OR60 Annual Conference - Keynote Papers and Extended Abstracts
Lancaster University. Lancaster. 11-13 September 2018

TWO EXAMPLE OPTIMISATION PROBLEMS FROM THE WORLD OF EDUCATION

Rhyd Lewis

School of Mathematics, Cardiff University
Cardiff CF24 4AG, WALES
LewisR9@cf.ac.uk

ABSTRACT

This work considers two distinct combinatorial optimisation problems related to education, namely lecture timetabling and school bus scheduling, both of which are known to be NP-hard. Our research into these problems has centred around the design of various high-performance heuristics that are able to produce good quality solutions to these problems in short amounts of time. To do this, we propose that it is necessary to "get to the heart" of these problems by identifying their underlying sub-problems. This, in turn, helps to inform the design of algorithmic operators that are able to exploit these structures and help to produce the solutions we need. In this extended abstract these problems are briefly considered in turn.

1 Lecture Timetabling

The first problem we consider is lecture timetabling, which has crossovers with the classical operational research problems of graph colouring and bipartite matching. According to Lewis (2008), university timetabling "can be considered the task of assigning a number of events, such as lectures, exams, meetings, and so on, to a limited set of timeslots (and perhaps rooms) in accordance with a set of constraints". For this research we consider the "post enrolment" timetabling model, where the constraints of the problem are specified by student enrolment data—that is, each student gives a list of the subjects they would like to take and the final timetable must seek to accommodate these choices. Specifically, we use the model proposed in the second international timetabling competition (http://www.cs.qub.ac.uk/itc2007/), which can be briefly summarised as follows:

The problem involves a number of "hard" constraints whose satisfaction is mandatory, together with three "soft" constraints, whose satisfaction is desirable, but not essential. We are then required to assign the set of events to 45 timeslots (five days, with nine timeslots per-day) according to these constraints. The hard constraints for this problem are as follows. First, for each event there is a set of students who are enrolled to attend—thus events should be assigned to timeslots such that no student is required to attend more than one event in any one timeslot. Next, each event also requires a set of room features (e.g. a certain number of seats, specialist teaching equipment, etc.), which will only be provided by certain rooms; thus each event needs to be assigned to a suitable room that exhibits the room features that it requires. The double booking of rooms is also disallowed. Hard constraints are also imposed stating that some events cannot be taught in certain timeslots. Finally, precedence constraints—stating that some events need to be scheduled before or after others—are also stipulated. The soft constraints for this problem, meanwhile, are as follows: first, students should not be asked to attend an event in the last timeslot of each day (i.e. timeslots 9, 18, 27, 36, or 45); second, students should not have to attend events in three or more successive timeslots occurring in the same day; third, students should not be required to attend just one event in a day.

Arguably, most of the current best-performing algorithms for this problem have been based on neighbourhood search methods (Lewis and Thompson 2015). To design such a method, three ingredients are needed: a *solution space*, which defines the set of all possible candidate solutions; a *cost function*, which measures the quality of each member of the solution space; and a *neighbourhood operator*, which allows us to move from one solution to another in the solution space. A neighbourhood search algorithm operates by using the neighbourhood operator (or operators) to move around the solution space, seeking the solution with the smallest cost. Examples of such algorithms applied to this particular problem include simulated annealing (Cambazard, Hebrard, O'Sullivan, and Papadopoulos 2012), tabu search (Lewis and Thompson 2015), ant colony optimisation (Nothegger, Mayer, Chwatal, and Raidl 2012) and evolutionary algorithms (Jat and Yang 2011).

Many of the most successful algorithms for this problem have also operated using a two-staged approach. Here, the soft constraints are effectively ignored in the first stage, and the algorithm only seeks a feasible solution (i.e., one satisfying all of the hard constraints). Once this is achieved, the second stage can then be invoked, which will seek to make changes to a solution in order to reduce the number of soft constraint violations, while not re-violating any of the hard constraints. Note in particular that many potential changes to a feasible solution (e.g. moving an event from one timeslot to another) will indeed cause a hard constraint to be violated and must be forbidden; hence, movements in the solution space can often be very restricted because only a small number of paths, if any, can be taken from a particular solution.

To look at this issue more closely, consider an algorithm whose solution space comprises all candidate solutions, feasible and infeasible, for a particular instance of this timetabling problem. Now consider the graph $G = (V,E)$ where each vertex $v \in V$ represents a different member of the solution space, and an edge between two vertices, denoted by $(u,v) \in E$, indicates the existence of a move under our neighbourhood operator that, when applied to solution u, will result in solution v. As noted, in Stage 2 of the two-staged approach infeasible solutions are not allowed, so our solution space now becomes the graph induced by V', where $V' \subseteq V$ is the set of all feasible solutions. This new graph will clearly contain far fewer edges than G; indeed, it may also be disconnected, meaning that if all of the "good" solutions to a particular problem instance happen to be in different components to that of our current solution, we will never be able to reach them. It is therefore in our interests to try to increase the connectivity of this solution space. In this research, this is achieved using two ideas that exploit the underlying structures of this problem. This allows a greater proportion of proposed neighbourhood moves to maintain solution feasibility and facilitates less restricted movements in the solution space. These are now outlined.

