Scheduling English Football Fixtures: Consideration of Two Conflicting Objectives

Graham Kendall, Barry McCollum, Frederico Cruz, Paul McMullan, and Lyndon While

1 Introduction

The English Premier League is one of the most high profile, and successful, football (soccer in the USA) leagues in the world. It comprises 20 teams which have to play each other both home and away (i.e. a double round robin tournament), resulting in 380 fixtures that have to be scheduled. The other three main divisions in England (the Championship, League One and League Two) each have 24 teams, resulting in 552 fixtures having to be scheduled for each division. Therefore, for the four main divisions in England 2036 fixtures have to be scheduled every season. The divisions

Graham Kendall
School of Computer Science, University of Nottingham, Malaysia Campus, Jalan Broga, 43500 Semenyih, Selangor Darul Ehsan, Malaysia
e-mail: graham.kendall@nottingham.edu.my

Barry McCollum
School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, BT7 1NN, UK
e-mail: B.McCollum@qub.ac.uk

Frederico Cruz
Departamento de Estatstica - ICEx - UFMG, Av. Antnio Carlos, 6627, 31270-901 - Belo Horizonte - MG, Brazil
e-mail: fcruz@est.ufmg.br

Paul McMullan
School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, BT7 1NN, UK
e-mail: p.p.mcmullan@qub.ac.uk

Lyndon While
School of Computer Science & Software Engineering, The University of Western Australia, Perth WA 6009, Australia
e-mail: lyndon@csse.uwa.edu.au

operate a system of promotion and relegation such that the teams in each division changes each year so it is not possible to simply use the same schedule every time.

Of particular interest are the schedules that need to be generated for the Christmas/New Year period. At this time of the year it is a requirement that every team plays two fixtures, one on Boxing Day (26th December) and one on New Years Day (1st January). Whilst scheduling these two sets of fixtures the overriding aim is to minimise the total distance that has to be travelled by the supporters. An analysis of the fixtures that were actually used, and also following discussions with the football authorities, confirm that this is a real world requirement and that the distances travelled by the supporters are the minimum when compared against other fixtures when all teams play. In addition, there are various other constraints that have to be respected, which are described in sections 3 and 4.

The problem we address in this chapter is to attempt to minimise two competing objectives to ascertain if there is a good trade off between them. The objectives we minimise are the distances travelled by the supporters and the number of *pair clashes*. Pairing matches two (or more) teams and dictates that these clubs should not play at home on the same day. If they do, this is termed a pair clash. In fact, a certain number of pair clashes are allowed. The exact number is taken from the number that were present in the published fixtures for a given season. Importantly, paired teams cannot play each other on the two days in question. This is treated as a hard constraint. It is this constraint that causes a problem. If we allow Liverpool and Everton (for example) to play each other, one set of supporters would only travel four miles. If these teams are paired (as they are) then they cannot play each other so the distances are likely to increase as either Liverpool or Everton would have to travel more than four miles. As pair clashes usually involve teams which are geographically close this gives rise to the conflicting objectives.

In [19], an initial study of the problem considered the 2003-2004 football season, suggesting that it may be possible to minimise both of these competing objectives but still produce results which are acceptable to both the supporters (who are interested in minimising the amount they travel) and the police (who are interested in having fewer pair clashes). In this chapter, we carry out a more in depth study by considering more seasons and carrying out statistical analysis of the results in order to draw stronger conclusions.

2 Related Work

Producing a double round robin tournament is relatively easy in that the algorithms are well known, with the polygon construction method being amongst the most popular [9]. The problem with utlising such an algorithm is that the fixtures it generates, although being a valid round robin tournament, will not adhere to all the additional constraints for a particular problem. Moreover, every problem instance will be subtly different and, often, a bespoke algorithm is required for each instance. This is even the case when faced with seemingly the same problem. For example, the English Football League consists of four divisions and 92 teams. It would be easy to assume that once an algorithm has been developed it can be used every season. This may indeed be the case but due to the promotion/relegation system the problem changes year on year and, perhaps, there are additional features/constraints in one season that were not previously present. Rasmussen and Trick [21] provide an excellent overview of the issues, methods and theoretical results for scheduling round robin tournaments.

