Computer Science & Information Technology

Faculty of Science

asap automated scheduling optimisation & planning

Investigating a Graph-Based Hyper-heuristic for Timetabling Problems

Rong Qu
ASAP Group, School of CSiT
University of Nottingham

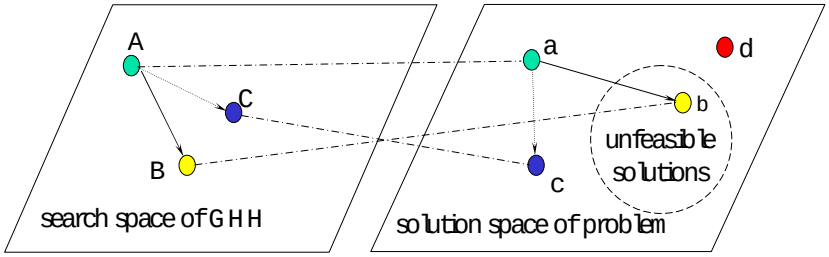
Content

- Hyper-heuristics
- A Graph Based Hyper-heuristic
  - Framework
  - Results
  - Observations & thoughts
  - Directions and extensions
- Conclusion

Hyper-heuristics

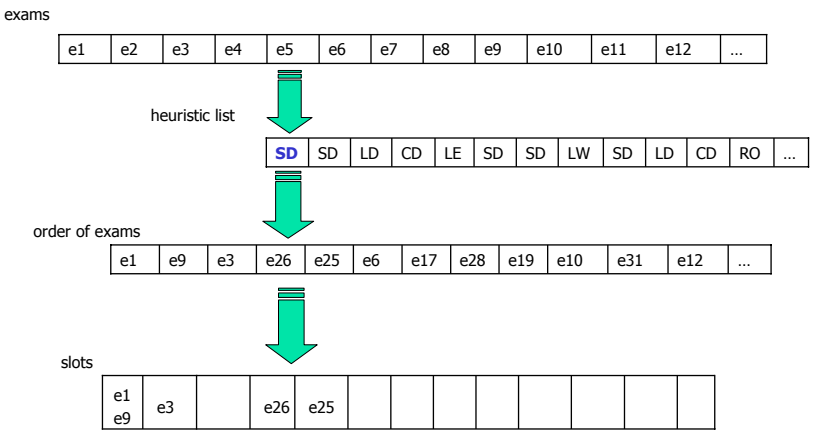
- High level heuristics: GAs, Tabu Search, Choice function, Ant Algorithm, CBR, etc
- Low level heuristics
  - different moving strategies (improve)
  - graph heuristics (constructive)
- Applications: educational timetabling, bin packing, personal scheduling, etc

A Graph-Based Hyper-heuristic -framework

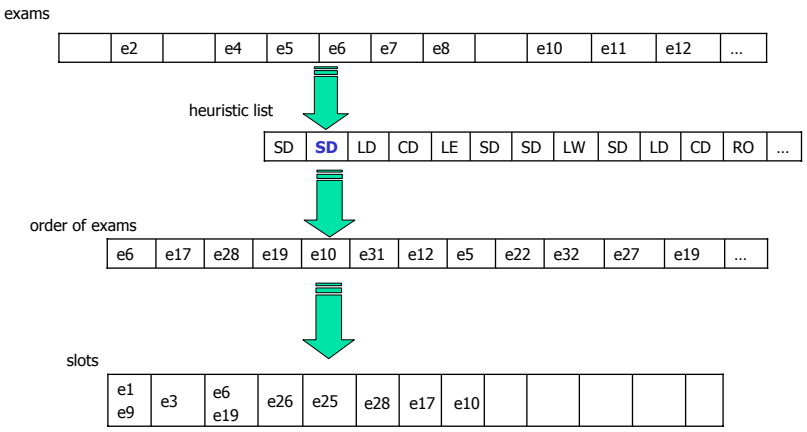


- Two search spaces
  - Search space of high level heuristics: permutations of low level heuristics
  - Solution space of problem: actual solutions

