

A COMPARATIVE STUDY OF DIFFERENT META-HEURISTIC METHODS FOR SCHEDULING THE CAF CHAMPIONS LEAGUE

By

KELETSO DITHEBE

DR TERRY OLIPHANT

A thesis submitted to
the University of the Witwatersrand
for the degree of
MASTER OF SCIENCE



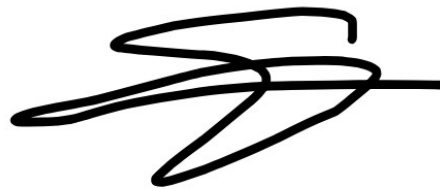
School of Computer Science and Applied Mathematics
Faculty of Science
University of the Witwatersrand
April 2020

DEDICATION

This dissertation/thesis is dedicated to my father, brother, daughter and my future wife.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text.



Keletso Dithebe:

30 July 2020

Date

ACKNOWLEDGMENTS

I would like to acknowledge a number of people who contributed towards the completion of this thesis. Firstly, I thank my supervisor Dr Terry Oliphant for her patience, enthusiasm, guidance and teachings. The skills I have gathered under your wings will always be with me. I'd also like to thank the CSIR for providing me with funding to pursue this research. Thank you to my future wife, and the mother of my child, Keamogetswe Makete for the support and encouragement she provided throughout this research. Many thanks to my father who is my pillar of strength and a source of wisdom whenever I'm in doubt. Lastly thank you to my friends, Kabelo Teffu, Bongani Maphiri, Mabone Kgosiemang, Tebogo Moshori and Karabo Molebatsi, whom I turned to when I needed to get away from the stress of doing an MSc.

ABSTRACT

The Confederation of African Football (CAF) Champions League, is an annual continental football tournament organised by CAF. The CAF Champions League has the capacity to feature 68 African teams, but on average only 57 teams participate in this league. Some of the reasons for teams and their affiliated football associations not participating, include excessive travel costs. In this dissertation, we highlight and tackle two problems which arise when generating fixtures for the CAF Champions League. These include the scheduling of football matches such that the total distance travelled by teams and match officials is minimized. These problems can be formulated as constrained optimization problems, where the constraints are dictated by the Confederation of African Football and other relevant stakeholders. We carry out a comparative study of the performance of three meta-heuristic methods, to solve the constrained problem. The methods under consideration are the Genetic Algorithm, Tabu search and Simulated Annealing. We then compare the results of the best performing meta-heuristic with that of existing CAF Champions League schedules.

Table of Contents

	Page
1 Introduction	1
1.1 Confederation of African Football	1
1.2 Motivation	2
1.2.1 Travel Cost	3
1.2.2 Match Rescheduling and Fatigue	4
1.2.3 Travelling Long Distances	7
1.3 Organisation of the dissertation	9
2 The CAF Champions League	10
2.1 Structure of the CAF Champions League	10
2.1.1 Former structure of the CCL	11
2.1.2 Current structure of the CCL	11
2.2 CAF Five-Year Ranking	12
2.3 CAF-Affiliated Associations	14
2.4 Prize Money	16
3 Review of Sports Scheduling Problems and Solution Techniques	17
3.1 Terminology	17
3.2 Sports scheduling problems	20
3.2.1 Model development	21
3.2.2 Traveling Tournament Problem	22
3.2.3 Balanced Tournament Design (BTD)	22
3.2.4 Minimizing Costs	23
3.2.5 Group-changing and Group-balanced schedules	23
3.3 Sports Scheduling Techniques	24
3.3.1 Decomposition Techniques	24
3.3.2 Integer Programming	25

3.3.3	Constraint Programming	26
3.3.4	Heuristic and Metaheuristics	26
3.4	Sports Scheduling Techniques Applied to Football	28
3.5	Benchmarks	30
4	The CCL Scheduling Problem	32
4.1	Problem description	32
4.2	Mathematical formulation of the Knock-out stage	35
4.2.1	Decision Variables	35
4.2.2	Parameters	36
4.2.3	Objective Function for the knock-out stage of the CCL	41
4.3	Mathematical formulation of the Group stage	43
4.3.1	Decision Variables	43
4.3.2	Parameters	43
4.3.3	Objective function	46
5	Meta-Heuristic Methods for Solving the CCL Scheduling Problem	48
5.1	Simulated Annealing	48
5.1.1	Important Parameters Used In SA	48
5.1.2	Acceptance Function	49
5.1.3	Temperature	50
5.2	Genetic Algorithm	51
5.2.1	Selection Operators	52
5.2.2	Crossover	55
5.2.3	Mutation	60
5.3	Tabu search	62
6	Implementation of Meta-Heuristic Methods for the CCL Scheduling Problem	65
6.1	Overview	65
6.2	Scope and Assumptions	66
6.2.1	Scope	66
6.2.2	Assumptions	67
6.3	Implementation of Simulated Annealing	68
6.3.1	Phase I of SA	68
6.3.2	Phase II of SA	69
6.3.3	Phase III of SA	70
6.4	Implementation of the Genetic Algorithm	72

6.4.1	Phase I of GA	73
6.4.2	Phase II of GA	74
6.4.3	Phase III of GA	75
6.5	Implementation of Tabu Search	78
6.5.1	Phase I of TS	78
6.5.2	Phase II of TS	80
6.5.3	Phase III of TS	81
7	Results	85
7.1	Results and discussion on the performance of SA, GA and TS	85
7.1.1	Solution Quality	86
7.1.2	Speed	89
7.2	Comparison of Existing Schedules and TS Generated	
	Schedules	93
7.2.1	Knock-Out Stage Comparison	93
7.2.2	Group Stage Comparison	98
8	Conclusion	107
8.1	Summary	108
8.2	Further Work	110
8.2.1	Improved Requirements	111
8.2.2	Match Official Criterion	111

List of Tables

1.1	The Egypt Premier League table as at 28/11/2018.	5
1.2	The ABSA PSL table as at 1/02/2017.	6
2.1	Points awarded to each association for the progress of its respective teams in the CCL and CCC competitions, prior to 2017.	13
2.2	Points awarded to each association for the progress of its respective teams in the CCL and CCC competitions, post 2016.	13
2.3	The weights used to rank teams and associations.	14
2.4	The 56 member associations which are able to enter a team into the CCL.	15
2.5	Money awarded for progress made in the competition.	16
4.1	Distance matrix between teams in the preliminary stage of the CCL.	39
4.2	Distance matrix between teams in the first knock-out stage of the CCL.	40
4.3	Distance matrix between teams and associations providing match officials in the preliminary stage of the CCL.	40
4.4	Distance matrix between teams and association providing match officials in the first knock-out stage.	41
4.5	Distance matrix between teams in the group stages of the CCL.	45

4.6	Distance matrix between teams and associations providing match officials in the group stages of the CCL.	46
7.1	The distances generated by SA, GA, and TS, respectively, for the knock-out stages of the CCL.	86
7.2	The combined distances obtained by TS, SA and GA, respectively, for the knock-out stages of the tournament.	87
7.3	The best performing meta-heuristic for the knock-out stages of the tournament. . .	87
7.4	The distances generated by SA, GA, and TS, respectively, for the group stages of the CCL.	88
7.5	The combined distances obtained by TS, SA and GA, respectively, for the group stages of the tournament.	88
7.6	The best performing meta-heuristic for the group stages of the tournament.	89
7.7	A summary of the CPU time taken by SA, GA, and TS, respectively, for the knock-out stages of the CCL.	90
7.8	The fastest meta-heuristic based on data from the knock-out stages of the tournament.	91
7.9	A summary of the CPU time taken by SA, GA, and TS, respectively, using data from the group stages of the CCL.	91
7.10	The fastest meta-heuristic based on data from the group stages of the tournament. .	92
7.11	Existing 2017 Preliminary Stage fixture list.	95
7.12	TS-generated 2017 Preliminary Stage fixture list.	96
7.13	Knock-out stage distance difference between TS and existing fixtures.	98
7.14	2017 Group stage seeding arrangement.	99
7.15	Existing 2017 Group Stage fixture list.	101

7.16 TS-generated 2017 Group Stage fixture list.	102
7.17 Existing Groups for the 2017 edition of the CCL	104
7.18 TS-generated Groups for the 2017 edition of the CCL	105
7.19 Group stage distance difference between TS and existing fixtures.	106

Chapter One

Introduction

The aim of this chapter is to provide a brief introduction to the work contained in this dissertation. Section 1.1 provides a brief introduction of the Confederation of African Football and some of the issues associated with their schedules. In Section 1.2, we provide motivation for scheduling CAF Champions League matches more efficiently. In Section 1.3, we outline the layout of the rest of the thesis.

1.1 Confederation of African Football

Football is the most popular sport played across Africa. The Confederation of African Football (CAF) is the continent's administrative and controlling body for professional football in Africa. CAF is responsible for promoting and developing the game of football in Africa, and increasing its popularity among Africans. This is done by hosting continental, national, and club competitions. CAF controls media rights (or access) to these events in order to optimize revenue and exposure.

The CAF Champions League (CCL) is the premier continental club competition organised by CAF. It is an annual international club football tournament arranged for the best teams in Africa. The competition is open to winners of all CAF-affiliated national leagues, as well as the winner of the competition from the previous year. In 2004, CAF decided to increase the representation of the

12 highest ranked countries¹, by allowing the runner up of these national leagues to take part in the competition as well. This decision saw 68 teams competing for the best African football team title. The winner of the CCL is also afforded the opportunity to represent Africa in the FIFA Club World Cup, an international club football tournament contested between the champion clubs from all six continental federations.

In terms of logistics, CAF assigns an Inter-clubs Organising Committee (IOC) to establish all fixtures for the CCL. The committee is responsible for the organization and supervision of all inter-club competitions, in accordance with the regulations which govern them. The organising committee is required to take into account, insofar as is possible, the geographic, economic and sports conditions within the football clubs respective countries, when establishing fixtures [6].

Some coaches, however, have expressed disappointment in the fixture lists produced by the IOC, as they have had to travel to remote locations in distant countries. Furthermore, most African football teams cannot afford the luxury of private planes. Teams usually travel using commercial flights and may be required to take connecting flights in other continents such as Europe or Asia in order to make it to the assigned venues. We now provide motivation for solving the travel distance issue.

1.2 Motivation

We briefly mentioned some reasons why the current method of scheduling CCL fixtures, as described in the regulation documents [6], maybe be inefficient. We offer further motivation for solving this problem.

¹CAF ranks countries according to the performance of their respective clubs in the previous 5 CAF Champions League tournaments.

1.2.1 Travel Cost

For most, if not all teams, the stress of costs and travel expenses associated with the CCL is overwhelming. This is because Africa is notoriously known for being an expensive continent to tour. For example, the price of a ticket for a 4-hour flight from Johannesburg, South Africa, to Luanda, Angola, costs more than a 10-hour flight from Johannesburg to London [32].

When Kaizer Chiefs, a soccer team based in Johannesburg, had to travel to Moroni in the Comoros Islands, they had to stop over at either Madagascar or Mauritius, because there are no direct flights to Moroni [32]. In an attempt to avoid too many days in the Comoros Islands, the club decided to charter a plane. This cost the team R750 000, which cut their budget of R1 500 000, granted by the ABSA Premier Soccer League (PSL), in half. The ABSA PSL grants R1 500 000 to all South African football clubs competing in CAF related tournaments. The Comoros based team, on the other hand, could not afford to make the trip to South Africa for the second leg fixture against Kaizer Chiefs. Hence, Kaizer Chiefs won the match by *walkover* and got an easy passage through to the next round. A *walkover* happens when one team withdraws from the competition, in which case the opposing team automatically qualifies for the next round of the competition. In our study, which looks at the last four years of the CCL, we have observed that a handful of teams win their preliminary or first knock-out round matches by *walkover*. *Walkovers* in other continental competitions such as the UEFA Champions League, the European equivalent of the CCL, are not as prevalent as they are in the CCL.

Ajax Cape Town did not want to compete at all in the CAF Confederation Cup (CCC)², but they eventually did travel to Angola to partake in the first round. This was after the South African Football Association (SAFA) insisted that they partake in the tournament. Despite their best efforts to keep expenses low, Ajax's journey into Angola consisted of a flight with Air Namibia which

²The CCC is a second tier football competition, which ranks below the CCL, and is also organised by CAF.

cost the team R215 000. Visas and accommodation costs for the team totalled at R410 000. An additional non-refundable expense of R140 000 was incurred for an extra day at the hotel which Ajax had to forfeit to get home early. For the second leg in Cape Town, Ajax paid R134 000 for flights for the match officials from the Seychelles, and spent R62 000 in allowances for match officials and commissioners. The total cost for this expedition was roughly R960 000 [32].

1.2.2 Match Rescheduling and Fatigue

CAF requires priority, over other national tournaments, when football associations and their respective teams participate in CAF organised tournaments. This is done to ensure that games organized by CAF take precedence in the event of clashes. These clashes can occur between CAF matches and other regional/sub-regional games, or matches which are organised on special occasions [6]. This has consequences for teams as they are often required to reschedule their national league and tournament matches. Al-Ahly of Egypt and Mamelodi Sundowns of South Africa are just two examples of teams who have reached the finals of the CCL in recent years. In doing so, these teams had to reschedule over 5 domestic league matches so as to participate in the CCL. Table 1.1 and Table 1.2 are league tables of the Egypt Premier League table as well as the ABSA PSL table.

Table 1.1, shows Al-Ahlys low position in the Egypt Premier League, after reaching the finals of the 2018 edition of the CCL. Their low position is due to the rescheduling of national league matches, as shown by the column MP, in Table 1.1. The column shows that Al-Ahly had only played seven matches out of 16. This meant that Al-Ahly had to then play every three days in order to catch up with other teams.

Egypt Premier League Table						
Position	Team	MP	Wins	Draws	Losses	Points
1	Zamalek	13	10	2	1	32
2	Misr Elmaqasah	15	8	2	5	26
3	Pyramids	12	6	5	1	23
4	El-Entag El-Harby	16	5	8	3	23
5	Arab Contractors	14	6	2	6	20
6	Smouha	14	4	8	2	20
7	Al Ittihad	16	4	7	5	19
8	El Gaish	15	4	6	5	18
9	Haras El Hodood	16	4	6	6	18
10	El Daklyeh	14	4	4	6	16
11	Nogoom FC	14	4	4	6	16
12	Wadi Degla	16	4	4	8	16
13	El Gounah	15	3	7	5	16
14	Petrojet	15	2	9	4	15
15	Enppi	14	3	5	6	14
16	El Masry	11	2	7	2	13
17	El Ismaily	13	3	4	6	13
18	Al Ahly	7	3	2	2	11

Table 1.1 The Egypt Premier League table as at 28/11/2018.

ABSA Premier League Table						
Position	Team	MP	Wins	Draws	Losses	Points
1	SuperSport United	14	6	7	1	25
2	BidVest Wits	13	7	4	2	25
3	Cape Town City	15	7	4	4	25
4	Kaizer Chiefs	15	6	6	3	24
5	Polokwane City	15	5	6	4	21
6	Golden Arrows	15	5	5	5	20
7	Mamelodi Sundowns	9	6	1	2	19
8	Maritzburg United	15	4	7	4	19
9	Ajax Cape Town	15	5	3	7	18
10	Orlando Pirates	13	4	6	3	18
11	Chippa United	14	4	4	6	16
12	Bloemfontein Celtic	14	3	6	5	15
13	Platinum Stars	14	3	5	6	14
14	Free State Stars	15	3	5	7	14
15	Highlands Park	15	2	7	6	13
16	Baroka FC	15	2	6	7	12

Table 1.2 The ABSA PSL table as at 1/02/2017.

Mamelodi Sundowns is another instance of a team who had to reschedule domestic football matches, due to reaching the latter stages of the CCL. Mamelodi Sundowns were considered by many as the best South African football team in recent years. They proved this by winning the CCL in 2016. However, after achieving the CCL title, Mamelodi Sundowns were six games behind in their domestic league, as per column MP in Table 1.2. As such, the team had to make up for the rescheduled games, which was not easy because injuries and fatigue were some of the challenges which they had to face.

1.2.3 Travelling Long Distances

Home advantage is the phenomenon where sports teams perform better at their *home* ground than away from *home*. Its existence is prevalent in a wide range of team sports including association football [37]. Although there is a general consensus, in the existing literature, around the prevalence of *home advantage*, there is some disagreement around the influence and magnitude of *travel-fatigue*. *Travel-fatigue* is tiredness caused by travelling long distances.

Courneya and Carron [27] published a literature review in which they concluded, after extensive research, that the effects of travel on *home advantage* were minimal. Contrary to these findings, Snyder and Purdy [12] revealed that teams won 58.8% of their matches when the visiting team travelled less than 200 miles (322 km). They showed that teams won 84.6% of their matches when the visiting team travelled more than 200 miles [12]. Pollard [39] found evidence that showed that *home advantage* in European Cup (present day UEFA Champions League) games was higher than in national football league matches. This could be a result of the obvious difference in distances between teams in national and international leagues. Stefani and Clarke [42] supported Pollard's [39] findings, and stated that teams had a 15% greater chance of winning their *home* games in European cup matches than in national league matches [42]. They concluded that the shorter distance travelled in national league matches could possibly be a contributing factor for the

performance of teams [42].

Clarke and Norman [44] were the first to find a significant correlation between *home advantage* and distance travelled by the visiting team. They argue that the effect of travel distance on the visiting team is particularly prevalent for geographically isolated teams like Plymouth, Exeter and Carlisle. Goddard [22] had similar findings. He found that geographical distance had a significant influence on matches. Asimakopoulos and Goddard [19] found a positive and significant correlation between distance travelled and *home* team advantage.

Pollard et al [38] found that, after controlling for team ability, visiting teams concede about 0.115 goals per 1 000 km travelled. They state that the largest distance between two teams in Brazil is 4 172 km. When they looked at individual team results, they found that Paysandú FC, who are based in Belem, performs best at *home*. A reason for this is that Paysandú FC is based in the most remote location in Belem, with its nearest neighbour, Fortaleza FC, being around 1 000 km away.

One could argue that distance travelled, and the effects thereof are balanced out when considering a two match fixture, where each match is played at a *home* venue of the two participants. However, this may not be the case for CCL participants. These teams are often required to travel over 5 000 km to an opponent on the same continent, and have to travel back the same distance to play a domestic league match. For these teams, one can argue that they are consistently disadvantaged due to travel distances.

We have provided motivation for our research, which includes issues with the current method being used to schedule CCL matches. Furthermore, nothing exists in the literature, on scheduling fixtures for the CCL, and it is not trivial to adapt the existing literature on sports scheduling to our problem. We thus propose the use of heuristic methods to schedule matches, and assign associations. This will be done, such that the distance travelled by teams and match officials is minimised.

1.3 Organisation of the dissertation

The remainder of the dissertation is organised as follows. In Chapter 2, we describe important aspects of the CCL. Chapter 3 contains a literature review on different sports scheduling problems and solution techniques. Chapter 4 outlines the CCL problem. In Chapter 5, we discuss the three meta-heuristic methods which we have selected to solve the CCL problem. We then discuss our implementation of these methods in Chapter 6. In Chapter 7, we present numerical results for the CCL problem, and lastly, some concluding remarks are made in Chapter 8.

Chapter Two

The CAF Champions League

In this chapter, we describe the structure of the CCL in Section 2.1. We discuss the five-year ranking system used by CAF in Section 2.2. We formally introduce the associations affiliated with CAF, as well as associations which have FIFA-approved match officials, in Section 2.3. Lastly, we present a discussion on the prize money awarded to teams, which progress in the CCL, in Section 2.4.

2.1 Structure of the CAF Champions League

The CCL is a prestigious continental tournament, and is open to winners of all CAF-affiliated national leagues, as well the previous winner of the competition. The CCL usually commences in February and ends in November of every year. However, changes to the competition were made to ensure that the CCL runs parallel with most of the major African national leagues, as well as European club competitions. As a result, the tournament will now commence in September and end in May of the following year. This will be beneficial to most participants who were previously not afforded an opportunity to rest after competing in their respective national leagues. We also note that in addition to the calendar changes, the structure of the CCL has changed as well. We now describe the former and current structures of the CCL.

2.1.1 Former structure of the CCL

In the past, the first phase of the CCL was divided into three knock-out stages. These were the preliminary stage, the first knock-out stage, and the second knock-out stage, respectively. The number of participants in the preliminary stages depended on the number of teams which chose to participate in the tournament, as well as the number of teams which were granted a *bye* by the IOC. A *bye* is an automatic qualification for the team to advance to the next round of the tournament. The first knock-out stage of the CCL, had 32 participants, while the second knock-out stage comprised of 16 teams. Each knock-out stage was a *two-legged-tie*, which means that a contest between two teams comprises two matches, with each team hosting the opposing team in one match. Here, the team which scored a higher aggregate number of goals in the two matches qualified for the next stage of the tournament.

The second phase of the competition comprised of eight teams which were grouped into two pools of four teams. Each group was self-contained, in that it held its own *double round-robin (2RR)* tournament. A 2RR is a tournament where every pair of opponents meets twice; once at each opponents *home* venue. Here, the top two teams from each group qualify for the semi-finals of the competition.

2.1.2 Current structure of the CCL

In the current structure of the CCL, the first phase is divided into two knock-out stages, as opposed to three. The two knock-out stages are the preliminary stage, and the first knock-out stage, respectively. The number of participants in the preliminary stages is determined the same way as that used in the former structure of the CCL, while the first knock-out stage has 32 participants. Each knock-out stage is a *two-legged-tie*. Again, the team with the highest aggregate number of goals qualifies for the next stage.

The second phase of the competition comprises of 16 teams, which are divided into four groups. Each group comprises of four teams. Teams are initially ranked according to their past performance or strength. The four strongest teams, according to their ranking, head each group. Four second tier teams are allocated to each group, as well as four third and fourth tier teams are allocated respectively, until all four groups are filled. Each group is self-contained, in that it holds its 2RR tournament. The top two teams from each group qualify for the quarter-finals.

The difference between the former and current structures of the CCL is that the former structure has a second knock-out round and features eight teams in the group stages, whereas the current structure does not have a second knock-out round, and the group stages comprise of 16 teams. In this research, we use the current structure of the CCL.

We now discuss the CAF five-year ranking system.

2.2 CAF Five-Year Ranking

The CAF five-year ranking system determines the number of teams which an association may enter into the CCL. Currently, only the 12 highest ranked member associations get to enter two teams into the CCL, while the other remaining associations are only given one spot. The ranking system calculates each associations points, according to the past performance of their football clubs in CAF club tournaments over the last five years, and assigns a weighting factor accordingly. The ranking system is also used to place teams in their respective strength groups or tiers, as discussed in Section 2.1.