The Travelling Tournament Problem (TTP) [11] is probably the most widely used test bed in sports scheduling. The problem was inspired by work carried out for Major League Baseball [11]. The aim of the TTP is to generate a double round robin tournament, while minimising the overall distance travelled by all teams. Unlike the problem studied in this paper, it is possible to minmise the overall travel distance as teams go on *road trips* so, with a suitable schedule, the length of these trips can be reduced. The TTP is further complicated by the introduction of two constraints. The first says that no team can play more than three consecutive home or away games. The second stipulates that if team i plays team j in round, r, then team j cannot play team i in round $r+1$. These constraints add sufficient complexity to the problem so as to make it challenging, but it still does not reflect all the constraints that are present in the real world problem.

The TTP has received significant research attention. Some of the important papers being [12, 2, 8, 22, 25]. A recent annotated bibliography of TTP papers can be found in [18]. An up to date list of the best known solutions, as well as details of all the instances, can be found at the web site maintained by Michael Trick [23].

With respect to minimising travel costs/distances, previous studies have considered a variety of sports. Campbell and Chen [6] and Ball and Webster [3] both studied basketball, attempting to minimise the distance travelled. Bean and Birge [4] also studied basketball, attempting to minimise airline travel costs. Minimising travel costs was also the focus of [5], for baseball. Minimising travel distances for hockey [16] and umpires for baseball [15] have also been studied. Wright [28], as one part of the evaluation function, considered travel between fixtures for English cricket clubs. Costa [7] considered the National Hockey League, where minimisation of the distance travelled by the teams was just one factor in the objective function.

Urrutia and Ribeiro [24] have shown that minimising distance and maximising breaks (two consecutive home games (home break) or two consecutive away games (away break)) is equivalent. This followed previous work by de Werra [26, 27] and Elf et al. [14] who showed how to construct schedules with the minimum number of breaks.

The scheduling problem that we are considering in this chapter is minimising the distance travelled for two complete fixtures (a complete fixture is defined as a set of fixtures when every team plays) while, at the same time, minimising the number of pair clashes. These two complete fixtures can then be used over the Christmas holiday period when, for a variety of reasons, teams wish to limit the amount of travelling undertaken. Note, that this is a different problem to the Travelling Tournament Problem as the TTP assumes that teams go on road trips, and so the total distance travelled over a season can be minimised. In English football, there is no concept of road trips. Therefore, over the course of a season, the distance cannot be minimised. However, we can minimise the distance on particular days. Kendall [17] adopted a two-phase approach to produce two complete fixtures for this problem. A depth first search was used to produce fixtures for one day, for each division. A further depth first search created another set of fixtures for the second day. This process produced eight separate fixtures (two sets of fixtures for each division) which adhered to some of the constraints (e.g. a team plays at home on one day and away on the other) but had not yet addressed the constraints with regards to pair clashes (see [17] for a detailed description). The fixture lists from the depth first searches were input to a local search procedure which aimed to satisfy the remaining constraints, whilst attempting to minimise the overall distance travelled. The output of the local search, and a post-process operation to ensure feasibility, produced the results presented in the paper.

Overviews of sports scheduling can be found in [13, 9, 10, 21, 29, 20, 18].

3 Problem Definition

In previous work [17] the only objective was to minimise the total distance travelled by the teams/supporters. The aim of that study was to investigate if we were able to generate better quality solutions than those used by the football league. We demonstrated that it was possible. As stated in the Introduction, the police also have an interest in the fixtures that are played at this time of the year. If we are able to generate acceptable schedules, with fewer pair clashes then the policing costs would be reduced.

The purpose of this chapter is to investigate if there is an acceptable trade off between the minimisation of distance and the minimisation of pair clashes. In order to do this we will utilise a multi-objective methodology.