Maximum Matching: As noted above, an important constraint with this problem is ensuring that events are assigned to suitable rooms. Given a set of events assigned to a particular timeslot, the existence of a suitable room allocation can actually be determined in polynomial time by solving a maximum bipartite matching problem. If a neighbourhood move seeks to insert a new event into a particular timeslot, we can therefore use this property to determine whether the room allocations of this timeslot can be appropriately rearranged to allow this extra event to be accommodated.

Graph Colouring: In graph colouring, a Kempe chain is a connected sub-graph containing vertices of just two colours. It is also known that, given a proper colouring, the interchanging of vertex colours in a Kempe chain will result in a new proper colouring with different colour assignments (Lewis 2016). For this timetabling problem, the ideas behind Kempe chains can be used to achieve additional movements in the solution space that are not possible under more simple operators. More specifically, if the movement of an event from timeslot A to timeslot B were to cause a solution to become infeasible (e.g., by requiring a student to be in two places at once), Kempe chains might allow us to overcome this by identifying some events in timeslot B that, when moved to timeslot A, will maintain solution feasibility.

Our research has clearly indicated that when the connectivity of the underlying solution space is increased through the application of these operators, the quality of the returned solutions in also enhanced. We have also looked at some other related techniques, such as using additional "dummy rooms" to help aid the search, though the results of these are much more mixed. Further details are reported by Lewis and Thompson (2015).

2 School Bus Scheduling

The second problem we consider in this paper concerns the design of school transport schedules (Lewis, Smith-Miles, and Phillips 2018). This involves compiling a list of eligible school children and then organising their transport to and from school; in particular, it requires the selection of a suitable set of pick-up points (bus stops), the assignment of students to these bus stops, and the design of bus routes that visit these stops while getting students to school on time. In doing this, a number of constraints should be considered:

1. We should try to reduce economic costs by minimising the number of vehicles used;
2. Each student should be picked up from a bus stop within walking distance of their home address;
3. We cannot assign more students to a bus than there are available seats on the bus;
4. Journey times of students should not exceed a specified limit (e.g., one hour).

School bus routing problems belong to a wider family of vehicle routing problems (VRPs), which involve identifying routes for a fleet of vehicles that are to serve a set of customers. Traditionally, VRPs can be expressed using an edge-weighted directed graph $G = (V,E)$, where the vertex set $V = \{v_0, v_1, \ldots, v_n\}$ represents a single depot and n customers (v_0 and $v_1, \ldots, v_n$ respectively), and the weighting function $d(u,v)$ gives the travel distance (or time) between each pair of vertices $u, v \in V$. As we might expect for a such an important and pervasive problem, VRP formulations come in very many guises and can include many additional features such as time-windows, limitations on routes lengths, ensuring suitable vehicles for each route, partitioning customers into pick-up and delivery locations, and the dynamic recalibration of routes subject to the arrival of new customer requests during the transportation period (Laporte 2009, Pillac, Gendreau, Guéret, and Medagila 2013).

One of the main differences between this school bus routing problem and more "traditional" VRPs is that, when planning school bus routes, it is not always clear which set of stopping points we should actually use. In general, a schoolchild may live within walking distance of a number of bus stops, and it makes sense to choose the stop that suitably balances the child's walking distance with the resultant time added to the bus's journey. It may also be advantageous to assign larger groups of students to the same bus stop in order to reduce the number of times that a bus needs to slow down, pick up children, and then rejoin the traffic stream. This problem therefore involves the additional complication of choosing a feasible subset of bus stops from the set of all available bus stops. A VRP along the more traditional lines mentioned above might then be considered using this subset.

The task of choosing a feasible subset of bus stops can be viewed as a set covering problem. Set covering involves taking a set of integers $U = \{1, 2, ..., n\}$ known as the "universe" together with a set S whose elements are subsets of the universe. The task is to then identify a subset $S' \subseteq S$ whose union equals the universe. With regards to the bus scheduling problem, S corresponds to the set whose elements are the addresses within walking distance of each bus stop. A feasible subset

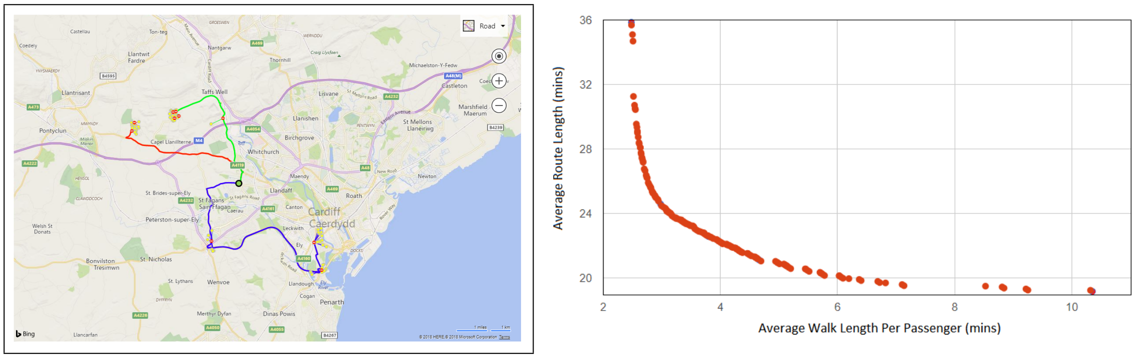

Figure 1: An example solution using three buses for a school in Cardiff, Wales (left). The chart on the right shows an approximate Pareto front for the same problem.

