A Graph Based Hyper-heuristic Framework
- research issues and extensions

Rong Qu

Automated Scheduling, Optimisation and Planning Group
School of Computer Science
University of Nottingham

The GHH Framework

- High level search
  - Any meta-heuristics

- Low level heuristics*: order events by how *difficult* to schedule them
  - Saturation Degree: least available slots
  - Colour Degree: most conflicted with those scheduled
  - Largest Degree: most conflicted with the others
  - Largest Weighted Degree: LD + students involved
  - Largest Enrolment: students enrolled

* Educational timetabling is used in this research as problem domain

The GHH Framework

events

e1	e2	e3	e4	e5	e6	e7	e8	e9	e10	e11	e12	...

heuristic list

SD	SD	LD	CD	LE	SD	SD	LW	SD	LD	CD	RO	...

order of events

e1	e9	e3	e26	e25	e6	e17	e28	e19	e10	e31	e12	...

slots

e1 e9	e3		e26	e25								

The GHH Framework

events

	e2		e4	e5	e6	e7	e8		e10	e11	e12	…

Heuristic list

SD	**SD**	LD	CD	LE	SD	SD	LW	SD	LD	CD	RO	…

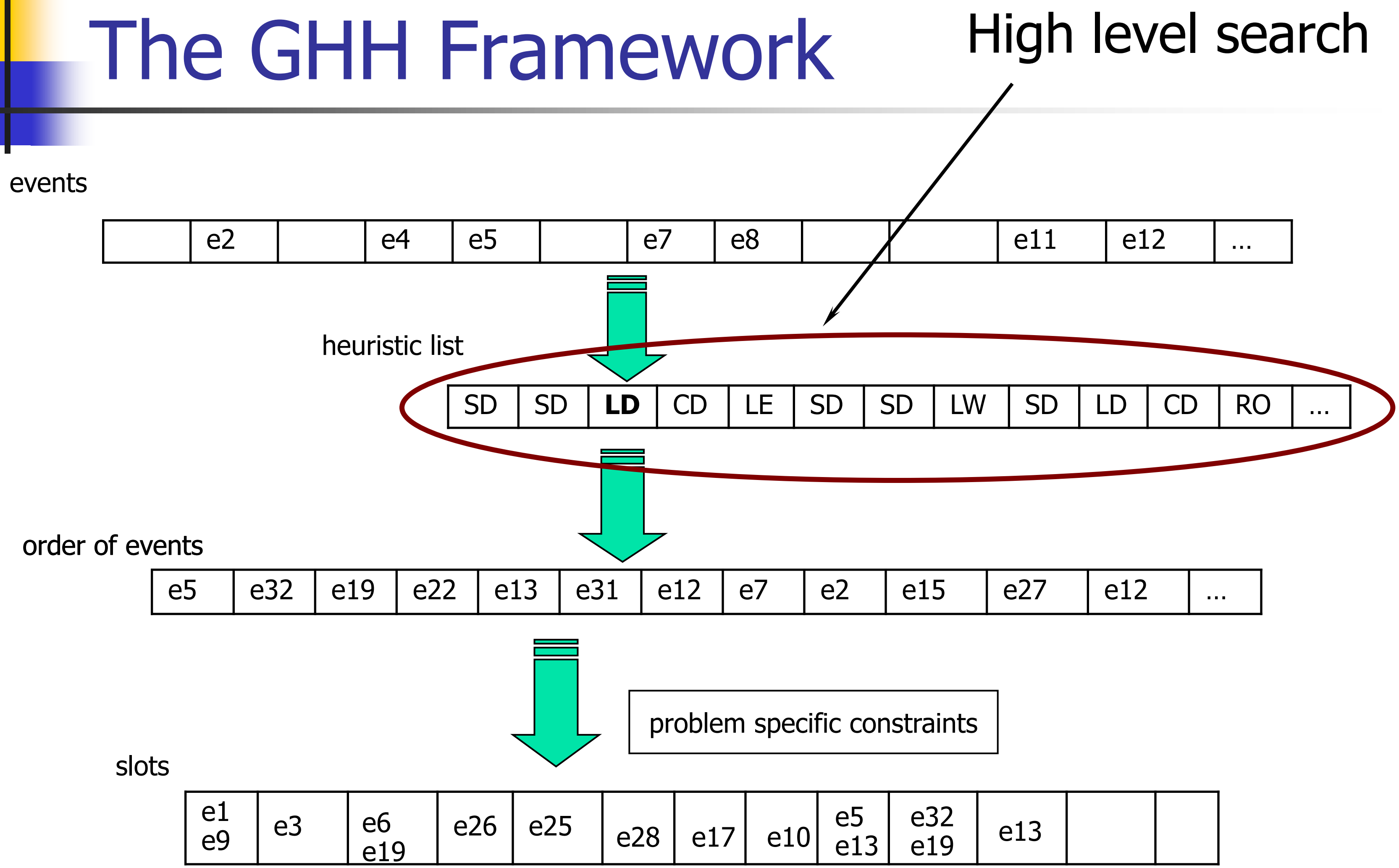
order of events

e6	e17	e28	e19	e10	e31	e12	e5	e22	e32	e27	e19	…

slots

e1 e9	e3	e6 e19	e26	e25	e28	e17	e10					

The GHH Framework



Educational Timetabling Problems

- Carter, Laporte & Lee (1996): 11 real world exam timetabling problems
  - Hard constraint: conflicts between exams
  - Objective function: *min* time slots (graph colouring)
  - Soft constraints: spread out exams over time slots
  - Objective function: $C(t) = \left(\sum_{s=0}^{4} w_s \times N_s \right) / S$
- Meta-heuristic Network: 11 derived course timetabling problems
