatively better than the teams with consecutive away matches. Clearly, the strength of the teams according to the position throughout the tournament can be expressed more smoothly without the patterns that contain many breaks.

Demand for alternative patterns with the minimum number of breaks has encouraged researchers to develop sophisticated solution approaches facing various constraints for practice. Constructive methods are widely used heuristics that create a complete solution systematically by sequentially adding components to the partial solution. However, this method cannot handle constraints. Therefore, in the last decades, instead of constructive methods, the researches concentrate on the decomposition methods that are able to deal with all the necessary constraints [3]. The main objective of the decomposition methods is to decompose the problem into subproblems which relatively easy to solve.

Sport	**Number of studies**	**Number of matches**	**Home winning percentage (draw games excluded)**
Baseball	1	133560	54.3
American football	5	2592	57.3
Ice Hockey	5	5312	61.2
Basketball	9	13686	64.4
Football	3	40492	68.3

Table 1.3 The home winning percentages for the popular team sports

Minimizing the travel distance preferred over minimizing the breaks when the tournament area is too big for teams to return home venue after each away match. When a team plays an away match, instead of returning home, they play another away match with one of the nearby team. This result in a considerable saving in traveling costs and time. After the 1970s oil crises, the cost of traveling increased greatly. Since this extra cost put especially amateur teams and tournaments in a difficult position, researcher started to focus on minimizing travel distance or in other word minimizing travel costs. The first published study on distance minimization, written by R.T. Campbell and D.S. Chen, was basketball schedule in 1976 [4].

Sports scheduling embrace a wide research area with different kind of problems and applications. Sport scheduling problems are hard combinatorial optimization problems' they yield a great opportunity for developing and testing different solution method. Combinatorial optimization problems' feasible region is generally too large to search comprehensively just with pure computation power. Therefore, in the literature, combinatorial optimization problems often solved by integer programming (IP), constraint programming (CP) and heuristic approaches. Elf et al. show that the break minimization problem, in general, is NP-hard [5]. In order to solve such problems, algorithms must be designed to meet the needs of the application. Many different approaches are widely discussed in the literature, but the number of constraints increased faster than the algorithms' performances. Thus, sports scheduling problems still remains very hard to solve. Whichever method we choose, algorithms usually need serious computational time to reach the optimal solution. Over the past 40 years, a massive amount of research has been done in order to find effective solution methods in sports scheduling. Although methods like Benders decomposition, CP, hybrid methods combining IP and CP, and metaheuristic methods become more and more effective, creating a schedule for twelve teams is still very hard. Especially when the total number of the team is large, finding the optimal solution is not always possible nor easy.

As the total number of teams increases, it becomes difficult to find an optimal solution. In such case, heuristic algorithms are often preferred over exact algorithm. Heuristic algorithms give us best possible solutions in short times but they do not always find the optimum solution.

The rest of the thesis is organized as fallows;

In chapter 2, we talk about sports scheduling terminology and we review the literature in details. Although sports scheduling literature mainly are divided into two categories: break minimization and distance minimization. In each category, a chronological review of existing papers is presented. In Chapter 3, we purpose a general mathematical model for a single round robin tournament with many constraints. In Chapter 4, propose a hybrid algorithm, which combines a heuristic approach and integer programming. The approach successfully solved the model within the proven lower bound for the break minimization. Chapter 5 is based on the model we developed for scheduling the Turkish Super League. According to data provided by FIFA, Turkey is in the top 20 countries with the highest number of football audiences around the world. Deloitte Annual Review of Football Finance (2017) announced that Turkish Super League is the 6th biggest revenue generating football league with the €734 million annually in Europe [6].

rank	**League**	**County**	**Estimated revenue**
1.	Premier League	England	€ 4888 Million
2.	Bundesliga	Germany	€ 3375 Million
3.	La Liga	Spain	€ 2526 Million
4.	Serie A	Italy	€ 2004 Million
5.	Ligue 1	France	€ 1485 Million
6.	Campeonato Brasileiro	Brazil	€ 958 Million
7.	Major League Soccer	USA	€ 851 Million
8.	Super League	Turkey	€ 734 Million
9.	EFL Championship	England	€ 723 Million
10.	Russian Premier League	Russia	€ 701 Million

Table 1.4 Football league by revenue in the World (2017)

Turkish Super League's TV right has increased significantly in recent years and total value reached $ 489 million in 2017, thus the schedule in Turkish Super League becomes more crucial. Table 1.4 shows the ranking of the football leagues by revenue in the world. Computational results are given in Chapter 6. We compare the results with the current schedule and will present our conclusion. We created a more fair schedule for the Turkish Super League which respond to the needs of all stakeholders.

Chapter 2

Literature Review

The growth of the interest in the sport has forced organizers to seek better schedule for the sports organizations, whether the organization is professional or not. After 90's, sports scheduling has become a popular research area within computer science and operations research. Although it may seem easy to create a schedule for a tournament, it can also transform into a very difficult combinatorial optimization problem with the special needs of the participants. In fact, a tournament with more than 20 teams is considered large-scale scheduling problems and heuristic approach often required to find a decent schedule [3].

Sport scheduling problems are hard combinatorial optimization problems, providing a great opportunity for developing and testing different solution methods. With the years of development of solution methods, today, it is possible to find optimal or near-optimal solutions to difficult scheduling problems. In the literature, we can find methods ranging from pure combinatorial approaches to every aspect of discrete optimization, including integer programming, constraint programming, metaheuristic approaches, and various combinations of them.

In this chapter, we talk about sports scheduling terminology and review the literature in detail. Broadly sports scheduling literature is mainly divided into two categories: break minimization and distance minimization, recently, minimization of carry-over effect is discussed in a few papers.

As sports scheduling becomes a wider research area, the number of published papers exceed hundreds. Nevertheless, we mainly focus on papers about the round-robin tournaments. We summarize the main contributors and outlines for each objective chronologically.

In the literature, there is a term named ***balanced tournament***. In the balanced tournament scheduling, there are external factors that make some games different from others, and these external effects are desirable to affect all teams similarly. For example, if all teams participating in the tournament plays in the common venues, in order to schedule a fair tournament, the teams' matches must be equally distributed to all venues. The problem about balanced tournament scheduling is not taken into account in the literature review.

2.1 Sport Scheduling Terminology

In the sports scheduling literature, the terminology is not consistent and sometimes notions have more than one meaning. To avoid confusions, notions we defined in this section are used throughout the thesis.

In a ***round robin tournament***, all participants play a fixed number of matches with each other. Generally, almost all of the sports leagues or tournaments play a double round robin (DRR) tournament where all teams play against each other twice in a season, but single (SRR) or triple round robin (TRR or 3RR) tournaments also exist. Table 2.1.1 shows round robin tournament preferences for football leagues in Europe.

A ***tournament*** is a sports event where n number of teams play against each other according to a given timetable. A ***timetable*** shows the opponent of each team in each slot. Table 2.1.2 shows a timetable example for a tournament with eight teams. Each row of the timetable represents a team, while columns represent slots. It is called match when two teams play against each other. The matches must be allocated to time slots, which is usually week or day, in such a way that each team can only play one match in each slot. If n is even number, at least $(n-1)$ slots are required to schedule single round robin tournament. However, when n is an odd number, dummy team must be added to the league. In this case, at least n slots are required to schedule single round robin tournament. When a team is opponent of the dummy team, it is called a bye in that slot and match does not take place.

County	n	Format	Symmetry	Canonical
England	20	DRR	None	No
Germany	18	DRR	Mirror	No
Spain	20	DRR	Mirror	Yes
Italy	20	DRR	Mirror	No
France	20	DRR	French	No
Russia	16	DRR	French	Yes
Slovakia	12	3RR	Mirror	Yes
Turkey	18	DRR	Mirror	Yes
Switzerland	10	4RR	Mirror + Inverted	Yes
Portugal	16	DRR	Mirror	Yes

Table 2.1.1 Round robin tournament preferences for football leagues in Europe

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Team 1	*5*	*3*	*2*	*8*	*4*	*6*	*7*
Team 2	*4*	*6*	*1*	*7*	*3*	*5*	*8*
Team 3	*7*	*1*	*8*	*5*	*2*	*4*	*6*
Team 4	*2*	*8*	*7*	*6*	*1*	*3*	*5*
Team 5	*1*	*7*	*6*	*3*	*8*	*2*	*4*
Team 6	*8*	*2*	*5*	*4*	*7*	*1*	*3*
Team 7	*3*	*5*	*4*	*2*	*6*	*8*	*1*
Team 8	*6*	*4*	*3*	*1*	*5*	*7*	*2*

Table 2.1.2 Timetable for a tournament with eight teams

If the number of available time slots are equal to the lower bound $(n - 1)$, each team plays a single match in every slot, that schedule is called a ***compact schedule***, if the number of available time slots greater than lower bound, that schedule is called a ***relaxed schedule*** [17]. Table 2.1.3 shows a compact schedule example for tournament with eight teams.

Every team has a home stadium and when a team plays a match in its home stadium it is called the ***home match*** for that team; otherwise, it is called as an ***away match***. The sequence of home (H), away (A) and bye (B) games for a given team is called a ***pattern***. Bye does not occur in a compact schedule with an even number of teams and every team play in each slot once.

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Team 1	*1-5*		*1-2*		*1-4*		*1-7*
Team 2		*2-6*		*2-7*		*2-5*	
Team 3		*3-1*	*3-8*		*3-2*		*3-6*
Team 4	*4-2*			*4-6*		*4-3*	
Team 5		*5-7*		*5-3*	*5-8*		*5-4*
Team 6	*6-8*		*6-5*			*6-1*	
Team 7	*7-3*		*7-4*		*7-6*	*7-8*	
Team 8		*8-4*		*8-1*			*8-2*

Table 2.1.3 Compact schedule example for tournament with eight teams

While scheduling a tournament, it may be beneficial to put off the assignments of games until a schedule has been obtained. In such cases, instead of actual teams, ***placeholders*** represent the teams in the pattern set and in the timetable. In compact schedule home (H) and away (A) are often replaced with binary numbers 1 and 0, respectively. ***Pattern set*** is a set of n patterns and each of them is associated with one of the teams. Table 2.1.4 shows a pattern set for a tournament with eight teams. Timetable can only be created from a feasible pattern set. Otherwise, the pattern set is infeasible and new pattern set needs to be created to find a feasible schedule if any exist.

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7
Pattern 1	*1*	*0*	*1*	*0*	*1*	*0*	*1*
Pattern 2	*0*	*1*	*0*	*1*	*0*	*1*	*0*
Pattern 3	*0*	*1*	*1*	*0*	*1*	*0*	*1*
Pattern 4	*1*	*0*	*0*	*1*	*0*	*1*	*0*
Pattern 5	*0*	*1*	*0*	*1*	*1*	*0*	*1*
Pattern 6	*1*	*0*	*1*	*0*	*0*	*1*	*0*
Pattern 7	*1*	*0*	*1*	*0*	*1*	*1*	*0*
Pattern 8	*0*	*1*	*0*	*1*	*0*	*0*	*1*

Table 2.1.4 A pattern set example for a tournament with eight teams

Break appears when a team play more than one home or away matches in consecutive two slots. In Table 2.1.4, team 3 has a break at slot 3. A sequence of consecutive home and away games (breaks) is called ***homestand*** and ***trip***, respectively. ***Tour*** is an entire row of the schedule for the respective team [3]. When the two patterns are compared, if the first pattern has an opposite binary number of what second has in every slot, those two patterns are called ***complementary patterns***. Table 2.1.5 shows a pair of complementary patterns.

