



HAL
open science

A new relaxation of the Resource-Constrained Project Scheduling Problem

Pierre-Antoine Morin, Christian Artigues, Alain Haït

► **To cite this version:**

Pierre-Antoine Morin, Christian Artigues, Alain Haït. A new relaxation of the Resource-Constrained Project Scheduling Problem. 15th International Conference on Project Management and Scheduling - PMS'2016, Apr 2016, Valencia, Spain. pp.9-12. hal-01875789

HAL Id: hal-01875789

<https://laas.hal.science/hal-01875789>

Submitted on 17 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A new relaxation of the Resource-Constrained Project Scheduling Problem

Pierre-Antoine Morin^{1,2}, Christian Artigues² and Alain Haït¹

¹ ISAE-Supaéro, Université de Toulouse
 pierre-antoine.morin@isae.fr, alain.hait@isae.fr

² LAAS-CNRS, Université de Toulouse
 artigues@laas.fr

Keywords: Resource-constrained project scheduling, time/resource aggregation, lower bound

In this paper we propose to partially relax the resource constraints in order to get lower bounds of the RCPSP. To this aim, by analogy with tactical project planning, we consider the resource constraints on aggregated periods while the precedence constraints still hold. Hence, according to period duration, the resource constraints will be more or less strong. An iterative bounding scheme is presented and some results are given.

1 A time/resource relaxation of the Resource-Constrained Project Scheduling Problem

Given a set of activities $V = \{0, \dots, n + 1\}$ linked by precedence constraints E , and such that each activity $i \in V$ requires an amount $b_{i,k} \geq 0$ on each resource k of a set of resources R , the Resource-Constrained Project Scheduling Problem (RCPSP) can be defined as the problem of finding a start time S_i for each activity $i \in V$ while satisfying the precedence constraints E and not exceeding the resource availabilities B_k , $k \in R$, such that the makespan, given by the difference between the start time of the dummy end activity $n + 1$ and the dummy start activity 0 , is minimized. We consider a relaxation of the RCPSP (P_Δ) for any $\Delta \in \mathbb{R}_+^*$.

$$(P_\Delta) \quad \min S_{n+1} - S_0 \tag{1}$$

$$S_j \geq S_i + p_i \quad (i, j) \in E \tag{2}$$

$$\sum_{i \in V} p_{i,t} b_{i,k} \leq \Delta \cdot B_k \quad t \in T, k \in R \tag{3}$$

$$p_{i,t} = |[t\Delta, (t+1)\Delta] \cap [S_i, S_i + p_i]| \quad i \in A, t \in T \tag{4}$$

$$S_i \geq 0 \quad i \in A \tag{5}$$

where $T = \{0, \dots, |T| - 1\}$ is a set of periods t of length Δ and $p_{i,t}$ denote the length of the time interval under which activity i is executed in period t . We assume that $(|T| - 1)\Delta$ is an upper bound on the makespan. To obtain a correct formulation for the RCPSP, one should take $\Delta = 1$ and enforce integrality on the start times, i.e. replace constraints (5) by $S_i \geq 0$ and integer, $\forall i \in A$. Obviously (P_Δ) is a relaxation of the RCPSP (even for $\Delta = 1$) as it can be seen that any schedule S feasible for the RCPSP, is also feasible for (P_Δ), for any $\Delta \in \mathbb{R}_+^*$. Furthermore, in the figure below with three activities having requirements (3,4,3) on a single resource of availability 4 a makespan of 6 is obtained thanks to non integer start times in the optimal solution of (P_2).

Note that besides being a valid relaxation of the RCPSP, the problem (P_Δ) can be used in a project planning context, when resource consumption have to be aggregated over several periods in an upper decision level while precedence constraints between activities have

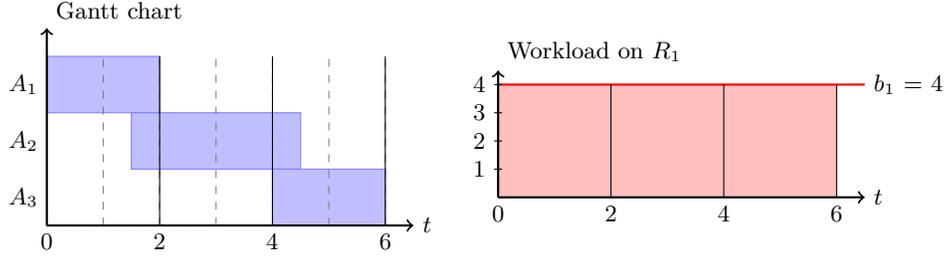


Fig. 1: Non-integer unique optimal solution ($\Delta = 2$).

to be tackled in a precise way. Another real-life application can be found in the cases where resource consumption cannot be computed in unit-time intervals for technical reasons. In (Haït and Artigues 2011), scheduling problem in a foundry involves electricity consuming activities with the constraint to keep total consumption below a prescribed maximal power, except that the consumption is only averaged each 15 minutes slots.