  - Hard constraints: conflicts between exams, room feature, room capacity
  - Soft constraints: minimise only one class a day, class in the last slot of a day, more than two classes in a row
  - Objective function: *min* sum of the costs for soft constraints

	car91	car92	ear83	hec92	kfu93	lse91	sta83	tre92	ute92	uta93	york83
GHH	5.36	4.53	37.92	12.25	15.2	11.33	**158.19**	8.92	28.01	3.88	41.37
Abdullah et al	5.21	4.36	34.87	10.28	**13.46**	10.24	159.28	**8.13**	**24.21**	3.63	**36.11**
Asmuni et al	5.20	4.52	37.02	11.78	15.81	12.09	160.42	8.67	27.78	3.57	40.66
Burke & Newall	4.6	**4.0**	37.05	11.54	13.9	10.82	168.73	8.35	25.83	**3.2**	36.8
Burke, Bykov et al	**4.2**	4.8	35.4	10.8	13.7	10.4	159.1	8.3	25.7	3.4	36.7
Caramia et al	6.6	6.0	**29.3**	**9.2**	13.8	**9.6**	**158.2**	9.4	24.4	3.5	36.2
Carter et al	7.1	6.2	36.4	10.8	14.0	10.5	161.5	9.6	25.8	3.5	41.7
Di Gapero & Schaerf	6.2	5.2	45.7	12.4	18.0	15.5	160.8	10.0	29.0	4.2	41.0
Merlot et al	5.1	4.3	35.1	10.6	13.5	10.5	157.3	8.4	25.1	3.5	37.4

	GHH	**Burke et al 2003**	**Socha et al. 2002**	**Socha et al 2002**
Small1	6	**1**	8	**1**
Small2	7	**2**	11	3
Small3	3	**0**	8	1
Small4	3	**1**	7	**1**
Small5	4	**0**	5	**0**
Medium1	372	**146**	199	195
Medium2	419	**173**	202.5	184
Medium3	359	267	77.5% Inf	**248**
Medium4	348	169	177.5	**164.5**
Medium5	**171**	303	100% Inf	219.5
Large	1068	80% Inf 1166	100% Inf	**851.5**

Research Questions/Issues

- Which high/low level search heuristics?
- Two search spaces
- Search in two spaces
- Other extensions

Which high level search method?

