Time windows and constraint boundaries for public transport scheduling

Ignacio Laplagne, Raymond S K Kwan, and Ann S K Kwan

School of Computing, University of Leeds,
Leeds LS2 9JT, UK
{ignacio, rsk, ann}@comp.leeds.ac.uk

Public transport driver scheduling is the problem of determining the composition of a set of driver shifts (a *schedule*) for a day's transport operation requiring coverage by drivers, while minimising the operational cost (and/or robustness) of the schedule [2]. A *relief opportunity* (RO) is a (time, location) pair where drivers can be relieved. In the case of rail driver scheduling, most relief opportunities occur when a train stops at a station. It is frequently the case that trains will stop for some time before continuing; this gives rise to *windows of relief opportunities* (WROs).

Driver scheduling models usually approximate windows of relief opportunities by their arrival time. WROs could be expanded into sets of 1-minute-apart ROs, but the resulting model is unsuitable to be solved using the generate-and-select (GaS) approach [1], because the number of valid shifts becomes unmanageable in size [3]. However, it is expected that not all of the ROs derived from WROs will be vital for yielding more efficient solutions. For example, some of these new ROs that are close together are likely to be redundant.

Applying the 1-minute expansion to a typical instance of the rail driver scheduling problem in the UK is likely to result in several hundred new ROs. The problem is then how to select which of these potential ROs to include in the expanded model; a brute-force approach (say, trying all subsets of size n, one at a time) is clearly unsuitable, even for a fixed number n of ROs. In this work we present a set of heuristics to select these ROs.

Many scheduling constraints can be looked at in terms of the *boundaries* they define. Figure 1 depicts such an example: given an RO r on vehicle v at time t, a maximum work spell length of x minutes will define a boundary in vehicle v at time $t-x$, such that any spell on vehicle v ending at r will satisfy this constraint if the spell starts at or after $t-x$, and will break the constraint otherwise. If $t-x$ falls inside a WRO w at vehicle v (but not at its arrival time), then considering relieving inside w at $t-x$ leads to forming a spell which was invalid on the simplified, relief-on-arrival model. This would indicate that the RO at vehicle v and time $t-x$ is a good candidate to be included in the extended model, because it allows for a new spell to be generated. There may be more than one potential RO within w that fall on or after $t-x$, and all of these could in principle be included in the expanded set, although this would probably result in too many ROs being selected.

We derive a general framework to look at scheduling constraints in terms of time boundaries. We show its application using different constraints, including

E. K. Burke, H. Rudová (Eds.): PATAT 2006, pp. 458–459. ISBN 80-210-3726-1.

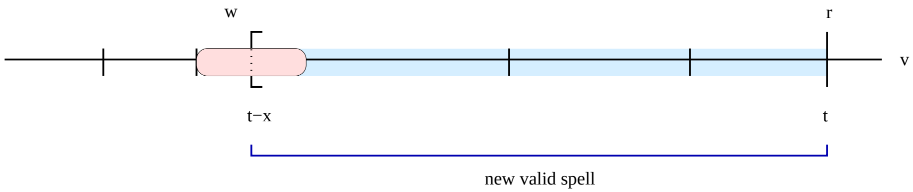

Fig. 1. Looking at boundary conditions for the *maximum spell length* constraint. A maximum spell length of x defines an interval $[t - x, t)$ for the start of a spell ending at time t. A new spell can be formed if the RO at time $t - x$ is added to the model.

the existence of a feasible travel link and the maximum spell length. In particular, we apply this analysis to the first constraint on a set of real-life driver scheduling instances from four different UK railway operations. By adding these selected ROs, we are able to improve on best-known solutions. A further study on one of these instances shows that the same result could have been achieved by adding just one of the about 90 ROs added by our heuristic. This reinforces our claim that a careful selection of the ROs to add may be crucial in achieving the best solutions.

Analysing a set of scheduling constraints simultaneously opens up a range of possible algorithms/heuristics. A straightforward way of doing so is to look at each constraint separately, and then somehow merge the sets of ROs obtained. A completely opposite approach is to observe that the structure of the new spells/shifts arising from the consideration of scheduling constraints is usually similar across different constraints, e.g. new spells are obtained by adding a piece of work at the start of a valid spell in the simplified model, and making the new spell start properly inside a WRO. Therefore, a possible algorithm consists in forming new spells/shifts with that common structure, then test whether these are valid and suggest new ROs to be considered.

We develop two such algorithms: one working at spell level; the other, at shift level, using the shifts created in the generation phase of a GaS solver as a base for creating new shifts. Results show that this kind of approach can effectively encompass several scheduling constraints simultaneously.

References

1. Sarah Fores, Les Proll, and Anthony Wren. TRACS II: a hybrid IP/heuristic driver scheduling system for public transport. *Journal of the Operational Research Society*, 53:1093–1100, 2002.
2. Raymond S. K. Kwan. Bus and train driver scheduling. In J. Y-T Leung, editor, *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, chapter 51. CRC Press, 2004.
3. Ignacio Laplagne, Raymond S. K. Kwan, and Ann S. K. Kwan. A hybridised integer programming and local search method for robust train driver schedules planning. *Lecture Notes in Computer Science*, 3616:71–85, 2005.