Lancaster University
MANAGEMENT SCHOOL

SCHEDULING CRICKET UMPIRES USING NEIGHBOURHOOD SEARCH – THE DRAMATIC IMPACT OF A SIMPLE CHANGE IN NEIGHBOURHOOD DEFINITION

Mike Wright

Lancaster University Management School
Department of Management Science

Working Paper 2012:6

SCHEDULING CRICKET UMPIRES USING NEIGHBOURHOOD SEARCH – THE DRAMATIC IMPACT OF A SIMPLE CHANGE IN NEIGHBOURHOOD DEFINITION

Mike Wright

Abstract

This paper reports on experiences using a neighbourhood search approach for scheduling umpires for two amateur cricket leagues in England. Experimental analysis shows that the inclusion of a new type of perturbation in addition to the two types already in use leads to much improved results. Further analysis suggests that the perturbation types used are all of distinct meta-types, all of which are valuable in producing good solutions for complex combinatorial problems of this type.

Keywords: *scheduling, timetabling, neighbourhoods, sport, cricket*

Introduction

Sports scheduling/timetabling is a rapidly expanding discipline in both theory and practice. Kendall *et al.*'s survey paper (2010) on the topic refers to 162 papers on the subject, of which 66% were published since 2000, and more have appeared since then.

One important application of sports timetabling involves the allocation of officials to matches in sports leagues, without which these leagues cannot function. Implementations have been published relating to baseball (Evans *et al*., 1984; Evans, 1988; Trick *et al.*, 2012), to football (Zakarya *et al.*, 1989; Gil Lafuente, 2004; Yavuz *et al*., 2008), to softball (Saur *et al.*, 2012), to volleyball (Linfati, 2012) and to cricket (Wright, 1991; Wright, 2007).

Umpire scheduling can often be a very complex problem, as there are frequently many objectives, preferences and constraints (hard and soft) to accommodate. While one can sometimes use optimising software for such problems, frequently heuristics, especially metaheuristics, are necessary. This often involves neighbourhood search.

There is an enormous variety of types of neighbourhood search, but all types adhere to the same general pattern. An initial solution (which may or may not be feasible) is created by some quick method, after which a very large number of "iterations" or "perturbations" are considered in turn. Each perturbation gives rise to a new solution in the "neighbourhood" of the current solution. Some of these perturbations are "accepted", hence creating a new current solution, while others are "rejected", meaning that the current solution remains unchanged. The decision as to whether to accept or reject follows rules specific to the variation of neighbourhood search being considered. See, for example, Glover and Kochenberger (2003) for further descriptions, examples, analyses and discussions of some of the more commonly used types of neighbourhood search.

Thus for any implementation of neighbourhood search it is necessary to define the way in which a solution may be perturbed so as to produce a neighbouring solution.

In the example discussed in this paper, the initial definition of a perturbation was one of two simple manoeuvres: a "swap" or a "replacement". While the resulting neighbourhood search proved successful in terms of satisfying the customers for the implementations involved, it appeared that it could probably be improved. After analysing the solutions produced, a further type of perturbation was included, with dramatic consequences. The reasons are examined, together with speculations as to what may make a set of perturbations successful more generally.

Scheduling cricket umpires

A typical cricket league, such as exists throughout England and many other countries, involves several teams, usually divided into groups known as divisions, covering a reasonably compact geographical area. There may be several divisions, which may be organised hierarchically or geographically or a mixture of the two. Teams play several matches over a season – at amateur level this typically consists of between 16 and 24 rounds. The most common pattern in English cricket is for matches to be played every Saturday over a season which runs from April to September.

The most common fixture pattern is a double round robin, i.e. such that every team plays against every other team in its division twice during a season, with one match at each team's home ground. Timetabling the fixtures is often carried out using simple patterns, though it can in some cases be much more complicated, especially at professional level – see, for example, Wright (1994) and Willis and Terrill (1994).

Another task which must be undertaken for any cricket league is the scheduling (or rostering or timetabling – terminology varies) of umpires. The umpires are vital officials in charge of applying the laws of cricket and the rules of the tournament. Two umpires are required for every match. Scheduling umpires for a league can be considered as a combinatorial optimisation problem, subject to constraints and objectives.

