



**HAL**  
open science

## Model-dependent task assignment in multi-manned mixed-model assembly lines with walking workers

S. Ehsan Hashemi-Petroodi, Simon Thevenin, Sergey Kovalev, Alexandre Dolgui

► **To cite this version:**

S. Ehsan Hashemi-Petroodi, Simon Thevenin, Sergey Kovalev, Alexandre Dolgui. Model-dependent task assignment in multi-manned mixed-model assembly lines with walking workers. *Omega*, 2022, 113, pp.102688. 10.1016/j.omega.2022.102688 . hal-03793680

**HAL Id: hal-03793680**

**<https://hal.science/hal-03793680>**

Submitted on 1 Oct 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Model-dependent task assignment in multi-manned mixed-model assembly lines with walking workers

S. Ehsan Hashemi-Petroodi<sup>a,\*</sup>, Simon Thevenin<sup>a</sup>, Sergey Kovalev<sup>b</sup>, Alexandre Dolgui<sup>a</sup>

<sup>a</sup>*IMT Atlantique, LS2N - CNRS, La Chantrerie, 4 rue Alfred Kastler, Nantes, France*

<sup>b</sup>*INSEEC Grande Ecole - INSEEC U. Research Center, 25 rue de l'Université, Lyon, France*

---

## Abstract

Due to mass customization and extensive market changes, manufacturing companies seek to enhance the flexibility and reconfigurability of their assembly lines. For instance, to adjust and adapt the line's capacity to different products and production requirements, workers may move along the stations, or the tasks may be re-assigned. This paper studies the impact of model-dependent task assignment, workforce reconfiguration, and equipment duplication in mixed-model assembly lines. The studied line is paced, and it can process different product models with different sets of tasks and precedence relations. Task and worker assignments to stations may change in each takt, and the goal is to design a line able to handle a pre-defined set of situations corresponding to different flows of products entering the line. The paper provides a new Mixed Integer Linear Programming (MILP) formulation to minimize the workforce and equipment costs in mixed-model assembly lines with model-dependent task assignment. We provide an efficient reformulation of the MILP by relying on the dualization approach commonly used in robust optimization. In addition, we employ a constructive matheuristic (CM) and a fix-and-optimize heuristic (FOH) to deal with large-scale instances. Extensive computational experiments performed with well-known benchmarks from the lit-

---

\*Corresponding author

*Email address:* `seyyed-ehsan.hashemi-petroodi@imt-atlantique.fr` (S. Ehsan Hashemi-Petroodi)

erature show that the suggested approaches perform well in terms of solution quality and computational time. In addition, the results reveal that model-dependent task assignment reduces significantly the equipment cost and the number of workers when compared to the classical mixed-model assembly lines with fixed task assignment and walking workers.

*Keywords:* Combinatorial optimization, Mixed-model assembly line, Workforce assignment, Walking worker, Model-dependent task assignment, Heuristic

---

## 1. Introduction

Mass customization and extensive changes in the market push manufacturing companies to employ mixed-model assembly lines (Kucukkoc and Zhang, 2014; Manzini et al., 2018). Short product life cycles, development of new technologies, frequent introduction of new products, and market fluctuations urge manufacturers to increase their adaptability and responsiveness. In this context, companies need to improve their flexibility (Hallgren and Olhager, 2009; Schmid et al., 2021), and turn towards robust and efficient concepts of production system organization (Battaïa et al., 2018), like reconfigurability (Koren et al., 1999). With a reconfigurable line, manufacturers can easily add, remove, or move manufacturing resources like machines, mobile robots, equipment, and workers. A manual mixed-model assembly line (MMAL) with walking workers can benefit from the concept of reconfigurability to adjust and adapt the line’s capacity to production requirements.