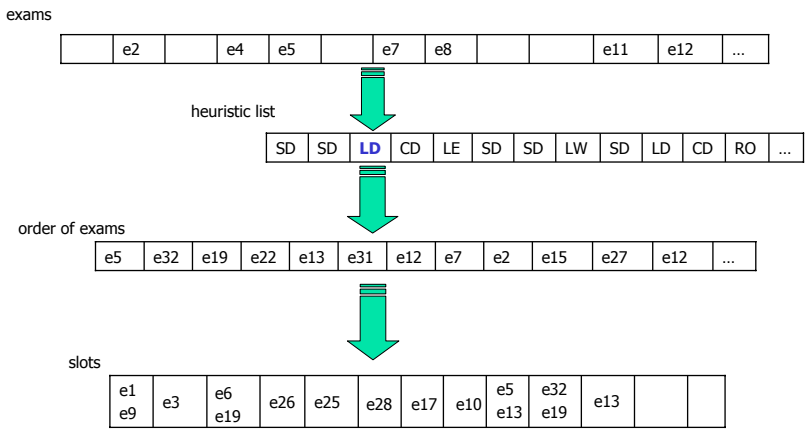
A Graph-Based Hyper-heuristic -framework



A Graph-Based Hyper-heuristic -framework



A Graph-Based Hyper-heuristic
-framework



A Graph-Based Hyper-heuristic
- framework: general high/low level heuristics

- High level heuristics: tabu search, steepest descent, iterated steepest descent
- Low level heuristics (graph heuristics): sequential methods that order events by the difficulties of assigning them into slots
  - Saturation Degree: least available slots
  - Colour Degree: most conflicted with those scheduled
  - Largest Degree: most conflicted with the others
  - Largest Weighted Degree: students involved
  - Largest Enrolment: students enrolled
  - *Random Ordering*

A Graph-Based Hyper-heuristic
- framework: time complexity

- Time complexity
  - Within each step of high level search, a solution needs to be built
  - Feasibility is not known until the whole heuristic list is applied on the problem
  - Search space contains a large number of heuristic lists that generate unfeasible solutions
- Computational time – key issue here

A Graph-Based Hyper-heuristic
- framework: mechanisms

- 3 mechanisms within high level heuristic
  - Failed list: heuristic lists that generated unfeasible solutions

    For example, if h1h1h2h3… is stored, h1h1h2h3h4.., h1h1h2h3h5… will be ignored in the search
  - Each heuristic in heuristic lists assigns 5 (instead of 1) exams at each step

    Best lists found in pattern $h_1h_{1\ldots}h_1h_3h_3\ldots h_3$
  - Initial heuristic list set as SD SD SD …

A Graph-Based Hyper-heuristic
- Results on timetabling problems

- Course timetabling
  - Metaheuristics network: 11 benchmark course timetabling problems
  - The same problem format/structure as the International Competition on Timetabling
  - GHH obtained results comparable with the best results reported (by ant algorithm, tabu search based hyper-heuristic, iterated local search). (with one medium instance the best result)

A Graph-Based Hyper-heuristic
- Results on timetabling problems

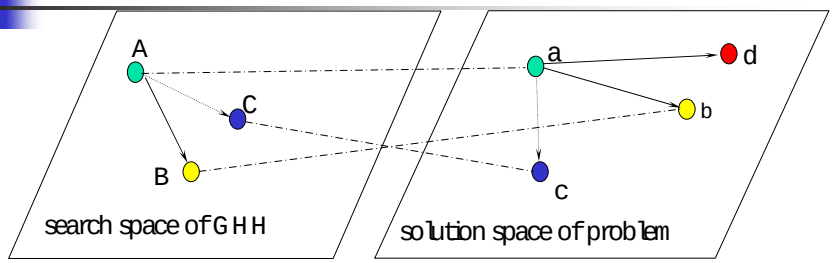
- Exam timetabling
  - Carter, Laporte & Lee (1996): 11 benchmark exam timetabling problems
  - State-of-the-art approaches employing different "fine-tuned" techniques reported different results
  - GHH obtained results within the range of best results reported (with one instance sta83 the best result*)
  - Constructive vs. improvement approaches

*due to some inconsistence on 11 benchmark problems, better result has been found on sta83 with less constraints

A Graph-Based Hyper-heuristic
- observations: why it works well?

- Local search based algorithms
  - Make moves within a limited search areas
  - Easily stuck to local optima
  - Different mechanisms developed
  - Chaotic attractor: a limited portion of search space
- GHH framework
  - Change the way of building the solutions at high level
  - Coverage of the solution space with the same searching effort

A Graph-Based Hyper-heuristic
- observations: why it works well?