	Slot 1	**Slot 2**	**Slot 3**	**Slot 4**	**Slot 5**	**Slot 6**	**Slot 7**
Pattern 1	1	0	1	0	1	0	1
Pattern 2	0	1	0	1	0	1	0

Table 2.1.5 Two complementary patterns

The ***schedule*** is a combination of the pattern set and the timetable. Table 2.1.6 shows an example of the mirrored DRR tournament schedule for eight teams. In literature, DRR tournament is often scheduling as SRR tournament and for the second half of DRR, mirror of SRR is used. When first and the second half are identical, the schedule can be mirrored, in this case home and away games are exchanged. In the ***French system***, the first slot of the first round, and the last slot of the second round have identical matches. Beside of that, the Professional Swiss League consists of four rounds (4RR); the first two rounds are mirrored, and the last two rounds are inverted. Table 2.1.7 shows the most popular matching system for the second half of DRR.

	S 1	**S 2**	**S 3**	**S 4**	**S 5**	**S 6**	**S 7**	**S 8**	**S 9**	**S 10**	**S 11**	**S 12**	**S 13**	**S 14**
Team 1	1-5		1-2		1-4		1-7		1-3		1-8		1-6	
Team 2		2-6		2-7		2-5		2-4		2-1		2-3		2-8
Team 3		3-1	3-8		3-2		3-6	3-7			3-5		3-4	
Team 4	4-2			4-6		4-3			4-8	4-7		4-1		4-5
Team 5		5-7		5-3	5-8		5-4	5-1		5-6			5-2	
Team 6	6-8		6-5			6-1			6-2		6-4	6-7		6-3
Team 7	7-3		7-4		7-6	7-8			7-5		7-2			7-1
Team 8		8-4		8-1			8-2	8-6		8-3		8-5	8-7	

Table 2.1.6 Example of the mirrored DRR schedule for tournament with eight teams

	The first half of DRR						The second half of DRR					
Mirroring	1	2	3	…	n-2	n-1	1	2	3	…	n-2	n-1
French System	1	2	3	…	n-2	n-1	2	3	4	…	n-1	1
Inverted System	1	2	3	…	n-2	n-1	n-1	n-2	n-3	…	2	1

Table 2.1.7 The most popular matching system for the second half of DRR

In the rest of the thesis, we let n denote the number of teams, I the set of teams and K the set of slots, B the major teams, SC the same city teams, ST the strong teams. In consideration of almost all of the sport scheduling literature concentrating on compact tournaments which have an even number of teams, this is presumed to be the case in this thesis except otherwise stated.

2.2 Constraints in Sports Scheduling

In sports scheduling, there are many stakeholders like teams, football players, fans, television channels, investors, local committees and governments, who often have conflicting needs with one another. Different countries and different leagues usually have their own special requirements that need to be taken into account. Furthermore, round robin tournament has its own mandatory constraints. Consequently, a large number of constraints need to fulfill for a fair schedule for all stakeholders. In this section, we provide a brief summary of the most typical constraints.

I. Global Round Robin constraints

In order for a tournament to be considered as a round-robin tournament, it must satisfy the following constraints.

- Each team has to play a match with each other.
- Each team has to play a match in each slot.

II. Break-slot constraints [12, 17, 19, 20, 21, 28, 29]

Generally, leagues want to avoid a schedule with first and last slots breaks. Moreover, in some special cases, breaks may not be desirable in a certain slot.

III. Consecutive break constraint [4, 32, 33, 34, 37, 38, 39, 42, 43]

Consecutive breaks occur when a team plays three or more consecutive home/away matches. Consecutive breaks are strictly forbidden in the minimum number of breaks problem. However, in the minimum traveling distance problem, Consecutive Break Constraint provides a balance between "long break" and "traveling distance" as the total number of breaks is inversely proportional to the total traveling distance [37].

IV. Place constraints [11, 16, 18, 19, 20, 22, 26, 28, 29, 33, 34, 35, 38, 39, 46]

When there is another event that is going to take place in the venue like local sports tournaments, concert or restoration, this constraint makes it certain that a team plays home or away in a certain slot.

V. Top team and bottom team constraints [16, 17, 18, 19, 20, 21, 26, 28, 29]

In some tournaments, the teams who complete the previous season at the top or bottom may need extra constraints for their special situations. For example, the teams who have finished the previous season in the top two cannot play a match against each other in the first five weeks of the tournament.

VI. Game constraints [5, 17, 21, 23, 25, 28, 29, 33, 39]

Game constraints either fix particular games to certain slots or prevent the match from being played in a specific period. These constraints are also named as a Big Game Constraint, since TV networks often prefer a big game on the predetermined slots, which is going to make the match more accessible and exciting for fans and media.

VII. Complementary constraints [12, 18, 19, 26, 28, 29, 33]

When two team share a venue, they cannot play a home match at the same slot due to venue availability. In this case, a complementary constraint can prevent this from happening. When they need to play with each other, this match is going to take place in the venue which both team own, but revenue usually earned by the official home team in that slot.

VIII. Geographical constraints [8, 16, 26, 28, 29]

This constraint prevents many matches to take place in a small geographical area in the same slot. In some sense, geographical constraints separate the matches in the whole tournament region. Same city constraints are also geographical constraints, which prevent many home games occurring in a slot when there is more than one team which participates tournament from any city.

IX. Pattern constraints [4, 17, 18, 19, 20, 21, 22, 26, 28, 29, 33, 34, 35, 37, 38, 39]

In some applications, there are different prerequisites for patterns. For example, a fixed number of breaks are allowed in a pattern, or a fixed consecutive break occurrences in a pattern or pattern set which all patterns have an equal number of breaks.

X. Separation constraints [17, 21, 27, 28, 29, 35, 38, 39]

If teams have to play a match more than one in non-mirrored tournaments, often a lower bound is needed for the number of slots between two matches with the same teams. Separation constraints are not necessary for the mirrored tournaments because there are always $n-2$ slots between two matches with the same teams.

XI. Traditionally strong teams constraints [17, 19, 21, 29]

In most of the leagues, there are ***traditionally strong teams*** that dominate their league and often finish the season in the top positions. Clearly, playing a series of matches with traditionally strong teams may cause fatigue and loss of joy for the weaker team because of possible consecutive losses. Traditionally strong teams constraints prevent this from happening and create a fairer schedule for all participants.

XII. Traditionally strong teams and Strong teams balance constraints

In a season, most of the points that traditionally strong teams and ***strong teams*** lose are lost in major away matches. Strong teams are the teams who completed the previous season at top positions. The ***major match*** is played between two traditionally strong teams. Especially at the end of the season, when the championship race becomes more exciting, the teams without a balanced traditionally strong team schedule comes to a disadvantaged position. In order to maintain the balance, every traditionally strong team must play half of their major matches in the home venue within the first half of the mirrored tournament.

XIII. Balanced team distribution constraints

Round robin tournament schedules contain patterns for every team that shows opponents in a sequent. The carry-over effect occurs when team X play a match against team Y and team Z consecutively, team Y will affect X's condition against team Z. In other words, team Z obtains a carry-over effect from team Y. As can be easily predicted, the carry-over effect may create an advantageous situation for team Z. If team Y's physical or tactical power is much stronger than team X, then team X's strength may be weakened by team Y since team X has to make more efforts to win. Consecutively playing matches with very strong teams may cause consecutive losses for weaker teams. Moreover, consecutive losses may result in an undesirable situation for any team and their supporters due to loss of motivation and disappointment. However, Goossens and Spieksma [49] used an approach to evaluate the effect of carry-over effects and they found out that the influence of carry-over on the score a match are negligible.

Instead of the carry-over effect, we use balanced team distribution constraints that prevent teams to play consecutive matches with top teams or bottom teams from previous season's data. Balanced team distribution constraints aims to provide a more balanced match distribution in leagues involving teams of different strengths. Considering a balanced team distribution as a hard constraint may cause invisibility, therefore it should be considered as a soft constraint.

In the sports scheduling literature, constraints are divided into two groups, which are hard constraints and soft constraints. Failing to satisfy a hard constraint make the schedule infeasible therefore all hard constraints must be satisfied to get a feasible schedule. On the other hand, failing to satisfy a soft constraint is penalizing the objective function. Converting a soft constraint to hard constraint may cause infeasibility.

Most common objectives in sports scheduling are minimizing the number of breaks or the traveling distance. However, minimizing the number of violated soft constraints, which is usually carry-over constraint, are also popular.

2.3 Minimizing Breaks

In the traveling tournament, teams play a couple of consecutive away matches without returning home venue. However, when this is not the case, teams usually return to the home venue after each away match. Therefore, alternative patterns without many consecutive away match preferred because of regular income and for the supporters who want to watch their team's matches in the home venue regularly. At the start of the league, teams with consecutive home matches might do relatively better than the teams with consecutive away matches. Clearly, the strength of the teams according to the position throughout the tournament can be expressed more smoothly without the patterns that contain many breaks.

Demand for alternative patterns with the minimum number of breaks has encouraged researchers to develop sophisticated solution approaches facing various constraints for practice. Constructive methods are widely used heuristics that create a complete solution systematically by sequentially adding components to the partial solution. However, this method cannot handle constraints. Therefore, in the last decades, instead of constructive methods, the researches concentrate on the decomposition methods that are able to deal with all the necessary constraints [3]. The main objective of the decomposition methods is to decompose the problem into subproblems which are relatively easy to solve.

2.3.1 Constructive Methods

In the 80s, the first sport scheduling studies were carried out using constructive methods. Rosa and Wallis [7], de Werra [8,9,10,11] and de Werra at al. [12] have done some studies on the relationship between graph theory and sport scheduling.

In [8], de Werra was the first person who considers a tournament with constraint, which is a geographical constraint. De Werra gave complementary patterns to teams, which close to each other, to satisfy geographical constraint if possible. In his paper, De Werra defined the term of ***Canonical Schedule***. In canonical schedule, teams play matches with the same opponents in the same order as shown in Table 2.3.1.1

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5
Team 1	*2*	*3*	*4*	*5*	*6*
Team 2	*1*	*4*	*5*	*6*	*3*
Team 3	*5*	*1*	*6*	*4*	*2*
Team 4	*6*	*2*	*1*	*3*	*5*
Team 5	*3*	*6*	*2*	*1*	*4*
Team 6	*4*	*5*	*3*	*2*	*1*

Table 2.3.1.1 Timetable example of canonical scheduling technique for six teams

In the last decade, many countries like Czechia, Poland, Ireland, Belgium, Germany, and Norway have stopped using canonical schedule because using canonical scheduling prevents the implementation of many necessary constraints.

