Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

ELSEVIER

IFAC PapersOnLine 50-1 (2017) 7970–7974

Long-term production planning problem: scheduling, makespan estimation and bottleneck analysis.

Dmitry I. Arkhipov * **Olga Battaïa** ** **Alexander A. Lazarev** ***

* *Department of Complex Systems Engineering, ISAE-SUPAERO, Universite de Toulouse, 10 avenue Edouard Belin - BP 54032 - 31055 Toulouse Cedex 4 France; V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russian Federation. (e-mail: miptrafter@gmail.com)*

** *Department of Complex Systems Engineering, ISAE-SUPAERO, Universite de Toulouse, 10 avenue Edouard Belin - BP 54032 - 31055 Toulouse Cedex 4 France. (e-mail: olga.battaia@isae.fr)*

*** *V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russian Federation; Lomonosov Moscow State University, Moscow, Russian Federation; Moscow Institute of Physics and Technology, Dolgoprudny, Russian Federation; International Laboratory of Decision Choice and Analysis, National Research University Higher School of Economics, Moscow, Russian Federation. (e-mail: jobmath@mail.ru)*

Abstract: In this paper, a long-term production planning problem is considered with the objective criteria C_{max} and T_{max} and under resource capacity and precedence constraints. The case study presented is characterized by four years planning horizon, 3552 operations, 51 workers and 57 units of equipment. The solution method elaborated in this study is a heuristic algorithm. Its performances are evaluated in numerical experiments. New procedures for makespan and resource load estimation are developed in order to identify bottleneck resources. The makespan estimation algorithm is tested on the well-known PSPLIB benchmark library where the best known lower bounds are improved for 5 instances. This procedure can also be used for the estimation of the gap from optimal value of the makespan time provided by the heuristic algorithm.

Keywords: Scheduling, resource-constrained project scheduling, timetabling, heuristics, makespan, bottleneck analysis

1. INTRODUCTION

We consider the problem of long-term planning for an engine assembly company. This problem can be considered as a special case of Resource Constrained Project Scheduling Problem (RCPSP) with a huge number of operations and two types of renewable resources. However, the considered problem has also some particularities which make it even harder than the classical formulation of RCPSP. First difference lies in the fact that the resource consumption rate for an operation depends on the resource type. Another difference is due to the need to create timetables for resources. Because of long-term planning horizon, the problem instances can be very large with a huge number of operations, workers and pieces of equipment. This study was motivated by the real-data problem with 3552 operations, 51 workers, 57 units of equipment and four years planning horizon. Taking into account such voluminous data that can be even more significant for some instances, heuristic approach was chosen to guarantee an optimization result in reasonable solution time.

RCPSP is a well-studied classical combinatorial optimisation problem. Garey & Johnson (1975) proved that the decision variant of the RCPSP is NP-complete in the strong sense even without precedence constraints and only one resource, by reduction to the 3-partition problem. A comprehensive survey on project scheduling problems formulations and solution methods was presented by Kolish & Padman (1997). Most of makespan lower bound estimation approaches are listed in the surveys written by Neron et.

* This research is supported by the Russian Science Foundation (grant 17-19-01665).

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2017.08.991

al. (2006) and Knust (2015). To compare algorithms performance and obtained lower bounds, benchmark library PSPLIB was created by Kolisch & Sprecher (1997). For the most of PSPLIB instances the best known makespan lower bounds were provided by the approaches presented by Brucker & Knust (2000), Schutt et. al. (2013) and Berthold et. al. (2010). Experimental evaluations of existed algorithms were presented in Hartmann & Kolisch (2000) and Kolisch & Hartmann (1998).

Since the problem considered in this paper differs from the classical formulation and in particular by its large scale, a new efficient heuristic approach is developed. This paper is organised as follows. The problem formulation and heuristic algorithm are presented in Sections 2 and 3 respectively. Section 4 describes the approach to estimate makespan and identify bottleneck resources. The results of numerical experiments are presented in Section 5 and some concluding remarks are given in Section 6. In the Section the table of most used notations is presented.

2. PROBLEM FORMULATION

The following optimization problem is considered. There is a set of orders O. For each order $i \in O$ a due date D_i is given. All orders are presented at time $t = 0$. For each ordered product there is a set of operations N_i and an assembly scheme presented by direct tree graph $G_i(N_i, TI_i)$ of operations to be done. All vertices of the graph G_i have only one outgoing edge, except the *final* vertex, which has only incoming edges. We denote the union of such graphs for all orders of the set O by $G(N, I)$, i.e. N is the set of operations to be done to perform all ordered products. For each operation $j \in N$, the following parameters are defined:

- set of required worker's skills RW_j;
- set of necessary equipment RE_j;
- worker occupation time p_j^w;
- equipment consumption time p_j^e.

