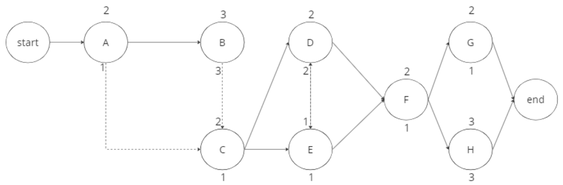
COMBINING SAT SOLVERS WITH HEURISTIC IDEAS FOR SOLVING RCPSP WITH LOGICAL CONSTRAINTS

AN EXPLORATION OF VARIABLE ORDERING HEURISTICS IMPACT ON SOLVING RCPSP-LOG

Author: Iarina Maria Tudor I.M.Tudor@student.tudelft.nl

1.RCPSP-LOG

- Resource-Constrained Project Scheduling Problem (RCPSP): a set of activities with durations and renewable resources, each having a constant availability per period. Each activity requires an amount of resources. The goal is to minimize the makespan [1].
- The extension with logical constraints, RCPSP-log, enables the modeling of complex relationships and dependencies between activities [2].
- The problems are known as NP-hard [8]
- AND constraint: job scheduled if all of its predecessors have finished.
- OR constraint: only one predecessor needs to be finished before being able to schedule the successor.
- BI constraint: two activities cannot be executed in parallel.



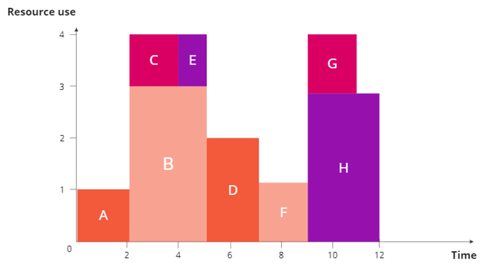


Figure 1. An example of project network and the optimal schedule

2.MOTIVATION

- Project scheduling is a complex issue that affects many sectors [4].
- Past work in the field is problem specific algorithms [3].
- Satisfiability (SAT) solving is a powerful general technique in ongoing development [9], but it has limitations compared to the state-of-the-art, which incorporates problem-specific knowledge.
- Variable selection heuristics can incorporate problem information, aim to reduce the solving time, and provide insight on partial solutions.

3.RESEARCH QUESTION

- How can the integration of heuristic ideas into SAT solvers enhance the solution process of RCPSP with logical constraints such as OR, AND and BI constraints?

4.METHOD

1. A weighted conjunctive normal form (CNF) encoding is generated with PYSAT [6], which represents the input of the MaxSAT solver.
2. MaxSAT solver finds an assignment that minimizes the makespan.
3. The variable selection guides the solver into finding solutions using some predefined order rules, where heuristics come in handy and prune the search space.

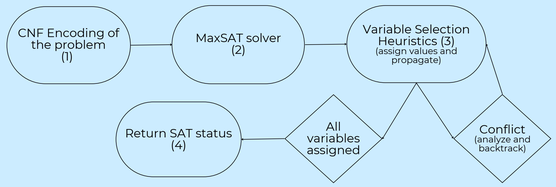


Figure 2. Method flow graph for SAT solving.

5.HEURISTIC IDEAS

- The two heuristics are based on:
  - the intuition of minimizing the makespan by trying to schedule each activity closer to the start of the project.
  - variable activity and conflict analysis, known as good practice in SAT solving.
- The first heuristic is the **greedy schedule each activity as early as possible (EST).** Each activity has a weight based on start time, fixing the variable selection order.
- The second approach combines the **EST with Variable State Independent Decaying Sum (VSIDS) [5]**, used by default by the solver.
  - the EST weights guide the start of the search and the conflict analysis takes over as the algorithm advances.
- Considering the contribution of VSIDS in SAT solving, the EST + VSIDS approach could incorporate more problem-specific knowledge and keep the advantages of the best-practice method [5].