We now present a summary of the ranking criteria used in the CCL, and the CCC. Given the structural changes in 2017, we present two separate tables which capture the discrepancies. The ranking criteria used before the structural change to the competition, can be found in Table 2.1:

Points Awarded		
Position	CAF Champions League	CAF Confederation Cup
Winner	5 Points	4 Points
Runner-up	4 Points	3 Points
Losing Semifinalist	3 Points	2 Points
3rd Place in Groups	2 Points	1 Points
4th Place in Groups	1 Points	1 Points

Table 2.1 Points awarded to each association for the progress of its respective teams in the CCL and CCC competitions, prior to 2017.

The current criteria for ranking associations is tabulated in Table 2.2:

Points Awarded		
Position	CAF Champions League	CAF Confederation Cup
Winner	6 Points	5 Points
Runner-up	5 Points	4 Points
Losing Semifinalist	4 Points	3 Points
Losing quarter-finalist	3 Points	2 Points
3rd Place in Groups	2 Points	1 Points
4rd Place in Groups	1 Points	0.5 Points

Table 2.2 Points awarded to each association for the progress of its respective teams in the CCL and CCC competitions, post 2016.

The points awarded in Table 2.2 are then weighted according to the number of years which have lapsed since the achievement, as shown in Table 2.3. These weightings were valid for the 2018 edition of the CCL.

Weights awarded	
Year	Weight
2017	5
2016	4
2015	3
2014	2
2013	1

Table 2.3 The weights used to rank teams and associations.

We now discuss CAF-affiliated associations and present the associations which are able to provide match officials.

2.3 CAF-Affiliated Associations

CAF currently has 56 member associations which are able to enter a team into the CCL. The 12 highest ranked associations, according to the CAF five-year ranking system, are given an additional spot in the competition. The additional spot is given to the CAF-affiliated national league runner-up. Therefore, a maximum of 68 teams could enter the CCL, however, this level of participation has never occurred.

Table 2.4 presents a list of all member associations, obtained from the official CAF website [5]. Associations which are marked with an asterisks represent countries with FIFA-approved match officials. These are the associations which provide referees for matches throughout the tournament.

There are only 51 associations which are able to provide match officials.

Country List		
Algeria *	Angola *	Benin *
Botswana *	Burkina Faso *	Burundi *
Cameroon *	Cape Verde *	Central African Republic *
Chad *	Comoros *	Congo *
Cote d'Ivoire *	Djibouti *	Egypt *
Equatorial Guinea *	Eritrea *	Ethiopia *
Gabon *	Gambia *	Ghana *
Guinea *	Guinea-Bissau *	Kenya *
Lesotho *	Liberia *	Libya *
Madagascar *	Malawi	Mali *
Mauritania *	Mauritius *	Morocco *
Mozambique *	Namibia *	Niger *
Nigeria *	RD Congo *	Reunion Island
Rwanda *	Sao Tome e Principe	Senegal *
Seychelles *	Sierra Leone *	Somalia *
South Africa *	South Sudan	Sudan *
Swaziland *	Tanzania *	Togo *
Tunisia *	Uganda *	Zambia *
Zanzibar	Zimbabwe *	

Table 2.4 The 56 member associations which are able to enter a team into the CCL.

We now present a brief discussion on the prize money awarded to teams, which depends on how far along a team has progressed in the CCL.

2.4 Prize Money

CAF first introduced prize money for the eight participants in the group stages of club competitions in 1997. Since then, the prize money has hardly risen enough to ease the financial burden on teams. In 2016, CAF announced TOTAL as the new official sponsor of African football. Details of the sponsorship were not revealed until recently. Table 2.5 below presents prize monies awarded over the past 22 years, as well as the projected amount for 2020.

CAF Champions League Prize Money			
Position	1997-2008	2009-2016	2017-2020
Winner	\$1 000 000	\$1 500 000	\$2 500 000
Runner-up	\$750 000	\$1 000 000	\$1 250 000
Semi-finalist	\$427 500	\$700 000	\$875 000
Quarter-finalist	Not Applicable	Not Applicable	\$650 000
3rd Place in Group	\$261 250	\$500 000	\$550 000
4rd Place in Group	\$190 000	\$400 000	\$550 000
Total Amount	\$2 628 750	\$4 100 000	\$6 375 000

Table 2.5 Money awarded for progress made in the competition.

As per Table 2.5, prize monies are only distributed once the group stages of the CCL have concluded. This means that each team participating in the competition has to cover their own travel costs during the tournament, as well as make travel arrangements and provide allowances for match officials who referee their respective *home* matches. For this, and other reasons mentioned in Section 1.2, we aim to reduce travel distances for teams and match officials. The obvious alleviation of the financial burden on teams is one of our main objectives.

Chapter Three

Review of Sports Scheduling Problems and Solution Techniques

In this chapter, we introduce some terminology which is vital to the discussion in Section 3.1. We discuss sports scheduling problems in Section 3.2, and list some sports scheduling techniques in Section 3.3. We then discuss football-related sports scheduling problems, and the techniques used to solve them in Section 3.4. In Section 3.5, we provide the reader with available benchmarks for various sports scheduling problems.

3.1 Terminology

Sports scheduling has generated interest from multiple disciplines such as operations research, scheduling theory, constraint programming (CP), graph theory, combinatorial optimisation, and applied mathematics [43]. In an annotated bibliography by Kendall et al. [43], 160 papers on sports scheduling were reviewed. The annotated bibliography combines sports scheduling papers, highlights the different types of sports scheduling problems and lists the scheduling techniques used to solve them.

The terminology used in Kendall et al. [43] conforms to that used in *Round-robin scheduling*; a survey by Rasmussen and Trick [40]. Rasmussen and Trick noted how inconsistent the terminology was in existing literature and thus presented a unified list which shall be used here, and adapt in this dissertation.

- *Venue, Home Games and Away Games*: Teams often have a *home* venue. A team plays *home* games at their *home* venue and *away* games at an opponent's venue.
- *Trip or Home Stand*: A sequence of consecutive *away games* is called a *trip*, while a sequence of consecutive *home games* is called a *home stand*.
- *Round-Robin (RR)*: A round-robin tournament is a tournament where each team meets every other team for a fixed number of times (a *1RR* refers to a single round-robin).
- *Double Round-Robin (2RR)*: This is a *RR* tournament where every pair of opponents meet twice. It is often required that the two games between every pair of opponents occurs at each teams respective *home* venue.
- *Slots*: When scheduling a tournament, the games must be allocated to a number of timeslots, called *slots*, in such a way that each team plays at most one game in each *slot*.
- *Timetable Set (T Set)*: A *T set* is the allocation of games to *slots*. Each row t of the timetable corresponds to a team, while the column s corresponds to a *slot*. Each entry is the opposition of team t in *slot* s .
- *Home and Away Pattern Set (HAP Set)*: A *HAP set* is the sequence of *home* and *away games* according to which a team plays during the tournament. Each entry in the *HAP set* contains either an H representing a *home game* or an A representing an *away game*. Alternatively a 1 or 0 could be used.
- *Schedule*: The superimposition of a *HAP set* and a corresponding *T set* constitutes a *schedule*.

- *Mirrored Schedule*: A *mirrored schedule* is when the first and the second half *RR* are identical, except that the *home* venues are reversed.
- *Compact or Relaxed Schedule*: The tournament is said to be *compact* when $(n - 1)$ *slots* are required to schedule a *IRR* for an even number of n teams. A *schedule* is *relaxed* when more *slots* are available.
- *Balanced Schedule*: For a *IRR*, the difference between the number of *home games* and *away games* played by each team is no more than one. A *2RR* is *balanced* if it is required that two games between every pair of opponents occur in opposite venues.
- *Break*: A tournament with an alternating pattern of *home* and *away games* is considered attractive. A *break* from (or alternative to) this corresponds to two consecutive *slots* of *home* or *away games*.
- *Irreducible Schedule*: A *schedule* for a *IRR* is irreducible when at most one team in each *slot* has a *break*.
- *Tour*: On a schedule, an entire row defines a *tour* for the corresponding team.
- *Complementary Patterns*: Two row vectors in a *HAP set* are said to be *complementary* if the first pattern has an *away game* when the second pattern has a *home game*, and vice versa.
- *Equitable Pattern Set*: An *equitable pattern set* is defined as teams having an equal number of *breaks*.
- *Feasible Pattern Set*: A *feasible pattern set* is a *HAP set* for which a corresponding *T set* exists.

Given that *RR* tournaments are compatible with graph theory, we also introduce a few graph theory concepts:

- *Graph (G)*: A graph $G = (V, E)$ where V is a finite set of vertices and E is the set of edges in G .
- *A matching in G*: A matching in G is a set of independent edges (non-adjacent edges).
- *A Perfect Matching in G*: A *perfect matching in G* is a matching in which all the vertices in V are incident to an edge.
- *One-Factor Graph*: The graph induced by a perfect matching is a one-regular graph (for all nodes the degree is 1, $\deg = 1$), called a *one-factor graph*.
- *One-Factorisation*: *One-factorisation* is the partitioning of a graph G into *one-factor graphs*.

Now that we have presented an exhaustive list of all the relevant terminology, we proceed with our discussion on sports scheduling.

3.2 Sports scheduling problems

Scheduling problems in sports leagues may be divided into two main classes, i.e., *temporally constrained* and *temporally relaxed* problems [43]. *Temporally constrained* scheduling problems consist of a minimum number of *slots* in which games can be scheduled. Hence, each team has to play exactly one game in each *slot*. Leagues following this pattern are said to be *compact*. *Temporally relaxed* scheduling problems generally have a higher number of *slots* than the minimum number of *slots* needed. Thus not every team necessarily has to play in each *round* or *slot*. This means that teams may have *slots* in which they do not have a scheduled game. We note that the words *slot* and *round* are used interchangeably within this dissertation.

Kendall et al. [43] formulates a *temporally constrained* scheduling problem as follows:

The league consists of an even number n of different teams indexed by $i \in 1, \dots, n$, where each team has to play against another team exactly $l \geq 1$ times. The number of slots available to schedule these $\binom{n}{2}l = \frac{n(n-1)l}{2}$ games is equal to $(n-1)l$, where each team has to play exactly one game in each round. Thus, one needs to determine which teams $i, j \in 1, \dots, n$ play against each other in each round $t = 1, \dots, (n-1)l$, and for each of these pairings, one also needs to determine whether the game is played in the home stadium of team i (home game for team i) or in the home stadium of team j (away game for team i).

In most cases, $l = 1$, representing a single *RR* tournament (SRRT) or $l = 2$, representing a double *RR* tournament (DRRT). We now highlight the most common types of scheduling problems.

3.2.1 Model development

Sports scheduling problems commonly consist of a set of constraints and an objective which needs to be achieved [43]. Most real-world sports scheduling problems appear to be similar, but contain differences. These differences can usually be found in the tailored constraints or objectives. For instance, the Belgian and German football leagues both consist of 18 teams in a mirrored *2RR*, however, they differ in their hard constraints. The Belgian league requires that teams do not have more than k consecutive *home* games, while the German league has no such constraint. A list of various other real world scheduling problems and their constraints can be found in [28].

Kendall et al. [43] observed that some problems without real world applications have been introduced to generate the interest of researchers. One such problem is the Traveling Tournament Problem (TTP), which offers a benchmark of instances for comparison. A second instance of such a problem is the Balance Tournament Design (BTD) problem. We begin by presenting a brief discussion on the TTP.

3.2.2 Traveling Tournament Problem

Easton et al. [14] formulated the travelling tournament problem when they investigated distances travelled by major league baseball teams in the United States. The problem is based on the assumption that teams go on road trips. A road trip means that a team leaves its home city and plays multiple games away from their home city before returning back home. Therefore, the aim of the problem is to minimize the total distance travelled by all teams. In the formulation of the problem, the authors restrict the minimum and maximum number of back to back games that can be played at *home* or played *away*. Furthermore, there are no repeat fixtures i.e., two teams cannot face each other in two consecutive *rounds*.

We now present a brief discussion on the Balanced Tournament Design problem

3.2.3 Balanced Tournament Design (BTD)

A balanced tournament design (BTD(n)) of order n is a schedule with n courts (or venues) and $2n$ teams (or players). The $2n$ teams (or players) are required to play in a *IRR* tournament in $2n - 1$ rounds. This is done such that no team plays more than twice at any given court [43]. It is for this reason that venues are regarded as an essential requirement when establishing a schedule within BTD problems. A slight variation of the BTD problem requires that the kick-off times for teams or referee assignments, are balanced over a season (as opposed to venues) [43].

Factored balanced tournament design FBTD(n) and the partitioned balanced tournament design PBTD(n) are variants of BTD. FBTD(n) requires each team to play in all assigned venues in the first n slots of the tournament. PBTD(n), however is a FBTD(n) with the additional requirement that each team plays once at each venue in the last n rounds. BTD and its variants are applicable in scheduling tennis tournaments such that the effects of playing in different courts are balanced. Dinitz and Dinitz [21] enumerate the designs for BTD(5), FBTD(5) and PBTD(5).

We now briefly discuss another important consideration in sports, which is to minimise costs.

3.2.4 Minimizing Costs

Cost (or benefits) are a strong driving force in any sport. For example, when team i is scheduled to play against team j at team i 's venue, in *round* r of a tournament, then a cost c_{ijr} is incurred. Hence, the objective in these problems is to find a feasible *IRR* or *2RR* schedule satisfying various constraints with the total cost minimised [43]. Briskorn [1] presents a branch-and-price algorithm which finds a feasible schedule for a *IRR* tournament with minimum breaks and total costs.

We now discuss one of the most overlooked aspects in sports; group changing and group balanced schedules.

3.2.5 Group-changing and Group-balanced schedules

If a given sports league is partitioned into $g \geq 2$ different strength groups, then the objective is to find a schedule which prevents teams from playing against extremely weak or extremely strong teams in consecutive *rounds*.

Kendall et al. [43] introduces two concepts to ensure some degree of group fairness. A *IRR* tournament is said to be *group changing* if no team plays back to back fixtures with an opposition from the same group g . A schedule is *group balanced* if no team plays more than once against teams of the same group within g consecutive games. Therefore, a team should play against opponents from every other group in consecutive *slots*. Briskorn and Knust [8] construct *group balanced* schedules based on graph models for an odd number of teams. They also construct *group changing* schedules for an even number of teams with $g = 3$ groups. Additional sports scheduling problems, can be found in the *Classification of literature on sports scheduling* [45] library.

Here, we present how some sports scheduling problems have been solved.

3.3 Sports Scheduling Techniques

Certain scheduling techniques work well for certain problems, and not so well for others. Therefore, a number of scheduling techniques exist. The most common approaches to scheduling are decomposition, integer programming, constraint programming, and meta-heuristics [43]. We outline a few of these below.

3.3.1 Decomposition Techniques

Decomposition is often used in sports scheduling problems. It entails separating a problem into sub-problems and sequentially solving each sub-problem using exact or heuristic methods. The following two approaches have been used extensively [43]:

- **First-schedule, then-break:** Here, pairs of teams are determined first i.e., within each *round*, pairs of opponents are determined. Thereafter a corresponding home-away pattern or *HAP set* with a minimum number of breaks is calculated [43].
- **First-break, then-schedule:** Here a feasible *HAP set* with a minimum number of breaks is determined. Thereafter pairs of teams for the corresponding pattern are calculated.

Trick [34] presents a first-schedule, then-break approach to scheduling sports timetables for leagues with up to 20 teams. Trick [34] notes that the second phase is much more difficult than the first.

Schreuder [26] uses the first-break, then-schedule approach to construct the real world Dutch professional football league schedule. Schreuder [26] develops a *HAP set* using oriented edge colourings of complete graphs, and then assigns the clubs to a *HAP set* with a clustering algorithm.

Even though decomposition techniques are effective, they are often used in conjunction with integer programming (IP).

3.3.2 Integer Programming

Integer programming has been used to tackle theoretical as well as real world scheduling problems [43]. Decomposition approaches often use IP methods to solve their sub-problems [43].

Kendall et al. [43] defines a basic IP model that can be used to schedule a *2RR* as follows:

Given n teams and r rounds

$$x_{i,j,t} = \begin{cases} 1, & \text{if team } i \text{ plays at } \textit{home} \text{ against team } j \text{ in round } t, \\ 0, & \text{otherwise,} \end{cases}$$

where $i = 1, \dots, n, j = 1, \dots, n$, (with $i \neq j$) and rounds $t = 1, \dots, r$. The constraints for a *2RR* tournament may be formulated as:

$$\begin{aligned} \sum_{t=1}^r x_{i,j,t} &= 1, & \forall 1 \leq i, j \leq n, \quad i \neq j, \\ \sum_{j=1}^n x_{i,j,t} &= 1, & \forall 1 \leq i \leq n, \quad 1 \leq t \leq r. \end{aligned}$$

The first constraint requires that every team i must play a *home* game against every other team, in each round. The second constraint states that team i must not play more than once in each *round*. Branch and Bound, Branch and Cut, Benders Decomposition, and Column Generation are some of the IP methods used in sports scheduling literature [43].

Easten et al. [13] uses a parallel implementation of a branch and price algorithm which implements IP and constraint programming (CP) to solve the TTP.

3.3.3 Constraint Programming

Kendall et al. [43] states that for CP techniques to be applicable to a problem, inter-related sets of variables and constraints must be used to model the optimisation problem. CP fixes variable values and derives new constraints through the process of eliminating possible values. This is achieved through logical conclusions [16].

Regin [25] presents CP approaches for the break minimisation problem in *IRR* tournaments. Regin [25] asserts that solutions for the problem are produced in 0.61 seconds when determining a schedule with $n - 2$ breaks for $n = 20$ teams, and less than a minute for $n = 60$ teams.

We now discuss two of the more popular solution techniques in sports scheduling, which are heuristic and meta-heuristic techniques.

3.3.4 Heuristic and Metaheuristics

Heuristic methods are approximate solution techniques, and have been used as an alternative to exact methods, which are not always applicable. Heuristic methods are increasingly used for solving, among others, NP-hard combinatorial problems.

Heuristics, are low-level search methods used to solve large and complex problems. They search local neighbourhoods by modifying the current solution and then replacing it with a better solution [43]. Heuristics generally find suboptimal solutions, but do so in a reasonable amount of computational time. A key advantage of heuristics, is their ability to be easily tailored to fit any (or almost any) problem. This is why they are so widely used.

The most popular heuristic approach is based on Local Search (LS) improvement techniques. LS can be summarized as an iterative search procedure which, starting from an initial feasible

solution, progressively improves on this solution by applying a series of local modifications. At each iteration, the search moves to an improved feasible solution which differs only slightly from the current one. The search terminates when a local optimum has been located. Often this local solution is of inferior quality [30].

Like heuristic methods, meta-heuristics apply heuristic neighbourhood search strategies. However, meta-heuristics employ local neighbourhood moves based on higher level strategies [16]. The high-level search procedure directs the low-level heuristic to escape a local optimum and thus conduct a larger exploration of the feasible region. The larger exploration often results in meta-heuristics finding better solutions than their counterparts, which use a local neighbourhood search.

In 1983, Kirkpatrick et. al [29] proposed a meta-heuristic approach called Simulated Annealing (SA), which could converge to an optimal solution of a combinatorial problem. The introduction of SA meant that one could look for other ways to tackle combinatorial optimisation problems, which sparked the interest of researchers. We discuss SA further in Section 5.1. Many different approaches were proposed in subsequent years, many of which were based on simulating natural phenomena.

Descriptions and classifications of popular meta-heuristics methods used to solve numerous scheduling problems can be found in Kyngäs [24]. Kyngäs [24] classifies and describes methods such as ant colony optimisation, genetic algorithms, co-operative local search, SA, ejection chains, memetic algorithms, hyper-heuristics, particle swarm optimisation, tabu search, and various other heuristic methods.

Di Gaspero and Schaerf [18] propose a family of tabu search solvers for the TTP. Di Gaspero and Schaerf [18] use complex combinations of neighbourhood structures. The varying neighbourhoods

are comprehensively analysed and experimentally compared. The solvers are assessed on three sets of available benchmarks and their outcomes are compared with previous results presented in Trick [35].

Kendall et al. [43] discovered that meta-heuristics benefit from the use of greedy randomised constructive heuristics. This is because greedy heuristics produce good initial solutions [43]. Kendall et al. [43] also highlight that the hybridisation of principles from different meta-heuristics can yield good results.

Costa [9] evaluates the hybridisation of genetic algorithms and tabu search in solving combinatorial optimisation problems. Costa [9] uses the National Hockey League as an example to demonstrate the power of the approach.

Other sports scheduling techniques such as Lagrange relaxation and SAT formulation can be found in the *Classification of literature on sports scheduling* library [45].

Kendall et al. [43] identify articles with scheduling problems in sports such as football, ice hockey, table tennis, referee assignment, cricket, baseball, tennis, basketball, and golf pairing. We now outline how scheduling techniques have been applied to football.

3.4 Sports Scheduling Techniques Applied to Football

Numerous exact and approximate optimisation methods have been applied to football scheduling problems [43]. We now present a few of these methods.

A semi-greedy heuristic algorithm, as well as a branch-and-bound algorithm are used by Bartsch et al. [3] to schedule the real-world German and Austrian Football Leagues. These methods

are compared with the semi-greedy algorithm which is preferred by both German and Austrian Football Associations. Bartsch et al. [3] worked with league authorities from each league and scheduled the Australian football League six times, and the German football league once, using the methods which they had developed.

Della Croce and Oliveri [7] attempted to schedule the Italian Football League. Their proposed procedure is divided into three phases, all of which are solved using IP. The first phase generates a *HAP set*, as per the requirements made by broadcasters, as well as several other constraints. A feasible *T set* which is compatible with the *HAP set* is found in the second phase. Lastly, the schedule is generated by assigning teams to the timetable. Della Croce and Oliveri [7] were in contact with the Italian Football League for the possible application of the proposed algorithm [43].

Kendall [17] considers the problem of minimising the total distance travelled by 92 English football teams in the top four divisions, over two complete fixtures. Here, each team is required to play one fixture at *home* and the other *away*. Kendall [17] considers that a team cannot play the same team twice during the considered period. The author uses a two-phase process. The first phase is a depth first search, while the other is a local search. Kendall [17] uses a depth-first search to create two consecutive complete fixtures and then uses a local search algorithm to satisfy the remaining constraints. Not only did Kendall reduce the distance travelled by English teams, but in some instances he was able to reduce pair-clashes [17]. Pair-clashes occur when two teams from the same town play their respective matches at *home*. This usually results in a security and crowd control issue for police in these towns due to the overwhelming number of supporters generated by these matches.

