Operations Research Perspectives 4 (2017) 1–11

Contents lists available at ScienceDirect

Operations Research Perspectives

journal homepage: www.elsevier.com/locate/orp

On the generalization of constraint programming and boolean satisfiability solving techniques to schedule a resource-constrained project consisting of multi-mode jobs

Alexander Schnell, Richard F. Hartl*

University of Vienna, Department of Business Administration, Chair of Production and Operations Management, Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

ARTICLE INFO

Article history:
Received 15 May 2016
Revised 16 January 2017
Accepted 16 January 2017
Available online 23 January 2017

Keywords:
Multi-mode resource-constrained project scheduling
Constraint programming
SAT solving
SCIP
Lazy clause generation
Exact algorithm

ABSTRACT

In our paper, we analyze new exact approaches for the multi-mode resource-constrained project scheduling (MRCPSP) problem with the aim of makespan minimization. For the single-mode RCPSP (SRCPSP) recent exact algorithms combine a Branch and Bound algorithm with principles from Constraint Programming (CP) and Boolean Satisfiability Solving (SAT). We extend the above principles for the solution of MRCPSP instances. This generalization is on the one hand achieved on the modeling level. We propose three CP-based formulations of the MRCPSP for the G12 CP platform and the optimization framework SCIP which both provide solution techniques combining CP and SAT principles. For one of the latter we implemented a new global constraint for SCIP, which generalizes the domain propagation and explanation generation principles for renewable resources in the context of multi-mode jobs. Our constraint applies the above principles in a more general way than the existing global constraint in SCIP. We compare our approaches with the state-of-the-art exact algorithm from the literature on MRCPSP instances with 20 and 30 jobs. Our computational experiments show that we can outperform the latter approach on these instances. Furthermore, we are the first to close (find the optimal solution and prove its optimality for) 628 open instances with 50 and 100 jobs from the literature. In addition, we improve the best known lower bound of 2815 instances and the best known upper bound of 151 instances.

1. Introduction

The multi-mode resource-constrained project scheduling problem (MRCPSP) is a generalization of the single-mode RCPSP (SRCPSP) where an additional mode-assignment step has to be considered. The aim is to find the best mode-assignment for a number of jobs subject to nonrenewable resource constraints such that the optimal schedule for the resulting SRCPSP (if existing) optimizes a specific objective function.

For the SRCPSP recent exact algorithms combine a Branch and Bound (BaB) algorithm with principles from Constraint Programming (CP) and Boolean Satisfiability Solving (SAT) (see [5], [15] and [30]). The idea of the CP-SAT algorithms is to combine the domain propagation processed through global constraints (Apt [3]) with the Conflict Analysis (CA) techniques of a SAT solver (Marques-Silva and Sakallah [19]). Therefore, the different propagators of the global constraints generate explanations, i.e. clauses consisting of Boolean literals, for their domain updates and the detected inconsistencies. The latter explanations are transfered to a SAT solving mechanism. The SAT mechanism constructs a conflict graph based on the explanations of the domain propagators and can possibly deduce *nogoods* and *backjumps* via CA.

Roughly speaking, nogoods are valid clauses for a SAT model, like e.g. cutting planes in Mixed-Integer Programming (MIP), which possibly prune branches of the BaB-tree. Backjumps are backtracking moves which lead from the actual node a to a preceding node p whereas $d(a) - d(p) > 1$ holds for the depth levels $d(a)$ and $d(p)$ in the BaB tree. Moreover, the branching strategy of the underlying BaB-algorithm uses conflict statistics of the literals forming the explanations. In general, the algorithms branch on the variables and values based on the number of conflicts the respective literals were involved in (Moskewicz et al. [21]). For a more detailed introduction to the principles of CP and SAT solving and the possible

* Corresponding author.
E-mail addresses: alexander.schnell@univie.ac.at (A. Schnell), richard.hartl@univie.ac.at (R.F. Hartl).

http://dx.doi.org/10.1016/j.orp.2017.01.002

combination of the both to one exact solution algorithm, we refer to Schutt et al. [31], Schutt [27] and Achterberg [1,2].

The lazy clause generation approach (LCG), a CP-SAT hybrid introduced by Ohrimenko et al. [23] and extended by Schutt et al. [28], is up-to-date the best exact approach for the SRCPSP with standard precedence relations and the aim of makespan minimization. Furthermore, LCG was also applied to variants of the SRCPSP with more general constraints and with objective functions differing from makespan minimization. Schutt et al. [31] successfully solve the SRCPSP with generalized precedence relations by LCG. They outperform the state-of-the-art exact approaches for this problem and also on average report better results compared to state-of-the-art heuristics. Moreover, Schutt et al. [29] outperform the state-of-the-art exact algorithm for the SRCPSP with discounted cash flows, again by generalizing LCG to this problem. One can conclude, that LCG is a robust approach for variants of the SRCPSP.

The aim of this paper is to provide a generalization of the CP-SAT hybrids for the SRCPSP to the MRCPSP. Exact approaches for the MRCPSP have been summarized and tested by Hartmann and Drexl [12], whereas they conclude that the approach of Sprecher and Drexl [32] is the exact method of choice. The most recent exact algorithm of Zhu et al. [37] outperforms the latter approach. They implemented a Branch-and-Cut procedure with a preprocessing and a heuristic step to generate good upper bounds as an input for their algorithm. A recent survey on heuristic approaches for the MRCPSP and a detailed experimental evaluation is given by Peteghem and Vanhoucke [35]. Their computational experiments show that the scatter search procedure of Peteghem and Vanhoucke [34] produces the best results. In this context, it is also important to mention the approach of Coelho and Vanhoucke [8] as they combine SAT solving techniques with a metaheuristic for the SRCPSP to solve the MRCPSP.