- High level search methods
  - Iterated Local Search
  - Tabu Search
  - Steepest Descent
  - Variable Neighbourhood Search

  - Objective function
    - heuristic lists → penalties (costs of timetables constructed)
  - "Walks" are allowed

Which high level search method?

- High level search methods
  - Similar performance within the same GHH framework (same total number of evaluations, same initials, etc)
  - ILS and VNS are slightly better
  - Results are comparable to state-of-the-art approaches on both course and exam benchmark problems

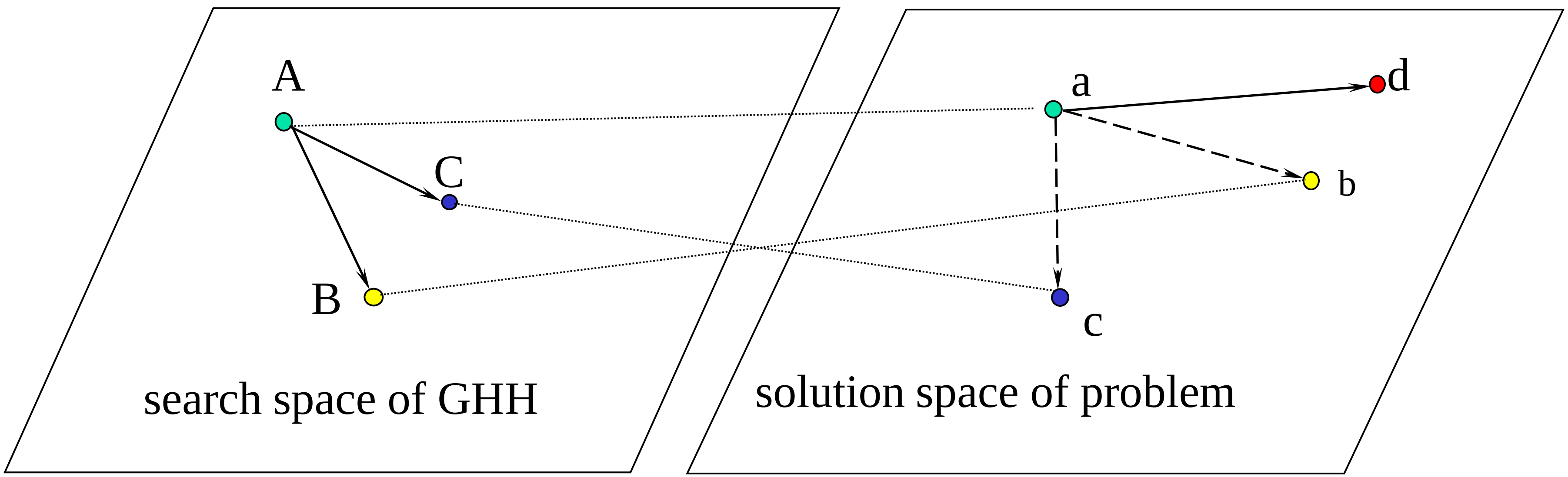
Which low level heuristics?

- Within GHH
  - Different subsets of graph heuristics (SD+LD, SD+LWD, SD+LE, SD+LWD+CD, etc)

  - With a limited computational time
    - SD + LWD performed the best
  - With more graph heuristics
    - Longer time given, the better the results
    - h^l (l: length of the sequence, h: number of graph heuristics
  - Random ordering also contributes the performance

