Just–in–Time Production of Large Assemblies Using Project Scheduling Models and Methods

Rainer Kolisch

Lehrstuhl für Technische Dienstleistungen und Operations Management
Technische Universität München

1 Introduction

Since the advent of just–in–time driven production planning and control at the Toyota manufacturing plants, the just–in–time paradigm has considered wide–spread consideration within production and operations management (cf., e.g., Schniederjans [22] and Cheng and Podolski [5]). While it was first employed for the high–volume–production of goods only, later there has been considerable research in the area of low–volume, make–to–order manufacturing (cf., e.g., Baker and Scudder [2], Neumann et al. [18], and Rachamadugu [21]). Agrawal et al. [1] considered a practical scheduling problem at Westinghouse ESG, where a number of customer–specific products have to be assembled subject to technological precedence and capacity constraints. The authors developed a MIP–formulation and – in the face of the $\mathcal{NP}$-hardness of the problem – a 'lead time evaluation and scheduling algorithm' with acronym LETSA.

In what follows we will show that the problem as considered by Agrawal et al. [1] – in line with many other well known scheduling problems – can be modeled as classical resource–constrained project scheduling problem (RCPSP). The remainder of the paper is organized as follows: In Section 2 we introduce the assembly scheduling problem and the heuristic proposed by Agrawal et al. [1]. Section 3 provides the resource–constrained project scheduling problem and outlines the serial scheduling algorithm. In Section 4 we show how the assembly scheduling problem can be modeled and solved as RCPSP. Finally, Section 5 outlines the impact of this result.

2 The Assembly Scheduling Problem

The assembly scheduling problem (ASP) can be depicted as follows (we use, with some minor modifications, the original notation proposed by Agrawal et al. [1]): There are $e = 1, \ldots, n_f$ customer–specific products. Each product e has to be assembled until its due date D_e. The assembly–structure of each product e is depicted by its bill of material (BOM). Figure 1 shows the

BOM of two products. Product $e = 1$ with due date $D_1 = 14$ comprises operations $O_1, \ldots, O_6$, product $e = 2$ with due date $D_2 = 10$ comprises operation O_7. Each rectangle depicts a make part and each circle depicts an operation. A make part is manufactured by a sequence of operations. Overall, there are n operations. Each product e has one final–assembly operation $f(e)$ which does not have any downstream operations. All other assembly operations O_i of product e have exactly one downstream operation $d(i)$. This gives for each product an assembly structure of the operations. In the assembly shop there are m different work–centers. In each work–center $W_{\mathcal{K}}$, $\mathcal{K} = 1, \ldots, m$, there are $f_{\mathcal{K}}$ functional identical machines. $I_{\mathcal{K}}$ is the set of operations which have to be processed by one of the machines in work–center $W_{\mathcal{K}}$. The processing of operation O_i takes t_i periods time. Once started, an operation cannot be preempted. When processed, operation O_i occupies one of the functional identical machines of the work–center where it has to be manufactured. Table 1 and Figure 1 give a two–product example which has been derived by adding product $e = 2$ to the example originally given in Agrawal et al. [1].

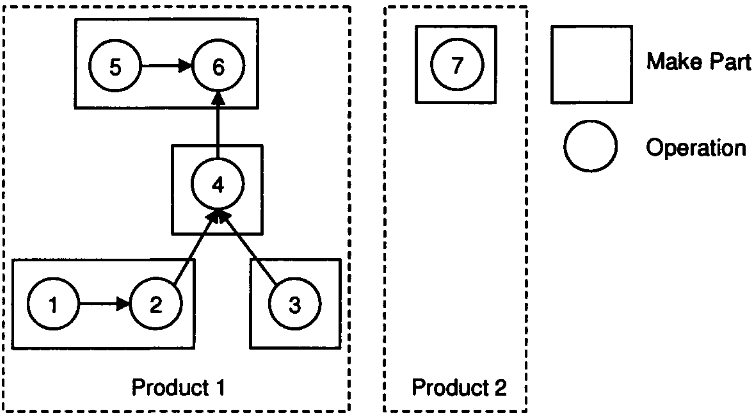

Figure 1: BOM of two products

In order to model the ASP, Agrawal et al. [1] introduce the following decision variables:

$$\delta_{i,j} = \begin{cases} 1, & \text{if operation } O_j \text{ precedes operation } O_i \\ 0, & \text{otherwise} \end{cases}$$