Our contribution is an extension of recent exact approaches combining CP and SAT techniques which are efficient for the SRCPSP to the MRCPSP, more precisely the MRCPSP with standard precedence relations. This extension can be partly achieved on the modeling level. We propose three CP models for the MRCPSP which can be formulated in optimization frameworks that integrate an exact solution approach combining CP, SAT and MIP techniques. Moreover, for one modeling formulation we implemented a new global constraint `cumulativemm` specially tailored to renewable resources in the context of multi-mode jobs. Note that we already successfully generalized and applied recent CP-SAT approaches to the MRCPSP with generalized precedence relations in [26]. The paper at hand can be seen as a predecessor of the latter paper.

In the remainder of the paper, we proceed as follows. In Section 2, we describe the MRCPSP and its computational complexity. Section 3 introduces three problem formulations in optimization frameworks which support the solution by a BaB algorithm integrating CP, SAT and MIP techniques. In Section 4, we describe the principles of our new global constraint `cumulativemm`. Section 5 discusses the results of our computational experiments and draws a comparison to the state-of-the-art exact approach of Zhu et al. [37]. The paper ends with a conclusion derived from the obtained results.

2. Problem description and complexity

The MRCPSP is a generalization of the SRCPSP, where every job $j \in J = \{0, \ldots, n+1\}$ can be processed in different modes $k \in M_j \subseteq \mathbb{N}$. The jobs 0 and $n+1$ are dummy jobs representing the start and the end of the complete project, i.e. in the beginning every job with no predecessor and every job with no successor is connected to the dummy job 0 and $n+1$ in the precedence network, respectively. Moreover, the jobs can not be preempted.

Moreover, a set of nonrenewable (renewable) resources $N(R) \subseteq \mathbb{N}$ with a maximal capacity of $C_r^{\nu},\ r \in N$ ($C_r^{\rho},\ r \in R$) is given. Every job's integer duration $d_{j,\,k} \geq 0$, nonrenewable (renewable) resource consumption $c_{j,k,r}^{\nu},\ r \in N(c_{j,k,r}^{\rho},\ r \in R)$ is dependent on the selected mode $k \in M_j$.

Nonrenewable resources $r \in N$ like e.g. a project budget or energy are available for the complete planning horizon. Once job j is processed in mode k, $C_r^{\nu} - c_{j,k,r}^{\nu}$ units of the nonrenewable resource $r \in N$ are still available for the remaining jobs. Moreover, a constant amount C_r^{ρ} of a renewable resource $r \in R$ like e.g. a number of machines or workers is available at every point in time.

Furthermore, a job $j \in J$ cannot end after a job from its successor set $\mathfrak{S}_j$ has started, i.e. in our paper we only consider standard precedence relations. As objective, we consider makespan minimization.

The solution of the MRCPSP can be divided into two steps. The first step consists of finding a feasible mode-assignment w.r.t. the nonrenewable resource capacities. The knapsack problem is polynomially reducible to the latter problem, i.e. already the mode-assignment step is **NP**-complete for $|N| \geq 2$ (Kolisch and Drexl [17]). The second step consists of finding an optimal schedule for a SRCPSP instance, i.e. of finding a schedule which minimizes the makespan and respects the precedence constraints and the renewable resource capacities for a given mode-assignment.[1] Note, that the SRCPSP with the objective of makespan minimization is strongly **NP**-complete (Blazewicz et al. [6]).

In total, one has to find a feasible mode-assignment at which the minimal makespan of the resulting SRCPSP is not larger than the minimal makespan detected for any other feasible mode-assignment.

As a preprocessing step one can remove redundant nonrenewable resources, inefficient and non-executable modes (see [32] and [11]). Furthermore, lower and upper bounds $lb(s_j)$ and $ub(s_j)$ can be deduced for the starting times s_j by applying forward (backward) recursion [7]. This approach is based on longest path calculations in the precedence network where the arc weights correspond to the minimal mode durations of every job $j \in J$ w.r.t. the remaining modes. For the evaluation of $ub(s_j)$, an upper bound T on the makespan is needed. T can be given by a problem specific heuristic or $T_{\max}$ defined in Section 4.

3. CP-models for the MRCPSP

There are two solution frameworks which provide a solution algorithm consisting of a combination of CP, SAT and MIP techniques.

The first is the Constraint Integer Programming framework SCIP, developed by Achterberg [2] and maintained and extended by members of the Zuse Institute in Berlin. SCIP provides a general BaB algorithm for optimization and allows the user to implement plugins, e.g. special branching strategies, primal heuristics and constraint handlers (i.e. global constraints). Moreover, default plugins exist to use SCIP as a stand-alone CP or MIP solver. Furthermore, when the formulated model only consists of default constraint handlers provided by SCIP, the solution algorithm integrates techniques from CP, SAT Solving and MIP.

The second framework is the G12 Constraint Programming Platform [9] provided by the NICTA research team [22]. The user can formulate a problem in the modeling language Zinc [20] and choose between different solution algorithms. Thereby, LCG can also be chosen for the solution of a model. With the G12 Con-

[1] Note that, it can happen that no feasible schedule for the resulting SRCPSP exists, if mode m has been chosen for job j and $c_{j,m,r}^{\rho} > C_r^{\rho}$.