The Devon Cricket League

The Devon Cricket League, in the county of Devon in south-west England, follows a common pattern. There are four hierarchical divisions, with ten teams in each division. At the end of each season, the top team in every division except the top one is promoted to the division above for the following season, and the bottom team in every division except the bottom one is relegated to the division below (though this can be complicated by teams ceasing to exist, as is common in amateur sport).

The competition within each division is in the format of a double round robin, i.e. every team plays twice against every other team, with one match at each team's ground. There are thus 18 rounds altogether in the competition, taking place on 18 consecutive Saturdays; 20 matches per round (five in each division); and thus 360 matches altogether, each of which requires two umpires. So the number of umpire allocations required is 720.

Typically there may be about 60 to 70 umpires available, who are therefore used on average about 10 to 12 times each, though with a wide variation.

The computer system devised to schedule these umpires operates as follows.

The hard constraints are:

- every match requires two umpires
- no umpire may be used on a date for which he has declared himself unavailable
- no umpire may be used for two matches on the same date
- some rules which are specific to the implementation in question: for example, some umpire-team combinations are considered infeasible

There are many other considerations, however, which are regarded as soft constraints or objectives. One such concerns the usage of each umpire. The league organisers like to be able to specify exactly how many matches each umpire should cover within each division, or maybe at each divisional level. These targets are very important to the league organisers. The best umpires will of course be required to operate mainly in the top division(s), but it may be valuable for them to drop down to a lower division from time to time, perhaps to help the more junior umpires. Likewise it may be helpful for middle-ranking umpires to have occasional exposure to top division matches, etc.

Other objectives concern travel distances and spread of umpires between teams, home grounds and each other. Further adaptations weight travel more heavily at certain times of year and encourage situations where two umpires can travel together, thus reducing costs. All the objectives are combined into a single mega-objective using weights which have been developed iteratively to ensure that the resulting solutions are good ones in practice. Fuller details of the formulation are given in Wright (2007).

Neighbourhood search approach

The search starts from a quickly constructed random initial solution which obeys the hard constraints. It then orders the matches randomly, and then operates a Simulated Annealing procedure on the matches in this order, as described below. When the list of matches has been exhausted, they are randomly reordered and the process restarts. This continues until a fixed total number of iterations have been accomplished.

While this is not the standard form of Simulated Annealing, it is a well-established approach (see Dowsland, 1995) which ensures that all possible perturbations are considered at various temperature levels and makes it very likely that the final outcome is at a local optimum (though in this case the search always ends with a simple local descent so as to ensure this).

The original neighbourhood definition was simple: replace one umpire by another for a specific match (a *replacement* perturbation), or exchange two umpires between two matches (a *swap* perturbation). The perturbation was accepted or rejected according to a variation of simulated annealing developed for cases such as this where a single overall objective is made up of several subobjectives – see Wright (2001) for details. When a perturbation is accepted, it becomes the new current solution. A geometric cooling scheme is used between starting and ending temperatures determined through a certain amount of trial and error during the development stages.

Replacement perturbations were important to enable the total workload of an umpire to change, and thus help to achieve the target numbers of matches for the umpires. Swap perturbations were needed so as to allow for changes that don't change an umpire's workload. It was envisaged at this stage that these two types of perturbation would be sufficient.

For a given match and position (1 or 2), the search process goes through all possible replacements in a randomised order, then through all possible swaps in a randomised order, before then going on to the next match/position etc., except that, once a perturbation is accepted, the search immediately moved on to the next match/position so as to avoid instant reversals.

This method ensures that all possible perturbations of both types are considered at a wide range of temperatures from high to low. It also means that the final solution reached is likely to be very close to a local optimum (since the final temperature is low), though to make sure of this the search ends with a simple descent procedure, starting from the best solution found during the search.

The Home Counties League

Recently, the system was adapted for the Home Counties League in South-East England. The structure is slightly different, with one top division and two others of equal status, organised geographically. The computer system was amended so that two divisions could be considered at an equal *level*; thus the target numbers related to the number of matches at each level rather than specifically in each division. In other respects, however, the problem was very similar to that faced by the Devon League.

The system developed for the Devon League was applied for the Home Counties League and the resulting solution was implemented in practice, to the great satisfaction of the League's organisers.