Kendall et al. [36] revised the problem of minimising the total distance travelled by English football teams over two complete fixtures. They investigated if there were any acceptable trade-offs

between minimising distances and minimising pair-clashes. Minimising the distance travelled by football teams was found to conflict with the minimisation of pair-clashes. This is because a pair-clash reduces the total distance travelled by teams as there are two teams in the same town not travelling to other towns to play their respective matches. This results in a situation where an increase in pair-clashes leads to a decrease in the total distance travelled. The authors demonstrate that it is possible to reduce the number of pair-clashes without a statistical differences to the distance which has to be travelled by the teams. The proposed solution utilizes the CPLEX method to replace a rather time consuming depth-first search, while SA is used as a replacement for the local search. The methods adopted reduced the CPU time from around 30 hours to approximately 4 minutes.

Goossens and Spieksma [11] describe how they scheduled the Belgian Football League after they were invited by the Belgian FA to automate the schedule production of the Belgian Football League. They used IP in a first-break, then-schedule approach. The schedule developed was then implemented for the 2007 - 2008 season of the Belgian Football League.

In order to assess sports scheduling techniques, their solutions are generally compared with that of benchmark method/s. We now present a brief discussion on the benchmarks used for sports scheduling techniques.

3.5 Benchmarks

A standard framework for highly constrained sports scheduling problems can be found in Nurmi et al. [28]. Various professional sports leagues, all over the world, were studied in the creation of the framework. The numerous objectives and constraints observed were used in the construction of the framework. Sets of artificial and real-world football problems were proposed and their best known solutions were published. This was done so that new scheduling techniques proposed by

researchers could be evaluated.

Artificial problems in Nurmi et al. [28] are mainly based on either the TTP or break minimisation problem [41]. Real world instances look at the Austrian, German, Belgian, Italian and Brazilian football leagues and their unique constraints. The objective, for each of these football leagues, is to minimise the number of *breaks*.

There is currently no literature available on scheduling the CCL. Thus this is a novel instance of the problem. However, there are similarities between the CCL problem, which we discuss in Chapter 4 and the problem considered by Kendall [17]. The adaptation of solution techniques used in [17] are not used, as they take a long time to execute.

Chapter Four

The CCL Scheduling Problem

In this chapter, we describe the CCL scheduling problem in Section 4.1, and then mathematically formulate the knock-out stages and the groups stages in Section 4.2 and Section 4.3, respectively. To be consistent with sports scheduling literature, the constraints used in Section 4.1 are defined according to Nurmi et al. [28]. We have also included constraints to deal with the uniqueness of our problem.

4.1 Problem description

The CCL is a continental club competition, where the best soccer teams from all affiliated national leagues compete for the title of the best club football team on the continent. Given that the CCL is a continental competition, teams are often required to travel far distances to other African nations in order to compete. Thus the CCL scheduling problem is a different problem compared to other football scheduling problems. The main difference lies in the fact that this is a continental tournament which can feature up to 68 teams, whereas other football scheduling problems are for national leagues with a maximum of 20 teams. The CCL problem includes a knock-out stage and as well as group stages, where the group stage could be considered as a mini league with four teams. The task of scheduling football matches in the CCL is therefore a unique and complex one. Apart from scheduling the CCL, in terms of placing teams, we also look into efficiently assigning

associations to provide match officials.

In this research, we are focused on optimally scheduling the CCL to reduce travel distances of teams and match officials from designated associations. This is done by using meta-heuristic methods to generate knock-out stage fixtures and set-up group stage fixture lists for the 2015, 2016, 2017 and 2018 editions of the CCL. More specifically, we will be using, and comparing the performance of three different meta-heuristic methods. The three methods will be evaluated based on their performance in solving our minimisation problem. To judge the effectiveness of taking a heuristic approach, we compare our generated fixtures against official fixtures that were used in the 2015, 2016, 2017 and 2018 editions of the CCL. From here on-wards, we refer to these fixtures as the “existing fixtures”.

For the scheduling problem, we consider a number of constraints which address various stakeholder requirements. These stakeholders include CAF, IOC, national associations, clubs, and fans. An example of a typical stakeholder requirement is that teams from the same country should not play against each other in the knock-out stages of the CCL. This requirement and others listed below ensure that the solution generated, by the meta-heuristic methods, is both feasible and comparable to existing fixture lists.

We now present a descriptive list of the constraints which we will include in the formulation of our problem. We divide the constraints into knock-out stage constraints and group stage constraints:

Knock-out stage constraints

- The same teams cannot play each other in two separate knock-out stages
- Paired teams cannot play each other in the knock-out stages i.e., teams from the same country cannot play against each other

- An association can not provide match officials for teams from the same country as the association
- An association can only provide match officials for each team once, i.e., if the Tunisian association provides match officials for a Kenyan team, then the Tunisian association should not provide match officials for this Kenyan team again
- Paired teams should not play at *home* in the same round of matches, i.e., one of the paired teams should play an *away* game when the other paired team plays at *home*
- A team cannot visit the same country twice in two consecutive years
- A team cannot visit the same country twice in the knock-out stages of the tournament

Group stage constraints

- An association should not provide match officials for teams from the same country as the association
- An association can only provide match officials for each team once
- Teams in the same strength group should not be grouped together, i.e., each group in the group stages should be both *group changing* and *group balanced*.

We note that the constraints considered above are based on fixtures, where the date and time at which these fixtures will occur were not taken into consideration. This means that our fixture lists are not date or time dependent.

The mathematical formulation of the problem considered differs from stage to stage. We, therefore, present the mathematical formulation for each stage, starting with the mathematical formulation of the knock-out stage of the CCL competition.

4.2 Mathematical formulation of the Knock-out stage

We mathematically formulate the latter editions of the CCL, particularly the 2017 and 2018 editions of the CCL because these editions have a higher number of teams in the group stages. This formulation can, however, be easily adapted to the 2015 and 2016 editions of the CCL.

We begin with a description of the decision variables which will be used throughout the rest of this dissertation.

4.2.1 Decision Variables

A list of the decision variables used in our problem description is now presented:

- Preliminary stage fixtures are represented by $X_{i,j}$, where i and j represent *home* and *away* teams, respectively:

$$X_{i,j} = \begin{cases} 1, & \text{if team } i \text{ plays against team } j \text{ in the preliminary stage of the CCL,} \\ 0, & \text{otherwise,} \end{cases}$$

where $i = 1, \dots, n$ and $j = 1, \dots, n \forall i \neq j$. Here n is the number of participants in the preliminary stage of the tournament.

- First knock-out stage fixtures are represented by $Y_{h,g}$, where h and g denote *home* and *away* teams, respectively:

$$Y_{h,g} = \begin{cases} 1, & \text{if team } h \text{ plays against team } g \text{ in the first knock-out stage of the CCL,} \\ 0, & \text{otherwise,} \end{cases}$$

where $h = 1, \dots, 32, g = 1, \dots, 32 \forall h \neq g$.

- Association assignments for the preliminary stage of the CCL is represented by $R_{i,k}$, where

i represents the *home* team in the preliminary stage and k represents the association required to provide match officials:

$$R_{i,k} = \begin{cases} 1, & \text{if association } k \text{ provides match officials to } \textit{home} \text{ team } i, \\ 0, & \text{otherwise,} \end{cases}$$

where $k = 1, \dots, m$, and m is the number of associations which provide match officials.

- Association assignments for the first knock-out stage of the CCL is represented by $A_{h,k}$, where h represents the *home* team in the first knock-out stage and k represents the association required to provide match officials:

$$A_{h,k} = \begin{cases} 1, & \text{if association } k \text{ provides match officials to } \textit{home} \text{ team } h, \\ 0, & \text{otherwise,} \end{cases}$$

where $k = 1, \dots, m$, and m is the number of associations which provide match officials.

We now present a list of parameters which will be used throughout the rest of this dissertation.

4.2.2 Parameters

The associations/pairings described in this section are for illustrative purposes only, as real associations/pairings differ from year to year and in different stages of the game. The list of parameters used in the problem formulation is as follows:

- As we mentioned in Section 4.1, teams from the same country cannot play against each other. We denote team sets by $p_{1,u}$ and $p_{2,u}$, where corresponding elements in each vector represent teams from the same country:

$$p_{1,u} = \{t_1, \mathbf{t}_3, t_5, t_7, t_9, t_{11}, \dots\},$$

$$p_{2,u} = \{t_2, \mathbf{t}_4, t_6, t_8, t_{10}, t_{12}, \dots\}.$$

To illustrate this, consider $u = 2$, i.e., $p_{1,2}$ and $p_{2,2}$, which are paired teams. The corresponding teams in $p_{1,2}$ and $p_{2,2}$, are t_3 and t_4 , respectively, and are not allowed to play against each other since they are from the same country. The same applies for $u = 1, 3, \dots$

- In Section 4.1, we also stated that associations should not provide match officials for teams from the same country as the association. We use $r_{1,v}$ and $r_{2,v}$ to denote teams and their respective associations:

$$r_{1,v} = \{t_1, t_2, \mathbf{t}_3, t_4, t_5, t_6, \dots\}.$$

$$r_{2,v} = \{a_1, a_1, \mathbf{a}_2, a_2, a_3, a_3, \dots\}.$$

When $v = 3$, then we have $r_{1,3}$ and $r_{2,3}$, where $r_{1,3}$ is a team from the same country as association $r_{2,3}$. The corresponding entries in $r_{1,3}$ and $r_{2,3}$ are t_3 and a_2 respectively, which means that association a_2 should not provide match officials for team t_3 , whether this team plays at *home* or *away*.

- In order to ensure that teams do not visit the same countries, in two consecutive years, we introduce the following vectors:

$$cy_{1,q} = \{t_1, t_1, t_3, \mathbf{t}_4, \dots\}.$$

$$cy_{2,q} = \{t_{13}, t_{27}, t_2, \mathbf{t}_{19}, \dots\}.$$

Assuming that it is year O , let $cy_{1,q}$ represent teams which took part in the previous CCL tournament in year $O - 1$, and let $cy_{2,q}$ denote teams from countries which teams in $cy_{1,q}$ visited in year $O - 1$. For example, if Orlando Pirates, a South African team, participated in the CCL in year $O - 1$, then Orlando Pirates would be in $cy_{1,q}$. Furthermore, if Orlando Pirates played against Chicken Inn FC, from Zimbabwe, in year $O - 1$, then $cy_{2,q}$ would contain the current team from Zimbabwe. The current team from Zimbabwe will not necessarily be Chicken Inn FC, but could be another team from Zimbabwe. Orlando Pirates and this

Zimbabwean team should not compete against each other. This enforces the rule that teams should not visit the same countries for two consecutive years.

- In order to depict the first knock-out stage, consider the following vectors:

$$c_{1,l} = \{t_3, t_4, t_{13}, \mathbf{t}_{13}, \dots\}.$$

$$c_{2,l} = \{t_{16}, t_{22}, t_5, \mathbf{t}_6, \dots\}.$$

Let $c_{1,l}$ and $c_{2,l}$ represent *home* and *away* match combinations of teams which played against each other in the preliminary stage of the tournament. Furthermore, let $c_{2,l}$ include teams from countries which were visited by teams in $c_{1,l}$ in the preliminary stage of the tournament. When $l = \{3, 4\}$, we have $c_{1,3}$, $c_{1,4}$, $c_{2,3}$, and $c_{2,4}$, where $c_{1,3}$ and $c_{1,4}$ is a team which played against team $c_{2,3}$ in the preliminary stages and $c_{2,4}$ is from the same country as $c_{2,3}$. The corresponding entries in $c_{1,3}$ and $c_{1,4}$ are both t_{13} , while the corresponding entries in $c_{2,3}$ and $c_{2,4}$ are t_5 and t_6 respectively. This means that team t_{13} should neither play against team t_5 nor t_6 . For example, if Kaizer Chiefs participated in the preliminary stage of the CCL and visited a country like Zambia, which can have two teams in the tournament. Then $c_{2,l}$ will contain both teams from Zambia, i.e., the team which Kaizer Chiefs played against in the preliminary stage, and the second Zambian team. This ensures that teams are not allowed to visit countries which they visited earlier in the tournament. This also ensures that teams do not compete against teams which they have already played against in earlier stages of the CCL.

- In order to ensure that associations do not provide match officials for teams more than once in a tournament, we introduce the following vectors:

$$b_{1,w} = \{\mathbf{t}_1, \mathbf{t}_{16}, t_{16}, t_1, \dots\}.$$

$$b_{2,w} = \{\mathbf{a}_4, \mathbf{a}_4, a_{21}, a_{21} \dots\}.$$

Vector $b_{2,w}$ contains associations which provided match officials for teams in vector $b_{1,w}$. Here, when $w = \{1, 2\}$, we have that association a_4 provided *home* team t_1 and *away* team t_{16} with match officials in the first leg of the preliminary stages of the CCL. This association should, therefore, not provide either team t_1 or team t_{16} with match officials again, for the rest of the tournament.

- A representation of the distances between teams in the preliminary stage of the CCL is depicted in Table 4.1. Actual distance matrices are too large to be depicted here.

$\mathbf{t_{i,j}}$	$\mathbf{t_1}$	$\mathbf{t_2}$	$\cdots$	$\mathbf{t_{n-1}}$	$\mathbf{t_n}$
$\mathbf{t_1}$	0	$d_{1,2}$	$\cdots$	$d_{1,n-1}$	$d_{1,n}$
$\mathbf{t_2}$	$d_{2,1}$	0	$\cdots$	$d_{2,n-1}$	$d_{2,n}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{n-1}}$	$d_{n-1,1}$	$d_{n-1,2}$	$\cdots$	0	$d_{n-1,n}$
$\mathbf{t_n}$	$d_{n,1}$	$d_{n,2}$	$\cdots$	$d_{n,n-1}$	0

Table 4.1 Distance matrix between teams in the preliminary stage of the CCL.

Here $d_{i,j}$ represents the distance (in kilometres) between team i and team j .

- A representation of the distances between teams in the first knock-out stage of the CCL are shown in Table 4.2.

$\mathbf{t_{h,g}}$	$\mathbf{t_1}$	$\mathbf{t_2}$	$\cdots$	$\mathbf{t_{31}}$	$\mathbf{t_{32}}$
$\mathbf{t_1}$	0	$d_{1,2}$	$\cdots$	$d_{1,31}$	$d_{1,32}$
$\mathbf{t_2}$	$d_{2,1}$	0	$\cdots$	$d_{2,31}$	$d_{2,32}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{31}}$	$d_{31,1}$	$d_{31,2}$	$\cdots$	0	$d_{31,32}$
$\mathbf{t_{32}}$	$d_{32,1}$	$d_{32,2}$	$\cdots$	$d_{32,31}$	0

Table 4.2 Distance matrix between teams in the first knock-out stage of the CCL.

Here $d_{h,g}$ represents the distance (in kilometres) between team h and team g .

- A representation of the distances between teams and associations in the preliminary stage of the CCL are shown in Table 4.3.

$\mathbf{i,j}$	$\mathbf{a_1}$	$\mathbf{a_2}$	$\cdots$	$\mathbf{a_{m-1}}$	$\mathbf{a_m}$
$\mathbf{t_1}$	$rd_{1,1}$	$rd_{1,2}$	$\cdots$	$rd_{1,m-1}$	$rd_{1,m}$
$\mathbf{t_2}$	$rd_{2,1}$	$rd_{2,2}$	$\cdots$	$rd_{2,m-1}$	$rd_{2,m}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{n-1}}$	$rd_{n-1,1}$	$rd_{n-1,2}$	$\cdots$	$rd_{n-1,m-1}$	$rd_{n,m}$
$\mathbf{t_n}$	$rd_{n,1}$	$rd_{n,2}$	$\cdots$	$rd_{n,m-1}$	$rd_{n,m}$

Table 4.3 Distance matrix between teams and associations providing match officials in the preliminary stage of the CCL.

Here $rd_{i,k}$ represents the distance (in kilometres) between team i and association k .

- A representation of the distances between teams and associations in the first knock-out stage of the CCL are shown in Table 4.4.

$\mathbf{i,j}$	$\mathbf{a_1}$	$\mathbf{a_2}$	$\cdots$	$\mathbf{a_{m-1}}$	$\mathbf{a_m}$
$\mathbf{t_1}$	$rd_{1,1}$	$rd_{1,2}$	$\cdots$	$rd_{1,m-1}$	$rd_{1,m}$
$\mathbf{t_2}$	$rd_{2,1}$	$rd_{2,2}$	$\cdots$	$rd_{2,m-1}$	$rd_{2,m}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{31}}$	$rd_{31,1}$	$rd_{31,2}$	$\cdots$	$rd_{31,m-1}$	$rd_{31,m}$
$\mathbf{t_{32}}$	$rd_{32,1}$	$rd_{32,2}$	$\cdots$	$rd_{32,m-1}$	$rd_{32,m}$

Table 4.4 Distance matrix between teams and association providing match officials in the first knock-out stage.

Here $rd_{h,k}$ represents the distance (in kilometres) between team h and association k .

We now present the objective function which defines our minimisation problem during the knock-out stage of the CCL.

4.2.3 Objective Function for the knock-out stage of the CCL

Our objective is to minimize the distance travelled by teams and match officials in the CCL, subject to the constraints presented in Section 4.1. We describe the discrete binary optimisation problem which defines our problem, as follows:

Minimise

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=1}^n \left[d_{i,j} X_{i,j} + d_{j,i} X_{j,i} + \left\{ \sum_{k=1}^m rd_{i,k} R_{i,k} X_{i,j} + \sum_{k=1}^m rd_{j,k} R_{j,k} X_{j,i} \right\} \right] \\ & + \sum_{h=1}^{32} \sum_{g=1}^{32} \left[d_{h,g} Y_{h,g} + d_{g,h} Y_{g,h} + \left\{ \sum_{k=1}^m rd_{h,k} A_{h,k} Y_{h,g} + \sum_{k=1}^m rd_{g,k} A_{g,k} Y_{g,h} \right\} \right], \end{aligned} \quad (4.1)$$

Subject to

$$\sum_{u=1}^{|p_X|} X_{p1,u,p2,u} + \sum_{u=1}^{|p_Y|} Y_{p1,u,p2,u} = 0, \quad (4.2)$$

$$\sum_{v=1}^{|r_R|} R_{r_1,v,r_2,v} + \sum_{v=1}^{|r_A|} A_{r_1,v,r_2,v} = 0, \quad (4.3)$$

$$\sum_{q=1}^{|cy_X|} X_{cy_{1,q},cy_{2,q}} + \sum_{q=1}^{|cy_Y|} Y_{cy_{1,q},cy_{2,q}} = 0, \quad (4.4)$$

$$\sum_{i=1}^n \sum_{j=1}^n X_{i,j} + \sum_{l=1}^{|c_Y|} Y_{c_{1,l},c_{2,l}} \leq 1, \quad (4.5)$$

$$\sum_{w=1}^{|b|} R_{b_{1,w},r_{2,w}} + \sum_{w=1}^{|b|} A_{b_{1,w},r_{2,w}} \leq 1. \quad (4.6)$$

Equation (4.1) states that the distance between teams in the preliminary and first knock-out stage is to be minimised. Equation (4.1) also states that the distance between the association providing match officials and the *home* team is to be minimised as well. Equation (4.1) includes a summation from 1 to n , where the number n depends on the number of participants in the preliminary stages, as well as the number of teams which are granted a *bye*. Equation (4.1) also includes a summation from 1 to 32, because the first knock-out stage has 32 participants. Equation (4.2) restricts paired teams from playing against each other, while Equation (4.3) prohibits associations from providing match officials to teams which are from the same country as the association. Equation (4.4) stipulates that teams cannot play in any country which they have visited in the previous CCL year, while Equation (4.5) requires that teams should not play in any country which they have visited in earlier stages of the competition. Lastly, Equation (4.6) restricts associations from providing match officials for teams which they have already provided match officials for in earlier matches of the tournament.

4.3 Mathematical formulation of the Group stage

We now present a mathematical formulation for the group stage of the CCL. Again, we begin by presenting the decision variables, and then the parameters which define the problem.

4.3.1 Decision Variables

A list of the decision variables used in the group stages is now presented:

- Match fixtures across strength groups are represented by $F_{S_{I,G},S_{J,G}}$:

$$F_{S_{I,G},S_{J,G}} = \begin{cases} 1, & \text{if team } I \text{ plays against team } J \text{ within the same group } G, \\ 0, & \text{otherwise,} \end{cases}$$

where $I = 1, 2, 3, 4$ and $J = 1, 2, 3, 4 \forall I \neq J$.

- Association assignments in the group stages are represented by $B_{i,k}$, where i represents the *home* team in group stage fixtures and k represents the association required to provide match officials for the group stage match.

$$B_{i,k} = \begin{cases} 1, & \text{if association } k \text{ provides match officials to } \textit{home} \text{ team } i, \\ 0, & \text{otherwise,} \end{cases}$$

where $k = 1, \dots, m$.

We now present a list of parameters which will be used to formulate the group stages of the CCL.

4.3.2 Parameters

The list of parameters used in the problem formulation is as follows:

- Let the vectors $S_{1,G}, S_{2,G}, S_{3,G}, S_{4,G}$, which have been used for illustrative purposes only, denote different strength groups:

$$S_{1,G} = \{t_1, \mathbf{t}_3, t_4, t_9\},$$

$$S_{2,G} = \{t_2, \mathbf{t}_6, t_7, t_{10}\},$$

$$S_{3,G} = \{t_5, \mathbf{t}_8, t_{11}, t_{12}\},$$

$$S_{4,G} = \{t_{13}, \mathbf{t}_{14}, t_{15}, t_{16}\}.$$

In order to construct a group for the group stages, from these four vectors, we select teams in the same column position from each vector. For instance, if $G = 2$, then $S_{1,2}$, $S_{2,2}$, $S_{3,2}$, and $S_{4,2}$ form one of the four groups which make up the group stages. Therefore, teams $\mathbf{t}_3$, $\mathbf{t}_6$, $\mathbf{t}_8$, $\mathbf{t}_{14}$, are grouped together for the group stages of the CCL.

- In order to ensure that teams from the same country do not play at *home* in the same slot.

The following vectors denote the paired team set one and paired team set two:

$$p_{1,u} = \{t_1, \mathbf{t}_3, t_7, \dots\},$$

$$p_{2,u} = \{t_2, \mathbf{t}_4, t_8, \dots\}.$$

If $u = 2$, then $p_{1,2}$ and $p_{2,2}$ are teams from the same country which should not play at *home* in the same slot. In this case, team t_3 and t_4 should not play their respective *home* matches in the same slot.

- In Section 4.1, we stated that associations should not provide match officials for teams from the same country as the association. We use $r_{1,v}$ and $r_{2,v}$ to denote teams and their respective associations:

$$r_{1,v} = \{\mathbf{t}_1, t_2, t_3, t_4, t_5, t_6, \dots\},$$

$$r_{2,v} = \{\mathbf{a}_1, a_1, a_2, a_2, a_3, a_3, \dots\}.$$

When $v = 1$, then $r_{1,1}$ is a team from the same country as association $r_{2,1}$. The corresponding

entries in $r_{1,1}$ and $r_{2,1}$ are t_1 and a_1 respectively, which means that association a_1 should not provide match officials for team t_1 , whether this team plays at *home* or *away*.