4 Experimental Setup

We use a two stage algorithm. In [17] a depth first search (DFS) was used, followed by a local search. DFS was used as we wanted to carry out a preliminary study just to see if this area was worthy of further study. As we were able to produce superior solutions to the published fixtures we have now decided to utilise more sophisticated methods, due to the large execution times of DFS which were typically a few hours for each division. In this work we utilise CPLEX as a replacement for DFS and simulated annealing [1] as a replacement for the local search. This reduces the overall execution time from tens of hours to a few minutes.

4.1 Phase 1: CPLEX

The first phase uses CPLEX to produce an optimal solution to a relaxed version of the problem. In generating *relaxed* optimal solutions we respect the following constraints, whilst minimizing the overall distance.

1. Each of the 92 teams has to play on two separate days (i.e. 46 fixtures will be scheduled on each day).
2. Each team has to play at home on one day and away on the other.
3. Teams are not allowed to play each other on both days.
4. A team is not allowed to play itself.

The CPLEX model is executed four times. Each run returns the Boxing Day and New Years Day fixtures for a particular division. Each run takes less than 10 seconds.

In solving the CPLEX model we do not take into account many of the constraints that ultimately have to be respected. For example, pair clashes, geographical constraints such as the number of London or Manchester clubs playing at home on the same day etc. (see [17] for details).

4.2 Phase 2: Simulated Annealing

The schedules from CPLEX are input to the second phase, where we utilise simulated annealing. This operates across all the divisions in order to resolve any hard constraint violations whilst still attempting to minimise the distance.

The simulated annealing parameters are as follows:

Start_Temperature = 1000 The same value is used across all seven datasets and was found by experimentation. We could have used different values for each dataset but we felt that it was beneficial to be consistent across all the datasets.
Stop_Temperature The algorithm continues while the temperature is > 0.1.
Cooling Schedule $CurTemp = CurTemp * 0.95$.
Number of Iterations 2000 iterations are carried out at each temperature.

4.3 Evaluation Function

The evaluation function we use for simulated annealing is dynamic in that the hard constraint violations are more heavily penalised as the search progresses. This enables more exploration at the start of the search, which gets tighter as the temperature is reduced. The objective function is formulated as follows:

$$f(x) = d\_fb + d\_fy + w \times penalty \quad (1)$$

where:

d_fb = total distance travelled by teams on Boxing Day.
d_fy = total distance travelled by teams on New Years Day.
w = is a weight for the penalty (see below). It is given by (*Start_Temperature* - *CurTemp*). *Start_Temperature* is the maximum temperature for the simulated annealing algorithm. *CurTemp* is the current temperature of the simulated annealing algorithm. As the simulated annealing algorithm progresses, the weight of the penalty gradually increases, driving the search towards feasible solutions, but allowing it to search the infeasible region at the start of the search.
penalty = This is given by a summation of the following terms (the limits referred to are available in [17] and represent the values found by analyzing the published fixtures):

ReverseFixtures The number of *reverse* fixtures (the same teams cannot meet on both days).
Boxing Day Local Derby Clashes The number of paired teams playing each other on Boxing Day.
New Years Day Local Derby Clashes The number of paired teams playing each other on New Years Day.
Boxing Day London Clashes The number of London clubs playing at home on Boxing Day, which exceed a given limit.
New Years Day London Clashes The number of London clubs playing at home on New Years Day, which exceed a given limit.
Boxing Day Greater Manchester Clashes The number of Greater Manchester based clubs playing at home on Boxing Day, which exceed a given limit.
New Years Day Greater Manchester Clashes The number of Greater Manchester based clubs playing at home on New Years Day, which exceed a given limit.
Boxing Day London Premier Clashes The number of Premiership London clubs playing at home on Boxing Day, which exceed a given limit.
New Years Day London Premier Clashes The number of Premiership London clubs playing at home on New Years Day, which exceed a given limit.
Boxing Day Clashes The number of Boxing Day clashes greater than an allowable limit.
New Years Day Clashes The number of New Years Day clashes greater than an allowable limit.