Two Search Spaces

Heuristic space | Solution space



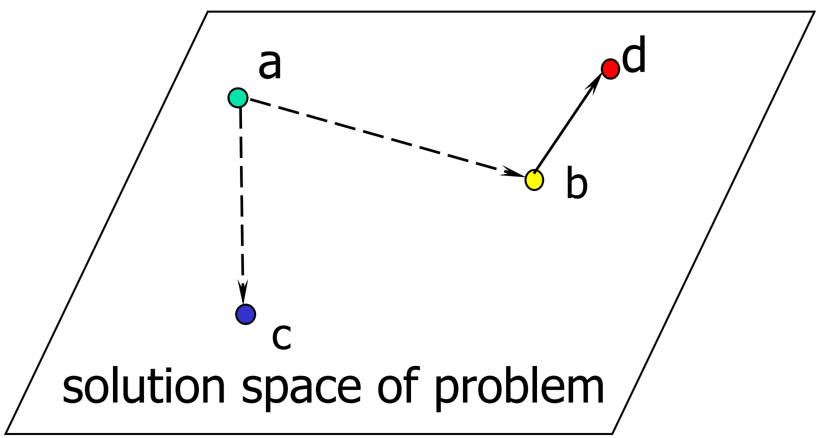
GHH: search is upon heuristics, not solutions
– not all the solutions in solution space are reachable?

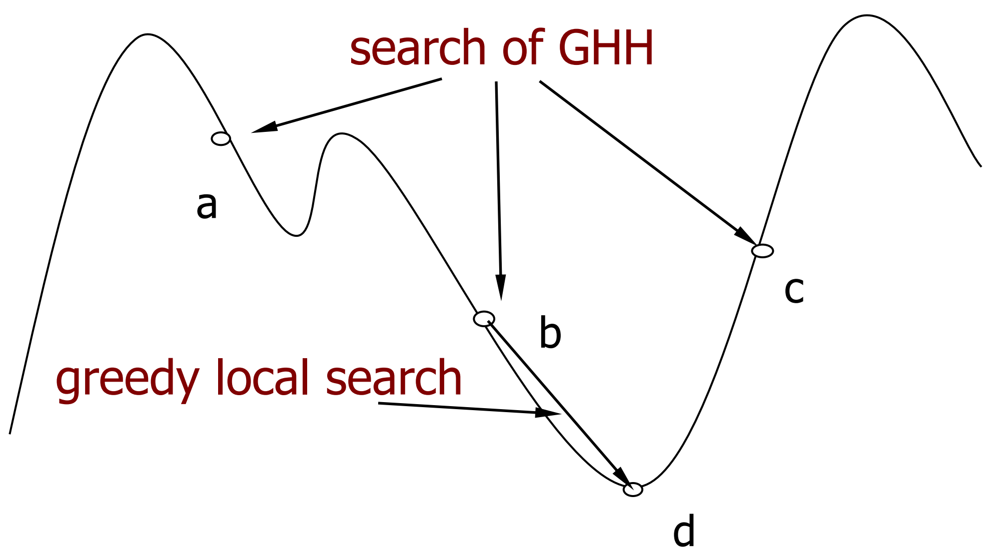
Two Search Spaces

	Heuristic space	Solution space
Representation		
Size (Upper Bound)		
Neighborhood Operator		
Objective Function		

Search in Two Spaces

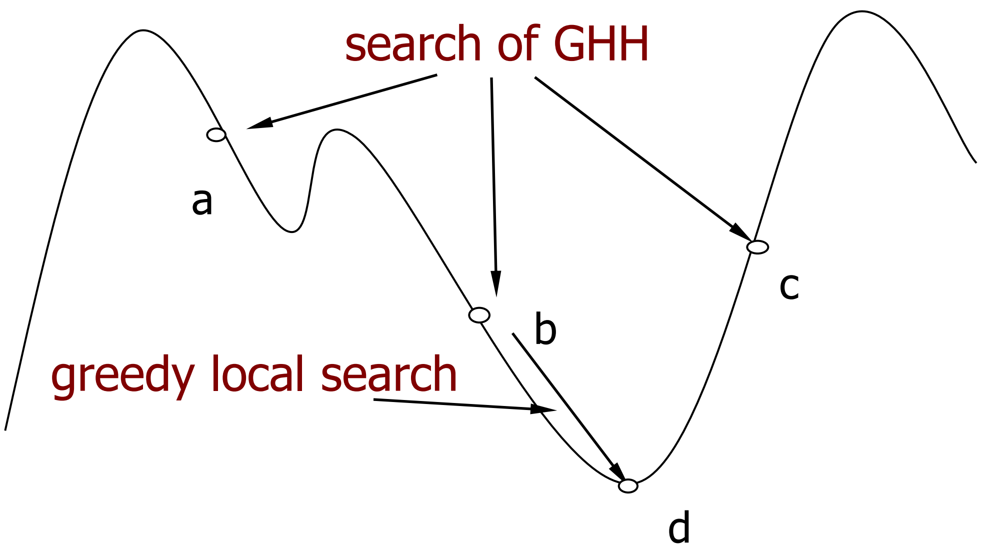
- Hybridisation of GHH with greedy search
  - High level search in heuristic space: a, b, c, …
  - Greedy search in solution space: b -> d, …
    - Coverage of the solution space





