A Continuous-Time Model for the Multi-Site Resource-Constrained Project Scheduling Problem

Mario Gnägi and Norbert Trautmann

Department of Business Administration, University of Bern, 3012 Bern, Switzerland
mario.gnaegi@pqm.unibe.ch, norbert.trautmann@pqm.unibe.ch

Keywords: Resource-Constrained Project Scheduling, Multiple Sites, Mathematical Programming.

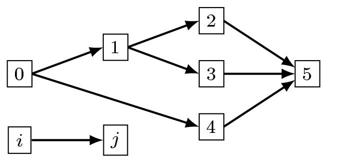
1 Introduction

In the Resource-Constrained Project Scheduling Problem (RCPSP), it is assumed that all project activities are executed at a single site, and consequently all resources are located at this site. In the multi-site RCPSP, which is a novel extension of the single-site RCPSP, it is possible to consider the execution of the activities at several alternative sites. Furthermore, some units of the various renewable resource types are assumed to be permanently located at a single site, whereas the other units can be moved between the sites. Hence, the multi-site RCPSP includes the management of a so-called resource pooling because some resource units can be shared among the different sites. Finally, the spatial distance between the sites gives rise to two different types of transportation times that must be considered in the project schedule. First, while a resource unit is moved between two sites, i.e., during the transportation time, it cannot be allocated to the execution of an activity. Second, if two activities that are interrelated by a precedence relationship are executed at different sites, then a minimum time lag between the completion of the first activity and the start of the second activity must be taken into account, which corresponds to the time required for transporting the first activity's output between the respective sites.

To the best of our knowledge, the multi-site RCPSP has only been treated by Laurent et al. (2017), who provide a binary linear programming (BLP) model. The model belongs to the class of discrete-time models, i.e., the planning horizon is divided into a set of equally-long periods, and it is assumed that an activity can be completed at the end of such a period only. Laurent et al. (2017) report results of a computational analysis performed with CPLEX on a set of small-sized self-generated instances, where the number of activities was varied between 5 and 30. It turned out that within a prescribed time limit of $3'600$ seconds of CPU time, none of the instances with 30 activities could be solved to optimality; therefore, Laurent et al. (2017) additionally propose four different metaheuristics.

Subject of this paper is a novel continuous-time mixed-binary linear programming (MBLP) model for the multi-site RCPSP. Besides an illustrative example, we provide new computational results for instances with 30 activities and a varying number of sites. We have tested the novel continuous-time model and the discrete-time model proposed by Laurent et al. (2017) on a set of 960 instances also generated by Laurent et al. (2017). It has turned out that when using the novel continuous-time model, feasible solutions are devised for all instances; a large number of these instances can even be solved to optimality, and the MIP gap for the remaining instances is relatively small. Moreover, when using the novel continuous-time model, for a considerable number of instances a feasible solution is devised which has a better objective function value than the best solution devised by the discrete-time model and all the metaheuristics presented in Laurent et al. (2017).

The remainder of this paper is organized as follows. In Section 2, we illustrate the multi-site RCPSP by an example. In Section 3, we explain the types of decision variables used



Activity i	0	1	2	3	4	5
p_i	0	2	2	3	2	0
r_{i1}	0	1	1	1	2	0
r_{i2}	0	1	0	0	0	0

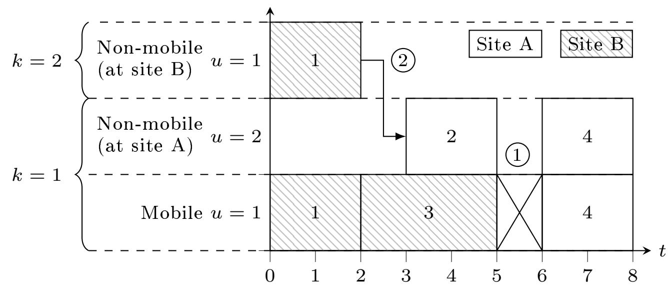


Fig. 1. Illustrative example: duration, resource requirements and activity-on-node network (left); optimal solution (right)

in the novel continuous-time model. In Section 4, we report our computational results. In Section 5, we provide some conclusions and give an outlook for future research.

