Solution Approaches for the Resource Constrained Project Scheduling Problem

By Dimitris Paraskevopoulos, Doctoral Candidate

Project scheduling involves the effective and efficient use of time to facilitate the execution of a project. In particular, it implicates the sequencing and scheduling of a set of operations, satisfying particular constraints (precedence, resource limited capacity, time windows etc) and minimizing a single or various objectives. There are various problems studied in the literature that share these characteristics (single machine problems, parallel machine problems, flow shops, job shops, time tabling problems, etc.). These problems can be modelled as the resource-constrained project scheduling problem (RCPSP) studied in this article.

The RCPSP can be given as follows. A single project consists of a set J = {0, 1, . . . , n, n+1} of activities which have to be processed. The activities are interrelated by two kinds of constraints. First, precedence constraints force activity j not to be started before all its immediate predecessor activities, comprised in the set Pj, have been finished. Second, performing the activities requires the use of resources with limited capacities. There are k resource types, given by the set K = {1, . . . , k}, each one with capacity Rk. While being processed, activity j requires rj,k units of resource type k during every period of its non–preemptable duration pj. The term preemptable duration means that the execution of an activity cannot be interrupted at one period and be continued at another time period. The objective of the RCPSP is to find precedence and resource feasible completion times for all activities so that the completion time of the project is minimized.

Let us consider the following RCPSP example with a single resource. Assume a project with 10 activities as illustrated in Figure 1. More specifically, each activity is represented by a node with a predefined time duration and resource requirements, as shown at the top and at the bottom of each node respectively. The arcs connecting particular pairs of nodes represent the precedence constraints while there is one resource available with capacity 8 units. The optimal solution for this problem instance is given in Figure 2. As one observes the capacity of the resource is not violated while the project completion time is 15 time units.

Figure 1 RCPSP network graph

Figure 2 Optimal solution

In practice, project managers might be perfectly willing to settle for reasonable project schedules that are obtained within short computation times. This requirement can only be obtained by applying good heuristic procedures, since RCPSP belongs to NP-hard class of problems and large (real life) scale instances cannot be solved by exact methods. Heuristic methods have been widely used in the last decades for solving a variety of combinatorial optimization problems. The most popular of them used to solve the RCPSP are the Tabu Search [2], the Simulated Annealing [4], the Genetic Algorithms [3] and the Evolutionary Strategies [1]. The first two, called trajectory methods, consider one single solution of the problem and perform operations to improve it. On the contrary, Genetic algorithms and Evolutionary strategies are based on a population of solutions.

The trend of the last years is to hybridize trajectory methods with evolutionary strategies. The main idea behind this hybridization is to combine strengths and eliminate weaknesses of each method into an integrated hybrid framework that will be able to achieve high standards of performance in terms of solution quality produced and run time consumption. Such hybrid methodologies for solving the RCPSP have been provided by Valls et al (2008), Debels et al (2006), and Tseng and Chen (2006).

References

[1] T. Back , U. Hammel and H.P. Schwefel, (1997). "Evolutionary Computation: Comments not the history and current state", *IEEE Transactions on Evolutionary Computation*. 1, pp 3-17.

[2] D. Debels, B. Reyck, R. Leus, M. Vanhoucke, (2006) "A hybrid scatter search/electromagnetism meta heuristic for project scheduling", *European Journal of Operational Research* 169, pp. 638–653.

[3] F. Glover, (1986). "Future Paths for Integer Programming and Links to Artificial Intelligence". *Computers & Operations Resources*, 13, pp. 533-549.

[4] J.H. Holland, (1992) Genetic algorithms. *Science Am.*, New York.

[5] S. Kirkpatrick, C.D. Gelatt and M. P. Vecchi (1983). "Optimization by Simulated Annealing". *Science*, vol. 220, pp. 671-680.

[6] Tseng L.Y. and S.C. Chen, (2006) "A hybrid metaheuristic for the resource-constrained project scheduling problem", *European Journal of Operational Research* 175, pp. 707–721.

[7] V. Valls, F. Ballestın, S. Quintanilla, (2008) "A hybrid genetic algorithm for the resource-constrained project scheduling problem" *European Journal of Operational Research* 185, pp. 495–508.