Search in Two Spaces

- Hybridisation of GHH with greedy search
  - Results greatly improved!
  - Hybrid GHH vs. Memetic Algorithms
    - Diversification vs. intensification



Extensions

- Heuristic hybridisations in GHH

  Question: best solutions → better/best ways of heuristic hybridisations?

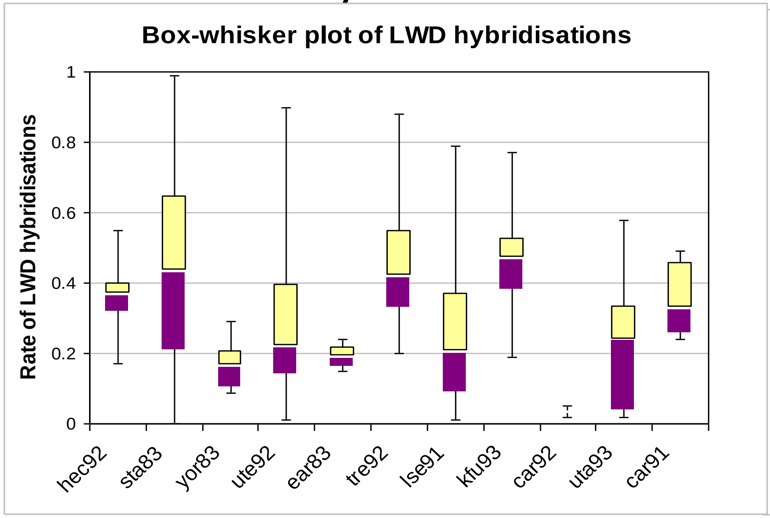
  I - Random GHH (SD+LWD, SD+LE, SD+LD)

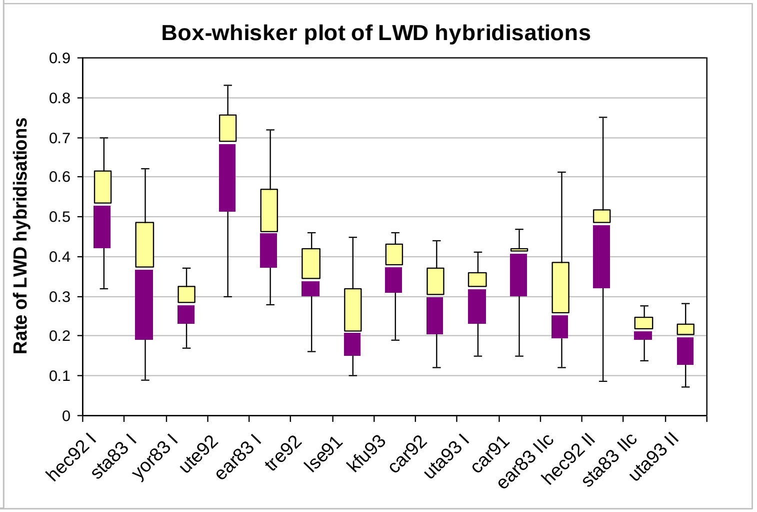
  A large collection of different heuristic sequences