However, while the client was happy, it was clear to the developer that the solutions reached may not have been as close to optimal as had been expected. In particular, on several occasions an umpire/team combination occurred twice within three weekends, which rarely happened for the Devon League, and there were other unexpectedly high costs.

Further inspection of the results, and comparisons between the two leagues, uncovered a possible reason, concerning the umpires' targets. The Devon League has sufficient umpires of sufficiently high standing to allow for some slack for most of them, enabling flexibility when schedules must be changed, e.g. because of illness. Thus, for example, a top-class umpire available for 17 of the 18 weekends might be given an overall target of only 15 matches. As a result, there was plenty of scope for swapping umpires between dates during the neighbourhood search process without breaking any hard constraints.

However, mainly because of a shortage of suitable umpires, the Home Counties League uses its best umpires as often as possible. Thus, once these umpires' targets have been met, swaps involving such umpires are feasible only between two matches taking place on the same date. This greatly restricts the search; moreover, if an umpire currently meets his targets there are few perturbations available which do not break these, i.e. only those between matches at the same divisional level and on the same date.

It seemed therefore that a redefinition of perturbation types could be useful here. Variable Neighbourhood Search (VNS) (Hansen & Mladenovic, 2001) was considered, but not implemented since it can increase neighbourhood size enormously and, in its simplest form at least, does not focus on the specific issue which may be causing substandard solutions to be produced. A more focused approach was sought which would detect potentially valuable types of perturbation to use throughout the search, and not just when progress was no longer being made, as is usually the case with VNS.

Experiments with a new type of perturbation

Thus, after a detailed inspection of the solutions being produced, a new type of perturbation was defined, involving two umpires, two dates and four matches, such that the umpires swap matches on both dates (a *double* perturbation). This allows umpires to change division levels on each date without breaking any soft constraint. It could be seen that this had the potential of improving some solutions.

100 experimental runs were made on the Home Counties data for each method: the "old" method, involving only replacement and swap perturbations, and the new one, which also included double perturbations. For each run the stopping criterion was if the total number of perturbations considered was greater than or equal to 20 million (this was only checked at the end of a loop through the matches).

The results were surprisingly dramatic, with the solutions produced by the new method being on average about 22.3% less costly than those produced by the old method. This was a much greater improvement than had been expected, and served to highlight the poor quality of the original solutions as implemented by the clients. Inspection showed that nearly all improvements related to the top division.

The significance of this improvement was emphasised by the fact that 68 of the 100 solutions produced by the new method were better than every single solution produced by the old method, and that every single solution produced by the new method was better than 94 of the solutions produced by the old method.

The same process was then carried out for Devon data, and the results were even more surprising, given that it had not been apparent that there was a problem. Here the improvement averaged about 20.4%. Inspection here showed that the improvements were spread more evenly throughout the divisions, which was perhaps why it had been less clear form initial inspection that major improvements were possible.

Here as many as 95 of the 100 solutions produced by the new method were better than every single solution produced by the old method, and every single solution produced by the new method was better than 98 of the solutions produced by the old method.

Further analysis was undertaken which looked at the prevalence and effects of each type of perturbation. The results below are given just for the Home Counties League runs, though similar patterns are shown throughout for the Devon runs as well.

Table 1 – prevalence and acceptance rate of each perturbation type

	% of total iterations	% accepted	% of total acceptances
Replacement	10%	0.128%	9.47%
Swap	16%	0.723%	83.44%
Double	74%	0.013%	7.08%
Total		0.135% (compared with 0.490% without doubles)	

This shows that the new double perturbations dominated proceedings, comprising nearly three-quarters of all perturbations. It also shows that the results of double perturbations are hardly ever accepted. However, we know from the results of these runs that their inclusion brings about substantial improvements to the final solutions!

Further analysis was carried out to determine the different impacts of the perturbation types at different stages of the search. In the following Figures, the x-axis gives each percentile of the search, so that, for example, a value of x =20 refers to what was happening on average between the 380,001st and the 400,000th perturbation. The x-axes go slightly beyond 100% because of the way that the stopping criterion was implemented and because of the final descent procedure.

In each Figure, the unbroken line relates to replacements, the dotted line to swaps and the dashed line to doubles.

