REPAIR-COST-GUIDED LOCAL SEARCH FOR 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

Keywords: Local Search, infeasible solutions, repair methods, scheduling, driver scheduling

Local Search methods have proved very useful in the field of combinatorial optimisation, particularly in the areas of scheduling and timetabling (Schaerf, 1999). One of the main tasks faced when implementing a local search approach for a specific problem is the design of the neighbourhood structure(s). In particular, it is frequently difficult to find neighbourhoods that are small enough to be computationally feasible to explore, and at the same time allow optimal solutions to be reached by moves inside these neighbourhoods. Local search frameworks generally request that intermediate solutions are *feasible*, i.e. they satisfy all the constraints imposed by the problem definition. Considering infeasible solutions as part of the solution space would often make it easier to define neighbourhood structures in which the optimal solutions are reachable, by allowing a feasible solution to be reached through a sequence of infeasible intermediate solutions; however, there usually is no natural, fair way of comparing feasible and infeasible solutions in terms of the original cost function. Moreover, the search might stray into the infeasible region an never be able to actually output a feasible solution to the user.

Many authors allow for infeasible solutions to be considered by augmenting the original objective function of the problem with a series of *penalty terms* (Hertz, 1992); through them, a solution becomes more unattractive (in terms of the augmented cost function) as the constraints get "more violated". A drawback of this approach is that the modeller must find the right way of weighing the original cost component against the "amount of constraint violation", which usually involves empirically determining individual weights for each type of constraint violation that may occur. Some authors provide heuristic methods of adaptively adjusting these weights during the search (Gendreau et al., 1994).

In this paper we present ongoing research on a local search framework that allows infeasible solutions to be part of the search space. This is based on the use of a fast *repair heuristic*, which transforms infeasible solutions into feasible ones. When considering the cost of an infeasible solution s, the repair operator is first applied, obtaining a feasible solution s_f; the cost of s is then defined to be the cost of its repaired, feasible equivalent s_f. Thus, although costs are computed on feasible solutions, the search may proceed on infeasible solutions. Similar ideas, based on the concept of the *Baldwin Effect* (Hinton and Nowlan, 1987), have been developed in the context of evolutionary algorithms (Kaige et al., 2004).

Working within this framework raises a number of questions. How much is the repaired solution an indication of the quality of the original infeasible solution, in terms of its contribution to the search process? What is the behaviour of the search process in the boundary between the feasible and infeasible regions? In particular, how can the transition from a feasible to an infeasible solution be encouraged if the repair heuristic is discouraging it? A number of tests and variants to the method are presented to tackle these questions.

We apply this technique to the driver scheduling problem. Generally modelled as a set covering problem, 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 of the schedule (Kwan, 2004). This problem is very well suited to this approach for two reasons: first, constraints imposed by labour union rules result in a situation where it is usually difficult to find feasible shifts, and even more so as a result of a "local move"; second, a fast repair heuristic can be easily derived from the generate-and-select approach used to solve set covering problems.

References

Gendreau, M., Hertz, A., and Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management Science*, 40(10):1276–1290.

Hertz, A. (1992). Finding a feasible course schedule using tabu search. *Discrete Applied Mathematics*, 35:255–270.

Hinton, G. E. and Nowlan, S. J. (1987). How learning can guide evolution. *Complex Systems*, 1:495–502.

Kaige, S., Narukawa, K., and Ishibuchi, H. (2004). Lamarckian repair and darwinian repair in emo algorithms for multiobjective 0/1 knapsack problems. In *GECCO 2004 (late breaking papers), Seattle, USA*.

Kwan, Raymond S. K. (2004). Bus and train driver scheduling. In Leung, J. Y-T, editor, *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*, chapter 51. CRC Press.

Schaerf, A. (1999). A survey of automated timetabling. *Artificial Intelligence Review*, 13(2):87–127.