

A Constructive Heuristic for the Travelling Tournament Problem

Graham Kendall¹, Wim Miserez², and Greet Vanden Berghe²

¹ University of Nottingham, School of Computer Science and IT, Jubilee Campus,
Nottingham NG8 1BB, UK
`gzk@cs.nott.ac.uk`

² KaHo St.-Lieven, Information Technology,
Gebr. Desmetstraat 1, 9000 Gent, Belgium
`Wim.Miserez@student.kahosl.be`, `Greet.VandenBerghe@kahosl.be`

Abstract. We present a two-stage heuristic for the travelling tournament problem. It consists of first creating patterns for good quality solutions for each team, after which these partial solutions are used/combined in an improvement heuristic to find overall solutions.

1 Introduction

The travelling tournament problem (TTP) involves minimising the total distance travelled when organising a double round robin tournament among a number of teams. Feasible solutions require that

- home and away games between two teams are not in consecutive timeslots (no repeaters),
- teams do not play more than three consecutive home/away games [7].

Sports scheduling has been the subject of many academic publications (e.g. [5, 6, 8, 9, 11–14]). Agnostopoulos et al. [1] present the current best results for the problems presented on the TTP website (<http://mat.gsia.cmu.edu/TOURN/>). Instead of tackling the entire tournament as one problem, we propose a two-stage approach in which we solve the problem for individual teams first. Similar decomposition techniques have been successfully applied to other timetabling domains (e.g. to nurse rostering [2–4] and university timetabling [10]). The main motivation for decomposing the problem is that it considerably reduces the search space, and thus the computation time.

The method proposed in this abstract includes solving the corresponding Travelling Salesman Problem (Section 2.1) and generating feasible patterns with that solution (Section 2.2).

2 Constructive Heuristic

2.1 Travelling Salesman Problem

The total travelling distance for a tournament equals the sum of the travelling distances for each team. We decompose the problem into sub problems for each

team. Each of these sub problems involves searching for the minimum travelling distance for a team. If we ignore the constraint on consecutive home and away games, the problem equals a Travelling Salesman Problem (TSP) for which the cities correspond to the teams in the TTP. Each team should solve exactly the same TSP problem.

The best solutions for the TSPs corresponding to the different TTPs are presented in Table 1. The sequence of numbers denotes a sequence of optimal distance, in which the numbers refer to the order in which the teams appear in the distance matrix on the TTP website referred to in Section 1.

n	Dist.	TTP Solution
4	2011	1-3-2-4
6	2971	5-1-6-4-2-3
8	3480	6-4-2-3-5-1-7-8
10	3834	6-4-2-3-5-1-9-10-8-7
12	5600	7-6-3-2-4-8-10-9-12-11-5-1
14	7427	1-5-11-14-13-12-9-10-8-4-2-3-6-7
16	7443	7-6-3-2-4-8-10-9-12-13-15-14-16-11-5-1
18	7426	9-11-15-13-16-14-18-12-2-10-7-4-17-1-3-6-8-5
20	7426	19-7-20-4-17-1-3-6-8-5-9-11-15-13-16-14-18-12-2-10
22	7989	9-11-10-2-12-18-14-16-13-15-22-21-8-3-1-4-17-20-7-19-6-5
24	8091	15-22-24-21-23-8-3-1-17-4-20-7-19-6-5-9-11-10-2-12-18-14-16-13
26	8274	17-4-1-3-6-8-23-5-11-9-21-24-22-15-13-16-14-18-12-2-26-10-25-19-7-20
28	8393	11-25-19-7-20-4-17-1-3-6-8-23-5-9-21-24-22-15-13-16-14-18-12-28-26-2-10-27
30	9393	9-5-11-30-15-13-29-16-14-18-12-28-26-2-10-27-25-19-7-20-17-4-1-3-6-8-23-24-22-21
32	9412	7-19-25-27-10-2-26-28-12-18-31-14-16-32-29-13-15-30-11-5-9-21-22-24-23-8-6-3-1-4-17-20

Table 1. Solutions for the TSP that corresponds to the TTP of size n

2.2 Feasible Patterns

After the shortest tour has been calculated, we apply a heuristic step that modifies the optimal solutions into feasible patterns. We therefore introduce a number of home-away edges in the optimal tour, in such a way that no more than three consecutive home/away games occur. We call the results patterns, since they do not specify any ‘home’ opponent of the team yet. An example of a pattern for team 3 is presented in Fig. 1. The top row presents the shortest tour for the TSP, starting from city/team 3 (which is the first and the last node in the bottom row). The shortest tour is interrupted with additional away/home and

home/away edges: $5 \rightarrow 3$, $3 \rightarrow 1$, $6 \rightarrow 3$ and $3 \rightarrow 2$ in the example. The pattern denotes a feasible solution for team 3, provided that the assignment of home opponents does not introduce any repeaters. Obviously, the presented pattern is not optimal since it contains more home-away edges than necessary.