2 Illustrative Example

In this section, we illustrate the multi-site RCPSP by means of an example project comprising $n = 4$ real activities $\{1, \ldots, 4\}$. Two resource types $k = 1$ and $k = 2$ are required for executing the activities. The project start and the project completion are represented by the fictitious activities 0 and $n+1$; both fictitious activities have a duration of 0 and do not require any unit of any resource type. The activity-on-node network of the illustrative example is depicted on the left-hand side of Figure 1. Each node of this network corresponds to exactly one activity of the project and vice versa, and an arc from a node i to a node j of the network indicates that a precedence relationship is prescribed between the completion of activity i and the start of activity j. The table below provides the duration p_i and the requirements r_{ik} for the two resource types $k = 1$ and $k = 2$ for each activity $i \in \{0, 1, \ldots, 5\}$. Furthermore, we assume that there are two sites A and B; a transport between these sites takes 1 unit of time in any direction. Moreover, we assume that there are two units of resource type $k = 1$; one of these units, e.g. unit $u = 1$, is mobile, i.e., it can be moved between sites A and B, and the other unit $u = 2$ is non-mobile, i.e., it is permanently located at site A. There is one unit of resource type $k = 2$, and this unit is non-mobile and permanently located at site B.

On the right-hand side of Figure 1, an optimal solution for our illustrative example is visualized. For each activity, rectangles indicate to which resource units it is assigned, and during which periods it is executed. First, there is a transportation time between the completion of activity $i = 3$ and the start of activity $j = 4$, because the mobile resource unit $u = 1$ of resource type $k = 1$, which is allocated to their execution, needs to be transported from site B to site A. Second, there is a transportation time between the completion of activity $i = 1$ and the start of activity $j = 2$, because these two activities are precedence-interrelated, but activity $i = 1$ is executed at site B, and activity $j = 2$ is executed at site A, i.e., the output of activity $i = 1$ must be transported from site B to site A.

3 Decision Variables Used in the Novel Continuous-Time Model

In this section, we explain the types of decision variables used in the novel continuous-time model by means of our illustrative example; for a detailed presentation of the model, we refer to Gnägi and Trautmann (2019). The notation used for the sets and parameters is as follows. The set V comprises all activities including the fictitious ones representing the

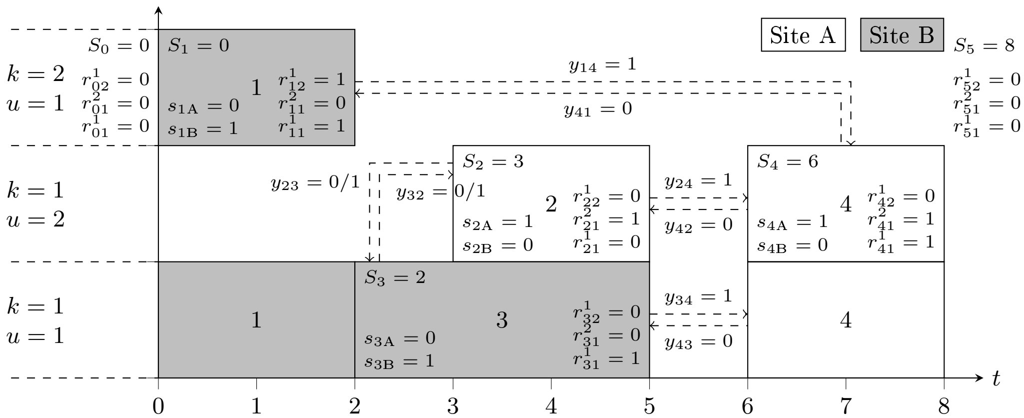


Fig. 2. Types of decision variables used in the continuous-time model

project start and the project completion; we also use a set $\dot{V}$ comprising all real activities only. Furthermore, the set L comprises the alternative sites, and the set R comprises the various resource types; for each resource type $k \in R$, we denote the available number of units as R_k.