4.4 Perturbation Operators

Simulated annealing often has a single neighborhood operator but we have defined sixteen operators in order to match the hard constraints within the model. The operators are as follows:

1. Examines the Boxing Day fixtures and if the number of clashes exceeds an upper limit, randomly select one of the clashing fixtures and swap the home and away teams.
2. Same as 1 expect that it considers New Years Day fixtures.
3. Examines the Boxing Day fixtures and if the number of London based clubs exceeds an upper limit, randomly select one of the fixtures that has a London based club playing at home and swap the home and away teams.
4. Same as 3 except that it considers Greater Manchester based clubs.
5. Same as 3 except that it considers London based premiership clubs.
6. Same as 3 except that it considers the New Years Day fixtures.
7. Same as 4 except that it considers the New Years Day fixtures.
8. Same as 5 except that it considers the New Years Day fixtures.
9. Examines the Boxing Day and New Years Day fixture lists, returning the number of reverse fixtures (where team i plays team j and team j plays team i). While there are reverse fixtures, one of the reverse fixtures on Boxing Day is chosen and the home team is swapped with a randomly selected home team, with the condition that the swaps must be made between teams in the same division. This operator iterates until all reverse fixtures have been removed from the fixture list.
10. Same as 9 except the swaps are made in the New Years Day fixtures.
11. This operator examines the Boxing Day and New Years Day fixture lists, returning the number fixtures where paired teams are playing each other. While this is the case, one of the Boxing Day fixtures is chosen and the home team is swapped with a randomly selected home team in the Boxing Day fixtures, with the condition that the swaps must be made between teams in the same division. This operator iterates until all local pair clashes have been removed from the fixture lists.
12. Same as 11 except the swaps are made in the New Years Day fixtures.
13. This operator chooses a random fixture from a candidate list (we use a candidate list size of 250) which represents the potential fixtures that have the shortest distances. Swaps are carried out in the Boxing Day fixtures in order to allow the two teams from the selected item in the candidate list to play each other. The necessary swaps are also done in the New Years Day fixture to ensure feasibility.
14. Same as 13 except that it considers the New Years Day fixtures.
15. Selects a random fixture in the Boxing Day fixture list and swaps the home and away teams.
16. Same as 15, but swaps a random fixture in the New Years Day fixture list.

At each iteration, one of the sixteen operators is chosen at random. $Start\_Temperature$ is initially set to enable infeasible solutions during the early stages of the algorithm, but they are more heavily penalised at lower temperatures (eq. 1), ensuring that the final solution is feasible.

4.5 Experimental Methodology

We are investigating this problem from a multi-objective perspective but rather than using a multi-objective algorithm we run the same algorithm a number of times, adjusting the parameters for each run. As an example, for the 2002-2003 season the number of pair clashes, in the published fixtures, was 10 and 8 for Boxing Day and New Years Day respectively. We denote this as 10-8 in the tables below. Therefore, the first experiment fixes the values as 10 and 8 as the number of pair clashes that cannot be exceeded. In this respect, these values represent hard constraints. The next experiment reduces one of these values so that the next experiment uses 10-6. We then reduce the other value to run a further experiment using 8-8. There are two points worthy of note. Firstly, we reduce the value by two as a pair clash of, say, Everton and Liverpool actually counts as two pair clashes as both teams are considered to be clashing. Secondly, we do not reduce the total number of pair clashes below 16.

5 Results

Tables 1 thru 7 shows the results of each of the seven seasons that we use. The *Clashes* column shows the number of pair clashes (see section 4.5 for the notation that we use). *Min* represents the best solution found. *Max* is worst solution found and *Average* and *Std Dev* are self-explanatory. All experiments were runs 30 times.

