Discrete Applied Mathematics 208 (2016) 123–136

Contents lists available at ScienceDirect

Discrete Applied Mathematics

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

Maximizing the robustness for simple assembly lines with fixed cycle time and limited number of workstations

André Rossi [a,*], Evgeny Gurevsky [b], Olga Battaïa [c], Alexandre Dolgui [d]

[a] LERIA, Université d'Angers, France
[b] IRCCyN, Université de Nantes, France
[c] Institut Supérieur de l'Aéronautique et de l'Espace, Toulouse, France
[d] IRCCyN, École Nationale Supérieure des Mines de Nantes, France

ARTICLE INFO

Article history:
Received 14 January 2015
Received in revised form 13 February 2016
Accepted 14 March 2016
Available online 11 April 2016

Keywords:
Assembly line design/balancing
Stability radius
Sensitivity
Robustness
Task time variability
MILP

ABSTRACT

This paper deals with an optimization problem that arises when a new paced simple assembly line has to be designed subject to a limited number of available workstations, cycle time constraint, and precedence relations between necessary assembly tasks. The studied problem, referred to as SALPB-S, consists in assigning the set of tasks to workstations so as to find the most robust line configuration (or solution) under task time variability. The robustness of solution is measured via its stability radius, i.e., as the maximal amplitude of deviations for task time nominal values that do not violate the solution feasibility. In this work, the concept of stability radius is considered for two well-known norms: ℓ_1 and ℓ_∞. For each norm, the problem is proven to be strongly $\mathcal{NP}$-hard and a mixed-integer linear program (MILP) is proposed for addressing it. To accelerate the seeking of optimal solutions, an upper bound on the stability radius is devised and integrated into the corresponding MILP. Computational results are reported on a collection of instances derived from classic benchmark data used in the literature for the Simple Assembly Line Balancing Problem.

1. Introduction

A simple assembly line is a typical flow-oriented manufacturing system (see, for example, [22,6]), which is used to fabricate a large quantity of a single type of product. It can be viewed as a set of linearly ordered workstations linked by a conveyor belt moving the product units. During manufacturing, the units pass through the workstations in the order of their location. Thus, they are sequentially injected at the beginning of the line, are transferred from one workstation to another, and are outputted at the end of the line. The workstations operate simultaneously. At each of them, its own set of tasks is repetitively carried out on the successive units.

In addition to the above, functioning simple assembly lines have also the following quite natural characteristics (see [3]):

- only one unit can be processed simultaneously at the workstation and only one workstation at a time can handle the unit;
- the tasks of any workstation are performed sequentially one by one without splitting;

* Corresponding author.
E-mail addresses: andre.rossi@univ-angers.fr (A. Rossi), evgeny.gurevsky@univ-nantes.fr (E. Gurevsky), olga.battaia@isae.fr (O. Battaïa), alexandre.dolgui@mines-nantes.fr (A. Dolgui).

http://dx.doi.org/10.1016/j.dam.2016.03.005
0166-218X/

- there is neither buffer stock nor parallel workstation and, as a consequence, the transfer of all the units situated on the line is implemented in a synchronized manner, i.e., all units are moved from their current workstations to the next ones simultaneously.

The design of such lines is an important problem, since it generally involves significant investments. This stage includes several important issues, one of which is named as *balancing problem*. In general, it consists in a partition of the set of necessary assembly tasks among workstations in an optimal way with respect to a given production goal. Mostly, supplementary restrictions can also be taken into consideration for this problem. For instance, some tasks are usually not executed in an arbitrary manner, but are subject to *precedence constraints*. The representation of these constraints is often done by a directed acyclic graph, where the set of nodes corresponds to the set of tasks and the arcs introduce a partial order over them. Thus, an arc (i, j) means that the task j cannot start before the task i is completed. The synchronized manner of the units transportation enforces that the total working time (or load) of any workstation is not greater than a certain given value determining a production rate of the line. Such a value is referred to as *cycle time*[1] and the corresponding constraint to as *cycle time constraint*. Finally, limitations of the available space for assembling may be naturally translated into restraints on the maximal number of workstations to be installed.

With regard to the objectives used, simple assembly line balancing problems (SALBP) are commonly classified into the following types (see, e.g., [22,2]): minimize the number of used workstations for a fixed cycle time (SALBP-1); minimize the cycle time with a given number of workstations (SALBP-2); and if neither the number of workstations nor the line cycle time is fixed, maximize the line efficiency (SALBP-E). The latter problem seeks a line configuration that minimizes the following expression: the number of used workstations multiplied by the working time on the most loaded one. For these problems, known to be $\mathcal{NP}$-hard (see [22], Chapter 2.2.1.5), a great number of exact and heuristic methods have been developed. Their comprehensive surveys can be found in [20,23,2].

Despite of all the attention given to SALBP, its classic formulation remains quite general and does not always reflect particular real-world situations in manufacturing. Frequently, more specific assumptions have to be taken into account. Thus, for instance, one of the important subjects to be considered is the task time variability. Indeed, as mentioned in [2], task times are often not exactly known at the preliminary design stage of the line and only their nominal (or estimated) values are used. This is caused by the following practical factors:

- for manual assembly lines, the performance of operators, implementing tasks, depends on their work rate, skill level, fatigue and motivation;
- product specifications as well as workstation characteristics may be changed during the line life cycle. It can be reasoned by a customer demand or updating the market of materials;
- various delays and micro-stoppages when tasks are executed.

Any of these events may occur in any moment of the line exploitation and can cause a costly line interruption if the cycle time is exceeded. As a consequence, to construct a robust line configuration for a long term usage, the task time variability should be anticipated at the line balancing stage. In what follows, we present an overview of the existing approaches dedicated to these aspects.

The choice of an appropriate approach for handling the processing time of tasks strongly depends on the available information dealing with its uncertainty. Thus, among the ways used in the literature, we can distinguish the following ones: *stochastic*, *fuzzy* and *robust approaches*.

For the *stochastic approach*, task processing times are represented as independent random variables with known probability distributions. As a consequence, for grouping the tasks into workstations, a particular technique supervising the cycle time constraint has to be used. Among the works dealing with this topic, those applying the so-called *chance-constrained method* are usually cited. This method consists in assigning the tasks so as to ensure that the probability of respecting the cycle time is greater, for each workstation, than a given value named as *confidence level*. For instance, such method was applied in [29,1] for a U-type assembly line balancing and in [18] for a two-sided assembly line design. In these articles, the authors use an integer linear programming (ILP) formulation of the corresponding problem that integrates the probabilistic cycle time constraint. Based on the information expressing the task times, they introduce new supplementary variables and use various linearization techniques in order to obtain again an equivalent deterministic ILP formulation for the probabilistic problem studied.

Concerning the *fuzzy approach*, the potential task processing time values are represented as a fuzzy set whose membership function describes their possibility distribution. Similarly to the stochastic case, for assigning such tasks to workstations, controlling the cycle time constraint is needed. To do this, a suitable fuzzy arithmetic and an appropriate method dedicated to comparing these fuzzy sets have to be introduced. An application of tasks with fuzzy times was presented in [28,9] for SALBP-1, in [13] for a mixed-model line balancing and in [30] for a bi-objective variant of SALBP-2.

However, it should be noted that the use of these two approaches in practice could be a difficult challenge. This is due to the fact that the available knowledge on the input data is not always sufficient to derive adequate probability or possibility

[1] In this paper, we suppose that the cycle time can be greater than or equal to the working time of the most loaded workstation.