- In order to ensure that associations do not provide match officials for teams more than once in the tournament, we introduce the following vectors:

$$b_{1,w} = \{t_1, t_{16}, \mathbf{t}_{16}, \mathbf{t}_1, \dots\},$$

$$b_{2,w} = \{a_4, a_4, \mathbf{a}_{21}, \mathbf{a}_{21} \dots\}.$$

Vector $b_{2,w}$ contains associations which provided match officials for teams in vector $b_{1,w}$. When $w = \{3, 4\}$ then association a_{21} provided *home* team t_{16} and *away* team t_1 with match officials, in the knock-out stages of the CCL. This association should not provide either team t_1 or t_{16} with match officials in the group stages.

- A representation of the distances between teams in the group stages of the CCL is shown in Table 4.5.

$\mathbf{t_{i,j}}$	$\mathbf{t_1}$	$\mathbf{t_2}$	$\cdots$	$\mathbf{t_{15}}$	$\mathbf{t_{16}}$
$\mathbf{t_1}$	0	$d_{1,2}$	$\cdots$	$d_{1,15}$	$d_{1,16}$
$\mathbf{t_2}$	$d_{2,1}$	0	$\cdots$	$d_{2,15}$	$d_{2,16}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{15}}$	$d_{15,1}$	$d_{15,2}$	$\cdots$	0	$d_{15,16}$
$\mathbf{t_{16}}$	$d_{16,1}$	$d_{16,2}$	$\cdots$	$d_{16,15}$	0

Table 4.5 Distance matrix between teams in the group stages of the CCL.

Here $d_{i,j}$ represents the distance (in kilometres) between team i and team j .

- A representation of the distances between teams and match officials in the group stages of the CCL is shown in Table 4.6.

$\mathbf{i,j}$	$\mathbf{a_1}$	$\mathbf{a_2}$	$\cdots$	$\mathbf{a_{50}}$	$\mathbf{a_{51}}$
$\mathbf{t_1}$	$rd_{1,1}$	$rd_{1,2}$	$\cdots$	$rd_{1,50}$	$rd_{1,51}$
$\mathbf{t_2}$	$rd_{2,1}$	$rd_{2,2}$	$\cdots$	$rd_{2,50}$	$rd_{2,51}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$\mathbf{t_{15}}$	$rd_{15,1}$	$rd_{15,2}$	$\cdots$	$rd_{15,50}$	$rd_{15,51}$
$\mathbf{t_{16}}$	$rd_{16,1}$	$rd_{16,2}$	$\cdots$	$rd_{16,50}$	$rd_{16,51}$

Table 4.6 Distance matrix between teams and associations providing match officials in the group stages of the CCL.

Here $rd_{i,k}$ represents the distance (in kilometres) between team i and association k .

We now present the objective function which defines our minimisation problem for the group stages of the CCL.

4.3.3 Objective function

For this stage of the CCL, the objective is to minimize the distance travelled by teams and match officials, subject to some constraints. The problem is defined as follows:

Minimise:

$$\begin{aligned} & \sum_{G=1}^4 \left[\sum_{I=1}^3 \sum_{J=2}^4 \left\{ d_{S_{I,G}, S_{J,G}} F_{S_{I,G}, S_{J,G}} + \sum_{k=1}^m rd_{S_{I,G}, k} B_{S_{I,G}, k} F_{S_{I,G}, S_{J,G}} \right\} \right] \\ & + \sum_{G=1}^4 \left[\sum_{J=4}^2 \sum_{I=3}^1 \left\{ d_{S_{J,G}, S_{I,G}} F_{S_{J,G}, S_{I,G}} + \sum_{k=1}^m rd_{S_{J,G}, k} B_{S_{J,G}, k} F_{S_{J,G}, S_{I,G}} \right\} \right], \end{aligned} \quad (4.7)$$

Subject to

$$\sum_{v=1}^{|r_B|} B_{r_{1,v}, r_{2,v}} = 0, \quad (4.8)$$

$$\sum_{w=1}^{|b|} B_{b_{1,w}, b_{2,w}} \leq 1. \quad (4.9)$$

Equation (4.7) is our objective function, and states that the distance d between team i and j , in group G should be minimised. The reason why the summations runs from 1 through to 4 is because the group stages of the CCL have four groups. Equation (4.7) also requires that the distance rd between team i and association k be minimised. The summation in the second line of Equation (4.7) states that the second round of matches in a $2RR$ should have a minimal distance, equal to the first round of matches, and that the distance between team j and association k should be minimised. The formulation of Equation (4.7) restricts a team from being grouped with other teams from its strength group, i.e., Equation (4.7) ensures that groups are both *group changing* and *group balanced*. Equation (4.8) prohibits associations from providing match officials for teams which are from the same country as the association. Finally, Equation (4.9) restricts associations from providing match officials for teams which they have already provided match officials for in earlier stages of the CCL. Match officials can only officiate a match for a team in any stage of the CCL once.

Chapter Five

Meta-Heuristic Methods for Solving the CCL Scheduling Problem

In this dissertation, we investigate three meta-heuristic methods, namely Simulated Annealing (SA), Genetic Algorithm (GA) and Tabu Search (TS) on their ability to solve the problem at hand. In this chapter, we present an overview of SA in Section 5.1. In Section 5.2, we provide a discussion on GA, and in Section 5.3, we present TS.

5.1 Simulated Annealing

Based on a Statistical Mechanics analogy, SA can be interpreted as a form of controlled random walk in the space of feasible solutions. SA works similarly to LS. However, while LS accepts only improving moves, SA allows for moves which do not necessarily result in an immediate improvement in the objective function. This is done so as to avoid or escape being trapped in a local minimum.

5.1.1 Important Parameters Used In SA

Parameters play an important role in random search techniques such as SA. Poorly chosen parameters affect the quality of the solution. Here, we take solution quality to mean an algorithms ability to

find a solution with the lowest total distance travelled. In this dissertation, parameter settings for SA are based on trial and error, as well as suggestions from Hillier and Lieberman [16], and Anagnostopoulos et al. [2]. Anagnostopoulos et al. [2] based the choice of their parameter values on experimental designs used to schedule the Major League Baseball TTP problem in the United States.

We now present a discussion on the parameters used in SA.

5.1.2 Acceptance Function

In Section 5.1, we mentioned that unlike LS which accepts only improving moves, SA allows for non-improving moves. These non-improving moves are accepted with the probability:

$$\Pr(\text{acceptance}) = e^{-\left(\frac{f_m - f_c}{T}\right)}, \quad (5.1)$$

where f_m is a trial solution, f_c is the current solution, and T is a temperature control parameter which approaches zero, over time. In this work, we take a trial solution to be the distance travelled within a fixture list. The trial solution is accepted if $f_m - f_c < 0$, i.e., if the total distance travelled in the trial solution is lower than the total distance travelled in the current solution. Conversely, if $f_m - f_c > 0$, i.e., if the total distance travelled in the trial solution is higher than the total distance travelled in the current solution, then the trial solution is interrogated. This trial solution is only accepted if some randomly generated uniform number is less than the *probability of acceptance* in Equation 5.1. Trial solutions accepted in this manner are called *uphill moves*. SA accepts *uphill moves* to allow for greater exploration of the solution space, by escaping local minima in search for the global minimum.

An important analogy used in SA is that of temperature control. We now present a discussion on the temperature parameter T , as well as the temperature schedule and epoch length used in SA.

5.1.3 Temperature

The temperature parameter T used in the *probability of acceptance* function described in Equation 5.1, determines the willingness of the SA solver to randomly search the solution space. With higher temperatures, the emphasis is on diversification to unexplored regions. This is achieved because there is a higher probability that non-improving moves will be accepted. Hence, high temperatures assist SA to identify the promising regions which may contain the global minimum. The probability of accepting non-improving moves decreases as the temperature control decreases [24]. This results in a decrease in random searches in unexplored spaces. The reduction in temperature, thus shifts the emphasis towards finding the local minima in the promising regions.

The initial temperature and cooling rate are essential parameters which need careful consideration. According to Kirkpatrick et al. [29], the temperature must be high enough to make the initial probability of accepting non-improving moves as high as possible. Anagnostopoulos et al. [2] suggests an initial temperature of $T_0 = 700$, while Hillier and Lieberman [16] suggest using an initial temperature of $T_0 = 0.2f_m$. They, however, make the assumption that the initial trial solution is of good quality [16]. We have, however, found that using a lower temperature than that suggested by Hillier and Lieberman [16] and a much higher temperature than that suggested by Anagnostopoulos et al. [2] produced improved solution quality.

Temperature Schedule

A proportional function is commonly used in reducing the temperature throughout the SA iterative process [33]. The cooling function used to reduce the temperature is defined as:

$$T_k = \alpha T_{k-1}, \quad (5.2)$$

where the k_{th} temperature is a function of the preceding temperature T_{k-1} , and the cooling rate $\alpha \in (0, 1)$. If α is set close to zero, then the temperature rapidly decreases and this could lead to poor solution quality. Conversely, if α is set closer to one, then the temperature gradually declines

and this could result in higher solution quality, albeit with a slower execution speed. Hillier and Lieberman [16] suggest a five temperature schedule with $\alpha = 0.5$. Anagnostopoulos et al. [2] used $\alpha = 0.98$ as their cooling rate, where the temperature is changed after a fixed number of consecutive non-improving moves are executed.

Epoch length

Epoch length is the number of iterations which are executed at each temperature T . There are a number of ways in which the epoch length can be chosen. Anagnostopoulos et al. [2] require that a fixed number of consecutive non-improving moves be executed before a decline in temperature is achieved, while Park and Kim [33] suggest that the epoch length be set proportional to the search space size of a problem. In this research, we chose our epoch length based on numerical testing.

5.2 Genetic Algorithm

The use of genetic algorithms, as an approach to solve optimisation problems, can be traced back to 1970 [10]. GA is a population based algorithm derived from evolutionary algorithms. GA generates solutions to optimisation problems using techniques inspired by natural evolution. The premise upon which GA is built, is that fitter individuals in a population have a higher chance of surviving their current environment than weaker individuals. Hence, newer generations in a population are assumed to be superior than older generations.

GA has the capacity to tackle a wide range of large complex problems, where other traditional methods have failed. A typical genetic algorithm consists of the following:

1. A population of guesses to the solution of the problem;
2. A strategy for evaluating individual solutions within a population;

3. A method for integrating a portion of the better individual solutions to make new and improved solutions for the new population;
4. An operator, which is used to increase diversity within the solutions

GAs ability to tackle complex search spaces with many local optima is one of the main reasons for its popularity. The ability to tackle these complex search spaces stems from having an initial population of random guesses, throughout the search space. GA then uses specific operators to steer the population towards convergence to a global optimum. We describe these operators in Sections 5.2.1 - 5.2.3.

5.2.1 Selection Operators

The process of *selection* influences the population in a manner similar to natural selection in biological systems. Unfit individuals within the population are weeded out, and fit individuals have a greater chance of reproduction to produce a new generation which is fitter.

Finding an appropriate *selection* operator to use in GA is important. *Selection* operators directly influence the overall performance of the algorithm, and the quality of the solution produced. *Selection* operators ensure that the right individuals are chosen for reproduction. Grefenstette [23], describes *selection* as the process in which one:

- maps the objective function to fitness
- creates a probability distribution based on fitness
- draws a sample from this distribution

Here we describe the ways in which the *selection* process is carried out, as outlined by Grefenstette [23]. A list of the most popular *selection* techniques is given below:

- Roulette Wheel Selection

- Elitism
- Rank Selection
- Tournament Selection

We present a discussion on each of these techniques.

Roulette Wheel Selection

In the roulette wheel procedure, the probability of selection is directly proportional to the fitness value of the individual in a population. The analogy of a roulette wheel is used because one can view the population as the formation of a roulette wheel, where the size of each slot is commensurate with an individuals fitness value. The analogy then goes on to spinning the wheel while throwing in a ball. The probability of the ball resting in any slot is equivalent to the arc of the slot i.e., the fitness value of the corresponding individual. Mathematically this can be represented by a probability function, which is defined as:

$$P(i) = \frac{f(i)}{\sum_{j=1}^n f_j}. \quad (5.3)$$

Equation 5.3 states that the probability of selecting an individual for a new population is directly proportional to the current fitness value of that individual. Hence, finding the individual that will be selected to form part of the new population is determined by summing up the fitness values of population members. This process is, however, terminated, when the sum exceeds some user defined number. The last individual added, before such termination, is chosen for reproduction. This procedure gives fitter individuals a higher chance of selection, as they occupy a larger portion of the roulette wheel. However, the risk of premature convergence to a local optimum is possible. Premature convergence to a local optimum stems from the presence of a dominant individual that is always selected due to its higher than average fitness value. This occurs because the dominant individual constitutes a bigger portion of the circumference of the roulette wheel, and thus has a greater chance of selection.

Elitism

We have observed that the use of roulette wheel selection may result in premature convergence to a sub-optimal solution. However, there is no guarantee that the best individual will always be selected, unless the dominant individual is much fitter than the rest. If the fittest individual is not selected, then it could die off and never be included to form part or contribute to a new population. In some cases, this may not have such a negative impact because this would slow the algorithm down, and consequently allow a more thorough exploration of the search space before convergence. On the other hand, if the best individual is chosen, then this individual could be lost if it is destroyed by genetic modification. There are several operators which are used to achieve this modification. These will be presented in Section 5.2.2 and 5.2.3, respectively.

The quality of a solution for many problems could be improved by retaining the best, or elite member between generations. Ensuring that the best individual is always selected, and never required to undergo genetic modification is called *elitism*. This means that the very best individual or the highest ranked individuals are copied directly to the next generation or population, while the other individuals may be required to undergo genetic modification.

Rank selection

Rank selection ranks individuals in the population from best to worst according to their fitness value. The best individual is ranked N and the worst is ranked 1. Rank selection was introduced by Baker [20], in an attempt to eliminate the scaling problems produced by dominant individuals in roulette wheel selection. A dominant individual constitutes a bigger portion of the circumference of the roulette wheel, and thus has a greater chance of selection. This results in a lack of diversity in this procedure, and could possibly result in premature convergence to a sub-optimal solution.

There are a number of different ways to rank a population from the best (fittest) individual to

the worst (weakest) individual. The most basic of these is linear ranking, with the probability of selection given by:

$$\Pr(i) = \frac{\text{rank}(i)}{n(n-1)}, \quad (5.4)$$

where n is the number of individuals in the population. Rank selection ensures a uniform distribution in the population. However, it slows convergence to a global optimum as the fitness levels of individuals participating in genetic modification, are almost similar.

Tournament Selection

In tournament selection, k individuals are selected from the population, and are required to compete with one another. The number of participants in each tournament is called the tournament size, and is user-defined. The fittest individual per tournament is selected for genetic modification. The biggest advantage of tournament selection is that it provides individuals with an equal opportunity to compete, and thus preserves diversity within the population.

5.2.2 Crossover

We mentioned, in Section 5.2.1, that certain operators are used to achieve genetic modification of individuals. This exchange of genes between selected solutions or parents, to produce offspring, is called *crossover*. There are a number of ways to perform crossover. The selection of a crossover technique depends on how a user chooses to represent individuals or solutions. For a classical binary GA, which makes use of vector notation, single point crossover is implemented on selected pairs of individuals by randomly selecting a single point within the binary vectors, and swapping all the genes to the right of this point.

Crossover provides a way for genes to be shared and melded to allow for deeper exploration of the search space. Some of the important crossover techniques are:

- Single-point order crossover

- Two-point order crossover
- Partially mapped crossover (PMX)
- Alternative edge crossover

We now present a discussion on each of these crossover techniques. For illustrative purposes, we make use of vector notated parent and child chromosomes, with numbers as genes. Crossover is done through transferring genes from parents to children. This is done in various ways, using the different crossover operators which we will now describe in detail.

Single point order crossover

Given parent chromosome P_1 and parent chromosome P_2 of length L , a randomly chosen position K is selected such that $(1 < K < L)$. Everything to the left of K in P_1 , known as the left schema of P_1 , is then copied into child C_1 , which is initially empty. Similarly, the left schema of P_2 is copied into child C_2 , which is also initially empty. The notation “|” is used to denote the single point K :

P_1	1	3	4	5	2	6	7	8	10	9
-------	---	---	---	---	---	---	---	---	----	---

P_2	5	4	3	10	9	8	2	1	6	7
-------	---	---	---	----	---	---	---	---	---	---

C_1	1	3	4	5						
-------	---	---	---	---	--	--	--	--	--	--

C_2	5	4	3	10						
-------	---	---	---	----	--	--	--	--	--	--

All genes, from the right schema of P_2 , are copied to the right schema of C_1 in the same sequence as they appear, provided that they are not already present in the left sequence of C_1 . If a gene is already present in C_1 , then this gene is skipped. The process is repeated for C_2 , from P_1 .

C_1	1	3	4	5	10	9	8	2	6	7
-------	---	---	---	---	----	---	---	---	---	---

C_2	5	4	3	10	1	2	6	7	8	9
-------	---	---	---	----	---	---	---	---	---	---

Two-point order crossover

This two-point order crossover is similar to the single point order crossover, but with two positions, K_1 and K_2 , instead of one. Here, K_1 and K_2 positions are randomly selected, such that $(1 < K_1 < K_2 < L)$. The middle schema, between position K_1 and K_2 , are copied from P_1 and P_2 into the empty child chromosomes C_1 and C_2 , respectively. The sections between position K_1 and K_2 are copied in the exact sequence and are placed in the same location as in the respective parent:

P_1	1	3	4	5	2	6	7	8	10	9
-------	---	---	---	---	---	---	---	---	----	---

P_2	5	4	3	10	9	8	2	1	6	7
-------	---	---	---	----	---	---	---	---	---	---

C_1					2	6	7			
-------	--	--	--	--	---	---	---	--	--	--

C_2					9	8	2			
-------	--	--	--	--	---	---	---	--	--	--

All genes in P_2 are copied into the empty spots in C_1 in the exact same order as they appear. If a gene is already present in the middle schema of C_1 , then this gene is skipped. The crossover process is repeated for parent P_1 and child C_2 .

C_1	5	4	3	10	2	6	7	9	8	1
-------	---	---	---	----	---	---	---	---	---	---

C_2	1	3	4	5	9	8	2	6	7	10
-------	---	---	---	---	---	---	---	---	---	----

Partially mapped crossover

Similar to two point order crossover, here the schema between position K_1 and K_2 in P_1 and P_2 are copied into the empty child chromosomes C_2 and C_1 , respectively. The schema is copied in the exact sequence and are placed in the same location as in the respective parent:

P_1	1	9	7	8	4	10	6	5	2	3
-------	---	---	---	---	---	----	---	---	---	---

P_2	4	3	9	7	5	8	2	1	10	6
-------	---	---	---	---	---	---	---	---	----	---

C_1				7	5	8	2			
-------	--	--	--	---	---	---	---	--	--	--

C_2				8	4	10	6			
-------	--	--	--	---	---	----	---	--	--	--

All genes in P_1 and P_2 are copied to C_1 and C_2 respectively. However this is done according to a set mapping relation. The mapping relation checks if genes that should be copied, already exist in the child. If this gene does not exist in the child, then it is copied directly to the child. If a

gene already exists in the child, then some adjustments need to be made. Consider C_1 for example, genes 1 and 9 are copied directly from P_1 to C_1 , however gene 7 already exists in C_1 . This gene cannot appear twice in C_1 , therefore a mapping is used to ensure that the gene in this position is unique. Particularly, we look for gene 7 in the other parent, P_2 , and map the corresponding gene in the same position in P_1 . We see that gene in this position is gene 8 which also already exists in C_1 . We thus continue by locating the position of gene 8 in P_2 and map the corresponding gene in P_1 . The corresponding gene is 10, which is not yet in C_1 . This gene is therefore copied to C_1 and the process of copying genes into children continues until another mapping relation is required. The entire mapping to complete the crossover process is as follows: $7 \Rightarrow 8 \Rightarrow 10$; $5 \Rightarrow 4$; $2 \Rightarrow 6$ for C_1 and $4 \Rightarrow 5$; $10 \Rightarrow 8 \Rightarrow 7$; $6 \Rightarrow 2$ for C_2 . The resultant child chromosomes C_1 and C_2 are therefore as follows:

C_1	1	9	10	7	5	8	2	4	6	3
-------	---	---	----	---	---	---	---	---	---	---

C_2	5	3	9	8	4	10	6	1	7	2
-------	---	---	---	---	---	----	---	---	---	---

Alternative edge crossover

Alternative edge crossover is an edge preserving operator, and is best illustrated with the use of an example. Edges are copied from both parents P_1 and P_2 in an alternating sequence, and distributed to each child chromosome.. An initial random edge (a, b) is located and copied from P_1 to C_1 , then edge (b, c) is located and copied from P_2 to C_1 , edge (c, d) is located and copied from P_1 to C_1 , etc. When an edge is already present in the child, then a new edge is randomly selected again to start the process. The process ends when all spaces in the child are filled. The same procedure is followed for C_2 . Now for P_1 and P_2 given below:

P_1	1	9	7	8	4	10	6	5	2	3
-------	---	---	---	---	---	----	---	---	---	---

P_2	4	3	9	7	5	8	2	1	10	6
-------	---	---	---	---	---	---	---	---	----	---

if edge (8, 2) is randomly selected first from P_2 , then the next edge selected from P_1 will be (2, 3) followed by (3, 9) from P_2 . The process continues until the child chromosome is filled or an unwanted cycle is reached. In the example above, an unwanted cycle is reached when the edge (5, 2) is obtained from P_1 . This is because gene 2 already exists in the child chromosome C_1 :

C_1	8	2	3	9	7	5				
-------	---	---	---	---	---	---	--	--	--	--

This is remedied by selecting a random edge in P_1 , which is not yet present in C_1 . A random edge in P_1 is chosen such that both its edges are not present in C_1 . The process continues and cycles are avoided until the child's genes are completely filled. The resultant child chromosome, C_1 , would then be:

C_1	8	2	3	9	7	5	10	6	4	1
-------	---	---	---	---	---	---	----	---	---	---

Even though crossover is a useful technique for selection, it is not necessary for a successful implementation of GA, as we were able to utilise GA without a crossover operator.