2 Schedule generation schemes and MILP formulation for P_Δ

We propose an adaptation of the serial and parallel schedule generation schemes (SSGS and PSGS), which are the basic constructive heuristics for the RCPSP. As in the RCPSP, an activity is candidate for being scheduled as soon as all its predecessors are scheduled. This yields an earliest start time $ES_i = \max_{(j,i) \in E} (S_j + p_j)$ for each such candidate activity i . Roughly, SSGS takes the candidate activities in a (predefined or dynamic) priority order and schedules the selected activity as soon as possible taking account of previously scheduled activities. PSGS proceeds by increasing time periods. Starting with $t = 0$, PSGS takes the first activity in the priority order that maximises $p_{i,t}$. If no activity can be scheduled in the period, t is incremented. In both cases we rely in a subproblem asking that given a set I of activities already scheduled, what is the earliest feasible start time of an activity $i \notin I$ inside a time period t ? This problem is solved as follows. Given a time period t , we denote by $\delta_{i,t}(I)$ the maximum possible value of $p_{i,t}$ given that activities I have already been scheduled. We have:

$$\delta_{i,t}(I) = \min_{k \in R} \delta_{i,t}^k(I) \text{ with } \delta_{i,t}^k(I) = \begin{cases} \min \left(\Delta, \frac{B_k \Delta - \sum_{j \in I} b_{j,k} p_{j,t}}{b_{i,k}} \right) & \text{if } b_{i,k} > 0 \\ \Delta & \text{otherwise} \end{cases}$$

For a candidate activity $ES_{i,t} = \max(ES_i, \Delta(t+1) - \delta_{i,t}(I))$ is in fact latest start time of i inside period t while maximizing its duration inside t , if $ES_i < \Delta(t+1)$. If $ES_{i,t} + p_i \leq \Delta(t+1)$, i can be feasibly scheduled at $S_i = ES_{i,t}$. Otherwise, there are $a_i = p_i - \Delta(t+1) + ES_{i,t}$ time units that have to be scheduled non-preemptively starting from $\Delta(t+1)$, which is possible if and only if $\delta_{i,t'}(I) = \Delta$ for $t' = t, \dots, t + \lceil \frac{a_i}{\Delta} \rceil - 1$ and $\delta_{i,t + \lceil \frac{a_i}{\Delta} \rceil}(I) \geq a_i \bmod \Delta$.

The model is adapted from the RCCP/RCPSVP model presented in (Haït and Baydoun 2012). It is based on a mixed time formulation: the continuous-time part is used to represent the precedence constraints and the discrete-time part for the resource constraints, aggregated over the periods. Binary variables $z_{s_{i,p}}$ represent the fact that activity i starts during period t or before (“step” variables). These variables are used to write the constraints that link activity start time S_i and $p_{i,t}$, the duration of activity i in period t , hence linking the continuous-time and the discrete-time parts. The constraints correspond to the six configurations of activity i and time interval $[\Delta.t, \Delta(t+1)]$ presented in Fig.2 (simplification of Allen’s 13 relations (Allen 1981)).

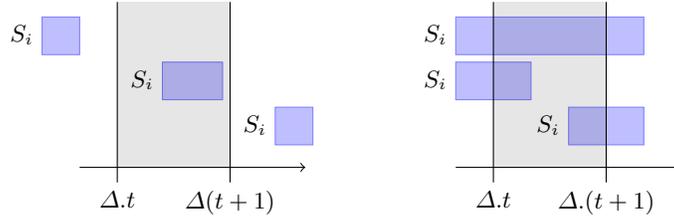


Fig. 2: Relative positions of an activity and a time interval.

The main difference with the original model of (Haït and Baydoun 2012) is that the durations of the activities are fixed in the RCPSP. Binary variables $z_{s_i,p}$ are sufficient to represent both the start and the end of an activity. Moreover, the resource requirements of an activity are the same all along its execution. For each period, the resource constraints are expressed as in Eq. (3).

3 An iterative lower bounding scheme for the RCPSP

The SGSs and the MILP developed for P_Δ are included in an iterative approach. The aim is to get lower and upper bounds of the RCPSP closer by progressively reducing period duration Δ .