  II - Analyze the best 5% heuristic sequences

  Rates of hybridisation at different positions of heuristic sequences

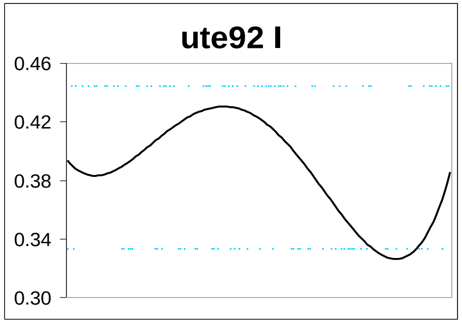
  Trends of hybridizations in the best sequences

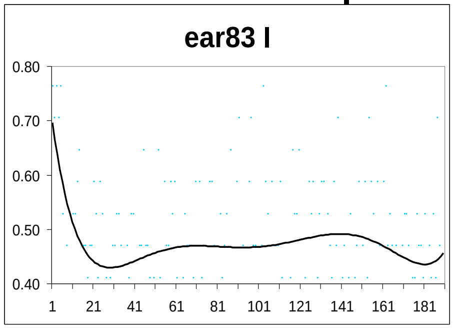


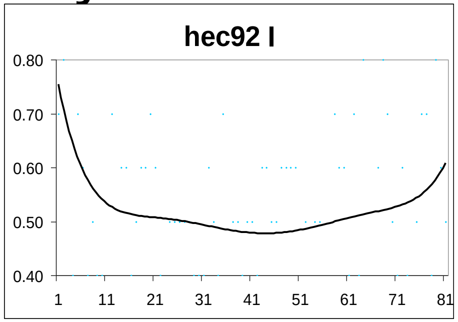


Extensions

- Heuristic hybridisations in GHH
  - Hybridising SD with LWD obtained better results compared with LE or LD
  - In the best 5% sequences
    - Higher percentage at early stage
    - High level of vibrancy at early stage
- Adaptive heuristic hybridization approach
  - GHH search focus on early stage of sequences
  - Rate of LWD hybridisation adaptively adjusted







Future Work

- Landscape of high level heuristic space
  - More likely to have plateau (neutral)
- Synchronise the search in two search spaces
  - Difficulty of landscape analysis in solution space
- Other recent extensions in the literature
  - Hierarchical hybridisation of graph heuristics
  - Tie breaking and timeslots ordering within GHH
  - Hybridising improvement based low level heuristics within GHH

Summary

- Search in two search spaces
  - Diversification by GHH in the heuristic space
  - Intensification by local search in the solution space

- Role of high level search methods

  To explore diversified solutions in the solution space by searching in the high level heuristic space

Some References

- E. Burke, A. Meisels, S. Petrovic and R. Qu A Graph-based Hyper-Heuristic for Exam Timetabling Problems. *EJOR* 176: 177-192, 2007. Top 1% cited in CS by ISI Essential Science Indicators
- R. Qu and E. K. Burke. Hybridisations within a Graph Based Hyper-heuristic Framework for University Timetabling Problems. *JORS*, 2008.
- N.R. Sabar, M. Ayob, G. Kendall, R. Qu, "A Graph Coloring Constructive Hyper-Heuristic for Examination Timetabling Problems" *Applied Intelligence*, 2011.
- E.K. Burke, N. Pham, R. Qu, J. Yellen. "Linear Combinations of Heuristics for Examination Timetabling". accepted by *Annals of OR*
- R. Qu, E. Burke and B. McCollum. Adaptive Automated Construction of Hybrid Heuristics for Exam Timetabling and Graph Colouring Problems. *EJOR*, 198(2): 392-404, 2009. Top 10% cited in CS by ISI Essential Science Indicators
- E. Burke, S. Petrovic, R. Qu, Case Based Heuristic Selection for Timetabling Problems. accepted by *Journal of Scheduling*, 9: 115-132, 2006. Top 1% cited in CS by ISI Essential Science Indicators
- http://www.cs.nott.ac.uk/~rxq/publications.htm
- http://www.cs.nott.ac.uk/~gxo/hhbibliography.html