In [9] De Werra suggested that if there is a SRR tournament, which consists of n teams, the tournament has at least $n-2$ breaks. Since teams with the same patterns cannot play a match with each other in any slot, all teams must have different patterns. Moreover, the observed result gives that only two teams can have a pattern without any break and De Werra's proof based on it. Even though his proof was very straightforward, the proposition is still crucial because it gives a lower bound for the total number of breaks in SRR tournaments.

In [9] De Werra also showed that mirrored DRR tournament has at least $3n - 6$ breaks. Every team who has a break in the first round also has a break in the second round and one more break between first round and second round. The breaks happen between rounds does not create an undesirable situation because there is a recess, which usually lasts at least 4 time-slots.

In [10] de Werra minimize the total number of breaks and distribute the breaks evenly between each team.

In [11] de Werra revive his previous works with extra place constraints.

Schreuder [13] stated that the binary variable could be used in the formulation of sports scheduling.

2.3.2 The Constrained Minimum Break Problem

In the 90s, instead of theoretical graph based studies, practical applications became popular because tournaments began to need better schedules. Constructive methods are incapable of handling constraints. Therefore, the necessity of developing new solution methods emerged when researchers had to take into account not only the number of breaks but also the constraints needed by stakeholders.

The problem may vary considerably for each application because of the different constraints. Some constraint like carryover and geographical constraints can make the problem very difficult if we cannot handle them in smart ways.

Simulated annealing is a metaheuristic method used by Willis et al. [14] for scheduling Australian cricket league. Wright [15] used another metaheuristic method called tabu search for scheduling England cricket league.

Despite other methods exist in the literature as mentioned above; most of the researchers have used decomposition approach in their studies. Although a sports scheduling problem is generally decomposed in four steps, the order of steps may vary and some steps may be combined. These steps are widely used to solve constrained break minimization problem in the literature.

Steps:

Step 1: Generating feasible patterns.

Step 2: Finding a set of patterns with cardinality equal to the number of teams.

Step 3: Finding timetables by assigning games to the pattern set.

Step 4: Assigning teams to patterns. (Together with the timetable, this gives a schedule.)

Schreuder [16] indicate that creating a timetable is NP-hard, so he divided the problem into two subproblems to be able to get the best possible approximate solution. Step 1 and Step 3 combination creates the canonical schedule for placeholders, then Schreuder allocated teams to placeholders. Heuristic solution methods presented to solve a mirrored DRR tournament for the Dutch football league.

Nemhauser and Trick [17] used all four steps to schedule the American college basketball tournament consisting of nine teams. They proposed a hybrid method, which combines the enumeration techniques and the IP to create pattern set. In the first step, they generated sufficient number of mirrored patterns, which has a moderate chance of being inside of the feasible pattern set. Specific two slots were replaced with each other in order to satisfy special constraint of the tournament. After generating the patterns, IP model chose 9 patterns among them that minimize the total number of breaks. Each slot of pattern set must have 4 home matches, 4 away matches and one bye in order for the pattern set to be feasible. The IP model found seventeen different feasible pattern sets. All feasible timetable for each pattern sets generated by the second IP model, and as a result 826 timetables were found. Eventually enumerating techniques used to allocate teams to placeholders and it leads to 9! different allocations. However, only seventeen of them were feasible and one of them was chosen as the final schedule.

After Nemhauser and Trick's integer programing and enumeration approach for solving sport scheduling, Schaerf [18], Henz [20, 21], and R´egin [22] proposed constraint programing approach.

Schaerf [18] considered mirrored DRR tournament scheduling problem with hard and soft constraints. The first phase of his 2-phase approach combined the first three steps of decomposition. Schaerf used a modified canonical schedule, which always gives lower bound on the break (3n-6) for mirrored DRR tournaments. Thus, phase 2 became an assignment problem and solved with CP. Although Schreuder's [16] heuristic approach gives a much faster solution, the CP model gives the optimal solution.

Henz [20] solved all four steps using constraint programming and emphasized that solving step 4 before step 3 often gives better results. In [21] Henz used the constraint programming approach explained in his previous paper [20]. Henz used the same basketball instance used by Nemhauser and Trick [17] to compare the effectiveness of his approach. Consequently, the constraint programming used by Henz outperforms the hybrid model proposed by Nemhauser and Trick.

Regin [22] also applies constraint programming approaches to solve SRR tournament scheduling problems. He showed that using the symmetry breaking constraints enhance the performance of the algorithm significantly. His problem is also known as the break minimization problem. The break minimization problem consists of finding a feasible pattern set which minimizes the total number of breaks within a fixed timetable.

Trick [23] has done work on the “order of the solution steps”. He stressed that the most critical parts of scheduling should be taken into account as early as possible in the solution and that the order of solution steps should be done in this direction. If there are a large number of place constraints, it is beneficial to solve steps 1 and 2, before step 3 and 4. Furthermore, if there are many timetable related constraints, it is advantageous to deal with steps 3 and 4, before step 1 and 2. Trick presented a two-phase solution approach in which the first phase solves step 3 and 4 together. He used a hybrid method, which combines constraint programming and integer programming, to solve the first phase then integer programming for the second phase.

Elf et al. [5] showed that in an undirected graph, break minimization problems is equivalent to maximum cut problems.

Miyashiro and Matsui [24] studied the problem of finding a feasible pattern set with exactly n-2 breaks for a given timetable. They proved that it could be solved in polynomial time whether there is feasible pattern set with an exactly n-2 break for a given timetable or not.

Miyashiro et al. [25] suggested mandatory condition for the feasibility of pattern sets that consist of N teams. His conditions of the feasible pattern set valid only for pattern sets with a minimum number of breaks up to twenty-six teams. For every subset of teams, $\overline{N} \subseteq N$ they let the functions $A(\overline{N}, s)$ and $H(\overline{N}, s)$ return the number of away games and home games $\overline{N}$ plays in slot s. Any subset of teams $\overline{N}$ must play $\frac{1}{2}|\overline{N}|(|\overline{N}|-1)$ matches in a SRR tournament in order to satisfy global round robin constraints and, in any slot s, teams cannot play more than $\min\{A(\overline{N}, s), H(\overline{N}, s)\}$ mutual matches.

The formulation of mandatory condition;

$$\sum_{s \in S} \min\{A(\overline{N}, s), H(\overline{N}, s)\} \geq \frac{1}{2}|\overline{N}|(|\overline{N}|-1) \qquad \forall\ \overline{N} \subseteq N \qquad (2.3.2.1)$$

Even though a given pattern set with a minimum number of breaks that meets the mandatory condition can be checked in polynomial time, they did not execute any program code beyond twenty-six teams because immense time requirements to solve integer programming.

Croce and Oliveri [26] scheduled professional Italian football league; Serie A. They considered many additional constraints to create a better schedule. Since there were more than one TV networks who had the right to broadcast matches in Serie A, the schedule was balanced according to the TV such that all TV networks have an equal number of home matches in each slot. Additionally, complementary constraints applied for each pair of teams who shares a stadium. Although 3-phase approach adapted by Croce and Oliveri, they used all four decomposition steps, since patterns that have a maximum of four breaks are generated before phase 1. Phase 1,2 and 3 correspond to steps 1,2 and 3 respectively. Croce and Oliveri solved all phases iteratively according to the following scheme with integer programming.

I. Two hundred pattern sets were generated.

II. For each pattern set, at least one feasible timetable was generated if any exists.

III. For each feasible timetable, teams were located to placeholders to find feasible schedules if any exists.

Croce and Oliveri created a few high-quality schedules with their solution method.

Rasmussen and Trick [27] developed a pattern generating Benders approach, which is iterative, logic-based Benders decomposition approach. The pattern generating Benders approach consist of four-phase. Phase 1 generated patterns while phase 2 generated pattern sets with the use of integer programming. Phase 3 located teams to placeholders after checking the feasibility of pattern set. Eventually, phase 4 generated timetable with constraint programming. When the feasibility check failed, a logic-based Benders cut added to the integer programming model and algorithm returns to phase 2. The cycle only ended after it found a feasible pattern set or after it proved the integer programming from phase 2 was infeasible. If the pattern set was infeasible algorithm returned to phase 1 and generated new patterns. The algorithm ended after an optimal solution found or infeasibility of model was proved. Figure 2.3.2.1 shows the flowchart of the pattern generating Benders approach. Their algorithm struggled to prove optimality, but they were able to create a better schedule for Danish football league, which consisted of twelve teams.

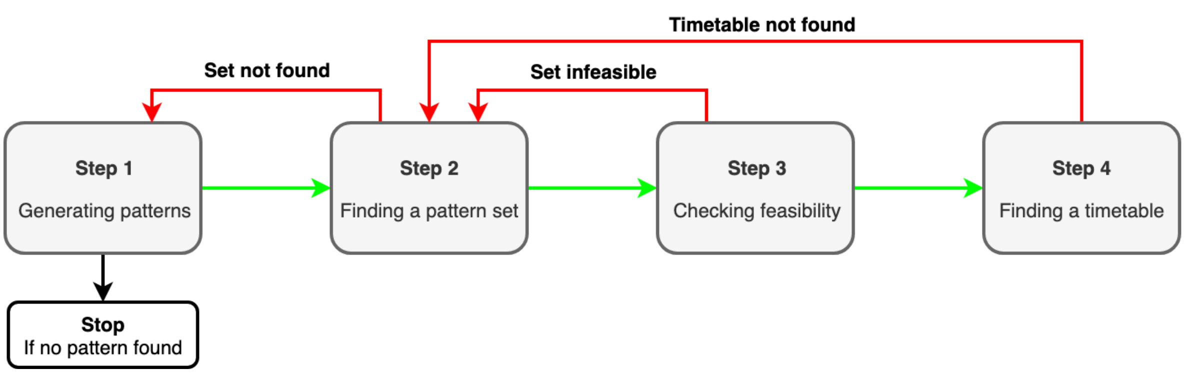

Figure 2.3.2.1 the flowchart of the algorithm

In [28], Rasmussen scheduled a TRR tournament for the Professional Danish Football League with the pattern generating Benders approach that was developed in [27], by Rasmussen and Trick. In contrast to [27], which only considered the place constraints, Rasmussen [28] considered various constraints related to the timetable. Extra added constraints made the problem much harder to solve because subproblem became an optimization problem instead of a feasibility problem. The master problem found pattern sets while the subproblem found timetables. Hence, optimality cut must be also added to master problem.

In [29] Bartsch et al. proposed a model that is a specific heuristic approach which consisted of 3 phases for the German and the Austrian leagues. In this approach, phase 1 generated pattern sets and timetable while phase 2 assigned teams to placeholders, and phase 3 set dates for each match. They used a constraint that distributed popular matches evenly over the tournament. The schedule created by Bartsch et al. was used in the German and Austrian leagues.

In [30] Carlsson et al. presented a constraint programming model for Swedish Professional Handball League that combined double round robin tournament with an extra divisional single round robin tournament. Unlike the decomposition approach, their model created a schedule in a single step, which overcomes the conditions that may lead to a sub-optimal solution. Carlsson et al. claimed that their approach require less computational effort to schedule the Swedish Professional Handball League than previous approaches.

