Flexible resource constrained scheduling problem (FRCPSP)

Jack Trainer
[1]STOR-i, Lancaster University

What is the FRCPSP?

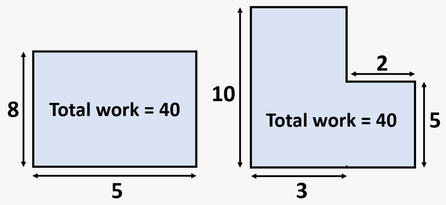

Figure 1: Visual representations of activities in a RCPSP (left) & a FRCPSP (right).

- A Project scheduling problem (PSP) is an optimization problem where the goal is to minimize the total duration (makespan) of a project consisting of a set of activities with resource & time requirements.
- When a PSP is Resource constrained (RCPSP), this means that there are limits imposed upon the number of resources available at a given time.
- In a RCPSP, the resource allocation for each activity is usually fixed. In a FRCPSP, the activities are given a work content which can be fulfilled flexibly.

Precedence constraints (PCs)

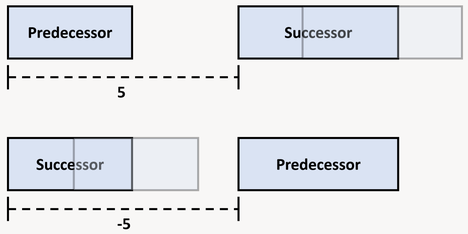

Figure 2: Visual representation of a positive PC (top) & a negative PC (bottom).

- A positive PC is the minimum lag between the start of a predecessor & the start of a successor.
- A negative PC is the maximum lead between the start of a predecessor & the start of a successor.

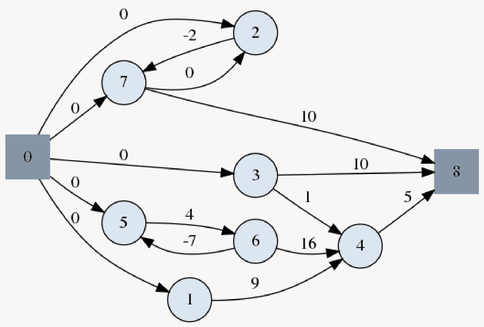

Figure 3: A network showing the PCs between activities in a project.

The greedy serial generation scheme (SGS)

- The SGS is the heart of the heuristic algorithm we use to solve the FRCPSP. We input lists of activities known as activity list representations (ALRs) & the SGS attempts to build a schedule of activities step-by-step.
- A greedy SGS schedules each activity at its earliest possible precedence- & resource- feasible start time & allocates the maximum number of available resources to the activity.

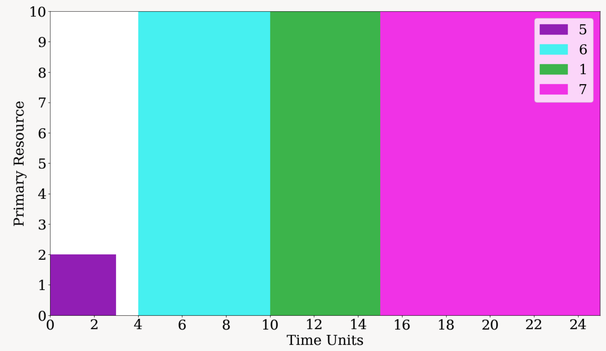

Figure 4: Example of a failed schedule generated by the greedy SGS.

Flaws of the greedy SGS

- In some projects, time constraints will require that certain activities need to be started whilst their predecessor is ongoing. Unless both activities can use their maximum resource the greedy SGS will not allow this.
- Always schedules activities as soon as possible leading to activities being stuck in gaps between other activities and not able to meet constraints.

The new SGS (greedy SGS with flaws corrected)

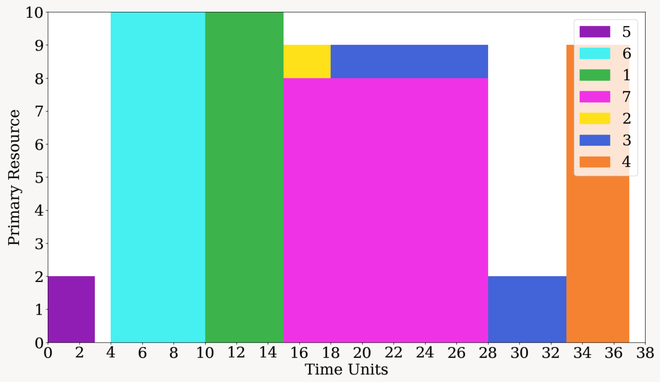

Figure 5: Example of a successful schedule generated by the new SGS.

Comparison of the greedy SGS with the new SGS

Old SGS	New SGS	Old SGS	New SGS	Old SGS	New SGS
NA	35	35	35	57	57
37	37	43	43	71	71
26	25	37	38	NA	NA
32	32	43	38	26	24
28	27	58	54	56	55
NA	38	23	23	42	45
36	36	46	40	51	55
31	35	33	33	43	48
34	29	86	86	57	57
NA	NA	29	29	27	27
22	21	28	28	NA	NA
28	33	40	40	47	47
31	31	34	33	33	33
28	33	40	40	40	40
25	25	NA	26	29	30
21	21	28	28	35	28
30	35	24	24	34	31
35	35	NA	NA	27	26
24	24	32	32	54	50
66	66	27	28	34	34
51	51	36	36	NA	NA
16	14	16	16	28	28
33	31	22	22	34	34
36	36	52	57	43	43
14	14	32	30	53	53
32	29	43	37	51	51
45	45	32	32	33	36
39	41	19	20	30	30
NA	49	33	30	14	14
33	33	19	22	23	23

Figure 6: Comparison between the best makespans achieved by the old SGS and the new SGS acting on 10 random ALRs for 90 different projects each consisting of around 10 activities each. NA indicates a failure to produce a schedule (some projects have no feasible solution).

- The average makespan of the old SGS is 35.74 and for the new SGS it is 35.65.

Conclusion & Future aims

- For certain projects (The three projects in the third column of the table), the SGS will only produce a successful schedule for a very specific ALR which is often not tested. The algorithm needs to be improved so it can recognise these situations and find the correct ALR quickly.
- Testing larger projects and data-sets to demonstrate that the new algorithm is watertight.
- Testing with real-life data e.g. from Sellafield Ltd.

References

Kolisch, R. & Hartmann, S. (1999). *Project scheduling*, 147–178.
Heuristic algorithms for the resource-constrained project scheduling problem: Classification and computational analysis.

Naber, A. & Kolisch, R. (2014). *European Journal of Operational Research*, 239335–348.
MIP models for resource-constrained project scheduling with flexible resource profiles.

j.trainer@lancaster.ac.uk