Table 1 2002-2003: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
10-8	5243	6786	5630	288.46
10-6	5674	7222	6183	410.71
8-8	5562	6797	6070	309.50

In tables 8 and 9 we analyse the results from table 1. Table 8 shows the results of independent two-tailed t-tests (at the 95% confidence level) to compare the means of each experiment against every other experiment for that season. Where two experiments are statistically significant the relevant cell shows “Yes”, otherwise the cell is empty. As an example, if we compare 10-8 (column) with 10-6 (row) in table 8 we see that the means (i.e. the travel distances from 30 independent runs) are statistically different. By comparing the means in table 1, 5630 and 6183 respec-

Table 2 2003-2004: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
8-14	5464	6173	5698	165.46
8-12	5412	6519	5827	228.66
8-10	5511	7093	6053	417.00
8-8	5887	7674	6535	433.83
6-14	5550	6334	5805	176.02
6-12	5559	6587	6036	289.75
6-10	5898	7416	6454	395.37
4-14	5592	6911	6059	274.61
4-12	5886	7848	6635	484.59
2-14	6028	7704	6704	448.87

Table 3 2004-2005: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
10-10	5365	6986	5644	318.33
10-8	5345	6348	5727	259.17
10-6	5812	7714	6431	421.63
8-10	5443	6982	5923	469.01
8-8	5645	7612	6428	550.67
6-10	5810	7824	6486	487.26

Table 4 2005-2006: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
12-14	5234	6046	5575	184.74
12-12	5335	6002	5596	153.90
12-10	5240	6511	5641	238.58
12-8	5334	6423	5754	231.81
12-6	5481	6958	6010	339.63
12-4	6041	6989	6468	271.99
10-14	5171	6683	5606	304.33
10-12	5308	6322	5610	204.96
10-10	5460	6674	5846	359.65
10-8	5595	6380	5872	216.82
10-6	6027	7561	6660	421.25
8-14	5335	6674	5680	286.00
8-12	5334	6133	5722	211.02
8-10	5608	7078	5979	356.15
8-8	6146	7277	6587	302.48
6-14	5500	6694	5843	254.23
6-12	5528	6655	5951	233.54
6-10	5884	7291	6529	382.80
4-14	5713	7391	6161	331.25
4-12	6032	7904	6662	434.72
2-14	6084	7551	6682	399.34

Table 5 2006-2007: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
14-8	5713	7040	6077	300.71
14-6	5735	7065	6117	270.59
14-4	5872	7000	6259	227.84
14-2	6110	7778	6741	402.35
12-8	5721	6784	6084	244.28
12-6	5714	6894	6234	326.99
12-4	6195	7546	6791	405.86
10-8	5762	7671	6209	411.02
10-6	5894	7376	6618	423.94
8-8	6071	6958	6513	251.33

Table 6 2007-2008: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
14-10	5366	5902	5595	145.26
14-8	5403	5975	5674	152.93
14-6	5425	7172	5870	372.17
14-4	5690	6995	6172	364.78
14-2	5905	7856	6698	435.98
12-10	5370	6506	5736	294.88
12-8	5321	7139	5850	338.15
12-6	5625	7394	6084	365.93
12-4	5961	7580	6575	411.41
10-10	5340	6552	5754	228.71
10-8	5616	6365	5944	183.52
10-6	6101	7468	6619	369.10
8-10	5536	7081	6056	369.47
8-8	6091	7884	6725	402.08
6-10	5951	7709	6647	381.12

Table 7 2008-2009: Summary of results from 30 runs

Clashes	Min	Max	Average	Std Dev
10-10	5564	6806	5833	246.11
10-8	5574	6235	5829	140.52
10-6	5736	6523	6106	208.78
8-10	5581	6817	5936	281.83
8-8	5790	6900	6148	230.42
6-10	5809	7194	6208	274.67

tively, we conclude that reducing the number of pair clashes from 18 (10-8) to 16 (8-8) the travel distances for the clubs/supporters increases by a significant amount. Looking at 10-6 and 8-8, there is no statistical difference. However, as both of these experiments represent 16 pair clashes it is, perhaps, not surprising that the average distance travelled over the 30 runs is (statistically) the same.