One of the important steps of assembly line reconfiguration is line balancing: assignment of tasks to workstations under a given criterion (minimizing takt time, number of workstations, total cost, etc.). Several restrictive assumptions are commonly made in the literature on assembly line balancing problems (Baybars, 1986; Scholl and Becker, 2006; Boysen et al., 2008; Battaïa and Dolgui, 2013): allocating only one worker to each station, producing only a single model in the line, workers and tasks are fixed, etc. However, for heavy tasks of large-size products, like in the automotive industry, assigning more than one worker to each station

is more realistic (Lopes et al., 2020). Multi-manned line balancing problems are often formulated using different restrictive assumptions (Dimitriadis, 2006; Michels et al., 2019; Becker and Scholl, 2009; Kellegöz, 2017). The movement of workers between stations adjusts stations' capacities to the production sequence (Sikora et al., 2017). The flexibility of assembly lines with multiple moving workers at workstations can be enhanced by a model-dependent task assignment to stations.

This paper deals with a multi-manned manual mixed-model assembly line balancing problem with walking workers (*MALBP-W*). The considered problem integrates the line design problem, consisting of equipment assignment to stations, and the task assignment problem, where tasks and workers are assigned to stations for a set of given product orders. Compared to the majority of studies in the literature, we consider that the production order entering the line is unknown, and we aim to design a line that can self-adjust to the products entering the line. More precisely, we provide a methodology to assign the equipment to the station and to select the number of workers so that the line meets the takt time for a prespecified set of pictures. As products enter the line, they consecutively occupy workstations creating different "pictures" of the line. By "picture" we mean the sequence of pairs station-product model that changes (product items shift towards the last station) every takt. As opposed to the concept of the line's picture, a product order can be defined as a sequence of product models entering the line whose number is not limited by the number of stations. For example, a product order can be  $(B - A - A - C - A - C - C)$ , where product B enters the line first, then two products A follow, and so on. The possible line pictures for the mentioned order of products are shown in Figure 1. Suppose that the line consists of four stations. One of the possible pictures of the line for such product order is: (station 1 - product C, station 2 - product A, station 3 - product A, station 4 - product B), see picture *Pic.4* in Figure 1. In the following takt, the picture of the line is (station 1 - product A, station 2 - product C, station 3 - product A, station 4 - product A), see picture *Pic.5* in Figure 1. The only product

B present in the product order has left the line.

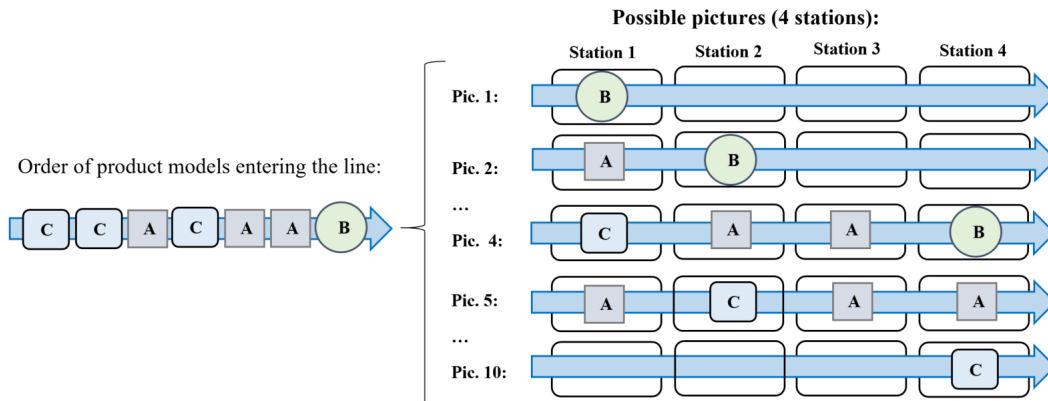


Figure 1: The possible pictures of a line for a given order of products.

The order of products entering a mixed-model line is often not controllable as it depends on the uncertainties of an upstream production step and variable demand. We aim to provide methods to design a line that can adapt to the picture of the line. In each takt, the worker can be reassigned to a different station, and the assignment of task change depending on the item present in the station. The objective is to assign the equipment to the station and to select the number of workers in order to guarantee the line can meet the takt time for a pre-specified set of pictures of the line. Therefore, the line is optimized for the worst picture of the line, since having enough resources for such picture guarantees a stable production flow delays. However, optimizing for the worst case may lead to over-conservative decisions. To control the conservatism, we assume that the user can specify the set of pictures the line may encounter by providing the maximum number of units of each item present in a picture. To define these restrictions, the user may rely on the demand mix, on historical data, or on expert knowledge.