of bus stops (i.e., one that features at least one bus stop within walking distance of each address) then corresponds to a subset $S' \subseteq S$ that covers all addresses.

In our experience, the main consideration of school transport planners is to minimise the number of vehicles used, which has the potential to offer significant financial savings. Our strategy for this problem involves using a fixed number of routes k and searching through the solution space defined by all solutions obeying Constraints 2 and 3 above. We then apply specialised VRP-based operators to try and shorten journey times, therefore also hopefully satisfying Constraint 4. If this cannot be achieved at a certain computation limit, k is increased by one, and the algorithm is repeated (initially k equals a lower bound, calculated as the total number of students divided by the bus capacity, rounded up to the nearest integer). This process is also interlaced with an operator based on set covering heuristics, which makes alterations to the set of bus stops being used by a solution, while ensuring that Constraints 2 and 3 remain satisfied.

Our experiments with this algorithm have involved both artificial and real-world problem instances (Lewis, Smith-Miles, and Phillips 2018). In general, we find that larger numbers of vehicles are needed when the maximum journey times are low and/or the maximum walking distances are low. This is quite natural because stricter journey limits imply the need for additional routes in feasible solutions, while shorter walk limits mean that more bus stops need to be visited. For real-world problem instances we have also found that, once a feasible solution using as few routes as possible has been achieved, stakeholders are also interested in making further adjustments in order to make the service both cost efficient and convenient for the users. From the financial point-of-view, administrators are interested in keeping the lengths of each route as short as possible because this will attract lower quotes from the commercial bus companies that want to provide the service. On the other hand, users want to be assigned to bus stops close to their homes, keeping walks short and discouraging parents from driving their children to the bus stop. Clearly these objectives are in conflict, because a bus route that visits many stops (allowing shorter walks) will also tend to be longer and therefore more expensive to run. Decision makers are therefore interested in looking at a set of non-dominating solutions that, for a fixed number of vehicles k, shows how these objectives influence one another, allowing them to choose a solution seen to be an appropriate compromise of the two. This naturally points to the suitability of multiobjective-based optimisation methods. A number of Pareto front approximations for our real world problem instances, generated by our bespoke methods, can be found at http://www.rhydlewis.eu/sbrp. Examples from this resource are provided in Figure 1.

REFERENCES

Cambazard, H., E. Hebrard, B. O'Sullivan, and A. Papadopoulos. 2012. "Local search and constraint programming for the post enrolment-based timetabling problem". *Annals of Operational Research* 194:111–135.

Jat, S., and S. Yang. 2011. "A hybrid genetic algorithm and tabu search approach for post enrolment course timetabling". *Journal of Scheduling* 14:617–637.

Laporte, G. 2009. "Fifty Years of Vehicle Routing". *Transportation Science* 43:408–416.

Lewis, R. 2008. "A Survey of Metaheuristic-based Techniques for University Timetabling Problems". *OR Spectrum* 30 (1): 167–190.

Lewis, R. 2016. *A Guide to Graph Colouring: Algorithms and Applications*. Springer International Publishing.

Lewis, R., K. Smith-Miles, and K. Phillips. 2018. "The School Bus Routing Problem: An Analysis and Algorithm". In *Combinatorial Algorithms*, Volume 10765 of *Lecture Notes in Computer Science*, 287–298. Springer-Verlag.

Lewis, R., and J. Thompson. 2015. "Analysing the Effects of Solution Space Connectivity with an Effective Metaheuristic for the Course Timetabling Problem". *European Journal of Operational Research* 240:637–648.

Nothegger, C., A. Mayer, A. Chwatal, and G. Raidl. 2012. "Solving the post enrolment course timetabling problem by ant colony optimization". *Annals of Operational Research* 194:325–339.

Pillac, V., M. Gendreau, C. Guéret, and A. Medagila. 2013. "A Review of Dynamic Vehicle Routing Problems". *European Journal of Operational Research* 225:1–11.

AUTHOR BIOGRAPHY

RHYD LEWIS is a senior lecturer in operational research at the School of Mathematics, Cardiff University, Wales. He holds a PhD in computing and operational research (Edinburgh Napier University, 2006) and is the author of the book *Graph Colouring: Algorithms and Applications* (Springer, 2015). Further publications are listed at his website http://www.RhydLewis.eu.