2.4 Minimizing Travel Distance

Minimizing the travel distance preferred over minimizing the breaks when the tournament area is too big for teams to return home venue after each away matches. When a team plays an away match, instead of returning home, they play another away match with one of the neighboring teams. This result in a considerable saving in traveling costs and time.

After the 1970s oil crises, the cost of traveling increased greatly. Since this extra cost put especially amateur teams and tournaments in a difficult position, the researcher started to focus on minimizing travel distance or in other word minimizing travel costs.

The First published study on distance minimization, written by R.T. Campbell and D.S. Chen, was basketball schedule in 1976 [4]. In 2001 Easton et al. [31] presented The traveling tournament problem: Description and benchmarks. Moreover, their paper, which is about minimizing traveling distance, attracted more attention than the work done by R.T. Campbell and D.S. Chen.

2.4.1 Practical Applications

R.T. Campbell and D.S. Chen [4] presented 2 phases solution approach for relaxed DRR basketball league consist of ten teams. In their study, teams are not allowed to play more than two consecutive away matches. The optimum trip was calculated for each team at phase-1. They showed that this could be done by separating rivals into groups of two such that distance between them are minimized. Moreover allowed trip length should be maximized. They use a constructive approach to convert optimal pairing into a number of feasible sequences. R.T. Campbell and D.S. Chen managed to find a feasible schedule that minimizes the total traveling distance with their approach. We should note that pairing approach is only useful when distances of venue locations for tournament teams are approximately equivalent.

In [32] Ball and Webster tried to apply the integer programming approach to a very similar problem. However, they failed to solve it because of the size of the problem. Afterward, they used a heuristic solution method which is almost the same model applied by R.T. Campbell and D.S. Chen.

Cain [33] presented a heuristic approach to schedule the basketball league, consisting of twelve teams and two divisions. This instance, which consists of 162 teams, is one of the biggest in the literature. In addition, the problem has become more difficult due to a large number of constraints which taken into account. Their solution method divided the problem into three stages, each generating a pattern set. After that, an optimal timetable found for the given pattern set with the use of the constructive method.

Bean and Birge [34] proposed a solution method to schedule the NBA tournament. Although they started with integer programming approach, they used a two-phase heuristic approach that similar to previous heuristic approaches due to the problem size. They relaxed the previous “at most two consecutive away matches” constraint to “at most five consecutive away matches” constraint which makes problem significantly harder. Since venues often used for other events, a large number of place constraint had to take into account. In phase 1, Bean and Birge used a heuristic approach that minimizes the total travel distance of each team. Since tournament teams' location of venue distances are very different from each other, it is not appropriate to use pairing approach here. Therefore, instead of the pairing approach, trips were scheduled one by one starting from the longest one in phase 2.

In [35] Ferland and Fleurent proposed a method which helps to schedule the National Hockey League (NHL). NHL is an ice hockey tournament where American and Canadian teams compete together. NHL is a relaxed tournament that consists of two conferences; each includes two divisions with a total of 21 teams. Since the mathematical formulation of the NHL is too large to handle, their proposed method only present a number of steps, which helps to create a schedule manually.

In [36] Ferland and Fleurent proposed an integer programming model that determines the number of matches that needed to play between divisions.

Russell and Leung [37] proposed 2 phases approach to schedule basketball league, which is a compact double round robin tournament. Phase 1 generates a feasible schedule for placeholders with the use of exchange heuristic and phase 2 assigns teams to placeholders with the use of the total enumeration. Phase 1 generates a feasible schedule for placeholders then the schedule is improved with the use of exchange heuristic. Phase 2 assigns teams to placeholders with the use of the total enumeration. Since the number of consecutive away matches restricted to two away matches, the pairing approach used. Even though their method presents a more flexible schedule, the schedule is rejected because of bye allowance and travel distance increment. Russell and Leung stated that minimizing total travel distance and maximizing the total number of break is connected. They proved the following theorem, which indicates that “If the maximum number of consecutive home or away matches is restricted to 2, in a round robin tournament that consists of an even number of teams, the maximum number of break strictly less than $n(\frac{n}{2} - 1)$ for $n \geq 6$.”

Costa [38] proposed a metaheuristic solution approach, which combines tabu search algorithm and genetics algorithm, to schedule National Hockey League. Afterwards, initial solutions is obtained by the algorithm that is used in [35], 3 phases algorithm is repeatedly used. In the reproduction phase, monotonically decreasing reproduction probability is assigned to schedules, which is related to a total number of violated constraints. In the crossover phase, new schedules are generated from existing schedules. Tabu search is used to obtain new schedules by replacing a match from one slot to another.

In [39] Wright presented a sub-cost guided simulated annealing approach to schedule relaxed double round robin tournament of the Professional New Zealand Basketball League that consist of ten teams. The objective function of mathematical model indicates the number of violated demands. Wright asserts that it is useful to relax the structural constraints during the search.

2.4.2 The Traveling Tournament Problem

The traveling tournament problem was first introduced by Easton et al. [31] who were motivated by the problems encountered in professional baseball leagues. When the traveling tournament problem used as a benchmark problem, solution methods can be developed and compared with each other.

Traveling tournament problem, which is expected to satisfy constraints and minimizes distance, can be formulated as follows.

Definition 1 (Easton et al. [31]) The traveling tournament problem is as follows:

Input of the travelling tournament problem: (n), the number of teams to participate in the tournament; (D), an nxn distance matrix; (L, U), Integer parameters that indicate the tradeoff between distance and break.

Output of the travelling tournament problem: A double round robin tournament on the n teams such that the number of consecutive home games and consecutive away games are between L and U inclusive and the total distance travelled by the teams is minimized

If $L = 1$ and $U = n - 1$, then teams may take a full trip like a traveling salesman tour. As the U gets smaller, teams need to return home more often, which increases total travel distance.

Scheduling the matches of the same opponent in consecutive slots creates an undesirable situation in most traveling tournament problems. Therefore, there are two additional required constraints, which are mirrored constraint and no-repeater constraint, which must be taken into account in the mathematical model to make the solution practically applicable. Since there is at least n-1 slot between the matches of the same opponents in the mirrored double round robin tournament, it is sufficient to use either the mirrored constraint or the no-repeater constraint.

In their paper Easton et al. [31] present two problem classes for traveling tournament problem: Circular distance instance is a traveling tournament problem that consist of n teams, which is obtained by generating a n node circle graph with unit distances (distance of one between all adjacent nodes). National league instances (NL) is a problem specific instance, which creates small instances that reflect the existent system within the ranges from 4 to 16 teams (NL4,…, NL16), designed for American League and National League.

In [40] Ribeiro and Urrutia presented another problem class: Constant distance instance, in this instance, the distance between all the teams is one. Therefore, Ribeiro and Urrutia proved that minimizing travel distances is equivalent to maximizing breaks when the distance is constant.

In [41] Trick calculated the upper bound and the lower bounds for all problem classes.

Different kind of solution methods presented for traveling tournament problem. Easton et al. [31] propose a solution method which generates pattern set with the maximum number of trips (maximize long sequence of breaks), afterward, related timetables are found as like that minimize traveling distance. They compared the result with the independent lower bound, which they calculate for each team separately, and performed the optimality check. As long as the independent lower bound is below the best solution found by the algorithm, the algorithm continues. They managed to solve NL4 and NL6 optimally with this method.

Benoist and Laburthe [42] proposed a hybrid algorithm, which combines constraint programming and Lagrange relaxation with the hierarchical structure. The constraint programming model solves the problem with the use of a global constraint that helps to improve bounds during the search. The global constraint contains a Lagrange controller to adjust the multipliers for perturbed subproblem for each team. Furthermore, the subproblem creates a schedule by minimizing the total distance individually for every team.

Easton at al. [43] proposed another solution approach that combines the integer programming and constraint programming. They used the branch and price algorithm in which columns represent the tour of the teams. The master problem assigns teams to tours with the use of integer programming, while the pricing problem generates tours with the use of constraint programming. Easton managed to schedule NL8 optimally however their mathematical model did not include no-repeater constraint. Thus, the solution only useful as a lower bound because it is not practically applicable.

Anagnostopoulos et al. [44] presented a simulated annealing algorithm to solve the traveling tournament problem. Their mathematical model contains hard constraint, which includes the global round robin constraints, and soft constraints, which include the no-repeater constraint. The model randomly chooses a step and performs with probability one as long as objective improves. When this is not the case, probability depends on the temperature and the worsening of the objective. Anagnostopoulos et al. improved the previous best solution for the National League.

In [45] Henz presented a hybrid algorithm, which combines neighbor search and constraint programming, to prevent the problem of stuck in local optima. The move he used not only exchange slots, but also relaxes corresponding variables in exchanged slots. The constraint programming used to generate schedule from an existing partial schedule.

In [40] Ribeiro and Urrutia presented upper bounds on the number of breaks of constrained and unconstrained single round robin tournaments. They also present an upper bound for the constrained single round robin tournament with maximum three consecutive breaks allowance.

Rasmussen and Trick [46] presented an algorithm that maximizes the total number of breaks for the constant distance traveling tournament problem presented by Ribeiro and Urrutia [40]. Eventually, they managed to reach optimal value for the problem up to 18 teams with mirrored constraint and up to 16 teams with no-repeater constraint. Furthermore, Hentenryck and Vergados [47] improved their best solutions for 20 teams with mirrored constraint and 18 teams with non-mirrored constraint using the same approach.

In [48] Lim et al. proposed an algorithm that combines simulated annealing and hill climbing for the traveling tournament problem. Their algorithm improves the best solutions for non-mirrored circular distance problem that consist of 10 teams and more.

Chapter 3

Mathematical Modeling

In this chapter, we propose an integer linear programming model for SRR scheduling. We need to create a schedule for the first half season then we will use a mirror of the first half season for the second half. Our goal is the minimization of the total number of breaks and balanced team distribution violations.

3.1. Formulation

Index Sets

$\boldsymbol{I} = \{\mathbf{1}, \dots, \mathbf{n}\}$: set of teams in the sports organization;

$\boldsymbol{K} = \{\mathbf{1}, \dots, \boldsymbol{n} - \mathbf{1}\}$: set of slots in the sports organization;

Parameters of the model

$\boldsymbol{n} =$ The number of teams that participates in the sports organization.

$\boldsymbol{M} = A\ constant\ number$;

$\boldsymbol{B} = \{\boldsymbol{b_1}, \boldsymbol{b_2}, \boldsymbol{b_3}, \dots\}$: Major teams in the sports organization;

$\boldsymbol{TST} = \{\boldsymbol{tst_1}, \boldsymbol{tst_2}, \boldsymbol{tst_3}, \dots\}$: Traditionally strong teams in the sports organization;

$\boldsymbol{ST} = \{\boldsymbol{st_1}, \boldsymbol{st_2}, \boldsymbol{st_3}, \dots\}$: Strong teams in the sports organization;

$\boldsymbol{SC_z} = \{\boldsymbol{sc_{z1}}, \boldsymbol{sc_{z2}}, \boldsymbol{sc_{z3}}, \dots\}$: The teams who participate in the sports organization from city Z;

$\boldsymbol{G} = \{\boldsymbol{g_1}, \boldsymbol{g_2}, \boldsymbol{g_3}, \dots\}$: *The teams that completed the previous season in group G*;

$\boldsymbol{nMM} = \{\boldsymbol{1}, \dots, \boldsymbol{nmm}\}$ **:** The first nMM (no-Major-Matches) slots of the tournament that major matches cannot be played.