The major contributions of this paper are fourfold. First, we formulate the problem as a robust scenario-based mixed-integer linear programming model (MILP), where the cost

related to workers and equipment is minimized for the worst picture. The line’s reconfigurability is achieved by moving workers between stations and equipment duplication at stations. To the best of our knowledge, this paper is the first to consider MALBP-W, and we provide a scenario-based MILP for the case with either model-dependent or fixed task assignment. Second, we show that the linear relaxation of the sub-problem that finds the number of workers required for the worst picture of the line yield an integer solution. As a consequence, we reformulate the MILP model using the dualization method commonly used in robust optimization. While the scenario-based MILP is unpractical because the number of scenarios/pictures is exponential in the number of stations, the reformulated MILP (RMILP) can solve practical size instances. Third, to solve large-size instances, we propose a constructive matheuristic (CM) and a fix-and-optimize heuristic (FOH). Finally, the performance of algorithms is evaluated in terms of solution quality and computational time through extensive computational experiments. In particular, we evaluate the impact of model-dependent task assignment to stations and compare it to the fixed task assignment. Our results suggest using the model-dependent task assignment because it results in a lower cost compared to a fixed task assignment. The difference in cost between these two assignments increases when the workforce cost and the problem size (the number of stations and types of products) increase.

The paper is organized as follows: Section 2 presents the literature review of the topic at hand. Section 3 formally defines the MALBP-W, and it provides a simple illustrative example as well as the mathematical model. Section 4 describes the proposed optimization approaches, including a transformed version of the mathematical model relying on the dualization approach, and two heuristics. Section 5 presents the computational results, discussion and managerial insights. The paper ends with the conclusion and future research directions in Section 6.

## 2. Literature review

This section reviews the literature related to the key concepts studied in the paper, namely, assembly line balancing, mixed-model assembly line design and balancing, fixed/walking workers in workforce assignments, fixed/dynamic task assignments. Finally, this section highlights the main contributions of the current work compared to the reviewed literature.

Assembly line balancing (task assignment) and design (resource assignment) are crucial steps for a mixed-model assembly line (MMAL) [Boysen et al. \(2009\)](#). These problems have been intensively studied ([Bukchin et al., 2002](#); [Choi, 2009](#); [Dolgui and Proth, 2010](#); [Alghazi and Kurz, 2018](#)). Workforce and task assignments are represented as decision variables whose values are needed to balance an MMAL ([Choi, 2009](#); [Gebennini et al., 2018](#); [Moreira et al., 2015](#); [Bukchin and Raviv, 2018](#); [Tiacci and Mimmi, 2018](#)). [Lee and Vairaktarakis \(1997\)](#) developed several heuristics for workforce minimization in a workforce planning problem using a MMAL. [Choi \(2009\)](#) proposed a goal programming approach to tackle a task assignment problem in a MMAL where the workers are not allowed to move between stations.

Overall, the literature on MMAL design and balancing is centered around workforce and task assignments. Workers can either be fixed at each station ([Biele and Mönch, 2018](#)) or move along the line and be re-assigned to other stations/tasks based on their skill sets or product requirements ([Naderi et al., 2019](#); [Al-Zuheri et al., 2016](#); [Şahin and Kellegöz, 2019](#)). The flexibility provided by walking workers in an assembly line improves the reconfigurability of the line ([Hashemi-Petroodi et al., 2021](#)). In several studies on workforce assignment problems, workers are allowed to move. [Battaïa et al. \(2015\)](#) assumed that workers are allowed to move between stations after finishing a task, while the processing time of a task depends on the number of workers assigned to the corresponding station. The authors built a linear programming model and constructive heuristics to solve a workforce minimization problem in a paced assembly line, motivated by an automotive industry case. Following

this study, [Dolgui et al. \(2018\)](#) developed a mixed integer linear program and constructive heuristics to solve a more general problem in which the precedence graph is disconnected. The problem was further generalized in [Delorme et al. \(2019\)](#) by including the search for an optimal product sequence. An integer linear programming model, an enumeration and a dynamic programming algorithms were proposed.