5.2.3 Mutation

Once crossover has been implemented, GA usually includes a mutation operator. The mutation operator improves the quality of a solution, by randomly altering the child's genetic make-up. This helps GA explore the search space even further. Mutation can also be viewed as a background

operator which maintains genetic diversity in the population. We now elaborate on how this is achieved. Given a child chromosome, we describe a few permutation-based mutation types:

- **Swap Mutation:** To perform a swap mutation, one selects two genes at random, and swaps their position. Swap mutation preserves most of the adjacency information, but the links that are broken disrupt the order in the chromosome. If we choose the genes in position 4 and 7, then child C_1 :

C_1	1	2	3	4	5	6	7	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

becomes:

C_1	1	2	3	7	5	6	4	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

- **Insert Mutation:** Here, two genes are selected at random. The second gene is moved to the position of the first gene and the first gene is permuted to the right along with all the remaining genes, in the exact order. Note that this preserves most of the order within the child chromosome. For example, randomly selecting genes at position 2 and 7 in C_1 :

C_1	1	2	3	4	5	6	7	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

C_1 becomes

C_1	1	7	2	3	4	5	6	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

- **Inversion Mutation:** In order to perform inversion, two genes are selected randomly, swapped and the substring between them is inverted. Inversion mutation preserves most adjacency information and only breaks two links, but it leads to the disruption of order in the chromosome. For example, if we randomly select genes at position 3 and 9 in C_1 :

C_1	1	2	3	4	5	6	7	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

then C_1 becomes:

C_1	1	2	9	8	7	6	5	4	3	10
-------	---	---	---	---	---	---	---	---	---	----

- **Scramble Mutation:** In this mutation, one has to pick a subset of genes at random and then randomly rearrange them. If a subset of genes from position 2 to 5 is chosen, then C_1 :

C_1	1	2	3	4	5	6	7	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

becomes:

C_1	1	3	5	2	4	6	7	8	9	10
-------	---	---	---	---	---	---	---	---	---	----

Here, the subset of genes from position 2 to 5 were randomly rearranged to produce the mutated child.

5.3 Tabu search

Tabu search (TS), like all other meta-heuristic methods, is motivated by the observation that not all local optimal solutions are good solutions. TS thus provides other iterative methods like LS with

a mechanism which helps it escape local optima and continue the search for a global optimum [31].

As such, TS is a meta-heuristic which can be superimposed on other heuristic search procedures. TS may be used as a guiding mechanism for procedures which iteratively move from one solution to the next, and have a function which measures whether the solution is improving or worsening. The guidance given by TS is very flexible in a sense that it motivates one to create new types of moves and evaluation criteria [15].

The core of TS lies in its short-term memory process known as the *Tabu List (TL)*. *TL* stores explored solutions, or some relevant attributes of these solutions. Once stored, these selected attributes are declared as *tabu*. This is done to ensure that certain moves are not reversed. For instance, when moving away from local optima, steps taken are recorded so as to prevent the search from tracing back along those steps. This allows TS to perform a more extensive exploration of the search space [30]. The recorded attributes remain in the *TL* for a particular number of iterations called *Tabu Tenure (TT)*, unless something called the aspiration criterion is satisfied. The aspiration criterion is a rule that only applies when a move, which has been declared as *tabu*, is involved in the build up of a solution which is better than the best solution found so far. In this case, the aspiration criterion would dismiss the *TL*, and the move would be accepted insofar as the solution found is better than the best recorded solution.

The *TL* is reinitialized every time a local search process is started. Therefore, if the random move defined in the previous phase was involved in the build up to the current local minimum, then the search will return to the previous solution and it will be incorporated into the *TL*. Otherwise, the local search will have a different trajectory.

A flow diagram is used to illustrate the TS procedure:

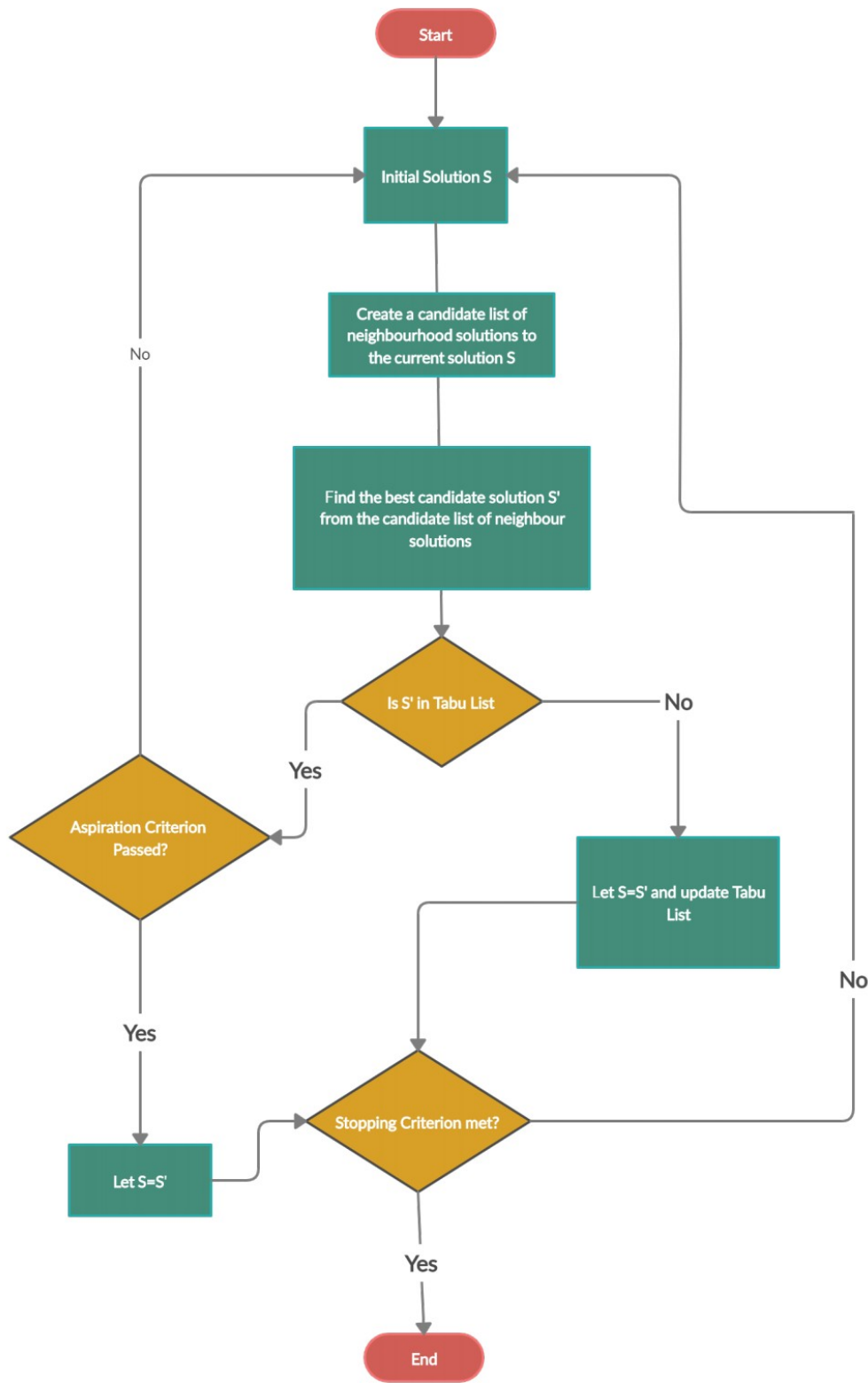


Figure 5.1 TS flow diagram

Chapter Six

Implementation of Meta-Heuristic Methods for the CCL Scheduling Problem

This chapter discusses the implementation of three different meta-heuristic methods which are used to solve the CCL scheduling problem. The objective function, as well as the constraints for the CCL scheduling problem were presented in Chapter 4. In this chapter, we start by presenting an overview of the model used to generate fixtures and assign associations, in Section 6.1. We then provide a summary of the scope and assumptions which we made when building the model in, Section 6.2. We provide the framework for our implementation of SA, GA and TS as well as their pseudo-codes, in Section 6.3, 6.4, and 6.5 respectively.

6.1 Overview

The model which we used for the CCL scheduling problem comprises of three parts:

1. The first part of the model identifies the fixture list which optimises travelling distances between teams.
2. The second part of the model identifies an association list which optimises travelling distances between the association providing match officials, and the *home* team. Recall that the reason why we need to optimise the distance between the association and the home team is because

the *home* team is required to make transport arrangements for the selected match officials who will referee the match. Decreasing the distance which will be travelled by match officials will decrease the cost of making transport arrangements for the match officials, on the *home* team.

3. The last part of the model is the verification phase, where the solution is checked for any constraint violations. The solution is saved if no violations are present and forms part of the constraints in the next knock-out stage or the next phase of the competition, which is the group stages.

When constructing a model to solve the CCL problem described in Chapter 4, we set out the scope of the problem and made some assumptions. The scope and assumptions made, may influence the quality of the solution generated by the meta-heuristic methods. We now present and discuss the scope as well as the assumptions we made when solving the problem at hand.

6.2 Scope and Assumptions

We begin with a discussion around the scope of the problem.

6.2.1 Scope

The scope of the problem refers to the extent to which our research problem is answered, and thus everything which we considered to be relevant in order to best solve the problem at hand.

Our model only considers the development of a list of fixtures, and the corresponding associations. We have not assigned dates to these fixtures, and we have not assigned a specific order to the fixtures. We made the assumption that CAF would consult TV broadcasters to find suitable dates and times in order to maximise TV viewership, and the revenue obtained from these affiliated TV broadcasters.

Furthermore, we did not consider the national leagues of affiliated countries, as well as all national tournaments played by affiliated teams in the CCL. We know that CAF requires that all affiliated associations prioritise the CCL. We, therefore, assumed that when affiliated associations schedule matches, then dates and times preferred by CAF would be identified and set aside.

We discuss some of the major assumptions made and how they influence the scope of the CCL problem.

6.2.2 Assumptions

FIFA

When scheduling CCL fixtures, FIFA international fixtures were considered for any clashes. However, we made the assumption that any of the dates which are assigned for FIFA International fixtures would be set aside by CAF authorities.

CAF Confederation Cup

Most teams participating in the CCL can be paired with teams in the lower ranked CCC. This can influence the fixture list and the assigning of associations. Since the CCL is the most prestigious club tournament in Africa, we assumed that the lower CCC can be scheduled around the CCL. For this reason, the CCC is not considered in the scheduling of the CCL, and is beyond the scope of this research.

Strength and Paired Teams

We assumed that the paired teams and strength teams, which are used, are accurate for each year, and will be accepted by CAF authorities. The paired teams used are based on affiliated associations and their respective teams, while strength groupings are adapted from CAFs official website [4]

Special Requests

We are aware that there may be some objections to the fixture lists which we generate. This includes that teams may request that their fixtures be played on different fields than the fields we used to construct the distance matrices. We are, however, confident that the schedules generated by each meta-heuristic method could still result in better fixtures than those currently used.

Other Considerations

The quality of our solution is reduced if there are important constraints which we did not consider due to a lack of disclosed information. This pertains to club, policing and geopolitical information, or any other information which may contribute to forming real world constraints. We, thus, acknowledge that the scope of the model was formulated around information which was disclosed and accessible. However, we believe that new constraints could easily be incorporated into the framework.

We now discuss how the meta-heuristic methods chosen to solve this problem were implemented.

6.3 Implementation of Simulated Annealing

As mentioned in Section 6.1, the meta-heuristic methods used are segmented into three different parts, in order to solve the CCL problem. We now describe these three parts for SA.

6.3.1 Phase I of SA

Simulated annealing requires an initial solution as well as an initial temperature. We begin by producing an initial fixture list, without taking travel distances, pair clashes or any other constraints into consideration. Alongside each match-up in our initial fixture list is a distance vector, which describes the distance between the two matched-up teams. We then swap two teams from two

different fixtures to generate a new fixture list, which either produces an improved (shorter) or worsened (greater) distance. The new fixture list is then checked for any constraint violations, and is only accepted if our solution is feasible, i.e., if the fixture list does not violate any constraints. Solutions which produce greater distances are accepted with the probability:

$$Prob(acceptance) = e^{-\left(\frac{f_m - f_c}{T}\right)}. \quad (6.1)$$

The simple swap operation is repeated epoch length L times. The temperature decreases after L swaps, and a whole new set of L swaps takes place again. The process continues until the temperature control decreases to our final temperature T_f .

6.3.2 Phase II of SA

Once *Phase I* of SA is complete, an optimal fixture list would have been generated. We then create a list of randomly sorted associations providing match officials, and assign these associations to the optimal fixture list obtained in *Phase I* of SA. Since this assignment is random, neither distances between paired associations and teams, nor constraint violations are taken into consideration. Alongside each pair of teams and associations is a distance vector, which contains the distance between the *home* team and the association. From this list of association assignments, two associations are randomly selected and swapped. The swap operation results in a new list of associations with corresponding distances from respective teams. The solution resulting from the swap operation is checked for any constraint violations, and is only accepted if it is feasible. Again, the solutions which yield greater distances are accepted with probability:

$$Prob(acceptance) = e^{-\left(\frac{f_m - f_c}{T}\right)}. \quad (6.2)$$

The simple swap operation is repeated epoch length L times. The temperature decreases after L swaps and the process is repeated. The process continues until the temperature control decreases to our final temperature T_f .

6.3.3 Phase III of SA

In *Phase III* of SA, we verify that the generated fixture lists, and corresponding list of associations are feasible, as they will be used to construct constraints for later stages of the tournament. The optimal fixture list and corresponding list of associations will also be used to build constraints for the subsequent year of the CCL. Once we have verified that both lists are feasible, these lists are then augmented into one and stored in MySQL. MySQL is a relational database management system based on Structured Query Language (SQL). The application is used for a wide range of purposes, including data warehousing, e-commerce, and logging applications. The stored solution is accessed when constructing the next stage of the competition, and will be utilised for the consecutive year of the CCL.

We now present a pseudo-code for the SA Algorithm as it has been implemented in this thesis.

```
1: procedure SIMULATED ANNEALING PHASE I
2:   Initialize parameters ( $T_{Max}, s, T_f$ )
3:    $T \leftarrow T_{Max}$ 
4:    $L \leftarrow s$ 
5:   best fixture  $\leftarrow$  Initial fixture list
6:   while  $T > T_f$  do
7:     for  $i \leftarrow 1 : L$  do
8:        $next \leftarrow$  NEIGHBOUR
9:       if constraint violation  $\in next$  then
10:         $next \leftarrow$  NEIGHBOUR
11:     end if
```

```

12:       $\Delta F \leftarrow \mathbf{TRAVEL\ DISTANCE}(next) - \mathbf{TRAVEL\ DISTANCE}(best\ fixture)$ 
13:      if  $\Delta F < 0$  then
14:           $best\ fixture \leftarrow next$ 
15:      else if  $\mathbf{RANDOM}() < e^{-(\frac{\Delta F}{T})}$  then
16:           $best\ fixture \leftarrow next$ 
17:      end if
18:  end for
19:   $T \leftarrow \mathbf{COOLING}(T)$ 
20: end while
21: return  $best\ fixture$ 
22: end procedure

```

1: **procedure** SIMULATED ANNEALING PHASE II

```

2:   Initialize parameters ( $T_{Max}, s, T_f$ )
3:    $T \leftarrow T_{Max}$ 
4:    $L \leftarrow s$ 
5:    $best\ ref \leftarrow \mathbf{Initial\ ref}$ 
6:   while  $T > T_f$  do
7:       for  $i \leftarrow 1 : L$  do
8:            $next \leftarrow \mathbf{NEIGHBOUR}$ 
9:           if  $constraint\ violation \in next$  then

```

```

10:       $next \leftarrow \mathbf{NEIGHBOUR}$ 
11:    end if
12:     $\Delta F \leftarrow \mathbf{TRAVEL\ DISTANCE}(next) - \mathbf{TRAVEL\ DISTANCE}(best\ ref)$ 
13:    if  $\Delta F < 0$  then
14:       $best\ ref \leftarrow next$ 
15:    else if  $\mathbf{RANDOM}() < e^{-(\frac{\Delta F}{T})}$  then
16:       $best\ ref \leftarrow next$ 
17:    end if
18:  end for
19:   $T \leftarrow \mathbf{COOLING}(T)$ 
20: end while
21: end procedure

```

```

1: procedure SIMULATED ANNEALING PHASE III
2:   if constraint violation  $\in$  Optimal Fixture and Association List then
3:     Break
4:   else
5:     Optimal Fixture and Association List  $\mapsto$  MySQL Database  $\triangleright$  Send to Mysql
6:   end if
7: end procedure

```

6.4 Implementation of the Genetic Algorithm

We now present and discuss our implementation of GA.

6.4.1 Phase I of GA

In order to initialise GA, a population of N randomly permuted individuals is created. Each individual within this population represents a fixture list containing randomly paired teams. These fixtures are created without taking travel distance, pair clashes, or any other constraints into consideration. Each individual is checked for constraint violations, and if any individual is found to violate a constraint, then this is remedied by finding an immediate feasible neighbourhood solution, i.e., a solution which does not violate any of the constraints.

When all of the individuals in the population are feasible, we then proceed to evaluate the strength of their corresponding solution, by calculating the total distance travelled by teams within each individual. The individual with the lowest cumulative travel distance is recorded as our current global optimum.

We then progress by partitioning our N individuals into groups of four. This is done, so that each group can undergo tournament selection. Within each group, we evaluate each individual and find the individual which yields the lowest total travelling distance. The other three individuals from the group of four individuals are discarded. The remaining individual is mutated in three different ways to create three new children using the *swap mutation*, *insertion mutation* and *inversion mutation*, described in Section 5.2.3. Each child is checked for constraint violations, and if any child is found to be infeasible, i.e., violate any constraints, then the child is replaced by an immediate feasible neighbourhood solution. Once all of this is done for all groups, we then evaluate the population as a whole, to find the child with the lowest travelling distance. If the total travel distance for this child is lower than the last recorded global optimum, then this becomes the new global optimum. The process is repeated N times, where N is a large positive number, to find the final optimal fixture list.

6.4.2 Phase II of GA

Once an optimal fixture list has been found in Phase I of GA, we create a population of N randomly permuted individuals again. However, each individual within this population will now make up a list of all associations. Each individual is assigned to the optimal fixture list found in *Phase I* of GA to check for any violations. One such violation is that the assigned association may be from the same country as either team playing in the assigned fixture. If any violations are found, then they are fixed by finding an immediate feasible neighbourhood solution.

When all of the individuals are deemed feasible, we then proceed to evaluate the strength of their respective solutions by calculating the total travel distance yielded by each individual. This total distance is comprised of the total distance between all associations, and all the *home* teams in the optimal fixture list. The individual which yields the lowest total travel distance between associations and *home* teams is saved and the corresponding distance is recorded as our global optimum.

We proceed by partitioning the N individuals into groups of four. Within each group, we evaluate each individual and identify the individual which yields the lowest distance within the group. The other three individuals from the group of four individuals are discarded. The remaining individual is mutated using the *swap*, *insertion* and *inversion* mutations, to create three new children. Each child is checked for feasibility. If any child is found to be infeasible, then it is replaced by an immediate feasible neighbourhood solution. Once all this is done for all groups, we then evaluate the population as a whole again and find the child which yields the lowest travelling distance. If the distance yielded by the child is lower than the last recorded global optimum, then this becomes the new global optimum. This process is repeated N times to find the final optimal association list.

6.4.3 Phase III of GA

In *Phase III* of GA, we verify that our fixture list and corresponding list of associations are feasible. Once we are satisfied that both lists do not violate any constraints, then these lists are augmented into one and stored. This solution is accessed when constructing the next stage of the competition, and will be utilised for the subsequent year of the CCL.

We now present a pseudo-code for the GA Algorithm.

```

1: procedure GENETIC ALGORITHM PHASE I
2:    $n \leftarrow \text{Number of teams}$ 
3:   Initialize  $\text{NumIter}$                                      ▷ Initialize number of iterations
4:    $\text{Pop} \leftarrow \{\}$ ,                                       ▷ create an empty population
5:   for  $i \leftarrow 1 : \text{PopSize}$  do
6:      $\text{Pop}_i \leftarrow \text{RANDPERM}(n)$                            ▷ Randomly populate population
7:     if  $\text{constraint violation} \in \text{Pop}_i$  then                 ▷ Check for violations
8:        $\text{Pop}_i \leftarrow \text{NEIGHBOUR}$ 
9:     end if
10:  end for
11:   $\text{GlobalMin} \leftarrow \infty$                                    ▷ Initialize Global Min
12:  for  $\text{iter} \leftarrow 1 : \text{NumIter}$  do
13:    for each  $\text{Pop}_i \in \text{Pop}$  do
14:      ASSESS SOLUTION STRENGTH( $\text{Pop}_i$ )

```

```
15:       $PopMin \leftarrow \mathbf{MIN}(Pop_i)$ 
16:      if  $PopMin \leq GlobalMin$  then
17:           $GlobalMin \leftarrow PopMin$ 
18:      end if
19:  end for
20:  for each  $Pop_G \in Pop$  do                                ▷ select four individuals
21:      for each  $Pop_j \in Pop_G$  do
22:           $\mathbf{ASSESS\ SOLUTION\ STRENGTH}(Pop_j)$ 
23:      end for
24:       $Best \leftarrow \mathbf{MIN}(Pop_G)$                                 ▷ find best of four
25:       $Children \leftarrow \mathbf{MUTATE}(Best)$ 
26:      for each  $Child_k \in Children$  do
27:          if  $constraint\ violation \in Child_k$  then                ▷ Violation Check
28:               $Child_k \leftarrow \mathbf{NEIGHBOUR}$ 
29:          end if
30:      end for
31:       $NewPop \leftarrow Children$                                 ▷ Children Create New Population
32:  end for
33:   $Pop \leftarrow NewPop$                                 ▷ New Population replaces Old Population
34:  end for
35: end procedure
```

```

1: procedure GENETIC ALGORITHM PHASE II
2:    $m \leftarrow \text{Number of Associations}$ 
3:   Initialize  $PopSize$ 
4:   Initialize  $NumIter$                                 ▷ Initialize number of iterations
5:    $Pop \leftarrow \{\}$ ,                                ▷ create an empty population
6:   for  $i \leftarrow 1 : PopSize$  do
7:      $Pop_i \leftarrow \text{RANDPERM}(m)$                 ▷ Randomly populate population
8:     if  $\text{constraint violation} \in Pop_i$  then          ▷ Check for violations
9:        $Pop_i \leftarrow \text{NEIGHBOUR}$ 
10:    end if
11:  end for
12:   $GlobalMin \leftarrow \infty$                         ▷ Initialize Global Min
13:  for  $iter \leftarrow 1 : NumIter$  do
14:    for each  $Pop_i \in Pop$  do
15:      ASSESS SOLUTION STRENGTH( $Pop_i$ )
16:       $PopMin \leftarrow \text{MIN}(Pop_i)$ 
17:      if  $PopMin \leq GlobalMin$  then
18:         $GlobalMin \leftarrow PopMin$ 
19:      end if
20:    end for
21:    for each  $Pop_G \in Pop$  do                        ▷ select four individuals
22:      for each  $Pop_j \in Pop_G$  do
23:        ASSESS SOLUTION STRENGTH( $Pop_j$ )
24:      end for
25:       $Best \leftarrow \text{MIN}(Pop_G)$                     ▷ find best of four
26:       $Children \leftarrow \text{MUTATE}(Best)$ 
27:      for each  $Child_k \in Children$  do

```

```

28:      if constraint violation  $\in$  Childk then                                ▷ Violation Check
29:          Childk  $\leftarrow$  NEIGHBOUR
30:      end if
31:  end for
32:      NewPop  $\leftarrow$  Children                                ▷ Children Create New Population
33:  end for
34:      Pop  $\leftarrow$  NewPop                                ▷ New Population replaces Old Population
35:  end for
36: end procedure

```

```

1: procedure GENETIC ALGORITHM PHASE III
2:   if constraint violation  $\in$  Optimal Fixture and Association List then
3:       Break
4:   else
5:       Optimal Fixture and Association List  $\mapsto$  MySQL Database    ▷ Send to Mysql
6:   end if
7: end procedure

```

6.5 Implementation of Tabu Search

We now discuss our implementation of TS.