The following basic mathematical model has been established by using the above mentioned decision variables and parameters.

Decision Variables

For any two teams $i, j \in I$ and week $k \in K$, let $\boldsymbol{x}_{(i,j,k)}$ be a binary variable;

$$\boldsymbol{x}_{(i,j,k)} = \begin{cases} 1 & \text{if team i plays a home game against team j in week k,} \\ 0 & \text{otherwise;} \end{cases} \qquad (3.1.1)$$

Our formulation for each constraint of the problem, which leads to a feasibility model with $O[n^2(n-1)]$ variables for n number of teams.

Objective Function

The minimizing of the total number of breaks is more important than the violations of the balanced team distributions. We used the big M method to prevent model to choose the fewer balanced team distributions over the fewer breaks.

$$\text{Min} \quad Z = m \sum_{i} \sum_{k} u_{(i,k)} + \sum_{g,k \mid k \le n-2} s_{g,k}^{-} \qquad (3.1.2)$$

Linearization of objective function

1. Defining a break

$$\sum_{j,i\neq j}[x_{(i,j,k)} + x_{(i,j,k+1)}] = t_{(i,k)} \qquad \forall \quad i \in I, k \leq n-2 \in K \qquad (3.1.3)$$

2. Linearization of objective function, Min $Z = \sum_i \sum_k |t_{(i,k)} - 1|$

$$t_{(i,k)} - 1 \leq u_{(i,k)} \qquad (3.1.4)$$

$$1 - t_{(i,k)} \leq u_{(i,k)} \qquad (3.1.5)$$

Constraints

Now we present the definition and the formulation of each constraint of the problem to make the problem more understandable.

1. Restricting a team to play exactly one game each week (Round Robin constraints)

$$\sum_{k\in K,i\neq j}[x_{(i,j,k)} + x_{(j,i,k)}] = 1 \qquad \forall \quad i,j \in I \qquad (3.1.6)$$

2. Each team must play a Home or an Away game with all the other teams (Round Robin constraints)

$$\sum_{j\in I,i\neq j}[x_{(i,j,k)} + x_{(j,i,k)}] = 1 \qquad \forall \quad i \in I, k \in K \qquad (3.1.7)$$

3. No team can play three consecutive Home or Away games

$$\sum_{i\in I,i\neq j}[x_{(i,jk)} + x_{(i,j,k+1)} + x_{(i,j,k+2)}] \leq 2 \qquad \forall \quad j \in I, (k \leq n-3) \in K \qquad (3.1.8)$$

$$\sum_{j \in I, i \neq j} [x_{(i,j,k)} + x_{(i,j,k+1)} + x_{(i,j,k+2)}] \leq 2 \qquad \forall \quad i \in I, (k \leq n-3) \in K \qquad (3.1.9)$$

No break allowed for the last three slots of the season: Last slots of the season are crucial for all the stakeholders. Most of the time which team will finish the tournament as a champion, or which teams will be relegated from the tournament are still unknown in last slots. Under these circumstances, any consecutive home games is a huge advantage and any consecutive away games is a huge disadvantage, for any teams who still have a chance to claim the championship title. The same thing also applies to the bottom of the league.

$$\sum_{j \in I, i \neq j} [x_{(i,j,n-2)} + x_{(i,j,n-1)}] = 1 \qquad \forall \quad i \in I \qquad (3.1.10)$$

$$\sum_{j \in I, i \neq j} [x_{(i,j,n-3)} + x_{(i,j,n-2)}] = 1 \qquad \forall \quad i \in I \qquad (3.1.11)$$

4. No break allowed for the first two slots of the season: Every team should play one home game and one away game in the first two slots.

$$\sum_{j \in I, i \neq j} [x_{(i,j,1)} + x_{(i,j,2)}] = 1 \qquad \forall \quad i \in I \qquad (3.1.12)$$

5. Same city constraints: assignation of the entire city teams to home or away in the same slot can not be accepted because of stadium availability, security reason or football fans who want to watch a live match in their city, every week. In our model, the same city constraints are hard constraints that cannot be violated. When the total number of teams, who participate in the tournament from the City Z, is an even number, half of the team should play home while the other half plays away. When the total number of same city team is two, the teams have often the complementary property. However, two teams who share the same stadium must have complementary property.

$$\sum_{sc_z \in SC_z, sc_z \neq i} \sum_{i \in I} x_{(sc_z,i,k)} \leq \lceil (sc_{z1} + sc_{z2} + \cdots)/2 \rceil \qquad \forall \quad k \in K \qquad (3.1.13)$$

$$\sum_{sc_z \in SC_z, sc_z \neq i} \sum_{i \in I} x_{(sc_z,i,k)} \geq \lfloor (sc_{z1} + sc_{z2} + \cdots)/2 \rfloor \qquad \forall \quad k \in K \qquad (3.1.14)$$

6. No major match allowed for the first nMM weeks of the season: Major matches, which followed by many fans, are an important source of football economy and for making it more exciting for fans and media, the matches between traditional strong teams should be placed after a certain number of weeks.

$$\sum_{nmm \in nMM, b \neq 2b} x_{(b,2b,nmm)} = 0 \qquad \forall \quad b, 2b \in B \qquad (3.1.15)$$

7. Major teams/traditionally strong teams lose most of the points they lost in a season when they face with the other Major teams/traditionally strong teams on the away matches. Teams without a balanced major team schedule may become disadvantaged towards the end of the season when the championship race became more exciting. To achieve a balance here, between major matches, a major team should play half of the major matches at home each round.

$$\sum_{k \in K, b \neq 2b} x_{(b,2b,k)} = \sum_{k \in K, b \neq 2b} x_{(2b,b,k)} \qquad \forall \quad b, 2b \in B \qquad (3.1.16)$$

$$\sum_{k \in K, tst \neq 2tst} x_{(tst,2tst,k)} = \sum_{k \in K, tst \neq 2tst} x_{(2tst,tst,k)} \qquad \forall \quad tst, 2tst \in TST \qquad (3.1.17)$$

8. Playing consecutive matches with very strong teams may cause consecutive losses for weaker teams. Moreover, consecutive losses may result in an undesirable situation for any team and their supporters due to loss of motivation and disappointment. Balanced team distribution constraints prevent teams to play consecutive matches with top teams or bottom teams from previous season's data. Balanced team distribution constraints aim to provide a more balanced match distribution in leagues involving teams of different strengths. In our model, balanced team distribution constraints are soft constraints.

$$\sum_{g \in G} [x_{(i,g,k)} + x_{(i,g,k+1)} + x_{(g,i,k)} + x_{(g,i,k+1)}] + s^{+}_{i,k} - s^{-}_{i,k} = 1 \qquad (3.1.18)$$

$$\forall \; i \in I, (k \leq n-2) \in K$$

9. Nonnegativity.

$$u_{(i,k)}, s^{+}_{i,k}, s^{-}_{i,k} \geq 0 \qquad (3.1.19)$$

Chapter 4

Solution Methodology

Sport scheduling problems typically are very hard combinatorial optimization problems with many constraints. Combinatorial optimization problems' feasible region is generally too large to search comprehensively just with pure calculation power. Therefore, in the literature, combinatorial optimization problems often solved by integer programming (IP), constraint programming (CP) and heuristic approaches. Whichever method we choose, algorithms usually need serious computational time to reach the optimal solution. Especially when the total number of the team is large, finding the optimal solution is not always possible nor easy. As the total number of teams increases in the tournament, it becomes difficult to find the optimum solution, and eventually, it is not possible to find the optimum solution. In such a case, heuristic algorithms are often preferred over the exact algorithm. Heuristic algorithms give us the best possible solutions in short times but they do not always find the optimum solution. In fact, heuristic algorithms often stuck in local optimum (does not explore all possible state of the problem).

At the proven lower bound for the single round robin tournament consists of n teams, when n is an even number, the tournament has at least n-2 breaks (Werra 1986). However, the lower bound calculated only with standard round robin constraints and we have extra constraints. Clearly, the extra constraints that we have added to create a more fair schedule may increase the lower bound. Nevertheless, searching a schedule that satisfies all of the constraints at the proven lower bound our priority in this thesis. Since we already know the lower bound of our problem, we can restrict our decision space greatly with using pattern sets that give us all possible lower bound solutions.

At lower bound for any mirrored DRR tournament, any schedule with n teams must have 2 zero-break patterns and n-2 one-break patterns. When all the teams have at most one break in the compact schedule, the schedule must have complementary property [11].

Number of teams (n)	Dummy team (dmy)	Number of slot SRR (n+dmy-1)	Zero- break pattern (always 2)	One-break pattern 2*(n-2)
2	0	1	2	0
3	1	3	2	4
4	0	3	2	4
5	1	5	2	8
6	0	5	2	8
7	1	7	2	12
8	0	7	2	12
9	1	9	2	16
10	0	9	2	16
11	1	11	2	20
12	0	11	2	20
13	1	13	2	24
14	0	13	2	24
15	1	15	2	28
16	0	15	2	28
17	1	17	2	32
18	0	17	2	32
19	1	19	2	36
20	0	19	2	36

Table 4.0.1 The number of tournament teams (n) and their corresponding values

In order to schedule the tournament consisting of n teams, we need a pattern set with n different patterns. The reason for this, the teams with the same pattern cannot play a match with each other, so standard SRR constraints cannot be satisfied and the schedule becomes infeasible. Therefore, the complement of each selected pattern must also be selected. Table4.0.1 shows the total number of tournament teams and their corresponding values.

Definition – 1: Fine pattern set: Let us call it a fine pattern set if the set only consisting of patterns that have less than two breaks. Table 4.2 shows an example of the fine pattern set for a league that consists of 10 teams, which can be used to find a solution at the lower bound. The pattern set has 16 one-break and 2 zero-break patterns.

\

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9
1. Pattern	1	0	1	0	1	0	1	0	1
2. Pattern	0	1	0	1	0	1	0	1	0
3. Pattern	***1***	***1***	0	1	0	1	0	1	0
4. Pattern	***0***	***0***	1	0	1	0	1	0	1
5. Pattern	0	***1***	***1***	0	1	0	1	0	1
6. Pattern	1	***0***	***0***	1	0	1	0	1	0
7. Pattern	1	0	***1***	***1***	0	1	0	1	0
8. Pattern	0	1	***0***	***0***	1	0	1	0	1
9. Pattern	0	1	0	***1***	***1***	0	1	0	1
10. Pattern	1	0	1	***0***	***0***	1	0	1	0
11. Pattern	1	0	1	0	***1***	***1***	0	1	0
12. Pattern	0	1	0	1	***0***	***0***	1	0	1
13. Pattern	0	1	0	1	0	***1***	***1***	0	1
14. Pattern	1	0	1	0	1	***0***	***0***	1	0
15. Pattern	1	0	1	0	1	0	***1***	***1***	0
16. Pattern	0	1	0	1	0	1	***0***	***0***	1
17. Pattern	0	1	0	1	0	1	0	***1***	***1***
18. Pattern	1	0	1	0	1	0	1	***0***	***0***

Table 4.0.2 The fine pattern set for a league that consists of 10 teams.