From another perspective, tasks assignment to stations can be either known ([Battaïa et al., 2015](#)), fixed ([Özcan et al., 2010](#); [Sikora et al., 2017](#)), or dynamic ([Kucukkoc and Zhang, 2014](#); [Hashemi-Petroodi et al., 2020](#)). A few publications exist on dynamic and/or model-dependent task assignment for a MMAL with fixed workers, such as parallel two-sided MMAL ([Kucukkoc and Zhang, 2014](#)), or traditional one-sided MMAL ([Choi, 2009](#)). For instance, in [Kucukkoc and Zhang \(2014\)](#), the assignment of tasks to stations depends on the production cycle, where a production cycle corresponds to a certain combination of product types present in the line. In the literature on MMAL balancing with moving workers, tasks assigned to stations are either fixed or given ([Battaïa et al., 2015](#); [Delorme et al., 2019](#); [Dolgui et al., 2018](#); [Hwang and Katayama, 2010](#)). Recently, [Hashemi-Petroodi et al. \(2020\)](#) studied the impact of dynamic task assignment on the total cost of equipment and workers. However, the concept of dynamic task assignment in [Hashemi-Petroodi et al. \(2020\)](#) is different from the task reassignment in this study. [Hashemi-Petroodi et al. \(2020\)](#) consider that tasks corresponding to any product model can be reassigned at each takt. As opposed to that study, in the present work, task reassignment is model-dependent, meaning that for each product model the assignment of tasks to stations can differ from task assignments for other product models. In addition, [Hashemi-Petroodi et al. \(2020\)](#) only present a scenario based model limited to small size instances, whereas the present work provides a set of more advanced optimization methods to solve larger instances. To the best of our knowledge, there is no existing work on MMAL balancing and design with walking workers and task reassignment. The present paper aims to fill this gap. Decisions on task and workforce



assignments have to be made simultaneously, since any change in task assignment may imply changes in workforce assignment (Cortez and Costa, 2015).

In this study, in addition to the MILP model, we present two solution techniques: matheuristic and fix-and-optimize heuristic. These approaches were rarely used in the literature on assembly line balancing. We may cite the work of Sun and Wang (2019), who applied a matheuristic approach to simple assembly line balancing problem. Lin and Ying (2016) proposed matheuristic approaches to solve a flowshop scheduling problem. Dang et al. (2021) developed a matheuristic for a parallel machine scheduling problem. Classical assembly line balancing problems are purely combinatorial as they often involve only integer or binary decision variables. As a result, meta-heuristics such as local search or population-based approaches perform well for classical assembly line balancing problems (Pereira and Álvarez-Miranda, 2018; Biele and Mönch, 2018; Dolgui et al., 2018; Li and Gao, 2014; Özcan et al., 2010; Kucukkoc and Zhang, 2014; Saif et al., 2019). In the considered MMAL with walking workers, we show that the sub-problem that consists in finding the worst picture of the line can be efficiently solved by a linear program. To leverage the efficient linear programming solver to solve this sub-problem, we propose to combine a heuristics to assign tasks to stations, and a MILP model to find the worst picture. Such heuristics have been successfully used for problems that include both binary and continuous variables. For instance, the fix-and-optimize heuristics was frequently used to efficiently solve lot-sizing problems (Chen, 2015; Lang and Shen, 2011; Sahling et al., 2009; Helber and Sahling, 2010). In this study, matheuristics and fix-and-optimize heuristics are harmoniously combined with an MILP model and provide good results.

### 3. Problem description and formulation

This section describes the multi-manned manual mixed-model assembly line balancing problem with walking workers (*MALBP – W*). In addition, it gives an illustrative example

and the scenario-based mathematical formulation for the cases with fixed ( $MALBP - W^{Fix}$ ) and model-dependent task assignment ( $MALBP - W^{Md}$ ).

