Computers & Operations Research 65 (2016) 139–148

Contents lists available at ScienceDirect

Computers & Operations Research

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

Balancing mixed-model assembly lines using adjacent cross-training in a demand variation environment

Caijun Yang [a,c], Jie Gao [b,*]

[a] The Department of Pharmacy Administration and Clinical Pharmacy, School of Pharmacy, Xi'an Jiaotong University, Xi'an Shaanxi 710061, China
[b] School of Management, Xi'an Jiaotong University, Xi'an Shaanxi 710049, China
[c] The Center for Drug Safety and Policy Research, Xi'an Jiaotong University, Xi'an Shannxi 710061, China

ARTICLE INFO

Available online 26 July 2015

Keywords:
Mixed-model assembly line
Demand variation
Cross-training
Branch, bound and remember

ABSTRACT

The internationalization of markets and increased sophistication of consumers have led to an increase in the variety and uncertainty of products demand. It spurs the wide use of flexible production systems in producers. In this study, we aim to present a flexible mixed-model assembly line with adjacent workforce cross-training policy to account for this issue. With the adjacent cross-training, the skill of each task can be learned by two workers in adjacent stations and then task reallocation is possible when demand varies. Whenever the production volume or product mix changes, the only modification of the line is shifting some tasks to the adjacent stations where the workers can deal with. In this way, the line can achieve quick response to demand variation with high efficiency without additional trainings or great changes (such as: employment or layoff). The problem is formulated and some important properties are characterized. Then, a branch, bound and remember (BB&R) algorithm is developed to solve the problem. The efficiency and effectiveness of the proposed algorithm and this policy are tested on 450 representative instances, which are randomly generated on the basis of 25 well-known benchmark problems.

1. Introduction

The intense competition of current marketplace coupled with increased pace of technological change has led to shortening product life cycles and growing demand for customized products. Industries are urgent to provide diversified product mix efficiently [35]. Therefore, mixed model assembly lines (MALs) are widely used by manufactures to replace the single model assembly lines (SALs). MALs can produce several models simultaneously with negligible setup cost, while SALs are characterized by mass production of the single standardized product. During the design stage, the most important issue for effective utilization of a MAL is the mixed model assembly line balancing problem (MALBP). The MALBP partitions the assembly work (tasks) among stations while regarding cycle time with respect to market demands and precedence constraints among tasks for all models [9]. Generally, there are two different types of MALBP: 1) type-I, minimizing the number of workstations for a given cycle time; 2) type-II: minimizing the cycle time for a given number of workstations.

Traditionally, the MALBP is considered as a middle-term planning decision with a typical planning horizon of several months or years [10]. Most of previous researches consider that the demand is static during the whole planning horizon (such as: [38,11,27,42,25]). And they attempt to design a line with known model mix and fixed cycle time for each model (or common cycle time). However, the demand may be unstable and change frequently during the planning horizon [35]. It is essential to consider the MALBP with demand variation in the context of the internationalization of markets and increased sophistication of consumers.

To cope with demand variation, there are three approaches in the literature, including robust balancing, reconfiguration and capacity adjustment. Robust balancing employs one single line configuration for all the future demand scenarios [12], which may cause inefficiency in some scenarios. Reconfiguration means rebalancing the line when demand varies, and it needs to remove equipment and retrain workers to perform the new set of tasks [2,39]. The learning process is quite time consuming. For example, in Nissan plant at Barcelona this learning time normally is two

---

* Corresponding author. Tel./fax: +86 29 8266 4646.
E-mail addresses: yangcj@mail.xjtu.edu.cn (C. Yang), gaoj@mail.xjtu.edu.cn (J. Gao).

http://dx.doi.org/10.1016/j.cor.2015.07.007
0305-0548/© 2015 Elsevier Ltd. All rights reserved.

weeks [12]. Therefore, reconfiguration usually causes high expenses and production losses. Capacity adjustment considers part of or all the possible demand information when the line is balanced and then adjusts the line when change occurs. Simaria et al. [36] try to meet demand variation by fitting the number of workers, and Li and Gao [23] attempt to satisfy the possible demand using overtime work. And both of the two works propose the same task assignment for all the scenarios.

The purpose of this paper is to present a contribution related to a straight-paced MAL with cross-training to meet the variation in production demand, which allows task reallocation when demand varies. Cross-training has been widely employed in Just-in-time (JIT) systems to improve efficiency and to gain flexibility [17], e.g. Toyota Sewn production system [4]. With a group of cross-trained workers, workloads can be easily balanced and the impact of absenteeism will be reduced largely compared to the same number of specialized workers. However, cross-training need be done judiciously as it is costly and even counterproductive when massive cross-training is executed. For this reason, Inman et al. [19] propose chained cross-training for assembly line workers to mitigate the impact of absenteeism. Anuar and Bukchin [3] and Bukchin and Sofer [8] use work sharing to improve the balancing efficiency of assembly lines. Workers in adjacent stations are cross-trained (refer to adjacent cross-training) in chained cross-training or work sharing. With adjacent cross-training, workers are convenient to help each and the line can be balanced and adjusted easily.