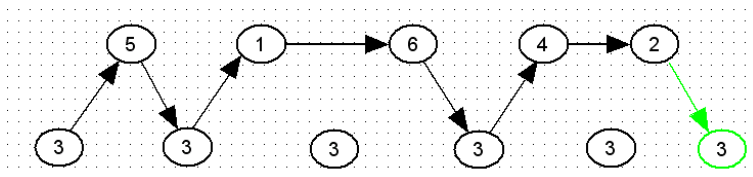


Fig. 1. Example of a pattern

In order to create the patterns, we first introduce a pattern array of length n . The elements in the array are 0 and 1. A 0 in the pattern means that the next team in the shortest path sequence should be visited without going home. A 1, on the contrary, means that the team should first go home before visiting the next team in the shortest path sequence. We do not put consecutive 1s in the patterns since the home games do not influence the travelling distance for the home team. The pattern corresponding to Fig. 1, for example, is 1001001. Any team always ends the competition at home, and thus the last element of the pattern array is always 1. This leaves $n - 2$ 0/1 elements to assign. The total number of possible assignments thus equals $2^{(n-2)}$. Table 2 indicates the number of feasible patterns (i.e. patterns representing solutions which have not more than 3 consecutive away games) for each of the TTP problems.

When mapping the possible patterns onto the shortest path, we can calculate the distance for each team-pattern combination. Although the shortest tours are equal for each team, the patterns are different. Those with the smallest distance are the most interesting patterns to select for the tournament. Obviously, the order in which home sequences or away sequences appear in a solution has no influence on the distance of the home team.

3 Optimising Heuristic

From the set of patterns, we select and combine those that lead to good quality solutions. In the optimisation step we improve the solutions with local search. Examples of the neighbourhood moves considered are:

- swap the first and the last opponent team in a sequence of three,
- swap entire sequences of opponent games,
- etc.

The results will be presented at the conference.

n	# feasible patterns	Best Dist.	Best feasible pattern
4	4	2134	0001
6	12	4065	010001
8	40	3727	00010001
10	137	4692	0001010001
12	463	6930	010010010001
14	1560	9986	01000100010001
16	5264	11810	01001000100010001
18	17945	10388	010001000100010001
20	60708	11480	01000101000100010001
22	205372	12198	0100010001001000100011
24	694769	14428	000100010001000100010001
26	2350385	15704	00100010001000100010001001
28	7951296	14192	0001000100011000100010010001
30	26899040	15475	010010001001000100010001001001
32	90998801	20377	01000100010001000100100010010001

Table 2. Number of feasible patterns per problem of size n , the best feasible distance and an example of a pattern corresponding to that distance, when mapped upon the TTP solution

References

1. A. Anagnostopoulos, L. Michel, P. van Hentenryck and Y. Vergados: A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 2006 (to appear).
2. U. Aickelin, K. Dowsland: An Indirect Genetic Algorithm for a Nurse Scheduling Problem. *Journal of Computers & Operations Research*, 31(5), 2004, 761-778
3. J.F. Bard, H.W. Purnomo: Preference scheduling for nurses using column generation, *European Journal of Operational Research* 164, 2005, 510-534
4. P. Brucker, R. Qu, E.K. Burke, G. Post: A Decomposition, Construction and Post-processing Approach for a Specific Nurse Rostering Problem, proceedings of MIS-TA 2005: The 2nd Multidisciplinary Conference on Scheduling: Theory and Applications, NY, USA, July 2005, 397-406
5. K. Easton, G. Nemhauser and M. Trick: Solving the Travelling Tournament Problem: A Combined Integer Programming and Constraint Programming Approach. In E. Burke and P. De Causmaecker (Eds). *Selected papers from the 4th International Conference on the Practice and Theory of Automated Timetabling. LNCS 2740*, 2003 100-109
6. K. Easton, G.L. Nemhauser and M. Trick: Sports scheduling, in *Handbook of Scheduling: Algorithms, Models and Performance Analysis*, Ch. 52, J.T. Leung (Ed.) 2004, 1-19. CRC Press
7. K. Easton, G. Nemhauser and M. Trick: The traveling tournament problem: Description and benchmarks. In T. Walsh, editor, *Principles and Practice of Constraint Programming, Lecture Notes in Computer Science, Vol. 2239*, Springer, 2001, 580-584
8. M. Henz, T. Mueller, and S. Thiel: Global constraints for round robin tournament scheduling. *European Journal of Operational Research*, Vol. 153, 2004, 92-101

9. M. Henz: Scheduling a major college basketball conference - revisited. *Operations Research*, 49, 2001, 163-168
10. J.H. Kingston: A Tiling Algorithm for High School Timetabling, Selected and Revised Papers of the 5th International Conference on Practice and Theory of Automated Timetabling, (PATAT 2004), Pittsburgh, Springer LNCS 3616, 2005, 208- 225
11. G.L. Nemhauser and M.A. Trick: Scheduling a major college basketball conference. *Operations Research*, Vol. 46, 1998, 1-8
12. J.A.M. Schreuder: Combinatorial Aspects of the Construction of Competition Dutch Professional Football Leagues. *Discrete Applied Mathematics*, Vol. 35, 301-312, 1992
13. M.B. Wright: Scheduling English Cricket Umpires. *Journal of the Operational Research Society*, 42:447-452, 1991
14. J.T. Yang, H.D. Huang and J.T. Horng: Devising a cost effective baseball scheduling by evolutionary algorithms. In *Proceedings of the 2002 Congress on Evolutionary Computation*, 2002, 1660-1665