### 3.1. Description of $MALBP - W$

The problem consists of designing a mixed-model manual (manned) assembly line. The line contains a set  $\mathcal{S} = \{1 \dots S\}$  of sequentially located stations. The line assembles a set  $\mathcal{I} = \{1 \dots I\}$  of product models, which flow in any order through the line. The line is paced, and the items move from one station to the next at a regular time interval  $C$ , called takt time. At each takt, there is only one item at each station. Items of different types enter the line one by one, and all of them pass through all the stations. We denote the set of all tasks as  $\mathcal{O}$ .  $\mathcal{O}$  refers to the unified set of tasks which some/all of them are common to different product models (possibly with different processing times). Each product model  $i$  requires a subset  $\mathcal{O}_i$  of the unified set of all tasks  $\mathcal{O}$ . Note that if a task is model specific, it appears only in one of the  $\mathcal{O}_i$ . There is a set  $A_i$  of precedence relations  $(o, o')$  for each product model  $i$ , where task  $o$  must be performed before task  $o'$  for this product model. Note that  $A_i$  can be the same for all product models in some use cases. The processing time  $p_{io}^l$  of task  $o$  performed on a model  $i$  depends on the number of workers  $l$  assigned to a station. In [Battaïa et al. \(2015\)](#), a real case from automotive industry was presented and the authors considered the same linear relation as in this paper. For instance, in the mentioned real case study, the processing time of assembling cylinders on different variants of engines linearly decreases depending on the number of workers. Depending on different types of tasks in different industrial use cases and how workers perform the assigned tasks, processing times may decrease not linearly. The model proposed in this work is able to tackle any type of relation between processing time and the number of workers, linear or non-linear. We assume that the tasks are non-preemptive, and that they cannot be divided into sub-tasks. There is a limit  $l_{max}$  on the number of workers assigned to the same station. In addition, each task requires a certain

equipment that has to be installed at a station. The set of equipment is denoted  $\mathcal{E}$ , and the requirements are represented with the parameter  $R_{oe}$ , whose value is equal to 1 if task  $o$  requires equipment  $e$ , and 0 otherwise. Each equipment has a certain ability to perform a set of tasks. If several tasks assigned to stations require the same type of equipment, it can be duplicated at stations (see [Askin and Zhou, 1997](#); [Tiacci and Mimmi, 2018](#), for example). The objective is to minimize the sum of equipment and workforce costs. Each equipment  $e$  has a cost  $c_{se}$  at each station  $s$ . The equipment cost can be station dependent because of the space restrictions, state of infrastructure at stations, proximity to utilities, removal of previous equipment and difficulty of installing the equipment. However, our model can also handle the special case in which equipment cost is not station dependent. As all workers are assumed to be identical and able to perform any task, the cost of workers  $\alpha$  is the same for all workers.

The line is reconfigurable in the sense that workers can move from one station to another at any takt time, thus adapting the production capacity to the current load in each station. We assume that the workers' walking times compared to task processing times are negligible. This assumption is valid in industrial situations in which walking times are sufficiently small compared to task processing times and cycle time, see for example [Battaïa et al. \(2015\)](#). In the present work, we only consider the line's design and balancing for the worst case without the prevision of the exact workers' movement. We can further assume that precise movements are decided at the control level. It is one of the possible future research directions in which a decision tool helps to select the best worker's movement to minimize the traveling distance between stations.

In this work, we consider two variants of the problem  $MALBP-W$ . In  $MALBP-W^{Fix}$ , the task assignment to stations remains fixed for all product models.  $MALBP-W^{Md}$  is similar to  $MALBP-W^{Fix}$ , but the task assignment in a station depends on the item in the station. In other words, in  $MALBP-W^{Md}$ , the task assignment may change from one

product model to another, but the task assignment is the same for several units of a given model.

The objective is to design a line that meets the takt time for the worst possible picture of the line. To define the set of pictures the line must cope with, we assume that the user can set the maximum number of units of each product model present on the line in any takt. This limit can be set based on a known ratio of demands for different products as in (Dolgui et al., 2018; Delorme et al., 2019), expert knowledge, or past data. For example, if at most one unit of product models  $A$ ,  $B$ , and  $C$  can be present at 3 stations, only one station may be occupied by model  $A$ , one station by model  $B$ , and one station by model  $C$  in each takt. However, the proposed optimization approach can also handle the non-restricted case, where the maximum number of units of each product model present on the line is infinite.