The complementary patterns are marked with the same color and breaks on the patterns are marked with grey.

4.1 Eliminating Infeasible Patterns

In this section, due to the complexity of the problem, we focused on reducing the computational time to make the problem manageable by eliminating infeasible pattern and pattern sets. Eliminating infeasible patterns is not new nor hard but eliminating pattern sets are require some additional work. We already know that we need n – 2 one-break patterns and there are total 2x(n - 2) one break patterns but, some of the patterns may already be infeasible due to extra constraints of the problem. Finding those patterns is not always easy or possible but finding and removing those patterns from the feasible set will make the problem easier because it will require less computational time.

No break allowed weeks (Break constraints): we are going to remove the patterns that have a break in certain weeks. In Turkish Super League, no break allowed weeks are 2, 16 and 17. Therefore, teams cannot play consecutive home or away games in the first two weeks and last three weeks.

- We find out that if we choose patterns in some combinations, the problem may be infeasible even for main SRR constraints. Removing those pattern combinations from the feasible set will also reduce computational time.

4.1.1 Ladder patterns:

Case 1:

Let us assume that we have six patterns consisting of three patterns and their complements, in which all patterns have exactly one-break on consecutive slots. Let us call it a ladder patterns (Table 4.1.1).

1. Pattern	***1***	***1***	0	1
2. Pattern	***0***	***0***	1	0
3. Pattern	0	***1***	***1***	0
4. Pattern	1	***0***	***0***	1
5. Pattern	1	0	***1***	***1***
6. Pattern	0	1	***0***	***0***

Table 4.1.1 Example of Ladder patterns

Proposition 1:

If the fine pattern set has a ladder patterns, the set is infeasible for SRR tournament schedule.

Proof of proposition 1:

Among of the six patterns let us choose three patterns with same first slot value.

1. Pattern	***1***	***1***	0	1
4. Pattern	1	***0***	***0***	1
5. Pattern	1	***0***	1	1

Teams correspond to these patterns can only play against each other on the slots that they have a break, because in all other slots if one plays a home match, others have to play a home match or vice versa. In order to satisfy standard SRR constraint, three teams corresponds to the three patterns have to play three matches with each other. (team 1 – team 2, team 1 – team 3, team 2 – team 3)

$\mathbf{C}(3,2) = 3$

1. Pattern	***1***	***1***	0	1
4. Pattern	1	***0***	***0***	1
5. Pattern	1	***0***	1	1
MNMS	0	1	1	0

$$\sum \boldsymbol{MNMS} = 0 + 1 + 1 + 0 = 2$$

Summation of the maximum number of matches that can be played in each slot (MNMS) is two. Therefore, the pattern sets containing ladder patterns are infeasible.

Case 2:

Let us assume that we have ten patterns consisting of five patterns and their complements from a fine pattern set, as in the following Table 4.1.2.

1. Pattern	**1**	**1**	0	1	0	1	0	1
2. Pattern	**0**	**0**	1	0	1	0	1	0
3. Pattern	0	**1**	**1**	0	1	0	1	0
4. Pattern	1	**0**	**0**	1	0	1	0	1
5. Pattern	0	1	0	**1**	**1**	0	1	0
6. Pattern	1	0	1	**0**	**0**	1	0	1
7. Pattern	1	0	1	0	**1**	**1**	0	1
8. Pattern	0	1	0	1	**0**	**0**	1	0
9. Pattern	1	0	1	0	1	0	**1**	**1**
10. Pattern	0	1	0	1	0	1	**0**	**0**

Table 4.1.2 A fine patter set example with 10 patterns (1)

Proposition 2:

The fine pattern set is infeasible for SRR tournament scheduling if the situation in the table 4.1.2 occur.

Proof of proposition 2:

Among of the ten patterns let us choose five patterns with same first slot value.

1. Pattern	**1**	**1**	0	1	0	1	0	1
4. Pattern	1	**0**	**0**	1	0	1	0	1
6. Pattern	1	0	1	**0**	**0**	1	0	1
7. Pattern	1	0	1	0	**1**	**1**	0	1
9. Pattern	1	0	1	0	1	0	**1**	**1**

Teams correspond to these patterns can only play against each other on the slots that they have a break, because in all other slots if one plays a home match, others have to play a home match or vice versa. In order to satisfy standard SRR constraint, five teams corresponds to the five patterns have to play ten matches with each other.

$\mathbf{C}(5,2) = 10$

1. Pattern	**1**	**1**	0	1	0	1	0	1
4. Pattern	1	**0**	**0**	1	0	1	0	1
6. Pattern	1	0	1	**0**	**0**	1	0	1
7. Pattern	1	0	1	0	**1**	**1**	0	1
9. Pattern	1	0	1	0	1	0	**1**	**1**
MNMS	0	1	2	2	2	1	1	0

$$\sum \boldsymbol{MNMS} = 0 + 1 + 2 + 2 + 2 + 1 + 1 + 0 = 9$$

Summation of the maximum number of matches that can be played in each slot (MNMS) is ten. Therefore, the pattern sets containing situation in Table 4.1.2 are infeasible.

Case 3:

Let us assume that we have ten patterns consisting of five patterns and their complements, as in the following figure.

1. Pattern	**1**	**1**	0	1	0	1	0	1
2. Pattern	**0**	**0**	1	0	1	0	1	0
3. Pattern	1	0	**1**	**1**	0	1	0	1
4. Pattern	0	1	**0**	**0**	1	0	1	0
5. Pattern	0	1	0	**1**	**1**	0	1	0
6. Pattern	1	0	1	**0**	**0**	1	0	1
7. Pattern	0	1	0	1	0	**1**	**1**	0
8. Pattern	1	0	1	0	1	**0**	**0**	1
9. Pattern	1	0	1	0	1	0	**1**	**1**
10. Pattern	0	1	0	1	0	1	**0**	**0**

Table 4.1.3 A fine patter set example with 10 patterns (2)

Proposition 3:

The fine pattern set is infeasible for SRR tournament scheduling if the situation in the Table 4.1.3 occur.

Proof of proposition 3:

Among of the ten patterns let us choose five patterns with same first slot value.

2. Pattern	**0**	**0**	1	0	1	0	1	0
4. Pattern	0	1	**0**	**0**	1	0	1	0
5. Pattern	0	1	0	**1**	**1**	0	1	0
7. Pattern	0	1	0	1	0	**1**	**1**	0
10. Pattern	0	1	0	1	0	1	**0**	**0**

Teams correspond to these patterns can only play against each other on the slots that they have a break, because in all other slots if one plays a home match, others have to play a home match or vice versa. In order to satisfy **standard SRR constraint, five teams corresponds to the five patterns have to play ten matches with each other.**

$\mathbf{C}(5,2) = 10$

2. Pattern	**0**	**0**	1	0	1	0	1	0
4. Pattern	0	1	**0**	**0**	1	0	1	0
5. Pattern	0	1	0	**1**	**1**	0	1	0
7. Pattern	0	1	0	1	0	**1**	**1**	0
10. Pattern	0	1	0	1	0	1	**0**	**0**
MNMS	0	1	1	2	2	2	1	0

$$\sum \boldsymbol{MNMS} = 0 + 1 + 1 + 2 + 2 + 2 + 1 + 0 = 9$$

Summation of the maximum number of matches that can be played in each slot (MNMS) is ten. Therefore, the pattern sets containing situation in Table 4.1.3 are infeasible.

4.2 Proposed approach

In order to solve the complex break minimization problems at the lower bound, we proposed a hybrid method, which combines a heuristic approach and integer programming. Figure 4.2.1 shows proposed hybrid algorithm. The approach consists of 5 steps are explained below;

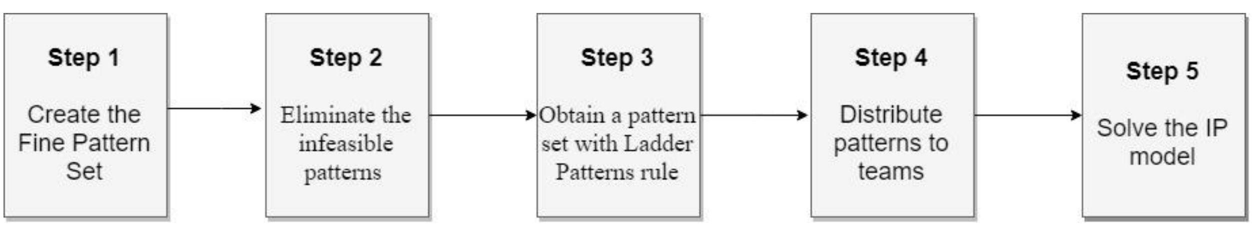

Figure 4.2.1 Proposed hybrid algorithm

Step 1: Find the Fine Pattern Set for the problem.

Step 2: Eliminate the infeasible patterns from the fine pattern set.

1. This elimination is based on constraint satisfaction. Eliminating constraints that can be satisfied by pattern elimination often reduces computational time required to solve the problem.
2. If a pattern is eliminated, also the complement of the pattern must be eliminated.

Step 3: From the remaining patterns, obtain a pattern set consisting of n patterns with the help of Ladder patterns rule.

1. If a pattern selected into the pattern set, also the complement of the pattern must be selected.
2. If a pattern set obtained by integer programming model, the computational time of step 5 might increase up to $C(the\ number\ of\ remaining\ patterns\ from\ Step\ 2, n)$ times.
3. If a pattern set obtained by manually; the computational time will be greatly decreased which may allow us to find a solution for a more complex problem. However;
   a. If there is more than one objective, the optimality of the other objectives may fail.
   b. Manually obtained pattern set cannot guarantee to satisfy the other constraints.

Step 4: Distribute patterns to teams from the obtained pattern set.

1. If the distribution is achieved with the help of integer programming computational time of step 5 might increase up to $n!$ times.
2. If the distribution is achieved manually, some of the constraints may lead to infeasibility. However, the same city constraint can be satisfied easily by the assignation of the complementary pattern.

Step 5: Solve the integer mathematical model with the remaining objectives and constraints.

Chapter 5

Application

In this chapter, we propose an integer linear programming model for Step 5 of our hybrid method to schedule the Turkish Super League. Our goal is the minimization of the total number of balanced team distribution violations.

In TFF Super League, there are 18 teams, three of them are major teams (Galatasaray, Fenerbahçe, Beşiktaş) and five of them are traditionally strong teams (Galatasaray, Fenerbahçe, Beşiktaş, Trabzonspor, Başakşehir).

In this model, all teams needed to have a constant number represent for modeling some of the constraints like the same city and major teams constraint, e.g. $i = 1$ is Galatasaray. Teams number 16,17 and 18 are new TFF Super League teams which promoted to the upper league from TFF 1. League at the end of the same season.