With one move

- Local search approaches
- Graph based hyper-heuristics

One bit different

One part different (from the different part of the heuristic list)

A Graph-Based Hyper-heuristic
- observations, low level heuristics

- Experiments: Hyper-heuristic employing different number of low level heuristics
  - SD + LD, SD + CD + LD with/without RO, SD + CD + Lx with/without RO
- Hyper-heuristic employing more number of *intelligent* low level heuristics obtains better results
  - RO does not contribute high performance of GHH
- Hyper-heuristic employing less number of low level heuristics occasionally obtains better results
  - GHH with less number of low level heuristics < GHH with more number of low level heuristic

A Graph-Based Hyper-heuristic
- observations, high level heuristics

- Neighbourhood structures within GHH
  - For each heuristic list, the number of its neighbourhoods is large
  - Only a subset of the neighbourhoods can be evaluated before a move can be made within the high level heuristic search
- Issues of high level search
  - How its performance can be affected?
  - Is it necessary to accept worse move to escape from local optima?
  - Is it worth taking time in areas where no (or very few) better solutions are nearby?

A Graph-Based Hyper-heuristic
- observations, high level heuristics

- Experiments: Hyper-heuristic employing tabu search and steepest descent
  - "Walks" are allowed
  - with the same number of computations
- GHH with steepest descent obtains similar results with that of tabu search based GHH
  - Tabu search: moves up and down
  - Steepest descent: similar solutions, after a number of long "walks", quickly, but then get stuck
- Landscape of high level space
  - Size of neighbourhoods is very large
  - More likely to have "walks" or plateau

A Graph-Based Hyper-heuristic
- observations, high level heuristics

- Experiments: Hyper-heuristic employing iterative steepest descent
- Aim
  - Get the best solutions quickly
  - Restart at very different point
  - More/larger portions of solution space searched
- Results on the same problems instances better than both tabu search and steepest descent based GHH

A Graph-Based Hyper-heuristic
- overall directions of GHH

- What we suggest here

Role of high level heuristics

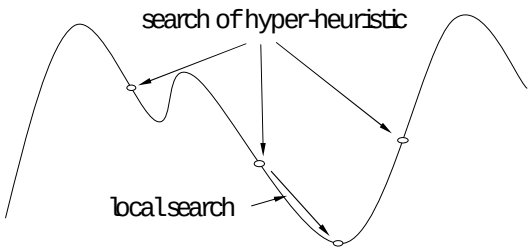
Exploring as large/wide areas of solution space as possible

by searching within high level search space

with a limited number of evaluations

A Graph-Based Hyper-heuristic
- extensions & future work

- Searching is upon the heuristics, not the solutions
- *Hypothesis*: not all the solutions in solution space are potentially reachable by search within high level search space.
  - GHH as a initialisation approach
  - Greedy local search afterwards

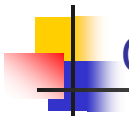


A Graph-Based Hyper-heuristic
- extensions & future work

- Case-based heuristic selection system
  - Extract/record knowledge of heuristic selection during problem solving
  - Learn to select good heuristics for particular situations
  - Suggesting good heuristics in different situations
  - Obtained good results on simulated problems, test on real-world problems

Conclusions

- A simple framework employs widely used general (graph) heuristics
  - Obtained results in the region of the fine-tuned (published) state-of-the-art approaches
  - Little domain knowledge required to easily extend to other timetabling/scheduling problems
- Future directions
  - Exploring as wider/much as possible solution space
  - Generality: more types of scheduling problems



Questions/Comments

- Computational time?
- Hypothesis
- Landscape analysis

References

- E.K. Burke, B. MacCarthy, S. Petrovic, R. Qu. Knowledge Discovery in Hyper-heuristic Using Case-Based Reasoning on Course Timetabling. PATAT'02, Aug, 2002.
- S. Petrovic, R. Qu. Case-Based Reasoning as a Heuristic Selector in a Hyper-Heuristic for Course Timetabling Problems. KES'02, Sep, 2002.
- E.K. Burke, S. Petrovic, R. Qu. Case-Based Heuristic Selection for Examination Timetabling. SEAL'02, Nov, 2002.
- E. K. Burke, A. J. Eckersley, B. McCollum, S. Petrovic and R. Qu Analysing Similarity in Exam Timetabling. 5th international conference on the Practice and Theory of Automated Timetabling, Pittsburgh, August 2004.
- E. K. Burke, A. Meisels, S. Petrovic and R. Qu A Graph-based Hyper-Heuristic for Exam Timetabling Problems. submitted to EJOR, 2004.
- E. Burke, S. Petrovic, R. Qu, Case Based Heuristic Selection for Timetabling Problems. accepted by Journal of Scheduling, 2005
- E. Burke, M. Dror, S. Petrovic, R. Qu, Hybrid Graph Heuristics within a Hyper-heuristic Approach to Exam Timetabling Problems. The 9th informs Computing Society Conference. Kluwer Academic Publishers. Jan 2005
- E. K. Burke and R. Qu Analysing High Level Heuristics within a Graph-Based Hyper Heuristic for Exam Timetabling Problems. submitted to EvoCOP.