Table 9 summarises the results from table 8 by only showing those experiments where there are statistical differences, AND when the total number of pair clashes is different (i.e. it will ignore 10-6 and 8-8). We can see from table 9 that there are no experiments where we can reduce the number of pair clashes that leads to no statistical difference in the distance travelled.

Tables 10 and 11 show similar analysis for the 2003-204 season. Again, it is not possible to reduce the number of pair clashes without an (statistically) increase in the distance travelled.

Tables 12 and 13 are more interesting. Table 12 shows that there is no statistical difference between the 10-10 (20 pair clashes) experiment and the 10-8 (18 pair clashes) experiment. Removing all the *noise* from the table (see table 13) we can see that it is possible to reduce the number of pair clashes from 20 to 18 without a significant rise in the distance travelled (the respective means from table 3 are 5644 and 5727).

For the remaining four seasons, we only present the summary tables. Where a“Yes” appears in these tables (tables 14 thru 17) it indicates that it is possible to reduce the number of pair clashes and not have an (statistical) increase in travel distance. The tables show that there are a number of opportunities to reduce policing costs. We are probably most interested in the top rows as they represent the fixtures that were actually used.

Table 8 2002-2003: Are the Results Statistically Different?

Clashes	10-8	10-6	8-8
10-8	X	Yes	Yes
10-6		X	
8-8			X

Table 9 2002-2003: Are different total clashes significantly different?

Clashes	10-8	10-6	8-8
10-8	X		
10-6		X	
8-8			X

Table 10 2003-2004: Are the Results Statistically Different?

Clashes	8-14	8-12	8-10	8-8	6-14	6-12	6-10	4-14	4-12	2-14
8-14	X	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
8-12		X	Yes	Yes		Yes	Yes	Yes	Yes	Yes
8-10			X	Yes	Yes		Yes		Yes	Yes
8-8				X	Yes	Yes		Yes		
6-14					X	Yes	Yes	Yes	Yes	Yes
6-12						X	Yes		Yes	Yes
6-10							X	Yes		Yes
4-14								X	Yes	Yes
4-12									X	
2-14										X

Table 11 2003-2004: Are different total clashes significantly different?

Clashes	8-14	8-12	8-10	8-8	6-14	6-12	6-10	4-14	4-12	2-14
8-14	X									
8-12		X								
8-10			X							
8-8				X						
6-14					X					
6-12						X				
6-10							X			
4-14								X		
4-12									X	
2-14										X

Table 12 2004-2005: Are the Results Statistically Different?

Clashes	10-10	10-8	10-6	8-10	8-8	6-10
10-10	X		Yes	Yes	Yes	Yes
10-8		X	Yes		Yes	Yes
10-6			X	Yes		
8-10				X	Yes	Yes
8-8					X	
6-10						X

Table 13 2004-2005: Are different total clashes significantly different?

Clashes	10-10	10-8	10-6	8-10	8-8	6-10
10-10	X	Yes				
10-8		X				
10-6			X			
8-10				X		
8-8					X	
6-10						X

Table 14 2005-2006: Are different total clashes significantly different?

Clashes	12-14	12-12	12-10	12-8	12-6	12-4	10-14	10-12	10-10	10-8	10-6	8-14	8-12	8-10	8-8	6-14	6-12	6-10	4-14	4-12	2-14
12-14	X	Yes	Yes				Yes	Yes				Yes									
12-12		X	Yes					Yes				Yes									
12-10			X	Yes			Yes						Yes								
12-8				X								Yes									
12-6					X				Yes												
12-4						X															
10-14							X	Yes				Yes	Yes								
10-12								X													
10-10									X	Yes		Yes		Yes			Yes				
10-8										X						Yes					
10-6											X										
8-14												X	Yes								
8-12													X								
8-10														X		Yes					
8-8															X						
6-14																X	Yes				
6-12																	X				
6-10																		X			
4-14																			X		
4-12																				X	
2-14																					X

Table 15 2006-2007: Are different total clashes significantly different?