One of the difference in comparison with classical RCPSP formulation is that for some operations $j \in N$ the inequality $p_j^w > p_j^e$ holds (Fig. 1). The sets of workers W and equipment E are given. One worker can have only one skill. The objective is to assign workers and equipment to

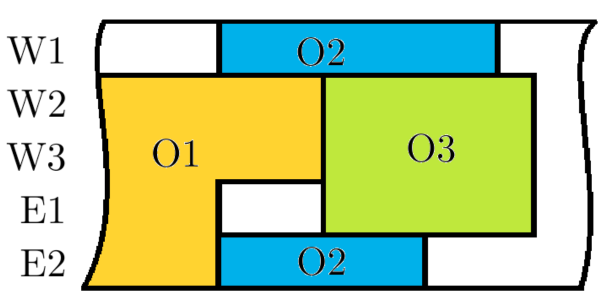

Fig. 1. Production of operations $O1$, $O2$, $O3$ by workers $W1$, $W2$, $W3$ using equipment $E1$ and $E2$.

operations and set up start time S_j and finish time F_j for each operation $j \in N$ subject to precedence relations and to minimize the bi-criteria of minimal tardiness and makespan objective functions

$$\min \max_{j\in N} F_j \,|\, \min \max_{i\in O}\{0, \max_{k\in N_i}\{F_k - D_i\}\}.$$

Heuristic approach is based on the idea of scheduling construction from right to left, minimizing the idleness of workers and equipment on each step, subject to critical paths of the graph G.

3. HEURISTIC APPROACH

3.1 Additional notations.

Let us introduce additional notations. For any operation $j \in N$ we define

- critical path $P_j = \max\limits_{K\in seq_j} \sum\limits_{l\in K} \max\{p_l^w, p_l^e\}$, where seq_j is a sequence of operations related to the path of the graph G incoming into vertex j and $j \in seq_j$;
- set of previous operations $prev_j$ is defined by graph G;
- one next operation $next_j$ is defined by graph G for all operations except final operations of orders of set O;
- state of operation s_j equals to:
  -1, if operation $next_j$ is not processed yet,
  0, if $next_j$ is processed, but s_j is still not processed,
  1, if s_j is already done.
  The initial state of operations equals to 0 for any final operation of the ordered product (if $next_j = \emptyset$) and equals -1 for others;
- operation ready time r_j equals D_i if j is a final operation of order O_i and equals S_{next_j} if $s_j \neq -1$ for other vertices. If $s_j = -1$, then r_j is not defined.

At each step of the algorithm, a set Q consists of operations with state $s_j = 0$.

For any equipment $k \in E$ or worker $l \in W$, we define a set of time intervals where it cannot process operations. We denote it as TI_k^e and TI_l^w respectively. Initially all sets consist of only one interval $[\max\limits_{i\in O} D_i, +\infty)$.

3.2 Algorithm description

Now let us present an approach to find suboptimal solution of the stated problem. It consists of two parts: inner cycle Algorithm ICA and main Algorithm MCA. ICA constructs the schedule for a set of heuristic parameters x_1, x_2, x_3, and the main Algorithm changes x_1, x_2, x_3 and chooses the solution with the best objective function value.

Inner cycle Algorithm description. Firstly we set states s_j of all operations $j \in N$: $s_j := 0$ if $next_j = \emptyset$ and $s_j := -1$ otherwise. Then we start an iteration cycle. At each iteration the set of candidates Q consists of all operations $j \in N$ with $s_j = 0$. Then operation $j \in Q$ with the highest value of criterion $CR(j)$ is chosen to be processed. Moments of time S_j and F_j, arrays of workers W_j and equipment E_j for processing operation j are found during the process of calculating $CR(j)$. After that algorithm changes the state of operation to $s_j := 1$ and for each previous operation $k \in prev_j$ set up state $s_k := 0$ and ready time $r_k := S_j$. For any worker $i \in W_j$ we add an interval $[S_j, S_j + p_j^w)$ to set TI_i^w. For any equipment $i \in E_j$ we add an interval $[S_j, S_j + p_j^e)$ to set TI_i^e. Then