The model employs a set of continuous start-time variables S_i $(i \in V)$, together with the following three sets of binary variables:

- site-selection variables s_{il} indicate whether an activity $i \in \dot{V}$ is executed at site $l \in L$,
- resource-assignment variables r^u_{ik} indicate the assignment of an activity $i \in V$ to the various units $u \in \{1, \ldots, R_k\}$ of resource type $k \in R$, and
- sequencing variables y_{ij} indicate the sequences between all pairs of activities $(i, j) \in \dot{V} \times \dot{V}$ $(i \neq j$ and $(i, j) \notin TE)$, where TE denotes the set of pairs of activities that cannot be executed in parallel due to the prescribed precedence relationships.

We illustrate these types of decision variables in Figure 2 by means of the example project introduced in Section 2. By convenience, the project starts at time 0, i.e., $S_0 := 0$.

4 Computational Results

In this section, we report the results of our experimental performance analysis. We tested the performance of both the discrete-time BLP model presented in Laurent et al. (2017), hereafter referred to as LDGN17, and the novel continuous-time MBLP model, hereafter referred to as GT20, on 960 test instances from the set MSj30 for the multi-site RCPSP, each of which consisting of $n = 30$ activities with $|L| = 2$ or $|L| = 3$ sites; these instances have been generated by Laurent et al. (2017) by adapting the well-known single-site RCPSP instances j30 from the PSPLIB (cf. Kolisch and Sprecher 1996). We implemented both models in Python 3.6, and we used the Gurobi Optimizer 8.1 as solver. For each test instance, we prescribed a maximum computation time of 300 seconds, and we limited the maximum number of threads to four. We did not change the default values for any other solver setting.

Table 1 summarizes the computational results. We report the results for all tested instances, but also for the subsets of instances for which both models devised at least a feasible solution. # Feas and # Opt correspond to the number of instances for which a

Table 1. Computational results

# Sites	Subset	Model	# Feas	# Opt	Gap^{LB} (%)	Gap^{BKS} (%)	# BKS^{+}	# Vars
2	All	LDGN17	453	268	30.64	13.34	59	8,826
		GT20	**480**	**316**	**13.17**	**0.56**	**121**	**3,187**
2	Feas	LDGN17	**453**	268	30.64	13.34	59	8,873
		GT20	**453**	**313**	**11.23**	**0.57**	**108**	**3,237**
3	All	LDGN17	443	214	45.91	19.75	61	8,858
		GT20	**480**	**277**	**21.09**	**1.56**	**149**	**3,217**
3	Feas	LDGN17	**443**	214	45.91	19.75	61	8,906
		GT20	**443**	**274**	**17.85**	**1.40**	**138**	**3,269**

feasible solution and to the number of instances for which a proven optimal solution, respectively, has been found within the prescribed maximum computation time. Gap^{LB} (%) corresponds to the average relative gap between the objective function value OFV and the lower bound LB obtained by the solver (calculated as $(OFV-LB)/LB$), and Gap^{BKS} (%) corresponds to the average relative gap between OFV and the best known solution BKS that has been reported by Laurent et al. (2017) for their proposed metaheuristics (calculated as $(OFV-BKS)/BKS$). # BKS^{+} corresponds to the number of instances for which a new best solution has been found. Finally, # Vars corresponds to the average number of variables used in the tested models (before Gurobi's preprocessing).

5 Conclusions and Outlook

We studied the multi-site RCPSP, which extends the single-site RCPSP by considering alternative sites for the activities' execution, the management of resource pooling among the sites, and the arising transportation times between the sites. For the studied problem, we have developed a novel continuous-time MBLP model, which turned out to outperform the only known model from the literature with respect to various performance measures on a set of standard test instances.

A promising direction for future research is to apply the novel continuous-time model for a detailed analysis of the advantages of resource pooling in project management.

References

Laurent, A., Deroussi, L., Grangeon, N., and Norre, S., 2017, "A new extension of the RCPSP in a multi-site context: Mathematical model and metaheuristics.", *Computers & Industrial Engineering*, Vol. 112, pp. 634–644.

Gnägi, M., and Trautmann, N., 2019, "A continuous-time mixed-binary linear programming formulation for the multi-site resource-constrained project scheduling problem.", In: Wang, M., Li, J., Tsung, F., and Yeung, A. (eds.): *Proceedings of the 2019 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Macau, pp. 382–365.

Kolisch, R., and Sprecher, A., 1996, "PSPLIB-a project scheduling problem library.", *European Journal of Operational Research*, Vol. 96(1), pp. 205–216.