European Journal of Operational Research 264 (2018) 200–211

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Production, Manufacturing and Logistics

Optimal workforce assignment to operations of a paced assembly line

Alexandre Dolgui[a,*], Sergey Kovalev[b], Mikhail Y. Kovalyov[c], Sergey Malyutin[a], Ameur Soukhal[d]

[a] IMT Atlantique, LS2N, CNRS, La Chantrerie, 4, rue Alfred Kastler - BP 20722, Nantes Cedex 3 F-44307, France
[b] INSEEC Business School, 25 rue de l'Université, Lyon 69007, France
[c] United Institute of Informatics Problems, National Academy of Sciences of Belarus, Minsk, Belarus
[d] Laboratoire d'Informatique, Ecole Polytechnique de l'Université de Tours, Tours, France

ARTICLE INFO

Article history:
Received 18 January 2016
Accepted 5 June 2017
Available online 10 June 2017

Keywords:
Workforce assignment
Scheduling
Complexity analysis
Mixed integer linear programming
Heuristics

ABSTRACT

We study a paced assembly line intended for manufacturing different products. Workers with identical skills perform non-preemptable operations whose assignment to stations is known. Operations assigned to the same station are executed sequentially, and they should follow the given precedence relations. Operations assigned to different stations can be performed in parallel. The operation's processing time depends on the number of workers performing this operation. The problem consists in assigning workers to operations such that the maximal number of workers employed simultaneously in the assembly line is minimized, the line cycle time is not exceeded and the box constraints specifying the possible number of workers for each operation are not violated. We show that the general problem is NP-hard in the strong sense, develop conventional and randomized heuristics, propose a reduction to a series of feasibility problems, present a MILP model for the feasibility problem, show relation of the feasibility problem to multi-mode project scheduling and multiprocessor scheduling, establish computational complexity of several special cases based on this relation and provide computer experiments with real and simulated data.

1. Introduction

We study a *paced unidirectional assembly line* consisting of m stations and manufacturing different products. Every station switches from processing a current product to the next one simultaneously. The time interval between two consecutive switches is called *cycle* and its duration is called *cycle time*. Cycle time remains the same for every cycle. Motivated by an industrial case, we study a workforce assignment problem for a single cycle of such a line. Without loss of generality, it is assumed that the cycle starts at time zero. In a given cycle, workers at station k execute a given set N_k of operations, $k = 1, \dots, m$. Parallel execution of operations is possible if these operations belong to different stations. The order of operations assigned to the same station should follow a given technological process characterized by *precedence relations* between the operations. If operation i is followed by operation j, then i must be completed before the start time of j. If operations i and j have no precedence relation, then they are called *independent* and can be performed in any order. Operations without predecessors can start at time zero. The set of precedence relations of operations at station k is represented by a *directed acyclic graph* $G_k = (N_k, U_k)$, where N_k is the set of operations assigned to station k and U_k, $U_k \subset N_k \times N_k$, is the set of oriented pairs of operations (i, j) of station k such that $(i, j) \in U_k$ if and only if operation i is followed by operation j. Let $N = \cup_{k=1}^{m} N_k$, $n = |N|$, and $U = \cup_{k=1}^{m} U_k$. Define graph $G = (N, U)$.

Operations are executed by at most $r_{\max}$ identical workers. *Processing time* $p_j(r)$ of an operation j is a positive non-increasing function of the number of workers r assigned to this operation. Operations are non-preemptive, and if a worker starts performing an operation, he or she cannot switch to any other operation before finishing the current one. Workers cannot execute more than one operation simultaneously. The time spent by a worker to move from one station to another is negligibly small compared to any processing time of any operation. Therefore, it is assumed that any worker can move from one operation to another in zero time. Workforce assignment consists in creating a *schedule*, in which the start time of each operation, its processing time and the sequence of operations for each worker are determined. Given a schedule,

* Corresponding author.
E-mail addresses: alexandre.dolgui@imt-atlantique.fr, alexandre.dolgui@mines-nantes.fr (A. Dolgui), skovalev@inseec.com (S. Kovalev), kovalyov_my@newman.bas-net.by (M.Y. Kovalyov), ameur.soukhal@univ-tours.fr (A. Soukhal).

http://dx.doi.org/10.1016/j.ejor.2017.06.017
0377-2217/

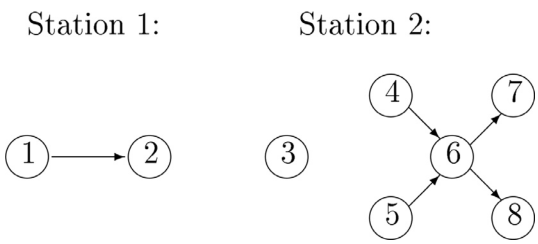

Fig. 1. Precedence graph.

Table 1
Processing times of operations.

Operations\ number of workers	1	2	3	4	5	6	7	8
1	50	10	10	9	11	24	20	
2	30		6	5	7	12	12	9
3						8	7	6

the number r_j of workers assigned to operation j, the operation *start time* S_j and the operation *completion time* C_j can be calculated for each operation j such that $C_j = S_j + p_j(r_j),\ j = 1, \dots, n$. The *makespan* of a schedule is defined as $C_{\max} = \max_{j\in N}\{C_j\}$. This value is equal to the line cycle time.

