European Journal of Operational Research 197 (2009) 492–508

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

A heuristic solution framework for the resource constrained (multi-)project scheduling problem with sequence-dependent transfer times

Doreen Krüger, Armin Scholl *

Friedrich-Schiller-Universität Jena, Fakultät für Wirtschaftswissenschaften, Lehrstuhl für Betriebswirtschaftliche Entscheidungsanalyse, Carl-Zeiß-Straße 3, D-07743 Jena, Germany

ARTICLE INFO

Article history:
Received 5 September 2007
Accepted 16 July 2008
Available online 5 August 2008

Keywords:
Project scheduling
Combinatorial optimisation
Mathematical model
Transfer times
Setup

ABSTRACT

We consider the problem of scheduling multiple projects subject to joint resource constraints. Most approaches proposed in the literature so far are based on the unrealistic assumption that resources can be transferred from one project to the other without any expense in time or cost. In order to contribute to closing this gap to reality, we generalise the multi-project scheduling problem by additionally including sequence- and resource-dependent transfer times, which represent setup activities necessary when a resource is removed from one project and reassigned to another (or from one job to another within the same project). In this paper, we define the modified resource constrained multi-project scheduling problem with transfer times (called RCMPSPTT), which aims at minimising the multi-project duration for the single-project approach or the mean project duration for the multi-project approach. We formulate both perspectives as an integer linear program, propose priority rule based solution procedures and present results of comprehensive computational experiments. Provided that the combination of scheduling scheme and priority rules is chosen appropriately, the procedures obtain good results. In particular, resource oriented priority rules are identified to be successful.

1. Introduction

1.1. Multi-project scheduling

Project scheduling has been playing a vital role in literature for some decades now. The common *resource constrained project scheduling problem (RCPSP)* has been studied extensively. However, single-project settings are rare in business today. Usually, companies run more than one project simultaneously. According to Payne (1995) up to 90% of all projects (measured by their value) worldwide are executed in a multi-project environment. This finding goes along with Lova and Tormos (2001), who questioned 202 Spanish companies and found that 84% of them run multiple projects in parallel. Nonetheless, single-project management concepts are by no means irrelevant as they provide a solid basis for multi-project concepts. For excellent introductions to resource constrained single-project scheduling, see Kolisch (1995), Klein (2000), Demeulemeester and Herroelen (2002), Neumann et al. (2003). For surveys of solution approaches see e.g. Brucker et al. (1999), Kolisch and Padman (2001), Kolisch and Hartmann (1999, 2006).

The *resource constrained multi-project scheduling problem* (*RCMPSP*) as an extension of the RCPSP is considered as the simultaneous scheduling of two or more projects which demand the same scarce resources. Precedence constraints are usually defined only within projects. However, precedence relations between projects are also possible which would result in a programme of interdependent projects as a special form of a sheer multi-project (Lycett et al., 2004, p. 289; Ireland, 2002, p. 23). Projects are linked by the usage of the same restricted resources of the company. An objective function on company level often has to be considered although objectives of single-projects may also be regarded (Kurtulus and Davis, 1982, p. 161). The company objective as e.g. maximising profit is aimed at by managing the whole project portfolio or multi-project of the company by a resource manager, whereas project targets are set by single-project managers. The latter aim to minimise project delay, project cost, etc.

* Corresponding author. Tel.: +49 3641943170.
E-mail addresses: d.krueger@wiwi.uni-jena.de (D. Krüger), a.scholl@wiwi.uni-jena.de (A. Scholl).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2008.07.036

Multi-project scheduling as considered here has been a research topic since the late 1960s. However, it has been studied not nearly as comprehensively as single-project scheduling. One may distinguish two main research fields in multi-project scheduling – the static and the dynamic project environment (Dumond and Mabert, 1988, p. 102). The *static environment* view assumes a closed project portfolio. All projects of the company are summarised to a super-project (portfolio) and scheduled once. The multi-project is unequivocal and no rescheduling necessary. After the last project of a multi-project has been completed, a new multi-project may start. On the contrary, the *dynamic environment* view considers an open project portfolio. While scheduled projects are executed, new projects arrive to the system and have to be integrated by rescheduling.