6.5.1 Phase I of TS

In *Phase I* of our implementation of TS, we first create an action list, by identifying all possible manipulations which will be applied to our fixture list, using operators. The action list will then contain an operator identifier (ID), as well as the teams which will be affected by this ID. These

operators are the *swap*, *insertion* and *inversion* operators. We also initialise the *Tabu List*, by creating an empty list.

We produce a random initial fixture list, and determine whether it is feasible. If the initial fixture list is found to violate any constraints, then this is remedied by finding an immediate feasible neighbourhood solution. This fixture list becomes our *current solution*, and is recorded as our *best solution*. We then initialise a temporary solution which is recorded as *best new solution* and assign a very large value to it.

Our action list is then employed on the *current solution* by taking the first action set out on the action list. The first action is a simple swap between the first two teams in the *current solution*. We then measure the resultant travelling distance in the current solution. If this distance is lower than the distance in our *best new solution*, which we expect since we assigned a large value to the *best new solution*, then this new fixture list is recorded as our *best new solution*. We again employ our action list on the *current solution*, by taking the second action set out on the action list. The second action is a simple swap between the first team and the third team on the *current solution*. Again, we measure the travelling distance in the current solution. If this distance is lower than our *best new solution*, then this new fixture list is recorded as our *best new solution*. This is repeated for the length of the action list, i.e., until all actions have been applied, to get the optimal *best new solution*.

The optimal *best new solution* replaces the *current solution*, and the *Tabu List* is updated accordingly, i.e., the one move which produced the optimal *best new solution* is recorded. Finally, we compare the new *current solution* with the *best solution*. If the distance travelled by teams in the new *current solution* is lower than the distance travelled by teams in the *best solution*, then we make the new *current solution* our *best solution*, and repeat the process again for a maximum amount of iterations. The maximum number of iterations is user prescribed.

6.5.2 Phase II of TS

In *Phase II* of TS, an action list is created, by recording all possible manipulations that could be done on a list of associations. An initial, random list of associations is produced and then attached to the optimal fixture list found in *Phase I* of TS. The paired lists of fixtures and associations are checked for feasibility. If the initial association list is found to violate any constraints, then this is remedied by finding an immediate feasible neighbourhood solution, within the association list. We then call the feasible combined list, our *current solution*. This current solution is recorded as our *best solution*. We then initialise a temporary solution and call it *best new solution*, and assign a very large value to it.

The action list is employed on the *current solution* by taking the first action. The first action is a simple swap between the first two associations on the *current solution*. We then measure the distance between the *home* teams and the resultant list of associations. If the distance is lower than the distance in the *best new solution*, which it should be, then the optimal fixture list, combined with the current list of associations is recorded as our *best new solution*. Our action list is again employed on the *current solution* by carrying out the second action on the action list, and measuring the distance between the *home* teams and the resultant list of associations. If the resultant distance is lower than the distance in the *best new solution*, then the optimal fixture list augmented with the new list of associations is recorded as our *best new solution*. This is repeated for the length of the action list to get an optimal fixture list as well as an optimal list of associations.

We then make the optimal *best new solution* our *current solution* and update the *Tabu List* accordingly. Finally, we compare the new *current solution* with the *best solution*. If the distance between associations and *home* teams in the new *current solution* is lower than the distance between associations and *home* teams in the *best solution*, then we make the new *current solution* our *best solution* and repeat the process again for a maximum amount of iterations.

6.5.3 Phase III of TS

In *Phase III* we verify that our fixture list, and corresponding list of associations are feasible. Once both lists are deemed feasible, then they are augmented and stored. This solution is accessed when constructing the next stage of the competition, and will be utilised when building the next edition/year of the CCL.

We now present a pseudo-code for the TS Algorithm.

```

1: procedure TABU SEARCH PHASE I
2:    $ActList \leftarrow \mathbf{CREATEACTLIST}(m)$  ▷ Create fixture Action List
3:    $nAct \leftarrow \mathbf{LENGTH}(ActList)$ 
4:    $TL \leftarrow \mathbf{ROUND}(0.5 \times nAct)$  ▷ Initialize Tabu List
5:    $TC \leftarrow \mathbf{ZEROS}(nAct, 1)$  ▷ Initialize Tabu Counter
6:    $Solution \leftarrow \mathbf{RANDPERM}(n)$  ▷ Randomly permute a solution
7:   if  $constraint\ violation \in Solution$  then ▷ Check for violations
8:      $Solution \leftarrow \mathbf{NEIGHBOUR}$ 
9:   end if
10:   $\mathbf{ASSESS\ SOLUTION\ STRENGTH}(Solution)$ 
11:   $BestSolution \leftarrow Solution$  ▷ Initialize Best Solution
12:  for  $iter \leftarrow 1 : MaxIter$  do
13:     $bestnewsol \leftarrow \infty$ 
14:    for  $i \leftarrow 1 : nAct$  do
15:      if  $TC(i) == 0$  then
16:         $newsol \leftarrow \mathbf{DOACT}(Solution, ActList(i))$ 

```

```

17:      if constraint violation  $\in$  newsol then                                ▷ Check violations
18:          newsol  $\leftarrow$  Solution                                           ▷ Reverse Action
19:      end if
20:      ASSESS SOLUTION STRENGTH(newsol)
21:      newsol.ActIndex  $\leftarrow$  i
22:      if newsol  $\leq$  bestnewsol then
23:          bestnewsol  $\leftarrow$  newsol
24:      end if
25:  end if
26:  end for
27:  Solution  $\leftarrow$  bestnewsol                                           ▷ Update Current Solution
28:  for i  $\leftarrow$  1 : nAct do
29:      if i == bestnewsol.ActIndex then
30:          TC(i)  $\leftarrow$  TL                                           ▷ Add to Tabu List
31:      else
32:          TC(i)  $\leftarrow$  MAX(TC(i) - 1, 0)                               ▷ Reduce Tabu Counter
33:      end if
34:  end for
35:  if Solution  $\leq$  BestSolution then
36:      BestSolution  $\leftarrow$  Solution
37:  end if
38:  end for
39: end procedure

```

```

1: procedure TABU SEARCH PHASE II
2:    $ActList \leftarrow \mathbf{CREATEACTIONLIST}(m)$                                 ▷ Create association Action List
3:    $nAct \leftarrow \mathbf{LENGTH}(ActList)$ 
4:    $TL \leftarrow \mathbf{ROUND}(0.5 \times nAct)$                                 ▷ Initialize Tabu List
5:    $TC \leftarrow \mathbf{ZEROS}(nAct, 1)$                                 ▷ Initialize Tabu Counter
6:    $Solution \leftarrow \mathbf{RANDPERM}(n)$                                 ▷ Randomly permute a solution
7:   if  $constraint\ violation \in Solution$  then                                ▷ Check for violations
8:      $Solution \leftarrow \mathbf{NEIGHBOUR}$ 
9:   end if
10:   $\mathbf{ASSESS\ SOLUTION\ STRENGTH}(Solution)$ 
11:   $BestSolution \leftarrow Solution$                                 ▷ Initialize Best Solution
12:  for  $iter \leftarrow 1 : MaxIter$  do
13:     $bestnewsol \leftarrow \infty$ 
14:    for  $i \leftarrow 1 : nAct$  do
15:      if  $TC(i) == 0$  then
16:         $newsol \leftarrow \mathbf{DOACT}(Solution, ActList(i))$ 
17:        if  $constraint\ violation \in newsol$  then                                ▷ Check violations
18:           $newsol \leftarrow Solution$                                 ▷ Reverse Action
19:        end if
20:         $\mathbf{ASSESS\ SOLUTION\ STRENGTH}(newsol)$ 
21:         $newsol.ActIndex \leftarrow i$ 
22:        if  $newsol \leq bestnewsol$  then
23:           $bestnewsol \leftarrow newsol$ 
24:        end if
25:      end if
26:    end for

```

```

27:    $Solution \leftarrow bestnewsol$  ▷ Update Current Solution
28:   for  $i \leftarrow 1 : nAct$  do
29:     if  $i == bestnewsol.ActIndex$  then
30:        $TC(i) \leftarrow TL$  ▷ Add to Tabu List
31:     else
32:        $TC(i) \leftarrow \mathbf{MAX}(TC(i) - 1, 0)$  ▷ Reduce Tabu Counter
33:     end if
34:   end for
35:   if  $Solution \leq BestSolution$  then
36:      $BestSolution \leftarrow Solution$ 
37:   end if
38: end for
39: end procedure

```

1: **procedure** TABU SEARCH PART III

```

2:   if  $constraint\ violation \in Optimal\ Fixture\ and\ Association\ List$  then
3:     Break
4:   else
5:      $Optimal\ Fixture\ and\ Association\ List \mapsto \mathbf{MySQL\ Database}$  ▷ Send to Mysql
6:   end if
7: end procedure

```

Chapter Seven

Results

In this chapter, we present our results for the meta-heuristic methods discussed in Chapters 5 and 6. We begin by presenting the results generated, based on data from the 2015, 2016, 2017 and 2018 editions of the CCL. For comparisons, two criteria are used to evaluate the performance of these methods. The criteria used are solution quality and speed, which are discussed in Section 7.1. The results generated by the best performing meta-heuristic, in terms of these criteria, is compared directly with existing fixtures, in Section 7.2.

All the tests in this chapter were performed on a PC with an Intel Core i5 CPU at 2.5 GHz with 8GB RAM. All Algorithms were coded in MATLAB 2015a 64 bit.

7.1 Results and discussion on the performance of SA, GA and TS

We compare the performance of SA, GA and TS by using the two performance criteria. Each criterion is used to evaluate the performance of the three meta-heuristics, in scheduling fixtures for the knock-out stages and the group stages of the CCL. We begin by presenting a discussion on the first criteria used to measure the performance of each meta-heuristic, solution quality.

7.1.1 Solution Quality

Solution quality refers to the magnitude of the total distance travelled by teams and match officials alike, as produced by each of the meta-heuristic methods. In Table 7.1, the travelling distance generated for teams and match officials in each stage of the CCL knock-out rounds, using data from the past four editions of the tournament, is tabulated. Data refers to the participating teams and associations, and geographical information of these participating teams and associations. This data was maintained, in order to make a fair comparison against existing fixtures for each year. The best solution, i.e., lowest distance produced, at each stage for every year, is highlighted in bold.

CCL Knock-out Stages						
	Total Travel distances (km) generated by SA		Total Travel distances (km) generated by GA		Total Travel distances (km) generated by TS	
	Team	Association	Team	Association	Team	Association
2015 Preliminary Stage	40 808	44 040	36 804	45 705	33 402	47 111
2015 First Knock-out Stage	42 120	33 974	38 476	38 824	39 610	33 260
2015 Second Knock-out Stage	38 414	24 879	38 390	30 391	20 710	24 975
2016 Preliminary Stage	37 450	41 414	37 864	40 701	35 528	34 434
2016 First Knock-out Stage	46 498	36 985	42 515	39 060	54 240	36 055
2016 Second Knock-out Stage	39 103	18 812	35 384	33 695	30 127	20 039
2017 Preliminary Stage	37 896	40 586	37 850	40 658	36 958	37 078
2017 First Knock-out Stage	44 936	34 299	43 338	38 692	47 906	32 439
2018 Preliminary Stage	41 512	48 538	35 604	49 819	47 106	42 663
2018 First Knock-out Stage	41 192	28 688	34 468	34 980	38 680	24 890

Table 7.1 The distances generated by SA, GA, and TS, respectively, for the knock-out stages of the CCL.

From the results found in Table 7.1, we notice that TS outperforms SA and GA in the key criteria of solution quality. The superior performance of TS is again evident when comparing the combined distance travelled by teams and match officials, as is illustrated in Table 7.2.

CCL Knock-out Stages			
	Combined Travel distance(<i>km</i>) generated by SA	Combined Travel distance(<i>km</i>) generated by GA	Combined Travel distance(<i>km</i>) generated by TS
2015 Preliminary Stage	84 848	82 509	80 513
2015 First Knock-out Stage	76 094	77 300	72 870
2015 Second Knock-out Stage	63 293	68 781	45 685
2016 Preliminary Stage	78 864	78 565	69 962
2016 First Knock-out Stage	83 483	81 575	90 295
2016 Second Knock-out Stage	57 915	69 079	50 166
2017 Preliminary Stage	78 482	78 508	74 036
2017 First Knock-out Stage	79 235	82 028	80 345
2018 Preliminary Stage	90 050	82 423	89 769
2018 First Knock-out Stage	68 880	69 448	63 570

Table 7.2 The combined distances obtained by TS, SA and GA, respectively, for the knock-out stages of the tournament.

We now present a summary of the best performing meta-heuristic for each knock-out stage of the CCL, in each year:

CCL Knock-out Stages	
	Best performing meta-heuristic
2015 Preliminary Stage	TS
2015 First Knock-out Stage	TS
2015 Second Knock-out Stage	TS
2016 Preliminary Stage	TS
2016 First Knock-out Stage	GA
2016 Second Knock-out Stage	TS
2017 Preliminary Stage	TS
2017 First Knock-out Stage	TS
2018 Preliminary Stage	GA
2018 First Knock-out Stage	TS

Table 7.3 The best performing meta-heuristic for the knock-out stages of the tournament.

We should note, however, that the superior performance of TS is not consistent in every stage of the tournament. In fact, we found that GA performed better in the group stages of the CCL, as shown in Table 7.4.

CCL Group Stages						
	Total Travel distance (km) generated by SA		Total Travel distance (km) generated by GA		Total Travel distance (km) generated by TS	
Year	Team	Association	Team	Association	Team	Association
2015	71 916	54 924	69 638	59 673	69 638	55 737
2016	100 160	34 492	94 906	46 689	94 906	40 172
2017	155 819	90 604	155 816	89 158	155 816	89 269
2018	91 522	82 490	91 522	76 651	92 314	77 974

Table 7.4 The distances generated by SA, GA, and TS, respectively, for the group stages of the CCL.

In Table 7.4, we see that the results for distances generated for teams by GA and TS are the same for 2015, 2016 and 2017. However, GA performed better than TS when allocating associations for the 2017 and 2018 editions of the CCL. The superior performance of GA is also illustrated when comparing combined travel distances in the group stages, as shown in Table 7.5.

CCL Group Stages			
Year	Combined Travel distance(<i>km</i>) generated by SA	Combined Travel distance(<i>km</i>) generated by GA	Combined Travel distance(<i>km</i>) generated by TS
2015	126 840	129 311	125 375
2016	134 652	141 595	135 078
2017	246 423	244 974	245 085
2018	174 012	168 173	170 288

Table 7.5 The combined distances obtained by TS, SA and GA, respectively, for the group stages of the tournament.

We summarize our findings for the best performing meta-heuristic method in the Group stages of the CCL, in Table 7.6:

CCL Group Stages	
Year	Best performing meta-heuristic
2015	TS
2016	SA
2017	GA
2018	GA

Table 7.6 The best performing meta-heuristic for the group stages of the tournament.

Based on the results presented in Tables 7.1 - 7.6, we can conclude that TS exhibited superior performance in the knock-out stages of the CCL, while GA performed better in the group stages. We now look at the second performance criteria which we used to measure the efficacy of each meta-heuristic, speed.

7.1.2 Speed

In the context of this research, we take *Speed* to mean the CPU time taken for an algorithm to produce a feasible solution. Table 7.7 shows the CPU time taken to produce feasible fixture lists, and efficient allocations of associations. The CPU time, in seconds, was measured for each stage of the CCL knock-out rounds, using data from the 2015, 2016, 2017 and 2018 editions of the tournament. The method which exhibited superior performance is highlighted in bold.

CCL Knock-out Stages						
	Time Taken (s) by SA		Time Taken (s) by GA		Time Taken (s) by TS	
	Team	Association	Team	Association	Team	Association
2015 Preliminary Stage	21.37	82.95	149.47	360.98	145.02	290.67
2015 First Knock-out Stage	12.51	82.71	397.68	819.5	59.99	307.38
2015 Second Knock-out Stage	25.69	329.76	1 201.75	598.04	13.32	298.55
2016 Preliminary Stage	20.92	80.73	1 591.88	1 181.66	129.57	318.3
2016 First Knock-out Stage	13.48	81.29	512.79	1 011.98	63.82	333.06
2016 Second Knock-out Stage	24.64	373.51	502.24	987.32	15.2	344.67
2017 Preliminary Stage	20.78	79.67	1 751.7	1 012.06	135.52	317.75
2017 First Knock-out Stage	13.32	80.54	501.25	979.34	64.11	317.86
2018 Preliminary Stage	22.74	90.15	1 661.2	1 038.44	181.4	356.48
2018 First Knock-out Stage	12.95	82.88	504.97	999.11	63.38	326.21

Table 7.7 A summary of the CPU time taken by SA, GA, and TS, respectively, for the knock-out stages of the CCL.

Based on the results in Table 7.7, we observe that overall, SA is by far the fastest meta-heuristic method. TS has the lowest CPU time, based on data from the 2015 and 2016 second knock-out rounds. We attribute this to the fact that the number of constraints increases as we progress through the tournament, and TS is able to navigate these constraints better than SA which requires more CPU time. GA is the worst performer in terms of CPU time. We believe that this is due to the number of individuals which we used in the population, and the fact that these individuals have to be thoroughly examined for constraint violations.

We now present a summary of the fastest meta-heuristic based on data for each knock-out stage of the CCL, in each year:

CCL Knock-out Stages	
	Fastest meta-heuristic
2015 Preliminary Stage	SA
2015 First Knock-out Stage	SA
2015 Second Knock-out Stage	TS
2016 Preliminary Stage	SA
2016 First Knock-out Stage	SA
2016 Second Knock-out Stage	TS
2017 Preliminary Stage	SA
2017 First Knock-out Stage	SA
2018 Preliminary Stage	SA
2018 First Knock-out Stage	SA

Table 7.8 The fastest meta-heuristic based on data from the knock-out stages of the tournament.

Table 7.9 displays the CPU time taken to produce feasible fixture lists, and efficient allocations of associations.

CCL Group Stages						
Year	Time Taken (sec) by SA		Time Taken (sec) by GA		Time Taken (sec) by TS	
	Team	Association	Team	Association	Team	Association
2015	0.29	327.86	0.67	98.64	0.06	272.29
2016	0.273	328.58	0.68	98.79	0.03	282.91
2017	0.413	327.6	1.09	1 033.15	0.14	314.89
2018	0.39	330.57	1.49	1 186.14	0.11	321.26

Table 7.9 A summary of the CPU time taken by SA, GA, and TS, respectively, using data from the group stages of the CCL.

In Table 7.9, we see that GA was the fastest meta-heuristic to assign associations based on data from the 2015 and 2016 edition of the CCL. However, for the 2017 and 2018 editions of the CCL it was the slowest. TS was the fastest at scheduling group stage fixtures for all the CCL editions examined. TS was also the fastest at assigning associations to these fixtures for the 2017 and 2018 editions of the CCL. Although, we observed that TS was the fastest at scheduling group stage fixtures, GA used less CPU time for the 2015 and 2016 editions of the CCL. This is based on summing both the time to schedule group stage fixtures and allocate associations.

Here, we present the fastest meta-heuristic for the group stages of the CCL, in each year:

CCL Group Stages	
Year	Fastest meta-heuristic
2015	GA
2016	GA
2017	TS
2018	TS

Table 7.10 The fastest meta-heuristic based on data from the group stages of the tournament.

When we take into account the performance criteria set out in the beginning of this chapter, we made no assumptions of the general importance of one criterion over the other. However, when choosing an overall best meta-heuristic, we found that TS generally found more optimal fixtures and association assignments for the knock-out stages of the CCL. This is shown in Table 7.1, confirmed in Table 7.2 and summarised in Table 7.3. In the group stages, however, we observed that TS was only able to outperform the other meta-heuristics for the 2015 edition of the CCL, as shown in Table 7.6.

Although, TS may not be the fastest meta-heuristic to construct optimal fixtures and association assignments, we observed that, it was the most consistent and best performing meta-heuristic method, over every stage and edition of the CCL. TS also demonstrated the ability to handle additional constraints as the tournament progressed without a proportionate increase in CPU time. We, therefore, conclude that TS exhibited superior performance over the other tested meta-heuristic methods. We compare the results generated by TS with the existing fixtures, since TS is the superior performing method.

7.2 Comparison of Existing Schedules and TS Generated Schedules

In this section, we compare the results generated by TS with existing fixtures, which were published and used for the 2017 edition of the CCL. We have specifically used the 2017 edition of the CCL, because the 2017 edition features 16 teams in the group stages as opposed to eight teams, as is the case for the 2015 and 2016 edition. These different structures were discussed in Section 2.1. We further motivate using the 2017 edition for comparison because this edition of the CCL has more constraints than the 2018 edition. This is due to the three knock-out stage fixtures from the 2016 edition of the CCL, which are used as constraints for the 2017 edition of the CCL. We begin with the comparison.

7.2.1 Knock-Out Stage Comparison

We present the fixture list produced by the IOC, and the TS-generated fixture list, in Tables 7.11 and 7.12, respectively. Each column in these Tables, is labelled descriptively. Column 1, labelled “Home” represents the *home* team, column 2 represents the *away* team, column 3 provides the list of associations required to provide match officials for each game, while columns 4 through 5 represents the distance between the *home* and *away* team, and the distance between the *home* team

and the association, respectively. Column 6 provides the total distance for each game, which is calculated as the sum of the distances in columns 4 and 5. In columns 1 and 2, the country of origin of each team is provided as a subscript.