Figure 1 – Acceptance rates by % of way through search

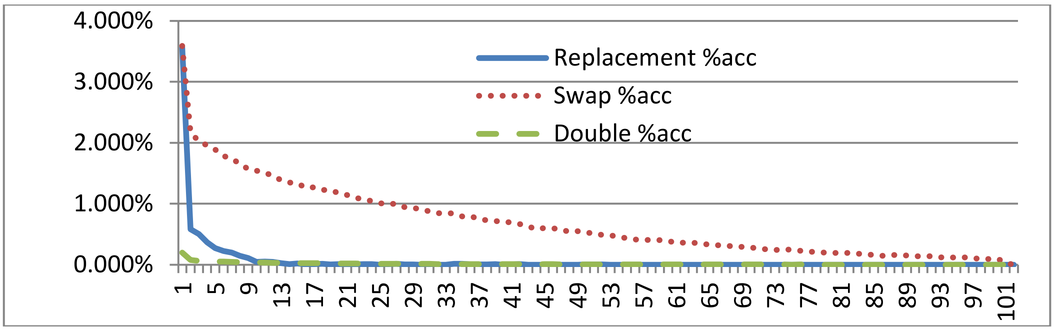


Figure 2 – Part of Figure 1 in close-up

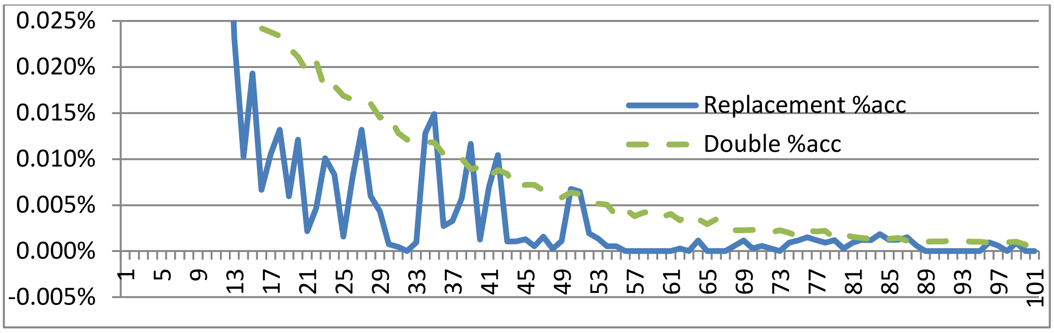


Figure 3 – % of total acceptances for each type

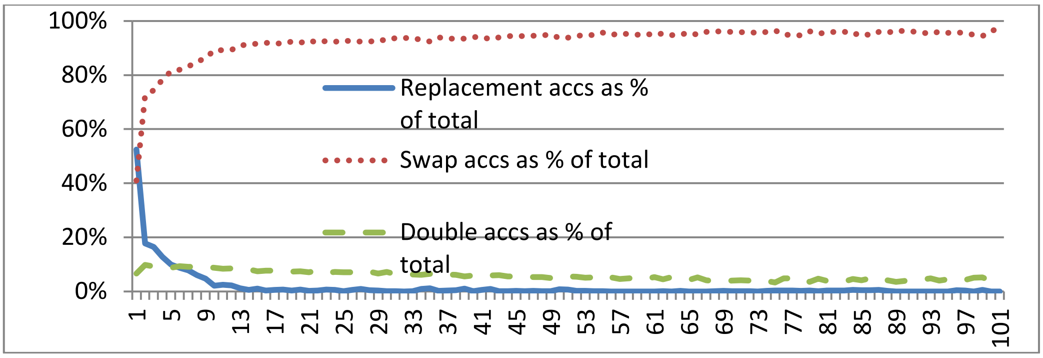
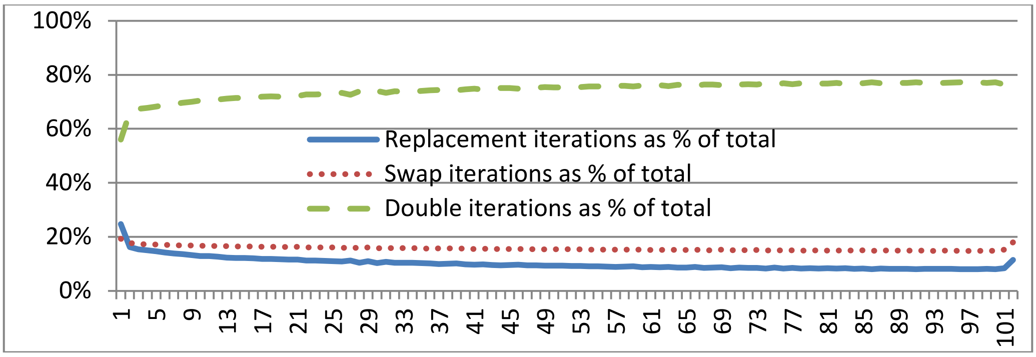