The ranking of the previous season is as follows;

1. Galatasaray A.Ş.
2. Fenerbahçe A.Ş.
3. Medipol Başakşehir Fk
4. Beşiktaş A.Ş.
5. Trabzonspor A.Ş.

6. Göztepe A.Ş.
7. Demir Grup Sivasspor
8. Kasimpaşa A.Ş.
9. Kayserispor
10. Evkur Yeni Malatyaspor
11. Teleset Mobilya Akhisarspor
12. Aytemiz Alanyaspor
13. Bursaspor
14. Antalyaspor A.Ş.
15. Atiker Konyaspor
16. Çaykur Rizespor
17. Ankaragücü
18. Erzurumspor

5.1 Formulation

Index Sets

$I = \{1, \ldots\ 18\}$: set of teams in the league;

$K = \{1, \ldots, 17\}$: Weeks;

Parameters of the model

$n = \{18\}$: The number of teams that participates in the sports organization.

$B = \{1,2,4\}$: Major teams;

$TST = \{1,2,3,4,5\}$: Traditionally strong teams in the sports organization;

$SC_{istanbul} = \{1,2,3,4,8\}$: Same city teams in İstanbul;

$SC_{antalya} = \{12,14\}$: Same city teams in Antalya;

$G1 = \{1, \ldots, 5\}$: Upper strong team;

$G1 = \{14, \ldots, 18\}$: Lower strong teams.

$nMM = \{1, \ldots, 5\}$: The first nMM slots of the tournament that major matches cannot be played.

Decision variables

For any two teams $i, j \in I$ and week $k \in K$, let $x_{(i,j,k)}$ be a binary variable;

$$\boldsymbol{x}_{(i,j,k)} = \begin{cases} 1 & \text{if team i plays a home game against team j in week k,} \\ 0 & \text{otherwise;} \end{cases} \qquad (5.1.1)$$

Objective function

Since we use a fine pattern set, all of the feasible solutions is at the lower bound for break minimization. Our new objective minimizes the balanced team distribution violations.

$$Min \quad Z = \sum_{g1,k \mid k \leq 16} s_{g1,k}^{1-} + \sum_{g3,k \mid k \leq 16} s_{g3,k}^{3-} \qquad (5.1.2)$$

Constraints of model

(1) Restricting a team to play exactly one game each week

$$\sum_{k \in K, i \neq j} [x_{(i,j,k)} + x_{(j,i,k)}] = 1 \qquad \forall \quad i, j \in I \qquad (5.1.3)$$

(2) Every team plays every other team. In other words, each team must play a Home or an Away game with all the other teams

$$\sum_{j \in I, i \neq j} [x_{(i,j,k)} + x_{(j,i,k)}] = 1 \qquad \forall \quad i \in I, k \in K \qquad (5.1.4)$$

(3) No break allowed for the last 3 weeks of the season

$$\sum_{j\in I, i\neq j}[x_{(i,j,15)} + x_{(i,j,16)}] = 1 \qquad \forall \quad i \in I \tag{5.1.5}$$

$$\sum_{j\in I, i\neq j}[x_{(i,j,16)} + x_{(i,j,17)}] = 1 \qquad \forall \quad i \in I \tag{5.1.6}$$

(4) No break allowed for the first 2 weeks of the season

$$\sum_{j\in I, i\neq j}[x_{(i,j,1)} + x_{(i,j,2)}] = 1 \qquad \forall \quad i \in I \tag{5.1.7}$$

(5) Same city constraint 1, İstanbul teams: At most 3 of the 5 and at least 2 of the 5 İstanbul teams should play in home (H) in same week

$$\sum_{ist\in SC_{istanbul}, ist\neq i}\sum_{i\in I} x_{(ist,i,k)} \leq 3 \qquad \forall \quad k \in K \tag{5.1.8}$$

$$\sum_{ist\in SC_{istanbul}, ist\neq i}\sum_{i\in I} x_{(ist,i,k)} \geq 2 \qquad \forall \quad k \in K \tag{5.1.9}$$

(6) Same city constraint 2, Major İstanbul teams: At most 2 of the 3 major teams and at least 1 of the 3 major teams should play in home (H) in the same week. Ps, In TFF Super League, all major teams are İstanbul teams and these teams are the teams with the most supporters in Turkey. Scheduling all major teams matches at home, may cause problems we already talked above.

$$\sum_{b\in B}\sum_{i\in I} x_{(b,i,k)} \leq 2 \qquad \forall \quad k \in K \tag{5.1.10}$$

$$\sum_{b\in B}\sum_{i\in I} x_{(b,i,k)} \geq 1 \qquad \forall \quad k \in K \tag{5.1.11}$$

(7) Same city constraint 3, Antalya teams: one team should play home while the other team plays away

$$\sum_{ant \in SC_{antalya}, ant \neq i} \sum_{i \in I} x_{(ant,i,k)} = 1 \qquad \forall \quad k \in K \tag{5.1.12}$$

(8) No major matches for the first nMM weeks

$$\sum_{nmm \in nMM} x_{(b1,b2,nmm)} = 0 \qquad \forall \quad b1, b2 \in B, b1 \neq b2 \tag{5.1.13}$$

(9) No team can play a consecutive game with major teams $b \in B$

$$\sum_{b \in B, i \neq b} [x_{(i,b,k)} + x_{(i,b,k+1)} + x_{(b,i,k)} + x_{(b,i,k+1)}] \leq 1 \quad \forall \quad i \in I, k \in K, k \leq 16 \tag{5.1.14}$$

(10) Major team balance: Every strong team must play half of their major matches in the home venue within the first half of the mirrored tournament.

$$\sum_{k \in K} [x_{(1,2,k)} + x_{(1,4,k)}] = 1 \tag{5.1.15}$$

$$\sum_{k \in K} [x_{(2,1,k)} + x_{(2,4,k)}] = 1 \tag{5.1.16}$$

(11) Traditionally strong teams balance: Every strong team must play half of the matches against other strong teams at their home venue within the first half of the mirrored tournament.

$$\sum_{k \in K} [x_{(1,3,k)} + x_{(1,5,k)}] = 1 \tag{5.1.17}$$

$$\sum_{k \in K} [x_{(2,3,k)} + x_{(2,5,k)}] = 1 \qquad (5.1.18)$$

$$\sum_{k \in K} [x_{(4,3,k)} + x_{(4,5,k)}] = 1 \qquad (5.1.19)$$

$$\sum_{k \in K} [x_{(1,3,k)} + x_{(2,3,k)}] = 1 \qquad (5.1.20)$$

(12) Balanced team distribution constrain, Top teams: if team i play with any team which belongs to g1 on (k)th week then team i should not play with g1 teams on (k+1)th week

$$\sum_{g1 \in G1} [x_{(i,g1,k)} + x_{(i,g1,k+1)} + x_{(g1,i,k)} + x_{(g1,i,k+1)}] + s^{1+}_{i,k} - s^{1-}_{i,k} = 1 \qquad (5.1.21)$$

$$\forall \; i \in I, k \in K \; (k \leq 16)$$

(13) Balanced team distribution constrain, Bottom teams: if team i play with any team which belongs to g2 on (k)th week then team i should not play with g2 teams on (k+1)th week

$$\sum_{g2 \in G2} [x_{(i,g2,k)} + x_{(i,g2,k+1)} + x_{(g2,i,k)} + x_{(g2,i,k+1)}] + s^{2+}_{i,k} - s^{2+}_{i,k} = 1 \qquad (5.1.22)$$

$\forall \; i \in I, k \in K \; (k \leq 16)$

(14) Nonnegativity

$$s^{1+}_{i,k}, s^{1-}_{i,k}, s^{2+}_{i,k}, s^{2-}_{i,k} \geq 0 \qquad (5.1.23)$$

Chapter 6

Conclusions and Future Prospects

6.1 Conclusions

For the purpose of checking the effectiveness of the hybrid algorithm, we have tested the efficiency of the algorithm on different instances of the mirrored round-robin tournament. For comparison, we compared computational results with the current situation on the TFF Super League.

In TFF Super League, there are 18 teams. A league that consists of 18 teams has 2 zero-break patterns and 32 one-break patterns. The fine pattern set for a league that consists of 18 teams showed in table 6.1.1. In step 2, Constraint (3) and (4) can be easily eliminated by removing the pattern number 3,4,31,32,33,34 from the fine pattern set.

After applying step 3, pattern set obtained by manually. On step 4, the teams are assigned to the patterns by manually because of the complexity of the problem.

	Slot 1	Slot 2	Slot 3	Slot 4	Slot 5	Slot 6	Slot 7	Slot 8	Slot 9	Slot 10	Slot 11	Slot 12	Slot 13	Slot 14	Slot 15	Slot 16	Slot 17
1. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
2. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
3. Pattern	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
4. Pattern	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
5. Pattern	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
6. Pattern	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
7. Pattern	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0
8. Pattern	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1
9. Pattern	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0	1
10. Pattern	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1	0
11. Pattern	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1	0
12. Pattern	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0	1
13. Pattern	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0	1
14. Pattern	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1	0
15. Pattern	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1	0
16. Pattern	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0	1
17. Pattern	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0	1
18. Pattern	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	0
19. Pattern	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1	0
20. Pattern	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1
21. Pattern	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0	1
22. Pattern	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0
23. Pattern	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1	0
24. Pattern	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	1
25. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0	1
26. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0
27. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1	0
28. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1
29. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0	1
30. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0
31. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1	0
32. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1
33. Pattern	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	1
34. Pattern	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0

Table 6.1.1 All patterns in the fine pattern set for a league that consists of 18 teams

On step 5, computational results obtained from GAMS (CPLEX) solver. Computation times announced in the tables are in seconds and all tests have been executed on a computer with 6th Generation Intel i5 processor (3.2 GHz) with 8 GB RAM. Table 6.1.2 shows the schedule we created for 2018-2019 TFF Turkish Super League.

	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15	W16	W17
1 - Galatasaray	11		14		6		7		2		5		13		9		16
2 - Fenerbahçe		3		5		8		17		14		15		4		10	
3 - Başakşehir	18		12	8		5		11		17		14		1		4	
4 - Sivas		1			7		6		10		13		16		17		8
5 - Trabzon	15		10		12		13	8		4		17		7		11	
6 - Göztepe		5		11		14			7		2		3		8		9
7 - Beşiktaş		8		9		11		14		15	3		10		12		2
8 - Kasımpaşa	14		18		10		12		13			1		11		15	
9 - Kayseri	4		15		18		2		3		8		5			14	
10 - Malatya		12		17		9		1		11		6		18	13		3
11 - Akhisar		9	4		14		15		18		12		2		16		17
12 - Bursa	2			13		16		9		1		4		6		18	
13 - Alanya	6		2		3	17		16		9		11		15		7	
14 - Antalya		17		4			10		15		18		12		5		13
15 - Konya		16		1		4		6			10		18		3		12
16 - Rize	10		7		2		3		5	8		9		14		6	
17 - Ankaragücü	7		6		15		18		12		16		8	9		1	
18 - Erzurum		13		16		1		4		6		7			2		5

Table 6.1.2 Proposed schedule for 2018-2019 TFF Super League

Model 1: 2x5 Balanced team groups **Model 2:** 2x4 Balanced team groups **Model 3:** No balanced team groups	**2017-2018**	**2018-2019**	**Model 1**	**Model 2**	**Model 3**
Number of breaks	16	16	16	16	16
No derby matches in first 5 weeks	1	0	0	0	0
Last 3 weeks breaks	2	2	0	0	0
Balanced team constraint violations	32	38	0	0	-
Consecutive matches with strong teams	12	8	0	0	0
Major teams H-A balance	No	No	Yes	Yes	Yes
Strong teams H-A balance	No	No	Yes	Yes	Yes
Proven Optimum	No	No	Yes	Yes	Yes
Time (second)	-	-	25	4	1

Table 6.1.3 The schedule we created compared with the current situations.