6.RESULTS

- Results have been found by comparing the performance of the 2 heuristics against the pumpkin SAT solver with VSIDS as default variable selection method.
- The experiments were done on PSPLIB [7] datasets, transformed for RCPSP-log following the method developed by Coelho and Vanhoucke [3].
- The highlighted entries in the table show the most significant findings in comparison to the baseline.
- Evaluations considered number of solutions, average time to solve, number of decisions and average best makespan over time, within a time limit of 15 seconds.

			OR				BI			
	k1	-	1	1	1	1	1	1	1	1
	k2	-	1	2	5	10	1	2	5	10
	Percent Log		100	50	20	10	100	50	20	10
j30										
Benchmark		196|32|0	184|44|0	192|36|0	195|33|0	195|33|0	162|66|0	179|49|0	190|38|0	191|37|0
EST + VSIDS		200|28|0	187|41|0	192|36|0	193|35|0	196|32|0	162|66|0	181|47|0	192|36|0	195|33|0
EST		187|10|31	173|9|46	178|13|37	180|9|39	184|8|36	148|15|65	166|14|48	181|11|36	184|13|31
j60										
Benchmark		45|25|0	39|31|0	36|34|0	40|30|0	43|27|0	18|52|0	25|45|0	38|32|0	42|28|0
EST + VSIDS		45|25|0	34|36|0	35|35|0	38|32|0	42|28|0	9|61|0	22|48|0	37|33|0	46|24|0
EST		48|5|17	41|11|17	45|7|18	48|7|15	46|9|15	33|12|25	41|9|20	42|8|20	49|7|14

Table 1. Solution results for the single-mode PSPLIB 30 and 60 job instances, reported as optimal | satisfiable | unknown solutions for each percentage of logical constraints controlled by k1 and k2, calculated as percent log.

7.CONCLUSIONS & LIMITATIONS

- Results provide insight for integrating variable selection heuristics, with the potential for more investigation into problem-specific ideas.
- The proposed approaches contribute to reducing the makespan and finding more optimal solutions for 60 jobs instances, specifically for BI constraints.
- There are limitations, such as struggling to find solutions for all instances when using the EST method.
- The approaches consider very limited problem specific information.

8. FUTURE WORK

- Critical path analysis and logical constraints information for more problem specific knowledge.
- Extend the datasets to more instances and increase the number of activities (90 and 120 jobs).
- Consider testing within different time limits.

Related literature

[1] C Artigues and R Leus. The resource-constrained project scheduling problem. *Handbook on Project Management and Scheduling*, 1:243–286, 2013.
[2] X Chen, K Yang, Y Zhang, and Y Guan. A comprehensive review on resource-constrained project scheduling problems. *Mathematical Problems in Engineering*, 2018:4801317, 2018.
[3] J Coelho and M Vanhoucke. An approach using sat solvers for the rcpsp with logical constraints. *European Journal of Operational Research*, 252(2):431–441, 2016.
[4] E Demeulemeester, R Kolisch, and A Salo. Project management and scheduling. *Flexible Services and Manufacturing Journal*, 25(1):1–5, 2013.
[5] J. Liang, V. Ganesh, E. Zulkoski, A. Zaman, and K. Czarnecki. Understanding vsids branching heuristics in conflict-driven clause-learning sat solvers, 2015.
[6] A Ignatiev, A Morgado, and J Marques-Silva. PySAT: A Python toolkit for prototyping with SAT oracles. *SAT*, pages 428–437, 2018.
[7] R Kolisch and A Sprecher. Psplib--a project scheduling problem library: Or software--orsep operations research software exchange program. *European Journal of Operational Research*, 96(1):205–216, 1997.
[8] N Nilsson and B Rolfsson. Complexity results for resource-constrained project scheduling problems with logical constraints. *Journal of Scheduling*, 14(4):317–332, 2011.
[9] M Järvisalo, D Le Berre, O Roussel, and L Simon. The international sat solver competitions. *AIMag*, 33(1):89–92, 2012

TU Delft
Responsible Professor: Emir Demirovic Supervisors: Konstantin Sidorov, Maarten Flippo EEMCS, Delft University of Technology, The Netherlands