Specifying and Solving Minimal Perturbation Problems in Timetabling

Jeffrey H. Kingston

The University of Sydney

Minimal perturbation problems

Adapt a timetable to unexpected new requirements, while minimizing changes

- Important when requirements change after timetable publication
- Every timetabling problem has its minimal perturbation variant
- Especially important in university course timetabling
- Very small literature

This paper

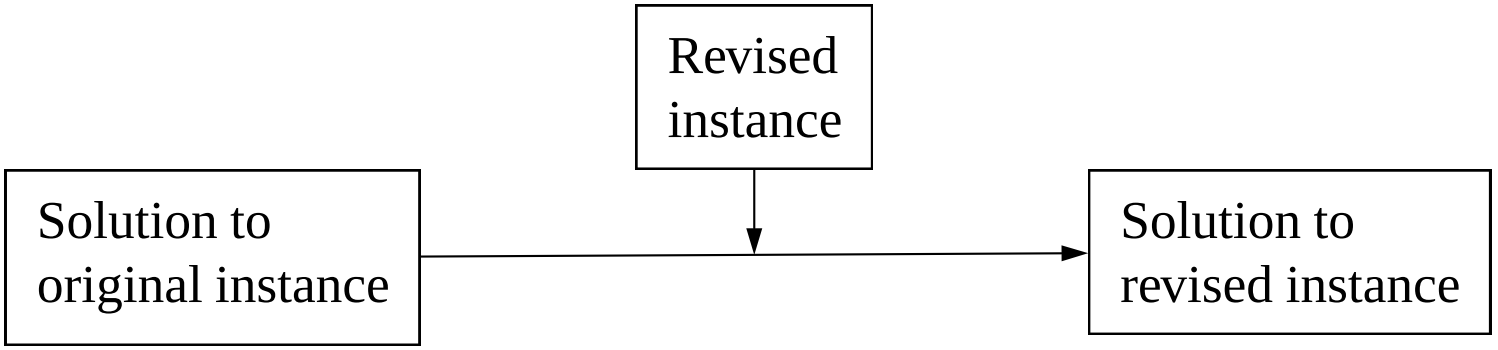
- A simple method of specifying minimal perturbation problems
- Allows arbitrary changes to requirements without listing all kinds
- Allows some repair algorithms to be re-used
- Works with all kinds of timetabling problems

First specification method

Original instance + Solution to original instance + Revised instance

- Allows arbitrary changes to requirements
- This is how the problem presents itself in practice
- But no obvious way to solve from here

Converting the original solution



- Solution is a list of assignments of times and resources to events
- Simply carry across all assignments that work in the revised instance
- Result may be incomplete and otherwise defective; but it’s legal

Second specification

Revised instance + extra constraints + Revised solution (optional)

Example of extra constraints

- Suppose revised solution assigns *Mon2* to event *Physics2*
- Then add ‘Penalty 20 if *Physics2* is not assigned *Mon2*’ to revised instance

Does it work with all kinds of timetabling problems?

- Instances must offer ‘prefer times’ and ‘prefer resources’ constraints
- Solutions must allow (but penalize) missing assignments

Penalty weights for extra constraints

There is an unavoidable tradeoff here, made explicit by the weights:

- High weights favour preserving existing solution
- Low weights favour finding a good revised solution

Weights can vary, e.g. high for times, low for rooms.

Consequences for solving

Revised instance + extra constraints + Revised solution (optional)

- Revised instance is a new instance of the original problem
- Solving it from scratch with original solver is correct but inefficient
- Some existing solvers can be both correct and efficient:

  (1) Start from revised solution (e.g. simulated annealing)

  (2) Focus on changed parts of solution (e.g. VLSN search)

Conclusion

- A simple method of specifying minimal perturbation problems
- Allows arbitrary changes to requirements without listing all kinds
- Allows some existing repair algorithms to be re-used