Table 6.1.3 shows the difference between the current situation (season 2017-2018 and season 2018-2019) and the schedule we created. Since different balanced team constraints affect the complexity of an instance, we created two different instances, one with 2x4 balanced team groups and one with 2x5 balanced team groups. Table 6.1.4. shows the effectiveness of the hybrid algorithm with different team numbers without balanced team distribution constraint.

	n=12	**n=14**	**n=16**	**n=20**	**n=22**
Number of breaks	10	12	14	18	20
Proven Optimum	Yes	Yes	Yes	Yes	Yes
Time (second)	0.01	0.03	0.5	532	3913

Table 6.1.4 The effectiveness of the hybrid algorithm with different team numbers.

The proposed hybrid method was successfully solved very complex break minimization problems that cannot be solved with exact methods. Moreover, we managed to reach optimality (zero gaps) for our model.

6.2 Future Prospects

In this MS thesis, we studied the break minimization problem in DRR tournaments, which is one of the well-known problem encountered in sports scheduling. The literature on sports scheduling has been extensively studied. We have created a mathematical model for the TFF Turkish Super League intending to obtain a strong schedule. We realized that the size and the complexity of the problem are too large to handle by the existing solution methods efficiently, which force us to seek a new method. After dozens of unsuccessful attempts, we developed a hybrid algorithm based on pattern elimination. We have defined a fine pattern set that limits the solution set to only the minimum break and we quickly eliminated the infeasible pattern sets with the help of the ladder patterns rule. As a result, we managed to solve the problem in seconds.

Our technique for Pattern elimination can be effective not only in sports scheduling but also in the fields of computer and electronics engineering where round-robin process scheduling is used.

BIBLIOGRAPHY

[1] www.atkearney.com/communications-media-technology/article?%2Fa%2Fthe-sports-market (20.05.2019)

[2] A. M. Nevill and R. L. Holder, "Home Advantage in Sport An Overview of Studies on the Advantage of Playing at Home," no. August, 1999.

[3] R. V. Rasmussen, "Hybrid IP/CP Methods for Solving Sports Scheduling Problems," *Building*, no. July, 2006.

[4] R.T. Campbell and D.S. Chen, "A minimum distance basketball-scheduling problem," In R.E. Machol, S.P. Ladany, and D.G. Morrison, editors, Management Science in Sports, vol. 4, pp. 15–26, 1976.

[5] M. Elf, M. J¨unger, and G. Rinaldi, "Minimizing breaks by maximizing cuts," *Operations Research Letters*, vol. 31, pp. 343–349, 2003.

[6] https://www2.deloitte.com/content/dam/Deloitte/ch/Documents/about-deloitte/ch-en-pr-report-deloitte-football-finance-07-06-2018.pdf (20.05.2019)

[7] A. Rosa and W.D. Wallis. Premature sets of 1-factors or how not to schedule round-robin tournaments. *Discrete Applied Mathematics*, 4:291–297, 1982.

[8] D. de Werra, "Geography, games and graphs," Discret. *Appl. Math.*, vol. 2, pp. 327–337, 1980.

[9] D. de Werra, "Scheduling in sports," *Studies on Graphs and Discrete Programming*, pp. 381–395, 1981.

[10] D. de Werra, "Minimizing irregularities in sports schedules using graph theory, " *Discrete Applied Mathematics*, vol. 4, pp. 217–226, 1982.

[11] D. de Werra, "Some models of graphs for scheduling sports competitions," *Discrete Applied Mathematics*, vol 21, pp. 47–65, 1988.

[12] D. de Werra, L. Jacot-Descombes, and P. Masson, "A constrained sports scheduling problem," *Discrete Applied Mathematics*, vol. 26, pp. 41–49, 1990.

[13] J.A.M. Schreuder, “Constructing timetables for sport competitions,” *Mathematical Programming Study*, vol. 13, pp. 58–67, 1980.

[14] R. V. Rasmussen, “Hybrid IP/CP Methods for Solving Sports Scheduling Problems,” *Building*, no. July, 2006.

[15] M.B. Wright, “Timetabling county cricket fixtures using a form of tabu search,” *Journal of the Operational Research Society*, vol.45, no. 7, pp. 758–770, 1994.

[16] J.A.M. Schreuder, “Combinatorial aspects of construction of competition dutch professional football leagues,” *Discrete Applied Mathematics*, vol. 35, pp. 301–312, 1992.

[17] G.L. Nemhauser and M.A. Trick, “Scheduling a major college basketball conference,” *Operations Research*, vol. 46, no. 1, pp. 1–8, 1998.

[18] A. Schaerf, “Scheduling sport tournaments using constraint logic programming,” *Constraints*, vol. 4, pp. 43–65, 1999.

[19] P. Van’t Hof, G. Post, and D. Briskorn, “Constructing fair round robin tournaments with a minimum number of breaks,” *Oper. Res. Lett.*, vol. 38, no. 6, pp. 592–596, 2010

[20] M. Henz, “Constraint-based round robin tournament planning,” *Proceedings of the International Conference on Logic Programming, Las Cruces*, MIT Press, *New Mexico*, pp. 545–557, 1999.

[21] M. Henz, “Scheduling a major college basketball conference – revisited,” *Operations Research*, vol. 49, pp. 163–168, 2001.

[22] J.-C. R´egin, “Minimization of the number of breaks in sports scheduling problems using constraint programming,” *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, vol. 57, pp. 115–130, 2001.

[23] M. Trick, “A schedule-then-break approach to sports timetabling,” *Practice and Theory of Automated Timetabling III*, vol. 2079 of, pp. 242–252, 2001.

[24] R. Miyashiro and T. Matsui. Round-robin tournaments with a small number of breaks. URL citeseer.ist.psu.edu/miyashiro03roundrobin.html. (20.05.2019)

[25] R. Miyashiro, H. Iwasaki, and T. Matsui, “Characterizing feasible pattern sets with a minimum number of breaks,” *Practice and Theory of Automated Timetabling IV*, vol. 2740, pp. 78–99, 2003.

[26] F.D. Croce and D. Oliveri, ”Scheduling the Italian football league: an ILP-based approach,” *Computers and Operations Research*, vol. 33, no. 7, pp. 1963–1974, 2006.

[27] R. V Rasmussen and M. A. Trick, “Discrete Optimization A Benders approach for the constrained minimum break problem,” vol. 177, pp. 198–213, 2007.
[28] R. V Rasmussen and R. V Rasmussen, “D EPARTMENT OF O PERATIONS R ESEARCH Scheduling a triple round robin tournament for the best Danish soccer league Scheduling a triple round robin tournament for the best Danish soccer league,” 2006.
[29] T. Bartsch, A. Drexl, and S. Kr¨oger, “Scheduling the professional soccer leagues of Austria and Germany,” *Computers and Operations Research*, vol. 33, no. 7, pp. 1907–1937, 2006.
[30] M. Carlsson, M. Johansson, and J. Larson, “Scheduling double round-robin tournaments with divisional play using constraint programming,” *Eur. J. Oper. Res.*, vol. 259, no. 3, pp. 1180–1190, 2017.

[31] K. Easton, G. Nemhauser, and M. Trick, “The traveling tournament problem: Description and benchmarks,” Principles and Practice of Constraint Programming, vol. 2239, pp. 580–585, 2001.
[32] B.C. Ball and D.B. Webster, “Optimal scheduling for even-numbered team athletic conferences,” AIIE Transactions, vol. 9, pp. 161–169, 1977.
[33] W.O. Cain, Jr, “The computer-assisted heuristic approach used to schedule the major league baseball clubs,” Optimal Strategies in Sports, Studies in Management Science and Systems, vol. 5, pp. 32–41, 1977.
[34] J.C. Bean and J.R. Birge, “Reducing travelling costs and player fatigue in the national basketball association,” Interfaces, vol. 10, no. 3, pp. 98–102, 1980.
[35] J.A. Ferland and C. Fleurent, “Computer aided scheduling for a sport league,” INFOR, vol. 29, pp. 14–25, 1991.
[36] C. Fleurent and J.A. Ferland, “Allocating games for the NHL using integer programming,” Operations Research, vol. 41, no. 4, pp. 649–654, 1993.
[37] R.A. Russell and J.M.Y. Leung, “Devising a cost effective schedule for a baseball league,” Operations Research, vol. 42, no. 4, pp. 614–625, 1994.
[38] D. Costa, “An evolutionary tabu search algorithm and the NHL scheduling problem,” INFOR, vol. 33, pp. 161–178, 1995.
[39] M.B. Wright, “Scheduling fixtures for basketball New Zealand,” Computers & Operations Research, vol. 33, pp. 1875–1893, 2006.

[40] S. Urrutia and C. C. Ribeiro, "Maximizing Breaks and Bounding Solutions to the Mirrored Traveling Tournament Problem," Discrete Applied Mathematics, vol. 154, no. 13, pp. 1932-1938, 2006.

[41] M. A. Trick, "M. Trick's Guide to Sports Scheduling", http://mat.gsia.cmu.edu/TOURN/ (20.05.2019)

[42] T. Benoist, F. Laburthe, and B. Rottembourg, "Lagrange Relaxation and Constraint Programming Collaborative schemes for Traveling Tournament Problems," 2001.

[43] K. Easton, G. Nemhauser, and M. Trick, "Solving the traveling tournament problem: a combined integer programming and constraint programming approach," Lecture Notes in Computer Science, vol. 2740, pp. 100–109, 2003.

[44] A. Anagnostopoulos, L. Michel, P. Van Hentenryck, and Y. Vergados, "A simulated annealing approach to the traveling tournament problem," In Proceedings CPAIOR'03, Montreal, 2003.

[45] M. Henz. Playing with constraint programming and large neighborhood search for traveling tournaments. In E. Burke and M. Trick, editors, Proceedings PATAT 2004, pages 23–32, 2004.

[46] Rasmussen, R., Trick, "*A Benders approach for constrained minimum break problem*," European Journal of Operational Research, vol. 177, no. 1, pp 198–213, 2007.

[47] P.V. Hentenryck and Y. Vergados, "Traveling tournament scheduling: A systematic evaluation of simulated annealing," Lecture Notes in Computer Science, vol. 3990, pp. 228–243, 2006.

[48] A. Lim, B. Rodrigues, and X. Zhang, *"A simulated annealing and hill-climbing algorithm for the traveling tournament problem*," European Journal of Operational Research, vol. 174, no. 3, pp. 1459–1478, 2006.

[49] D. R. Goossens and F. C. R. Spieksma, "The carry-over effect does not influence football results," no. 2, pp. 1–23, 2012.