Clashes	14-8	14-6	14-4	14-2	12-8	12-6	12-4	10-8	10-6	8-8
14-8	X	Yes			Yes	Yes		Yes		
14-6		X				Yes		Yes		
14-4			X							
14-2				X						
12-8					X			Yes		
12-6						X				
12-4							X			
10-8								X		
10-6									X	
8-8										X

Table 16 2007-2008: Are different total clashes significantly different?

Clashes	14-10	14-8	14-6	14-4	14-2	12-10	12-8	12-6	12-4	10-10	10-8	10-6	8-10	8-8	6-10
14-10	X														
14-8		X								Yes					
14-6			X			Yes					Yes		Yes		
14-4				X											
14-2					X										
12-10						X	Yes			Yes					
12-8							X				Yes				
12-6								X							
12-4									X						
10-10										X					
10-8											X				
10-6												X			
8-10													X		
8-8														X	
6-10															X

Table 17 2008-2009: Are different total clashes significantly different?

Clashes	10-10	10-8	10-6	8-10	8-8	6-10
10-10	X	Yes		Yes		
10-8		X				
10-6			X			
8-10				X		
8-8					X	
6-10						

6 Conclusion

We have demonstrated that it is sometimes possible to reduce the number of pair clashes without a statistical difference to the distance that has to be travelled by the club/supporters. This provides the police with the ability to reduce their costs for these two days, which might have included paying overtime. We hope that we are able to discuss these results with the football authorities and the police in order for them to validate our work and to provide us with potential future research directions. We already recognise that some pair clashes might provide the police with more *problems* than others and it might be worth prioritising certain clashes so that these can be removed, rather than removing less high profile fixtures. As a longer term research aim, we would like to include in our model details about public transport as some routes might be more difficult than other routes, even if they are shorter. We also plan to run our algorithms for every future season, as well as for previous seasons. Executing the algorithm is not the main issue. Data collection provides the real challenge due to the distance data that has to be collected. To date, this has been carried out manually by using motoring organisation's web sites but we have recently started experimenting with services such as Google Maps™ and Multimap which will speed up the data collection.

References

1. Aarts, E., Korst, J., Michels, W.: Simulated annealing. In: E.K. Burke, G. Kendall (eds.) Search Methodologies: Introductory Tutorials in Optimization and Decision Support Methodologies, 1st edn., chap. 7, pp. 97–125. Springer (2005)
2. Anagnostopoulos, A., Michel, L., Van Hentenryck, P., Vergados, Y.: A simulated annealing approach to the traveling tournament problem. Journal of Scheduling **9**, 177–193 (2006)
3. Ball, B.C., Webster, D.B.: Optimal scheduling for even-numbered team athletic conferences. AIIE Transactions **9**, 161–169 (1977)
4. Bean, J.C., Birge, J.R.: Reducing travelling costs and player fatigue in the national basketball association. Interfaces **10**, 98–102 (1980)
5. Cain, W.O.: The computer-aided heuristic approach used to schedule the major league baseball clubs. In: S.P. Ladany, R.E. Machol (eds.) Optimal Strategies in Sports, pp. 33–41. North Holland, Amsterdam (1977)
6. Campbell, R.T., Chen, D.S.: A minimum distance basketball scheduling problem. In: R.E. Machol, S.P. Ladany, D.G. Morrison (eds.) Management Science in Sports, *Studies in the Management Sciences*, vol. 4, pp. 15–25. North-Holland, Amsterdam (1976)
7. Costa, D.: An evolutionary tabu search algorithm and the NHL scheduling problem. INFOR **33**, 161–178 (1995)
8. Di Gaspero, L., Schaerf, A.: A composite-neighborhood tabu search approach to the traveling tournament problem. Journal of Heuristics **13**, 189–207 (2007)
9. Dinitz, J.H., Fronček, D., Lamken, E.R., Wallis, W.D.: Scheduling a tournament. In: C.J. Colbourn, J.H. Dinitz (eds.) Handbook of Combinatorial Designs, 2nd edn., pp. 591–606. CRC Press (2006)
10. Drexl, A., Knust, S.: Sports league scheduling: Graph- and resource-based models. Omega **35**, 465–471 (2007)