In our MALBP, we use adjacent cross-training to cope with demand variation. For each task, two workers in adjacent stations are cross-trained to perform it. Consequently, tasks can be reallocated to adjacent stations where workers can deal with when demand changes. In this way, the assembly line can meet the demand requirement very quickly and efficiently without too much moving or any new employment (or layoff) or training. The demand during the planning horizon can be characterized by several possible representative demand scenarios, just as the method used in [36] and Chica et al. [12]. This MALBP with adjacent cross-training concerns how to assign tasks to each station to satisfy the demands in all scenarios. The objective is to minimize the number of stations.

The remainder of the paper is organized as follows. In Section 2, we review the related literatures. The mathematical model is formulated and some important properties are characterized in Section 3. Section 4 analyzes the lower and upper bounds of the problem. Solution method is proposed in Section 5. Computational experiments are carried out in Section 6 before concluding remarks are provided in Section 7.

2. Literature review

Due to the prevalence of MALs in the modern industries, extensive studies have been conducted for MALBPs. Comprehensive literature reviews on MALBPs can be found in Ghosh and Gagnon [16], Erel and Gokcen [15], Becker and Scholl [6], Boysen et al. [9,10].

To balance a MAL, the common approach is transforming the MALBP into a single-model assembly line balancing problem (SALBP) [37,10,18]. It usually generates a joint precedence graph and assumes that a task common to multiple models should be assigned to a single station. Generally, the cycle time in a MAL is determined on average over all the models. Hence, for some stations the processing time may be longer or shorter than the cycle time, depending on which model they are performed. For example, a model with an electrically powered sunroof will take longer time on the task “mount sunroof” than that for a model with a manually operated one. If several models with electric sunroofs are scheduled successively, work overload may occur in that station. In order to avoid operating inefficiencies like work overload or idle time when the models are sequenced, two different approaches are proposed in the literature:

1) Introduce extra objectives when balance the line. On a MAL, there are two types of task assignment variability: a) variability of assembly times of a certain model on varied stations (model variability); b) variability of assembly times of different models at a particular station (station variability) [7]. To obtain efficient operation, station variability need to be minimized properly. Thomopoulos [38] considers the objective of minimizing the maximal deviation of a station time of any model from the average station time per unit. Merengo et al. [27] try to balance both the workload allocated to each station for different models (horizontal balancing) and the average workloads of the different stations on the line (vertical balancing). Bukchin [7] develops a full experimental study to evaluate the performance of different smoothing criteria on avoiding work overload. Emde et al. [14] test a multitude of smoothing criteria and find that some objectives are superior to others.

2) Consider the balancing and sequencing problem simultaneously because of their natural interaction. Mosadegha et al. [28] jointly optimize the balancing and sequencing problem using an evolution strategy. Ozturk et al. [29] consider both balancing and model sequencing within the same formulation by two different approaches: mixed integer programming (MIP) and constraint programming (CP). However, the two problems have completely different time frames. When mid-term line balancing decision is executed, the daily model mix is not known. Therefore, a simultaneous approach is only suitable under very special conditions [10].

Some works consider other configurations. Barutcuoglu and Azizoglu [5] study the MALBP with equipment selection decisions. They assume that the cheaper equipment gives no smaller task time for all tasks. Apinar and Bayhan [1] discuss the MALBP with sequence-dependent setup times between tasks in the context of parallel workstations and zoning constraints. Kara [20], Kara and Tekin [21] and Lian et al. [24] analyze the balancing problem for U-shaped MALs. Chutima and Chimklai [13] and Kucukkoc and Zhang [22] balance two-sided MALs.

All the above works assume that demand is static. Some researchers deal with MALBP for demand variation. In the literature, robust balancing, reconfiguration and capacity adjustment are used to cope with this problem as stated in Section 1.

1) Robust balancing, which considers all the demand information in the design stage. Chica et al. [12] introduce robustness functions to measure how robust the line configuration is when the demands of the mixed products change.
2) Reconfiguration, which rebalances the line when demand varies. Altemeier et al. [2] and Yang et al. [40] rebalance the line when the demand changes. Altemeier et al. [2] present a decision support approach to simplify the problem. They use an incomplete precedence graph and introduce new numerical indicators in an integrated heuristic optimization procedure to semi-automate the reconfiguration process.
3) Capacity adjustment, which considers part of or all the future possible demand information when the line is balanced and then adjusts the line when change occurs. Simaria et al. [36] assume one worker can take charge of several stations in one scenario on a U-line. They try to meet demand variation by fitting the number of workers. To achieve this, they firstly assign tasks to stations to gratify the maximal demand rate, and then for each scenario allocate workers to stations. Li and Gao [23] try to satisfy the possible demand using overtime work, and they concern how to allocate assembly tasks to stations and determine the amount of overtime in each