Figure 4 – % of total iterations for each type



Conclusion

These Figures show very distinct differences in the contribution patterns of each type of perturbation, suggesting that the three types of perturbation are representatives of three meta-types, as follows:

1. "Get rich quick", making strong early gains, eliminating most unnecessary major costs such as soft constraint violation penalties, but of little value when fine-tuning later in the search;
2. "Steady as she goes", valuable throughout the process, especially when fine-tuning at the end;
3. "Occasional gem", only rarely useful, but capable of producing dramatic improvements.

It is hypothesised that any neighbourhood search process is more likely to produce high-quality solutions to complex problems if it contains perturbations of all three meta-types.

References

Dowsland, K.A. (1995) "Simulated Annealing", in: C.R. Reeves (ed.) *Modern heuristic techniques for combinatorial problems*. McGraw-Hill, Maidenhead, Chapter 2, pp 20-69.

Evans, J.R. (1988) "A microcomputer-based decision support system for scheduling umpires in the American baseball league". *Interfaces* 18(6), 42-51.

Evans, J.R., Hebert J.E. and Deckro R.F. (1984) "Play ball – the scheduling of sports officials". *Perspectives in Computing* 4(1), 18-29.

Gil Lafuente, J. (2004) "The best systems for appointing referees" in: "Economics, Management and Optimization in Sports", eds. S. Butenko, J. Gil Lafuente and P.M. Pardalos, 101-120, Springer-Verlag, Berlin.

Hansen, P. and Mladenovic, N. (2001) "Variable Neighborhood Search: principles and applications". *European Journal of Operational Research* 130(3): 449-467.

Kendall, G. X., Knust S., Ribeiro C.C., & Urrutia, S. (2010). Scheduling in sports: an annotated bibliography. *Computers and Operations Research 37(1),* 1-19.

Linfati, R. (2012) "Referee Assignment Problem Case: Italian Volleyball Championships". PhD Thesis, University of Bologna, Italy.

Saur, M.C., Starr, K., Husted, M., Newman, A.M. (2012) "Scheduling softball series in the Rocky Mountain Athletic Conference". Interfaces 42(3), 296-309.

Trick, M.A., Yildiz, H. and Yunes, T. (2012) "Scheduling major league baseball umpires and the traveling umpire problem". Interfaces 42(3), 232-244.

Willis, R.J. and Terrill, B.J. (1994) "Scheduling the Australian state cricket season using simulated annealing". *Journal of the Operational Research Society* 45(3), 276-280.

Wright, M.B. (1991) "Scheduling English cricket umpires". *Journal of the Operational Research Society* 42(6), 447-452.

Wright, M.B. (1994) "Timetabling county cricket fixtures using a form of tabu search". *Journal of the Operational Research Society* 45(7), 758-770.

Wright, M.B. (2001) "Subcost-guided simulated annealing", in: "Essays and surveys in metaheuristics", eds. C.C. Ribeiro and P. Hansen, chapter 28, 631-639 (Kluwer Academic Publishers, Boston).

Wright, M.B. (2007) "Case study: problem formulation and solution for a real-world sports scheduling problem". *Journal of the Operational Research Society* 58(4), 439–445.

Yavuz, M., İnan, U.H. and Fığlalı, A (2008) "Fair referee assignments for professional football leagues". *Computers and Operations Research* 35(9), 2937-2951.

Zakarya, Z., Hertz, A. and de Werra, D. (1989) "Un système informatique pour les calendriers d'arbitrage d'une association sportive". Report of the Départment de mathématiques, École Polytechnique Fédérale de Lausanne, Switzerland.