Let $S^*(P_\Delta)$ be an optimum of P_Δ . Then the objective value of this solution is a lower bound for any $P_{\Delta'}$ with $\Delta' < \Delta$. If $S^*(P_\Delta)$ is feasible for $P'_{\Delta'}$, it is also an optimum of $P'_{\Delta'}$. Otherwise, we can apply one of the SGSs in order to repair $S^*(P_\Delta)$ and obtain a feasible solution for $P'_{\Delta'}$, upper bound of this problem. The MILP model is used to solve P_Δ . The bigger the period duration Δ , the less binary variables in the model. Starting with a big Δ guarantees a short computation time.

The proposed bounding scheme is applied to periods $\{\Delta_0, \Delta_1, \dots, \Delta_U\}$ where $\Delta_0 > \Delta_1 > \dots > \Delta_U$ and $\Delta_U = 1$. The sequence is given below:

1. Solve P_{Δ_0}
2. For each Δ_u in $\Delta_1, \dots, \Delta_U$
 - The lower bound is given by $S^*(P_{\Delta_{u-1}})$.
 - Repaired solutions from $P_{\Delta_{u-1}}, P_{\Delta_{u-2}}, \dots, P_{\Delta_0}$ define the upper bound of P_{Δ_u} . Some of them are used as initial solutions for MIP warm start.
 - Solve P_{Δ_u} .
3. The last iteration, for $\Delta_U = 1$, may still give a relaxation of the RCPSP if some of the activity start times are not integer. SGSs are again applied in that case in order to get an integer solution, optimum or upper bound of the RCPSP.

The number of iterations and the period reduction from one iteration to another are important parameters of this approach. The closer to period $\Delta_U = 1$, the less reduction should be applied, in order to keep efficient bounds for MILP solving.

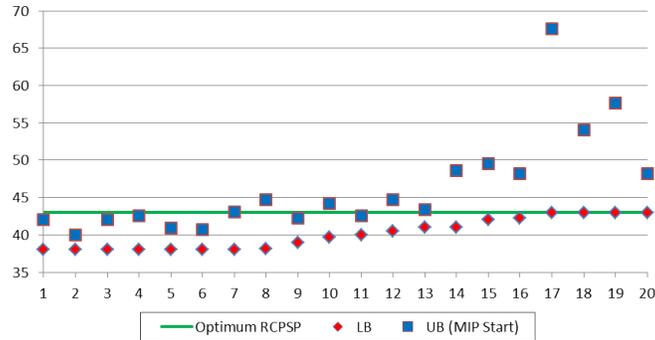
4 Experimental results and conclusion

Figure 3 represents the propagation of the bounds obtained by solving the instance 30.1.1 of the PSPLIB. The number of iterations and the lengths of the periods for each iteration was fixed *a priori*. One can note that the lengths chosen are such that the difference between two consecutive values for Δ progressively decreases throughout the iterative procedure. Time 1 is the resolution time when bounds are propagated inside the iterative

process, while time 2 corresponds to the resolution time without information propagation (no iterative process, P_Δ solved directly).

Iteration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Δ	8	7	6	5	4,5	4	3,5	3,25	3	2,75	2,5	2,25	2	1,8	1,6	1,4	1,3	1,2	1,1	1
Time 1 (s)	0,23	0,30	0,35	0,40	0,47	1,19	3,46	1,16	1,51	3,00	1,92	3,12	1,43	3,71	4,17	6,04	13,34	8,28	13,36	5,23
Time 2 (s)	0,23	0,53	0,96	1,28	2,03	4,11	6,60	7,99	10,46	16,33	16,75	21,44	26,61	33,74	38,29	57,01	77,42	80,60	128,11	137,53

(a) Values of Δ and computational times (with and without propagation)



(b) Propagation of the bounds

Fig. 3: Example: instance 30.1.1 (PSPLIB).

For one iteration, the upper bound is the smallest duration of feasible schedules for P_Δ , obtained by repairing previously generated solutions (given to the solver, that realizes a warm start), while the lower bound is the duration of an optimal schedule for P_Δ (returned by the solver).

The optimal solution found for P_1 (which is a relaxation of the RCPSP) is integer, so it is also optimal for the RCPSP. One can note that the lower bound increases continuously throughout the resolution process, and reaches the optimal duration for the RCPSP before the last iteration, i.e. with non-unit periods.

References

- Haït A., C. Artigues, 2011, “A hybrid CP/MILP method for scheduling with energy costs”, *European Journal of Industrial Engineering*, Vol. 5, pp. 471-489.
- Haït A., G. Baydoun, 2012, “A new event-based MILP model for the resource-constrained project scheduling problem with variable intensity activities (RCPSVP)”, *Proc. IEEE International Conference on Industrial Engineering and Engineering Management*, Hong Kong, 2012.
- Allen J.F., 1981, “An Interval-Based Representation of Temporal Knowledge”, in *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI'81)*, Vancouver, Canada, 1981.