The following constraints must be satisfied in a *feasible* schedule:

- *Box constraints*. For technical reasons, the number of workers assigned to an operation must be within certain limits: $a_j \le r_j \le b_j$, where a_j and b_j are given positive integer numbers, $j = 1, \dots, n$.
- *Cycle time constraint*. In order to achieve the desired level of productivity, the line cycle time must not be exceeded: $C_{\max} \le d,\ j = 1, \dots, n$, where d is a given upper bound on the line cycle time.

The criterion of the problem, that we denote as *MinNumber*, is the minimization of the maximal number of workers employed simultaneously in the line,

$$W_{\max} = \max_{0\le t\le d}\left\{\sum_{j\in N(t)} r_j\right\},$$

where $N(t)$ is the set of operations executed at time instant t. Let $W^*_{\max}$ denote the minimal $W_{\max}$ value. Assume without loss of generality that the number of available workers is such that $r_{\max} \le \Sigma_{j\in N}\, b_j$, because otherwise we can reset $r_{\max} = \sum_{j\in N} b_j$, and that $\sum_{j\in N_k} p_j(b_j) \le d$ for $k = 1, \dots, m$ and $\max\{\max_{j\in N}\{a_j\}, \lceil\sum_{j\in N} p_j(b_j)/d\rceil\} \le r_{\max}$, because otherwise the problem *MinNumber* has no solution.

For the sake of clarity, consider an example in which the assembly line consists of two stations. There are eight operations and four available workers. Precedence graph for this example is presented in Fig. 1.

Processing times of operations depending on the number of workers are given in Table 1.

An empty entry for a given number of workers and operation j means that this number of workers is either less than a_j or greater than b_j. Note that all four workers can be used on the line, but only one, two or three of them can be used to perform the same operation. A Gantt chart illustrating a feasible schedule with an upper bound on line cycle time $d = 44$ and maximum number of workers $r_{\max} = 4$ is given in Fig. 2.

Problem *MinNumber* appeared as a sub-problem on workforce planning for an assembly line that produces three automobile engine models in the European project amePLM. Each engine model

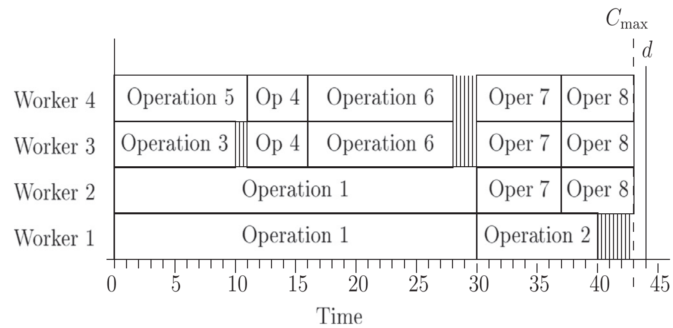

Fig. 2. A feasible schedule. Dashed rectangles represent idle times of workers.

visits stations in the same order and requires the same set of operations. Operation processing times depend on the engine model and they are also inversely proportional to the number of used workers. A detailed description of this industrial case is given in Battaïa et al. (2015). Preliminary results of our studies of the problem *MinNumber* are presented at the 15th IFAC Symposium (Dolgui, Kovalev, Kovalyov, Malyutin, & Soukhal, 2015). In Dolgui, Kovalev, Kovalyov, Malyutin, and Soukhal (2015), a special case of the problem *MinNumber* is addressed, in which graphs G_k are connected, as they are in the industrial case, though, this specificity is not precisely mentioned. The current paper studies a more general problem formulation where the graph is disconnected. Here we correct and improve the draft heuristics and MILP model suggested in Dolgui, Kovalev, Kovalyov, Malyutin, and Soukhal (2015). This paper additionally contains an updated and extended literature review, a classification of computational complexity of special cases and an extensive computational study. A particular result of this new research is that the number of workers in the real life problem of the project amePLM is decreased from 26, obtained by Battaïa et al. (2015), to 25.

The rest of the paper is organized as follows. The next section presents a literature review. In Section 3, we present the exact bisection search procedure, which consists in an iterative solution of feasibility problems *Feasible*(Q) for a given number Q of workers. This section also provides the description of a *Mixed Integer Linear Programming (MILP)* model for problems *Feasible*(Q). In Section 4, we show the relationship of the problem *Feasible*(Q) to the multi-mode project scheduling problems and multiprocessor scheduling problems and establish the computational complexity of several special cases of the problem *MinNumber* based on the relationship to the latter problems. Three constructive, two conventional and one randomized heuristics are given in Section 5. Computational studies of the heuristics and MILP model are described in Section 6. Section 7 contains a summary of the results and suggestions for future research.

2. Literature review

The assembly line design, balancing and scheduling problems are widely studied in the literature (Dolgui & Proth, 2010). The earliest studies of optimal workforce assignment problems concentrated on the two-dimensional assignment models with simple constraints. They were initiated in the 19th century and have become classics of combinatorial optimization, see the monograph of Burkard, Dell'Amico, and Martello (2009). Later on, a number of practical constraints were taken into consideration in timetabling, rostering, shift scheduling and resource constrained project scheduling models, as it was described, e.g., in Willemen (2002), Ernst, Jiang, Krishnamoorthy, and Sier (2004), Naveh, Richter, Altshuler, Gresh, and Connors (2007), Rocha, Oliveira, and Carravilla (2012), Miller (2013) and Artigues, Demassey, and Néron (2013).