11. Easton, K., Nemhauser, G.L., Trick, M.A.: The travelling tournament problem: Description and benchmarks. In: T. Walsh (ed.) Principles and Practice of Constraint Programming, *Lecture Notes in Computer Science*, vol. 2239, pp. 580–585. Springer (2001)
12. Easton, K., Nemhauser, G.L., Trick, M.A.: Solving the travelling tournament problem: A combined integer programming and constraint programming approach. In: E. Burke, P. de Causmaecker (eds.) The 4th International Conference on the Practice and Theory of Automated Timetabling, *Lecture Notes in Computer Science*, vol. 2740, pp. 100–109. Springer (2003)
13. Easton, K., Nemhauser, G.L., Trick, M.A.: Sports scheduling. In: J.T. Leung (ed.) Handbook of Scheduling, pp. 52.1–52.19. CRC Press (2004)
14. Elf, M., Jnger, M., Rinaldi, G.: Minimizing breaks by maximizing cuts. Operations Research Letters **31**(3), 343–349 (2003)
15. Evans, J.R.: A microcomputer-based decision support system for scheduling umpires in the American Baseball League. Interfaces **18**, 42–51 (1988)
16. Ferland, J.A., Fleurent, C.: Computer aided scheduling for a sport league. INFOR **29**, 14–25 (1991)
17. Kendall, G.: Scheduling English football fixtures over holiday periods. Journal of the Operational Research Society **59**, 743–755 (2008)
18. Kendall, G., Knust, S., Ribeiro, C.C., Urrutia, S.: Scheduling in sports: An annotated bibliography. Computers & Operations Research **37**, 1–19 (2010)
19. Kendall, G., While, L., McCollum, B., Cruz, F.: A multiobjective approach for UK football scheduling. In: E.K. Burke, M. Gendreau (eds.) Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling (2008)
20. Knust, S.: Classification of literature on sports scheduling (2010). Available online at http://www.inf.uos.de/knust/sportssched/sportlit_class/, last visited 15th July 2010
21. Rasmussen, R.V., Trick, M.A.: Round robin scheduling – A survey. European Journal of Operational Research **188**, 617–636 (2008)
22. Ribeiro, C.C., Urrutia, S.: Heuristics for the mirrored traveling tournament problem. European Journal of Operational Research **179**, 775–787 (2007)
23. Trick, M.: Traveling tournament problem instances (2010). Available online at http://mat.gsia.cmu.edu/TOURN/, last accessed 15th July 2010
24. Urrutia, S., Ribeiro, C.: Minimizing travels by maximizing breaks in round robin tournament schedules. Electronic Notes in Discrete Mathematics **18-C**, 227–233 (2004)
25. Urrutia, S., Ribeiro, C.C., Melo, R.A.: A new lower bound to the traveling tournament problem. In: Proceedings of the IEEE Symposium on Computational Intelligence in Scheduling, pp. 15–18. IEEE, Honolulu (2007)
26. de Werra, D.: Scheduling in sports. In: P. Hansen (ed.) Studies on Graphs and Discrete Programming, pp. 381–395. North Holland, Amsterdam (1981)
27. de Werra, D.: Some models of graphs for scheduling sports competitions. Discrete Applied Mathematics **21**, 47–65 (1988)
28. Wright, M.: Timetabling county cricket fixtures using a form of tabu search. Journal of the Operational Research Society **45**, 758–770 (1994)
29. Wright, M.: 50 years of OR in sport. Journal of the Operational Research Society **60**, S161–S168 (2009)