A Reactive Approach for the Multi-Skill Project Scheduling Problem

Odile Bellenguez-Morineau
IRCCyN/ Ecole des Mines de Nantes
4 rue Alfred Kastler - La Chantrerie
BP 20722
44307 Nantes Cedex 3
FRANCE
e-mail: odile.morineau@emn.fr

Abstract

This paper deals with the Multi-Skill Project Scheduling Problem (MSPSP) in case of disruptions. First, the problem is defined, then a way to build a first schedule is introduced and finally the reactive approach used to repare the schedule is presented.

Keywords: Multi-Skill Project Scheduling Problem, Reactive Approach, Parallel Schedule Scheme.

1 The Multi-Skill Project Scheduling Problem

This problem consists in scheduling a project in order to minimize the makespan. Activities $A_i, i \in \{1..n\}$, of the project are linked by precedence relations. The resources correspond to the persons $P_m, m \in \{1..M\}$, involved in the project. Each person masters some of the given skills $S_k, k \in \{1..K\}$. $MS_{m,k} = 1$ if P_m masters S_k, 0 otherwise. Those persons may also have some unavailability periods. Each activity A_i has specific requirements $b_{i,k}, i \in \{1..n\}, k \in \{1..K\}$, that correspond to a fixed number of required persons with skill S_k. So, a schedule corresponds to a starting time and a subset of employees for each activity. This subset must satisfy the skill requirement. Moreover, an employee must be available during the whole processing time and master the skill corresponding to the unique requirement he is assigned to.

2 Maximum Parallel Schedule Scheme

Different methods exist to solve MSPSP without perturbation (see [1]). We focus on the Parallel Schedule Scheme used in [2] (inspired from [3]).

At each step, t is the smallest time point where at least one activity can be scheduled according to resource and precedence constraints. At time t, a set EP is composed of the activities that may be scheduled according to precedence constraints. EP is sorted according to a priority rule. Then, we have to know if all the set EP can be scheduled at time t without violating the resource contraints. The test is based on a min-cost max-flow formulation. A graph Gp is built: a first layer is composed of a node for each activity of EP, a second layer with one node for each skill required by these activities, and a third layer contains a node for each available employee.

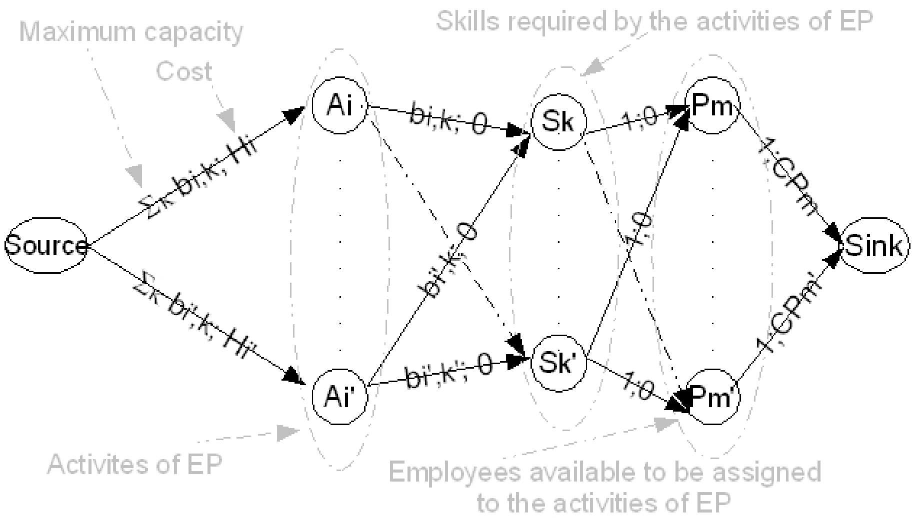

Figure 1: Graph G_p

For all arcs, there exists a maximum capacity and an associated cost. Notice that H_i is an index equal to the rank of the activity A_i in the sorted set EP multipled by a constant strictly greater than 1. Moreover, CP_m is a heuristic indicator, computed according to the total time each skill mastered by P_m is used along the project. This indicator is a ratio smaller than 1. Complete description of this indicator can be found in [2].

If the min-cost max-flow computed in this graph is equal to the sum of all the requirements of all the activities involved, they can be scheduled at time t. Moreover, the final flow gives a feasible way to assign employees to the requirements. However if the maximum flow computed is strictly smaller than the sum of the requirements, then it is not possible to schedule all those activities at time t. In this case, the first (according to the priority rule) activity not completly satisfied is removed from EP and the maximum flow is computed again, until a maximum set to schedule at time t is defined. Then t is increased. Procedure ends when all activities are scheduled.

3 Reactive approach

As explained in [4], once a project is scheduled, some unexpected events may occur such as additional unavailability periods of an employee or increase of some activities duration. Thus, we propose a way to repare the schedule in order to limit the increasing of the project duration without changing too many assignments.

So, at each time t a perturbation obliges to modify the project scheduling because precedence or/and resource constraints are not yet respected, a reschedule is done. The approach is based on the Parallel Schedule Scheme (see. part 2). At time t, a graph Gp is built as previously to reschedule as many activities as possible. But to avoid too many assignment changes that may lead to employees' discontent, once the graph is built, an **admissible flow** is added to this graph. Any previous assignment that does not lead to a conflict according to the reschedule is kept. So, a one-unit flow is added on the path corresponding to this assignment in the graph. By searching a min-cost max-flow in this graph, we determine the maximum subset of activities to schedule at time t. Moreover, the way that the maximum flow is built takes into account the different paths given in the admissible flow, and changes this flow only if this is the only way to improve the solution. That means that the assignments are kept as much as possible.

If some activities have to be delayed because they cannot be scheduled at time t anymore, their subset of employees assigned is also kept in the admissible flow proposed at the next time point. So, the starting time may be changed, but the set of assigned persons is kept as much as possible. t is increased until complete reschedule is done.

Using the Parallel Schedule Scheme helps limiting the increase of the project duration in regards of what is done if activites are delayed to keep every assignment. Moreover, assignments corresponding to the admissible flow are changed only if it is the only way to schedule the activity at the earliest time. Thus, activities are scheduled as early as possible, but the subset of employees assigned to each activity is as close to the original one as possible.

4 Conclusion

This paper deals with the MSPSP in case of disruptions. A way to repare a schedule when a disruption makes it unrealizable is introduced. This procedure is based on the Parallel Schedule Scheme. The aim is to have a better project duration than if the project is only delayed according to the disruption and to have less planning modifications for employees than if a total reschedule is done. Preliminary results seem to be promising. A complete study will be presented in order to establish if this aim is reached.

References

[1] BELLENGUEZ-MORINEAU O., *Methods to solve multi-skill project scheduling problem*, PhD thesis abstract, 4OR : A quarterly Journal of Operations Research, Online first ISSN n∘ 1614-2411 (2007).

[2] BELLENGUEZ-MORINEAU O. AND NÉRON E., *A Tabu Search for Multi-Skill Project Scheduling Problem*, submitted to Journal of Operations and Logistics (2007).

[3] BROOKS G.H., AND WHITE C.R., *An algorithm for finding optimal or near optimal solutions to the production scheduling problem*, Journal of Industrial Engineering, January-February issue (1965): 34-40.

[4] HERROELEN W., AND LEUS R., *Robust and reactive project scheduling: a review and classification procedures*, International Journal of Production research, vol 8 (2004): 1599-1620.