Existing 2017 Preliminary Stage fixtures					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Ngaya Club _(Comoros)	Young Africans _(Tanzania)	Seychelles	695	1 559	2 254
Africans _(Tanzania)	Ngaya Club _(Comoros)	Uganda	695	1 093	1 788
Horoya _(Guinea)	US Goree _(Senegal)	Mali	707	709	1 416
US Goree _(Senegal)	Horoya _(Guinea)	Guinea Bissau	707	373	1 080
Barack Young Controllers _(Liberia)	Stade Malien _(Mali)	Sierra Leone	771	360	1 131
Stade Malien _(Mali)	Barack Young Controllers _(Liberia)	Morocco	771	2 375	3 146
AC Leopards _(Congo)	UMS de Loum _(Cameroon)	Gabon	977	627	1 604
UMS de Loum _(Cameroon)	AC Leopards _(Congo)	Equatorial Guinea	977	114	1 091
AS FAN _(Niger)	AS Tanda _(Ivory Coast)	Nigeria	1 134	771	1 905
AS Tanda _(Ivory Coast)	AS FAN _(Niger)	Burkina Faso	1 134	832	1 966
CAPS United _(Zimbabwe)	Lioli _(Lesotho)	Botswana	1 327	922	2 249
Lioli _(Lesotho)	CAPS United _(Zimbabwe)	Namibia	1 327	1 286	2 613
APR _(Rwanda)	Zanaco _(Zambia)	Burundi	1 509	178	1 687
Zanaco _(Zambia)	APR _(Rwanda)	Zimbabwe	1 509	396	1 905
Ferrovioario Beira _(Mozambique)	Zimamoto _(Zanzibar)	Swaziland	1 592	815	2 407
Zimamoto _(Zanzibar)	Ferrovioario Beira _(Mozambique)	Uganda	1 592	1 035	2 627
Gambia Ports Authority _(Gambia)	Sewe Sport _(Ivory Coast)	Cape Verde	1 663	755	2 418
Sewe Sport _(Ivory Coast)	Gambia Ports Authority _(Gambia)	Niger	1 663	1134	2 797
AS Real Bamako _(Mali)	Rivers United _(Nigeria)	Benin	1 866	1 352	3 218
Rivers United _(Nigeria)	AS Real Bamako _(Mali)	Ghana	1 866	804	2 670
Atlabara _(South Sudan)	Coton Sport _(Cameroon)	Djibouti	2 073	1 436	3 509
Coton Sport _(Cameroon)	Atlabara _(South Sudan)	DR Congo	2 073	1 532	3 605
CF Mounana _(Gabon)	Vital'O _(Burundi)	Chad	2 251	1 442	3 693
Vital'O _(Burundi)	CF Mounana _(Gabon)	Tanzania	2 251	1 154	3 405
Cote d'Or _(Seychelles)	Saint George _(Ethiopia)	Mauritius	2 291	1 784	4 075
Saint George _(Ethiopia)	Cote d'Or _(Seychelles)	Sudan	2 291	1 143	3 434
CNaPS Sport _(Madagascar)	Township Rollers _(Botswana)	Mozambique	2 318	1 718	4 036
Township Rollers _(Botswana)	CNaPS Sport _(Madagascar)	Lesotho	2 318	547	2 865
1 de Agosto _(Angola)	Kampala City Council Authority _(Uganda)	Congo	2 379	567	2 946
Kampala City Council Authority _(Uganda)	1 de Agosto _(Angola)	Kenya	2 379	509	2 888

Existing 2017 Preliminary Stage fixtures Continued					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Diablos Noir _(Congo)	Rail Club du Kadiogo _(Burkina Faso)	Central African Republic	2 625	1 033	3 658
Rail Club du Kadiogo _(Burkina Faso)	Diablos Noir _(Congo)	Togo	2 625	755	3 380
Bidvest Wits _(South Africa)	Saint-Louisienne _(Reunion)	Zambia	2 837	1 203	4 040
Saint-Louisienne _(Reunion)	Bidvest Wits _(South Africa)	Comoros	2 837	1 678	4 515
Al-Merriekh _(Sudan)	Sony Ela Nguema _(Equatorial Guinea)	Somalia	2 915	2 068	4 983
Sony Ela Nguema _(Equatorial Guinea)	Al-Merriekh _(Sudan)	Cameroon	2 915	304	3 219
FUS Rabat _(Morocco)	Johansen _(Sierra Leone)	Algeria	2 917	942	3 859
Johansen _(Sierra Leone)	FUS Rabat _(Morocco)	Mauritania	2 917	1 108	4 025
Al-Ahly Tripoli _(Libya)	Wa All Stars _(Ghana)	Egypt	2 980	1 771	4 751
Wa All Stars _(Ghana)	Al-Ahly Tripoli _(Libya)	Liberia	2 980	1 145	4 125
AS Vita Club _(DR Congo)	Royal Leopards _(Swaziland)	South Africa	2 987	2 788	5 775
Royal Leopards _(Swaziland)	AS Vita Club _(DR Congo)	Angola	2 987	2 723	5 710
AS Port-Louis _(Mauritius)	Tusker _(Kenya)	Madagascar	3 091	1 063	4 154
Tusker _(Kenya)	AS Port-Louis _(Mauritius)	Ethiopia	3 091	1 155	4 246
Enugu Rangers _(Nigeria)	JS Saoura _(Algeria)	Ivory Coast	3 403	1 280	4 683
JS Saoura _(Algeria)	Enugu Rangers _(Nigeria)	Senegal	3 403	3 188	6 591
		Total	94 616	53 526	148 142

Table 7.11 Existing 2017 Preliminary Stage fixture list.

TS-generated 2017 Preliminary Stage fixtures					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Young Africans _(Tanzania)	Zimamoto _(Zanzibar)	Comoros	77	695	772
Zimamoto _(Zanzibar)	Young Africans _(Tanzania)	Kenya	77	607	684
Sony Ela Nguema _(Equatorial Guinea)	UMS de Loum _(Cameroon)	Nigeria	112	607	719
UMS de Loum _(Cameroon)	Sony Ela Nguema _(Equatorial Guinea)	Congo	112	1 111	1 223
Horoya _(Guinea)	Johansen _(Sierra Leone)	Guinea Bissau	128	336	464
Johansen _(Sierra Leone)	Horoya _(Guinea)	Liberia	128	367	495
APR _(Rwanda)	Vital'O _(Burundi)	Ethiopia	179	1 550	1 729
Vital'O _(Burundi)	APR _(Rwanda)	Uganda	179	548	727
AS Port-Louis _(Mauritius)	Saint-Louisienne _(Reunion)	Seychelles	246	1 755	2 001
Saint-Louisienne _(Reunion)	AS Port-Louis _(Mauritius)	Madagascar	246	868	1 114
Bidvest Wits _(South Africa)	Royal Leopards _(Swaziland)	Lesotho	318	357	675

Results

TS-generated 2017 Preliminary Stage fixtures Continued					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Royal Leopards _(Swaziland)	Bidvest Wits _(South Africa)	Mozambique	318	153	471
Rail Club du Kadiogo _(Burkina Faso)	Wa All Stars _(Ghana)	Niger	338	419	757
Wa All Stars _(Ghana)	Rail Club du Kadiogo _(Burkina Faso)	Ivory Coast	338	569	907
CAPS United _(Zimbabwe)	Zanaco _(Zambia)	Botswana	393	922	1 315
Zanaco _(Zambia)	CAPS United _(Zimbabwe)	DR Congo	393	1 886	2 279
Kampala City Council Authority _(Uganda)	Tusker _(Kenya)	Rwanda	507	378	885
Tusker _(Kenya)	Kampala City Council Authority _(Uganda)	Burundi	507	871	1 378
Lioli _(Lesotho)	Township Rollers _(Botswana)	Swaziland	541	490	1 031
Township Rollers _(Botswana)	Lioli _(Lesotho)	South Africa	541	274	815
1 de Agosto _(Angola)	AS Vita Club _(DR Congo)	Namibia	563	1 566	2 129
AS Vita Club _(DR Congo)	1 de Agosto _(Angola)	Central African Republic	563	1 036	1 599
AC Leopards _(Congo)	CF Mounana _(Gabon)	Cameroon	624	906	1 530
CF Mounana _(Gabon)	AC Leopards _(Congo)	Equatorial Guinea	624	381	1 005
Coton Sport _(Cameroon)	Enugu Rangers _(Nigeria)	Chad	726	361	1 087
Enugu Rangers _(Nigeria)	Coton Sport _(Cameroon)	Benin	726	538	1 264
Barack Young Controllers _(Liberia)	Sewe Sport _(Ivory Coast)	Sierra Leone	763	360	1 123
Sewe Sport _(Ivory Coast)	Barack Young Controllers _(Liberia)	Togo	763	587	1 350
AS Real Bamako _(Mali)	Gambia Ports Authority _(Gambia)	Guinea	946	696	1 642
Gambia Ports Authority _(Gambia)	AS Real Bamako _(Mali)	Senegal	946	162	1 108
FUS Raba _(Morocco)	JS Saoura _(Algeria)	Mauritania	950	1 992	2 942
JS Saoura _(Algeria)	FUS Rabat _(Morocco)	Tunisia	950	630	1 580
Stade Malien _(Mali)	US Goree _(Senegal)	Morocco	1 049	2 375	3 424
US Goree _(Senegal)	Stade Malien _(Mali)	Cape Verde	1 049	652	1 701
Al-Merriekh _(Sudan)	Saint George _(Ethiopia)	Egypt	1 161	1 614	2 775
Saint George _(Ethiopia)	Al-Merriekh _(Sudan)	Djibouti	1 161	718	1 879
AS Tanda _(Ivory Coast)	Rivers United _(Nigeria)	Ghana	1 222	423	1 645
Rivers United _(Nigeria)	AS Tanda _(Ivory Coast)	Gabon	1 222	561	1 783
Ferrovial Beira _(Mozambique)	Ngaya Club _(Comoros)	Zimbabwe	1 275	459	1 734
Ngaya Club _(Comoros)	Ferrovial Beira _(Mozambique)	Tanzania	1 275	705	1 980
CNaPS Sport _(Madagascar)	Cote d'Or _(Seychelles)	Mauritius	1 851	1 057	2 908
Cote d'Or _(Seychelles)	CNaPS Sport _(Madagascar)	Somalia	1 851	1 354	3 205
Atlabara _(South Sudan)	Diablos Noir _(Congo)	Sudan	2 084	1 191	3 275
Diablos Noir _(Congo)	Atlabara _(South Sudan)	Angola	2 084	553	2 637
Al-Ahly Tripoli _(Libya)	AS FAN _(Niger)	Algeria	2 426	1 024	3 450
AS FAN _(Niger)	Al-Ahly Tripoli _(Libya)	Burkina Faso	2 426	414	2 840
		Total	36 958	37 078	74 036

Table 7.12 TS-generated 2017 Preliminary Stage fixture list.

In Table 7.11, we observed that the largest distance travelled by an away team is 3 403 km. This distance is 28.7 % higher than the largest travel distance of 2 426 km, generated by TS, see Table 7.12. Furthermore, we found that there are at least 16 teams which were required to travel distances larger than 2 426 km, see Table 7.11. The combined distance travelled by these 16 teams in the existing fixture list, is 47 510 km, whereas the cumulative distance travelled by all 46 teams in the TS-generated fixture list is 36 958 km.

We also explored referee assignments, and found that *JS Saoura of Algeria*, which had to travel over 3000 km for their *away* match to *Enugu Rangers of Nigeria*, were also required to pay for match officials coming in from Senegal, which is 3188 km away. This distance is 25.5 % higher than the largest TS-generated distance of 2 375 km, see Table 7.12. We found four other instances, where the distance between the association and the *home* team, either exceeds or is equal to the highest distance generated by TS, in this category.

Table 7.13 shows the difference in distances generated by TS and the current method used to schedule CCL fixtures. Column 4 in Table 7.13, presents the distance travelled in the existing schedule less the distance generated by TS, and is seen as the distance saved by TS. If this number is positive, then the distance generated by TS is less than that of the existing fixtures. Conversely, if this is a negative number, then the distance generated by TS is greater than that of the existing fixtures.

CCL Knock-out Stages						
	Travel distance (km) in Existing Fixtures		Travel distance (km) generated by TS		Distance Saving (km)	
	Team	Association	Team	Association	Team	Association
2015 Preliminary Stage	105 396	59 239	33 402	47 111	71 997	12 128
2015 First Knock-out Stage	110 874	76 214	39 610	33 260	71 264	42 954
2015 Second Knock-out Stage	46 068	55 657	20 710	24 975	25 358	30 682
2016 Preliminary Stage	95 164	55 892	35 528	34 434	59 636	21 458
2016 First Knock-out Stage	109 102	73 075	54 240	36 055	54 862	37 020
2016 Second Knock-out Stage	63 280	49 774	30 127	20 039	33 153	29 735
2017 Preliminary Stage	94 616	53 526	36 958	37 078	57 658	16 448
2017 First Knock-out Stage	117 691	73 816	47 906	32 439	69 785	41 377
2018 Preliminary Stage	135 974	70 960	47 106	42 663	88 841	28 297
2018 First Knock-out Stage	92 528	86 542	38 680	24 890	53 848	61 652

Table 7.13 Knock-out stage distance difference between TS and existing fixtures.

It is evident that TS was able to achieve remarkable improvements for scheduling fixtures as well as assigning match officials.

We now compare the results for the group stages of the CCL.

7.2.2 Group Stage Comparison

Before we delve into the group stage comparison, we illustrate how the teams were seeded and why certain fixtures are considered infeasible. Table 7.14 illustrates the seeding of teams according to their performances in the CAF competitions for the previous five seasons. The ranking points, which were obtained using CAF's five-year ranking system, are shown in parentheses.

2017 Group Stage Strength Groups			
Strength group I	Strength group II	Strength group III	Strength group IV
Al Ahly (45 points) Egypt	Al Hilal (20 points) Sudan	Al Merreikh (14 points) Sudan	Saint George (2 points) Ethiopia
Zamalek (36 points) Egypt	Espérance de Tunis (20 points) Tunisia	Coton Sport (12 points) Cameroon	Ferroviário Beira (0 points) Mozambique
Etoile du Sahel (31 points) Tunisia	Wydad Casablanca (16 points) Morocco	As Vita Club (12 points) DR Congo	Zanaco (0 points) Zambia
Mamelodi Sundowns (25 points) South Africa	USM Alger (16 points) Algeria	Al Ahly Tripoli (5 points) Libya	CAPS United (0 points) Zimbabwe

Table 7.14 2017 Group stage seeding arrangement.

We now move on to present the 2017 group stage fixture lists, which are based on the group stage seeding, in Table 7.14. Tables 7.15 and 7.16 show the existing fixture list, and the TS-generated fixture list, respectively. Columns 1 through 6 are identical to Tables 7.11 and 7.12.

Existing 2017 Group Stages					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Al-Ahly _(Egypt)	Zanaco _(Zambia)	Rwanda	5 077	3 574	8 651
Al-Ahly _(Egypt)	Coton Sport _(Cameroon)	Ivory Coast	2 973	4 618	7 591
Al-Ahly _(Egypt)	Wydad Casablanca _(Morocco)	Algeria	3 615	2 728	6 343
Al-Ahly Tripoli _(Libya)	Zamalek _(Egypt)	Morocco	1 755	1 858	3 613
Al-Ahly Tripoli _(Libya)	CAPS United _(Zimbabwe)	Botswana	5 954	6 542	12 496
Al-Ahly Tripoli _(Libya)	USM Alger _(Algeria)	Ethiopia	1 020	3 742	4 762
Al-Hilal _(Sudan)	Etoile du Sahel _(Tunisia)	Madagascar	3 123	4 182	7 305
Al-Hilal _(Sudan)	Ferroviario Beira _(Mozambique)	Ivory Coast	3 957	4 146	8 103
Al-Hilal _(Sudan)	Al-Merriekh _(Sudan)	South Africa	1	4 683	4 684
Al-Merriekh _(Sudan)	Al-Hilal _(Sudan)	Cameroon	1	2 645	2 646
Al-Merriekh _(Sudan)	Etoile du Sahel _(Tunisia)	Morocco	3 123	4 429	7 552
Al-Merriekh _(Sudan)	Ferroviario Beira _(Mozambique)	Ethiopia	3 956	1 008	4 964

Existing 2017 Group Stages Continued					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
AS Vita Club _(DRCongo)	Esperance de Tunis _(Tunisia)	Botswana	4 603	2 529	7 132
AS Vita Club _(DRCongo)	Mamelodi Sundowns _(South Africa)	Gambia	2 748	4 042	6 790
AS Vita Club _(DRCongo)	Saint George _(Ethiopia)	Zambia	2 870	1 883	4 753
CAPS United _(Zimbabwe)	USM Alger _(Algeria)	Burundi	6 753	1 620	8 373
CAPS United _(Zimbabwe)	Al-Ahly Tripoli _(Libya)	Kenya	5 954	1 957	7 911
CAPS United _(Zimbabwe)	Zamalek _(Egypt)	Angola	5 336	2 161	7 497
Coton Sport _(Cameroon)	Wydad Casablanca _(Morocco)	Mauritania	3 438	3 316	6 754
Coton Sport _(Cameroon)	Al-Ahly _(Egypt)	Nigeria	2 973	651	3 624
Coton Sport _(Cameroon)	Zanaco _(Zambia)	Gambia	3 207	3 301	6 508
Esperance de Tunis _(Tunisia)	Mamelodi Sundowns _(South Africa)	Mauritania	7 206	3 303	10 509
Esperance de Tunis _(Tunisia)	Saint George _(Ethiopia)	Burundi	4 372	4 884	9 256
Esperance de Tunis _(Tunisia)	AS Vita Club _(DRCongo)	Zambia	4 603	6 105	10 708
Etoile du Sahel _(Tunisia)	Ferrovial Beira _(Mozambique)	Burkina Faso	6 700	2 881	9 581
Etoile du Sahel _(Tunisia)	Al-Merriekh _(Sudan)	Zambia	3 123	5 998	9 121
Etoile du Sahel _(Tunisia)	Al-Hilal _(Sudan)	Egypt	3 123	2 027	5 150
Ferrovial Beira _(Mozambique)	Al-Merriekh _(Sudan)	Angola	3 956	2 622	6 578
Ferrovial Beira _(Mozambique)	Al-Hilal _(Sudan)	Zambia	3 957	850	4 807
Ferrovial Beira _(Mozambique)	Etoile du Sahel _(Tunisia)	Mali	6 700	5 924	12 624
Mamelodi Sundowns _(South Africa)	Saint George _(Ethiopia)	Egypt	3 823	6 233	10 056
Mamelodi Sundowns _(South Africa)	AS Vita Club _(DRCongo)	Morocco	2 748	7 606	1 0354
Mamelodi Sundowns _(South Africa)	Esperance de Tunis _(Tunisia)	Gabon	7 206	3 536	10 742
Saint George _(Ethiopia)	AS Vita Club _(DRCongo)	Rwanda	2 870	1 369	4 239
Saint George _(Ethiopia)	Esperance de Tunis _(Tunisia)	Senegal	4 372	6 163	10 535
Saint George _(Ethiopia)	Mamelodi Sundowns _(South Africa)	Algeria	3 823	4 887	8 710
USM Alger _(Algeria)	Al-Ahly Tripoli _(Libya)	Tunisia	1 020	633	1 653
USM Alger _(Algeria)	Zamalek _(Egypt)	South Africa	2 722	7 487	10 209
USM Alger _(Algeria)	CAPS United _(Zimbabwe)	Senegal	6 753	3 190	9 943
Wydad Casablanca _(Morocco)	Al-Ahly _(Egypt)	Angola	3 615	5 220	8 835
Wydad Casablanca _(Morocco)	Zanaco _(Zambia)	Gambia	6 642	2 484	9 126
Wydad Casablanca _(Morocco)	Coton Sport _(Cameroon)	South Africa	3 438	7 647	11 085
Zamalek _(Egypt)	CAPS United _(Zimbabwe)	Ivory Coast	5 336	4 618	9 954
Zamalek _(Egypt)	USM Alger _(Algeria)	Mauritania	2 722	4 955	7 677
Zamalek _(Egypt)	Al-Ahly Tripoli _(Libya)	Madagascar	1 755	5 721	7 476

Existing 2017 Group Stages Continued					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Zanaco _(Zambia)	Coton Sport _(Cameroon)	Tunisia	3 207	6 121	9 328
Zanaco _(Zambia)	Wydad Casablanca _(Morocco)	Gabon	6 642	2 720	9 362
Zanaco _(Zambia)	Al-Ahly _(Egypt)	Kenya	5 077	1 838	6 915
		Total	178 092	188 162	366 254

Table 7.15 Existing 2017 Group Stage fixture list.