Research mainly focuses on the static environment. Scheduling in such a static environment has been researched amongst others by Fendley (1968), who was the first discussing the modelling of a complete multi-project scheduling system and proposing methods for assigning due dates to incoming projects and priority rules for sequencing individual jobs. Pritsker et al. (1969) present a zero–one integer program for the problem considering various possible constraints, e.g. job splitting or resource substitutability and objectives like, e.g., total throughput time, makespan or total lateness. Kurtulus and Davis (1982) introduce the single-project approach, whereas Kurtulus and Narula (1985) add the multi-project approach for the multi-project scheduling problem (see below). In both papers, special priority rule based solution procedures are developed and tested. Lawrence and Morton (1993) test resource price based priority rules to minimise weighted tardiness costs within a multi-project. Lova and Tormos (2001) analyse existing priority rules for the single- and multi-project approach. Moreover, they present new two-phase rules for the multi-project approach. Vercellis (1994) suggests a decomposition technique. However, these are only some examples for research on static multi-project scheduling, further references are, amongst others, Patterson (1973), Mohanty and Siddiq (1989), Wiley et al. (1998), and Lova et al. (2000).

Dynamic environments are researched by Dumond and Mabert (1988). Their study is based on priority rules for static environments including due date assignment rules. Dumond (1992) as well as Dumond and Dumond (1993) introduce different resource availability levels. Bock and Patterson (1990) allow resource preemption in the multi-project. Yang and Sum (1993, 1997) consider dynamic project environments with a dual-level management structure for assigning resources to projects on a higher level and scheduling projects on a lower level. Ash and Smith-Daniels (1999) put emphasis on the learning, forgetting and relearning cycle in dynamic multi-project environments while Anavi-Isakow and Golany (2003) apply queuing theory and adapt the production management concept of CONWIP (constant work in progress) to the multi-project environment.

This paper extends static multi-project scheduling which consists of a single- and a multi-project approach. The *single-project approach* merges all projects of the multi-project to an artificial super-project with a dummy start and end job for time and resource scheduling (Kurtulus and Davis, 1982, p. 162). Hence, in this case multi-project scheduling is identical to single-project scheduling of large projects. The multi-project duration, which is given by the realised finishing time of the last job of the latest project, is minimised. The *multi-project approach* keeps the projects separate for time scheduling to identify a critical path for each project. Afterwards, the projects are merged for resource scheduling. The objective is to minimise the mean project delay, i.e., average lateness with respect to due dates or, as a surrogate, critical path times.

The only scheduling procedures applied to both approaches so far are heuristics using priority rules. According to Lova and Tormos (2001), a multi-project consists of about 120 up to 480 jobs. Exact procedures cannot handle problems of this size so far since already the basic RCPSP is NP-hard. Hence, heuristics are the only realistic possibility for solving the RCMPSP.

Even though the static project environment is emphasised in literature, it still fails to pay attention to some important aspects of multi-projects. Virtually all papers neglect resource transfers between projects or assume them having zero duration. Time delays and costs caused by these transfers are not taken into account. In reality, transfers may take time

- when a resource is physically moved from one location to another, e.g. heavy machines, specialists that fly around the world, and/or
- when a resource has to be adjusted in respect to content, e.g. setups of machines or human resources that have to get acquainted with new projects.

1.2. Setup times in project scheduling

Setup times which are a variant of transfer times have already been investigated in production scheduling and lot sizing extensively. In single-project scheduling some research on the field of setup times has been done. Kaplan (1991) introduces sequence-independent setup times when a job is restarted after preemption and presents a dynamic programming procedure to solve the problem. However, Demeulemeester (1992) shows that Kaplan's algorithm may fail to find optimal solutions. Kolisch (1995) develops a zero–one integer program for a restricted version of RCPSP with sequence-independent setup times. Vanhoucke (2008) considers sequence-independent setups necessary to continue a preempted job. Neumann et al. (2003, chapter 2.14) extend the approach of Trautman (2001) and present the RCPSP with time windows and sequence-dependent changeover times. Unlike Trautmann, who assumes that resource requirements can only take values of 0 or 1, Neumann et al. allow for arbitrary resource capacities and resource requirements of jobs. They split the problem into two interdependent subproblems. In a first step, a precedence-feasible schedule is determined. In a second step it is checked whether this schedule is changeover feasible. Changeover feasibility is given when all resource constraints are met while setup times are considered. They also present a branch-and-bound algorithm that enumerates time- and changeover feasible schedules. Mika et al. (2006) give a more extensive literature review on setup times in project scheduling. They classify setups into several types from a job perspective but do not focus on resources which have to changeover to other jobs or projects.

In multi-project scheduling, setup or transfer times are rarely discussed in literature so far. Yang and Sum (1993, 1997) consider resource transfers in a dynamic multi-project environment with a dual-level management structure. A central resource pool manager assigns resources to projects, whereas each project manager schedules jobs within his project using the allocated resources. Resource changeovers can only be handled via a central pool. Dodin and Elimam (1997) present an audit scheduling problem considering sequence-dependent