TS-generated 2017 Group Stages					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Al-Ahly _(Egypt)	CAPS United _(Zimbabwe)	Chad	5 336	2 615	7 951
Al-Ahly _(Egypt)	USM Alger _(Algeria)	Djibouti	2 722	2 343	5 065
Al-Ahly _(Egypt)	Coton Sport _(Cameroon)	Libya	2 973	1 750	4 723
Al-Ahly Tripoli _(Libya)	Etoile du Sahel _(Tunisia)	Egypt	403	1 771	2 174
Al-Ahly Tripoli _(Libya)	Esperance de Tunis _(Tunisia)	Ivory Coast	506	3 540	4 046
Al-Ahly Tripoli _(Libya)	Zanaco _(Zambia)	Niger	5 610	2 427	8 037
Al-Hilal _(Sudan)	Al-Merriekh _(Sudan)	Central African Republic	1	1 972	1 973
Al-Hilal _(Sudan)	Saint George _(Ethiopia)	DR Congo	1 160	2 919	4 079
Al-Hilal _(Sudan)	Zamalek _(Egypt)	Rwanda	1 614	1 978	3 592
Al-Merriekh _(Sudan)	Saint George _(Ethiopia)	Burundi	1 161	2 143	3 304
Al-Merriekh _(Sudan)	Zamalek _(Egypt)	Ethiopia	1 615	1 008	2 623
Al-Merriekh _(Sudan)	Al-Hilal _(Sudan)	Somalia	1	2 068	2 069
AS Vita Club _(DRCongo)	Wydad Casablanca _(Morocco)	Cameroon	4 859	1 004	5 863
AS Vita Club _(DRCongo)	Mamelodi Sundowns _(SouthAfrica)	Equatorial Guinea	2 748	1 159	3 907
AS Vita Club _(DRCongo)	Ferrovial Beira _(Mozambique)	Gabon	2 731	838	3 569
CAPS United _(Zimbabwe)	USM Alger _(Algeria)	Madagascar	6 753	1 748	8 501
CAPS United _(Zimbabwe)	Al-Ahly _(Egypt)	Mozambique	5 336	913	6 249
CAPS United _(Zimbabwe)	Coton Sport _(Cameroon)	South Africa	3 589	981	4 570

TS-generated 2017 Group Stages Continued					
Team		Association	Distance		
Home	Away		Home - Away	Home - Association	Total
Coton Sport _(Cameroon)	CAPS United _(Zimbabwe)	Benin	3 589	1 228	4 817
Coton Sport _(Cameroon)	USM Alger _(Algeria)	Nigeria	3 232	651	3 883
Coton Sport _(Cameroon)	Al-Ahly _(Egypt)	Togo	2 973	1 387	4 360
Esperance de Tunis _(Tunisia)	Zanaco _(Zambia)	Algeria	6106	650	6 756
Esperance de Tunis _(Tunisia)	Etoile du Sahel _(Tunisia)	Guinea	107	3 857	3 964
Esperance de Tunis _(Tunisia)	Al-Ahly Tripoli _(Libya)	Morocco	506	1 583	2 089
Etoile du Sahel _(Tunisia)	Zanaco _(Zambia)	Burkina Faso	5 999	2 881	8 880
Etoile du Sahel _(Tunisia)	Al-Ahly Tripoli _(Libya)	Mali	403	3 186	3 589
Etoile du Sahel _(Tunisia)	Esperance de Tunis _(Tunisia)	Sierra Leone	107	3 890	3 997
Ferrovial Beira _(Mozambique)	Mamelodi Sundowns _(South Africa)	Comoros	960	1 273	2 233
Ferrovial Beira _(Mozambique)	AS Vita Club _(DRCongo)	Mauritius	2 731	2 368	5 099
Ferrovial Beira _(Mozambique)	Wydad Casablanca _(Morocco)	Zambia	7 443	850	8 293
Mamelodi Sundowns _(South Africa)	Wydad Casablanca _(Morocco)	Lesotho	7 607	403	8 010
Mamelodi Sundowns _(South Africa)	Ferrovial Beira _(Mozambique)	Swaziland	960	312	1 272
Mamelodi Sundowns _(South Africa)	AS Vita Club _(DRCongo)	Zimbabwe	2 748	936	3 684
Saint George _(Ethiopia)	Al-Hilal _(Sudan)	Kenya	1 160	937	2 097
Saint George _(Ethiopia)	Al-Merriekh _(Sudan)	Seychelles	1 161	2 294	3 455
Saint George _(Ethiopia)	Zamalek _(Egypt)	Uganda	2 673	993	3 666
USM Alger _(Algeria)	CAPS United _(Zimbabwe)	Gambia	6 753	3 251	10 004
USM Alger _(Algeria)	Coton Sport _(Cameroon)	Guinea Bissau	3 232	3 344	6 576
USM Alger _(Algeria)	Al-Ahly _(Egypt)	Tunisia	2 722	633	3 355
Wydad Casablanca _(Morocco)	AS Vita Club _(DRCongo)	Cape Verde	4 859	2 698	7 557
Wydad Casablanca _(Morocco)	Ferrovial Beira _(Mozambique)	Mauritania	7 443	1 985	9 428
Wydad Casablanca _(Morocco)	Mamelodi Sundowns _(South Africa)	Senegal	7 607	2 392	9 999
Zamalek _(Egypt)	Al-Merriekh _(Sudan)	Eritrea	1 615	1 819	3 434
Zamalek _(Egypt)	Al-Hilal _(Sudan)	Ghana	1 614	4 286	5 900
Zamalek _(Egypt)	Saint George _(Ethiopia)	Sudan	2 673	1 631	4 304
Zanaco _(Zambia)	Al-Ahly Tripoli _(Libya)	Angola	5 610	1 790	7 400
Zanaco _(Zambia)	Esperance de Tunis _(Tunisia)	Botswana	6 106	1 058	7 164
Zanaco _(Zambia)	Etoile du Sahel _(Tunisia)	Tanzania	5 999	1 526	7 525
		Total	155 816	89 269	245 085

Table 7.16 TS-generated 2017 Group Stage fixture list.

By investigating the existing group stage fixture list, in Table 7.15, we found that the highest distance travelled by a team was 7 206 km. This distance is less than the highest distance generated by TS, 7 607 km, for this category, see Table 7.16. However, the highest distance between an association and a *home* team in the existing group stage fixture list is 7 647 km, which is significantly higher than the maximum distance between an association and a *home* team generated by TS, 4 286 km, see Table 7.16. We found 19 other instances, where the distance between an association and a *home* team, exceeds the highest distance generated by TS, in this category. The combined distance travelled by the 19 match officials in the existing fixture list, is 109 841 km, whereas the cumulative distance travelled by all 48 match officials in the TS-generated fixture list is 89 269 km. In both existing and TS-generated group stage fixture lists, we found instances of teams travelling one kilometre. This is because two Sudanese teams qualified for the group stages, and these two teams were not in the same strength group when groups were created, see Table 7.14 for the different strength groups. These two Sudanese teams were, therefore, optimally allocated to the same group.

We now present Table 7.17, which shows the existing groups for the 2017 edition of the CCL. Within each group in Table 7.17, a team is listed in column 1, and all associated travel distances are given in column 2. The first distance is the distance travelled by the team in question, for all their *away* matches, and the second distance is the distance between the team in question and all the associations which provided match officials for this teams *home* games. Teams within each group are ordered according to their respective strength group allocation in Table 7.14. Thus, the first team within each group is from strength group one, the second team within each group is from strength group two, the third team within each group is from strength group three, and the fourth team within each group is from strength group four.

Existing 2017 Group A		
Team	Total Distance Travelled (km)	
	Team	Association
Étoile du Sahel	12 946	10 906
Al-Hilal	7 081	13 011
Al-Merrikh	7 080	8 082
Ferroviário Beira	14 613	9 396

Existing 2017 Group B		
Team	Total Distance Travelled (km)	
	Team	Association
Zamalek	9 813	15 294
USM Alger	10 495	11 310
Al-Ahli Tripoli	8 729	12 142
CAPS United	18 043	5 738

Existing 2017 Group C		
Team	Total Distance Travelled (km)	
	Team	Association
Mamelodi Sundowns	13 777	17 375
Espérance de Tunis	16 181	14 292
AS Vita Club	10 221	8 454
Saint George	11 065	12 419

Existing 2017 Group D		
Team	Total Distance Travelled (km)	
	Team	Association
Al-Ahly	11 665	10 920
Wydad Casablanca	13 695	15 351
Coton Sport	9 618	7 268
Zanaco	14 926	10 679

Table 7.17 Existing Groups for the 2017 edition of the CCL

We now present the groups generated by TS, in Table 7.18. We found that TS exhibited superior performance for allocating associations. Here, the maximum total distance travelled by match officials to one team, is 9 957 km, see Table 7.18. These results are significantly better than the maximum total distance travelled by match officials to one team in the existing fixture list, which is 17 375 km, see Table 7.17.

TS yielded worse results than the existing group stage fixture list, for 6 teams. *Wydad Casablanca of Morocco* was the worst of these in that the TS-generated grouping required *Wydad Casablanca of Morocco* to travel 19 909 km, as compared to only 13 695 km in the existing groups, see Tables 7.17 and 7.18, respectively.

TS-generated 2017 Group A		
Team	Total Distance Travelled (km)	
	Team	Association
Étoile du Sahel	6 509	9 957
Espérance de Tunis	6 719	6 090
Al-Ahli Tripoli	6 519	7 738
Zanaco	17 715	4 374

TS-generated 2017 Group B		
Team	Total Distance Travelled (km)	
	Team	Association
Zamalek	5 902	7 736
Al-Hilal	2 775	6 869
Al-Merrikh	2 777	5 219
Saint George	4 994	4 224

TS-generated 2017 Group C		
Team	Total Distance Travelled (km)	
	Team	Association
Mamelodi Sundowns	11 315	1 651
Wydad Casablanca	19 909	4 224
AS Vita Club	10 338	3 001
Ferrovário Beira	11 134	4 491

TS-generated 2017 Group D		
Team	Total Distance Travelled (km)	
	Team	Association
Al-Ahly	11 031	6 708
USM Alger	12 707	7 228
Coton Sport	9 794	3 266
CAPS United	15 678	3 642

Table 7.18 TS-generated Groups for the 2017 edition of the CCL

Table 7.19 illustrates the differences in distance between the existing group stage fixture lists and the TS-generated group stage fixture lists for the 2015, 2016 ,2017 and 2018 editions of the CCL. Here, we see that even though TS produced worse results for 6 teams, particularly in the 2017 CCL group stages, TS was still better overall at finding a grouping which reduced the total distance travelled by all teams.

CCL Group Stages						
Year	Travel distance (km) in Existing Fixtures		Travel distance (km) generated by TS		Distance Saving (km)	
	Team	Association	Team	Association	Team	Association
2015	71 866	86 437	69 638	55 737	2 228	30 700
2016	219 820	145 308	94 906	40 172	124 914	105 136
2017	178 092	188 162	155 816	89 269	22 276	98 893
2018	196 732	197 446	92 314	77 974	104 418	119 472

Table 7.19 Group stage distance difference between TS and existing fixtures.

Based on the results presented in this chapter, TS is a superior method for scheduling fixtures and assigning associations. We found that TS yields better schedules and assigns associations efficiently throughout the knock-out stages of the CCL as well as throughout the group stages of the CCL in all years under review. We observed that even though TS was the superior meta-heuristic, all the other meta-heuristics investigated produced better results than the existing method used to schedule fixtures and assign associations.

Chapter Eight

Conclusion

Our research objective is to optimally schedule fixtures for the CAF Champions League, which minimises the distance travelled by teams and match officials. To do this, we used, and compared the performance of three well-known meta-heuristics methods, namely Simulated Annealing, Genetic Algorithm, and Tabu Search. The existing data for the 2015, 2016, 2017 and 2018 editions of the CAF Champions League was used for this research. The performance of each meta-heuristic was evaluated and the overall best meta-heuristic method was compared with the published fixture lists from four editions of the CAF Champions League. Results show that Tabu Search is the most efficient method for scheduling CAF Champions League fixtures as well as assigning associations.

Although we found Tabu Search to be the meta-heuristic which yielded superior performance over the other methods, the results found in Chapter 7 demonstrate that all the meta-heuristics presented in this dissertation yield better results than the current methods used to schedule fixtures and assign associations, for the CAF Champions League.

Using Simulated annealing, Genetic algorithm, and Tabu search, we were able to generate feasible fixtures for the knock-out stages of the competition, which minimised travel distances significantly, as compared to the existing schedules. The manner in which we implemented these techniques allows for additional constraints to be included if need be.

In the current method, which is employed to schedule group stage fixtures, a random draw is made at a prestigious gala event, attended by the qualifying 16 teams. The draw is made to establish the groups by taking different strengths into consideration. Even though this promotes transparency, it is often inefficient in many other areas. Using the three meta-heuristic methods, we were able to generate fixtures for the group stage matches, which yielded far lower travel distances than the existing schedules. Furthermore, there were no biases or preferential considerations when generating these fixtures.

Overall, we successfully generated fixtures for the knock-out stages as well as the group stages of the CAF Champions League. We also assigned associations for these fixtures. Our generated fixtures and association assignments yielded far lower travel distances for teams and match officials.

8.1 Summary

In Chapter 1, we motivate why the CAF Champions League scheduling problem should be solved. The first reason is that travelling within Africa is expensive and as such, travel costs should be taken into consideration when scheduling fixtures. The second reason is based on the prerequisite that the CAF Champions League has to be made a priority over all national league games. Thus, when participating in the CAF Champions League, all teams need to reschedule their domestic league matches. This often means that teams need to set aside additional time to play make-up matches for their national league games. The final motivation for solving this problem, is the various impacts of travelling long distances, such as fatigue, which could affect the outcome of a match. As such, when scheduling fixtures, the distance travelled by teams should also be considered.

In Chapter 2, we discuss the former and present structure of the CAF Champions League, the different football associations affiliated with CAF, and the associations which have FIFA-approved

match officials. We also discuss the former and present five-year ranking system used by CAF, as well as the former and present earnings of progressing further into the CAF Champions League.

In Chapter 3, we present a review on some sports scheduling problems, as well as solution techniques for solving these problems. The review is divided into five parts. In the first part, we present important terminology which is used throughout this dissertation. The second part provides an overview of the most common types of existing scheduling problems. The third part presents some of the more popular scheduling techniques used to solve these scheduling problems, while the fourth part looks explicitly at existing football scheduling problems, and the solution techniques used to solve them. The last part of the review highlights some sports scheduling problems and their benchmark results.

In Chapter 4, we provide an informal description of the CAF Champions League scheduling problem. We also identify CAF, the Inter-clubs Organising Committee, national associations, clubs and football supporters as key stakeholders. We list some key requirements to take into consideration when scheduling fixtures for the CAF Champions League. These were obtained from available literature. These were incorporated into the problem, as constraints. Even if we were unable to identify an exhaustive list of constraints for the CAF scheduling problem, we are confident that additional constraints could easily be incorporated within our framework. Lastly, we mathematically formulate the CAF Champions League scheduling problem.

We considered many different scheduling techniques found in the literature, and concluded that meta-heuristic methods are the easiest to use in terms of applicability to our problem. We were careful to choose meta-heuristics which are very robust and popular. Simulated Annealing, Genetic Algorithm and Tabu Search were chosen and applied to our problem. We discuss these three meta-heuristic methods and their associated parameters in Chapter 5.

In Chapter 6, we discuss the implementation of Simulated Annealing, Genetic Algorithm, and Tabu Search, to solve the CAF Champions League scheduling problem. We present an overview of the model and then provide a summary of the scope and assumptions made when building the model. We make use of pseudo codes to further illustrate how these techniques were implemented.

In Chapter 7, computational results are presented, illustrating the effectiveness of the methods discussed in Chapters 5 and 6. The main objective of applying these techniques was to establish fixtures and assign associations for these fixtures such that the distances travelled by both teams and match officials alike throughout the CAF Champions League tournament, are minimised. We compare the performance of these methods on their ability to build a feasible fixture list and efficiently assign associations. We also investigated the speed with which this could be achieved. Results found in Section 7.1 provide proof that the meta-heuristic methods under consideration are a far better alternative to the current methods employed. After an in depth study of these methods, we found Tabu Search to be the superior meta-heuristic. A direct comparison of results generated by Tabu Search, and the published fixtures, is conducted in Section 7.2. We found that Tabu Search is by far more efficient at building optimal CAF Champions League fixtures, as well as assigning associations. We may not have had access to all information pertaining to stakeholder requirements and restrictions, and thus may have omitted some constraints in our model. However, the framework of the chosen meta-heuristic method can be easily adapted to incorporate additional constraints.

8.2 Further Work

The solutions presented herein provide a new and more efficient way for solving the real-world CAF Champions League scheduling problem. However, there is scope for further research into CAF Champions League scheduling. Here we present two recommendations for further research

8.2.1 Improved Requirements

The meta-heuristic approaches used could be adapted to include an improved list of requirements (constraints) from stakeholders. Making these requirements available to researchers could generate interest in finding optimal schedules for future editions of the tournament. Such research could improve the overall structuring, perception and experience of the CAF Champions League. These improvements could also inspire less teams to forfeit matches in the early stages of the tournament, and encourage greater overall participation in the CAF Champions League.

8.2.2 Match Official Criterion

We noticed a diversification of match official assignment in the early stages of the tournament. However, this is not necessarily the case as the tournament progresses. We assumed that this is because a certain level of expertise is required to officiate high-stake matches. Further work could be done to find an optimal assignment of match officials with the required credentials, in order to maintain diversification throughout the tournament.

Bibliography

- [1] Drexl A and Briskorn D. “A branch-and-price algorithm for scheduling sport leagues.” In: *Journal of the Operational Research Society* 60 (2009), pp. 84–93.
- [2] van Hentenryck P Anagnostopoulos A and Vergados Y. “A simulated annealing approach to the traveling tournament problem”. In: *Journal of Scheduling* 9.2 (2006), pp. 177–193.
- [3] Drexl A Bartsch T and Kröger S. “Scheduling the professional soccer leagues of Austria and Germany”. In: *Computers and Operations Research* 33.7 (2006), pp. 1907–1937.
- [4] CAF. *Draw procedure for CL group phase*. 2017. URL: <http://www.cafonline.com/en-US/NewsCenter/News/NewsDetails>.
- [5] CAF. *Member associations*. 2018. URL: <http://www.cafonline.com/en-us/memberassociations.aspx>.
- [6] CAF. *Regulation and official documents*. 2018. URL: <http://www.cafonline.com/en-us/caf/regulationsandofficial/statutes.acpx>.
- [7] Oliveri F Croce D and Oliveri D. “Scheduling the Italian football league: An ILP-based approach”. In: *Computers and Operations Research* 33.7 (2006), pp. 1963–1974.
- [8] Briskorn D and Knust S. “Constructing fair sports leagues schedules. Technical report”. In: *Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel* (2008).
- [9] Costa D. “An evolutionary tabu search algorithm and the NHL scheduling problem.” In: *INFOR: Inform Syst Opl Res* 33 (), pp. 161–178.
- [10] Goldberg D. “Genetic Algorithms in Search, Optimization, and Machine Learning”. unpublished. 1989.
- [11] Goossens D and Spieksma F. “Scheduling the Belgian Soccer League”. In: *Interfaces* 39.2 (2009), pp. 109–118.
- [12] Snyder E and Purdy D. “The home advantage in collegiate basketball.” In: *Sociology of sport journal* (1985), pp. 352–356.
- [13] Nemhauser G Easton K and Trick M. “Solving the Traveling Tournament Problem: A Combined Integer Programming and Constraint Programming Approach”. unpublished. 2001.

-
- [14] Nemhauser G Easton K and Trick M. "The traveling tournament problem: description and benchmarks." In: *Principles and Practice of Constraint Programming, Lecture Notes in Computer Science* 2239 (2001), pp. 580–584.
- [15] Glover F. "Tabu Search: A Tutorial." unpublished. 1990.
- [16] Hillier F and Lieberman G. "The timetable constrained distance minimization problem". In: *Introduction to Operations Research*. Singapore: McGraw Hill, 2010, pp. 607–646.
- [17] Kendall G. "Scheduling English Football Fixtures Over Holiday Periods." In: *Journal of the Operational Research Society* 59.6 (2008), pp. 743–755.
- [18] Schaerf L Gaspero D and Schaerf A. "A composite-neighborhood tabu search approach to the traveling tournament problem". In: *Journal of Heuristics* 13 (2007), pp. 189–207.
- [19] Asimakopoulos I and Goddard J. "Forecasting Football Results and the Efficiency of Fixed-odds Betting." In: *Journal of Forecasting* (2004), pp. 51–66.
- [20] Baker J. "Adaptive selection methods for genetic algorithms". In: *Proceedings of an International Conference on Genetic Algorithms and their applications*. 1985, pp. 101–111.
- [21] Dinitz J and Dinitz M. "Enumeration of balanced tournament designs on 10 points". In: *Journal of Combinatorial Mathematics and Combinatorial Computing* 52 (2005), pp. 51–63.
- [22] Goddard J. "Regression models for forecasting goals and match results in association football." In: *International Journal of Forecasting* (2005), pp. 331–340.
- [23] Grefenstette J. "Proportional selection and sampling algorithms". In: *Evolutionary computation* 1 (2000), pp. 172–180.
- [24] Kyngäs J. "Solving Challenging Real-World Scheduling Problems". unpublished. 2011.
- [25] Regin J. "Minimization of the number of breaks in sports scheduling problems using constraint programming". In: *Constraint Programming Large Scale Discrete Optimization, Volume 57 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science*. Ed. by E. Freuder and R. Wallace. 2001, pp. 115–130.
- [26] Schreuder J. "Combinatorial aspects of construction of competition Dutch Professional Football Leagues". In: *Discrete Applied Mathematics*. 1992, pp. 301–312.
- [27] Courneya K and Carron A. "The home advantage in sport competitions: A literature review." In: *Journal of Sport and Exercise Psychology* (1992), pp. 13–27.
- [28] Nurmi K and Goossens D. "A framework for a highly constrained sports scheduling problem". In: *Proceedings of the International MultiConference of Engineers and Computer Scientists* 3 (2010), pp. 14–28.
- [29] Gelatt Jr C.D Kirkpatrick S and Vecchi M.P. "Optimization by Simulated Annealing". unpublished. 1983.

- [30] Gendreau M. “An introduction to Tabu Search”. unpublished. 2002.
- [31] Gendreau M. “The Traveling Salesman Problem: A Case Study in Local Optimization”. unpublished. 1995.
- [32] Gleeson M. “Africa is Expensive.” In: *Kickoff*. 2016, p. 56.
- [33] Park M and Kim Y.D. “A systematic procedure for setting parameters in simulated annealing algorithms”. In: *Computers and Operations Research* 25.3 (1998), pp. 207–217.
- [34] Trick M. “A schedule-then-break approach to sports timetabling”. In: *The 3rd International Conference on the Practice and Theory of Automated Timetabling, Volume 2079 of Lecture Notes in Computer Science*. Ed. by E. K. Burke and E. Erben. Springer-Verlag: Berlin, Germany, 2001, pp. 242–252.
- [35] Trick M. *Challenge Traveling Tournament Instances*. URL: [URL:%20http://mat.gsia.cmu.edu/TOURN/](http://mat.gsia.cmu.edu/TOURN/).
- [36] Kendall G McMullanl P and McCollum B. “Scheduling English Football Fixtures: Consideration of Two Conflicting Objectives.” unpublished. 2010.
- [37] A Nevill and Holder R. “Home advantage in sport: an overview of studies on the advantage of playing at home”. In: *Sports Med* 28 (1999), pp. 221–236.
- [38] Silva C Pollard R and Medeiros N. “Home advantage in football in Bazil: differnces between teams and the effects of distance travelled.” In: *The Brazilian Journal of Soccer Science* (2008), pp. 3–10.
- [39] Pollard R. “Home advantage in soccer: a retrospective analysis.” In: *Journal of Sport Sciences* (1986), pp. 237–248.
- [40] Rasmussen R and Trick M. “Round robin scheduling – a survey”. In: *European Journal of Operational Research* 188.3 (2008), pp. 617–636.
- [41] Rasmussen R and Trick M. “The timetable constrained distance minimization problem”. In: *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. Lecture notes in Computer Science* 3990 (2006), pp. 176–181.
- [42] Stefani R and Clarke S. “Predictions and home advantage for Australian rules Football.” In: *Journal of Applied Statistics* (1992), pp. 251–261.
- [43] amd Kendall G Ribeiro C Knust S. “Scheduling in sports: An annotated bibliography”. In: *Computers and Operations Research*. 2010, pp. 1–19.
- [44] Clarke S and Norman J. “Home ground advantage of individual clubs in English soccer.” In: *The Statistician* (1995), pp. 509–521.
- [45] Knust S. *Classification of Literature on Sports Scheduling*. URL: <http://www.informatik.uniosnabrueck.de/knust/sportssched/sportlit-class.html..>