Unlike other similar works (Taube and Minner, 2018; Cortez and Costa, 2015), this study does not consider the sequencing problem while assuming an arbitrary order of products in the mixed-model line. This assumption is valid in many environments containing the mixed-model line (Becker and Scholl, 2006; Bukchin et al., 2002). Some existing works (Battaïa et al., 2015; Kucukkoc and Zhang, 2014; Delorme et al., 2019) provide methodologies to design and balance the line for a given set of production orders. As we consider fixed and model-dependent task assignments, given a set of possible orders of products, the number of workers is computed based on the pictures contained in the sequence. To give the user direct control over the line capacity, we let him/her restrict directly the possible line pictures.

Note that in the considered  $MALBP-W$ , product models can have different sets of tasks with different processing times and precedence graphs. The proposed solution approaches can also handle special cases, where product models require the same sets of tasks with the same precedence graph like in (Battaïa et al., 2015).

### 3.2. Illustrative example

This section illustrates the  $MALBP - W$  on a simple example with two sequential stations. The example illustrates the impact of model-dependent task assignment on the number of workers, equipment assigned to the stations, and total cost of the workforce and equipment, while workers can move between stations at the end of each takt.

A picture of the line is denoted as  $(1 - i, \dots S - j)$ ,  $i, j \in \mathcal{I}$ , and it determines the sequence of pairs station-product model in a certain takt. Since there are only two stations and two product models, the only possible pictures of the line (Pic.) are  $(1 - A, 2 - B)$ ,  $(1 - B, 2 - B)$ ,  $(1 - A, 2 - A)$ , and  $(1 - A, 2 - B)$ . Figure 2 shows precedence graphs and processing times for a common set of tasks  $\{1, 2, \dots, 5\}$  for the two products. In Figure 2, processing time values correspond to the task durations when they are performed by a single worker ( $p_{io}^1$ ). At most three workers can work at the same station simultaneously ( $l_{max} = 3$ ). In this small example, we assume that the values of  $p_{io}^l$  are calculated by dividing  $p_{io}^1$  by the number of workers  $l$  assigned to each task  $o$  of each product model  $i$ , but our model can handle any processing time function. Processing time calculations are marked in blue color in Figure 3.

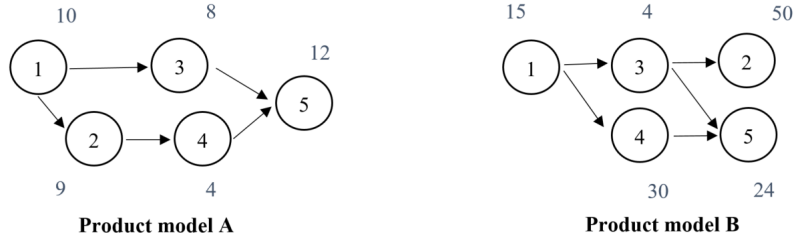


Figure 2: The precedence graphs and tasks processing times of the simple example.

Table 1 shows the compatibility between equipment and tasks, and the cost of using the equipment at stations. Each equipment is able to perform a certain set of tasks. Note that the cost of equipment increases as the ability to perform a higher number of tasks increases. The cost of a worker is  $\alpha = 500$ , and the takt time is  $C = 25$ .

Table 1: Compatibility between tasks and equipment, and the cost of equipment at each station.

	Task 1	Task 2	Task 3	Task 4	Task 5	Station 1	Station 2
<b>Equipment 1</b>	✓				✓	132	122
<b>Equipment 2</b>	✓		✓	✓	✓	172	148
<b>Equipment 3</b>	✓	✓	✓	✓	✓	224	200

Figure 3 presents the tasks, equipment and workers assigned to the stations for each picture of the line, the total processing time of each station marked in blue color, as well as the number of workers, equipment, and the total cost for the worst takt of both problems  $MALBP - W^{Fix}$  and  $MALBP - W^{Md}$ . The optimal solution of  $MALBP - W^{Fix}$  requires 6 workers and results in a total cost of 3372. The optimal solution to  $MALBP - W^{Md}$  requires only 5 workers and gives a total cost 2872.

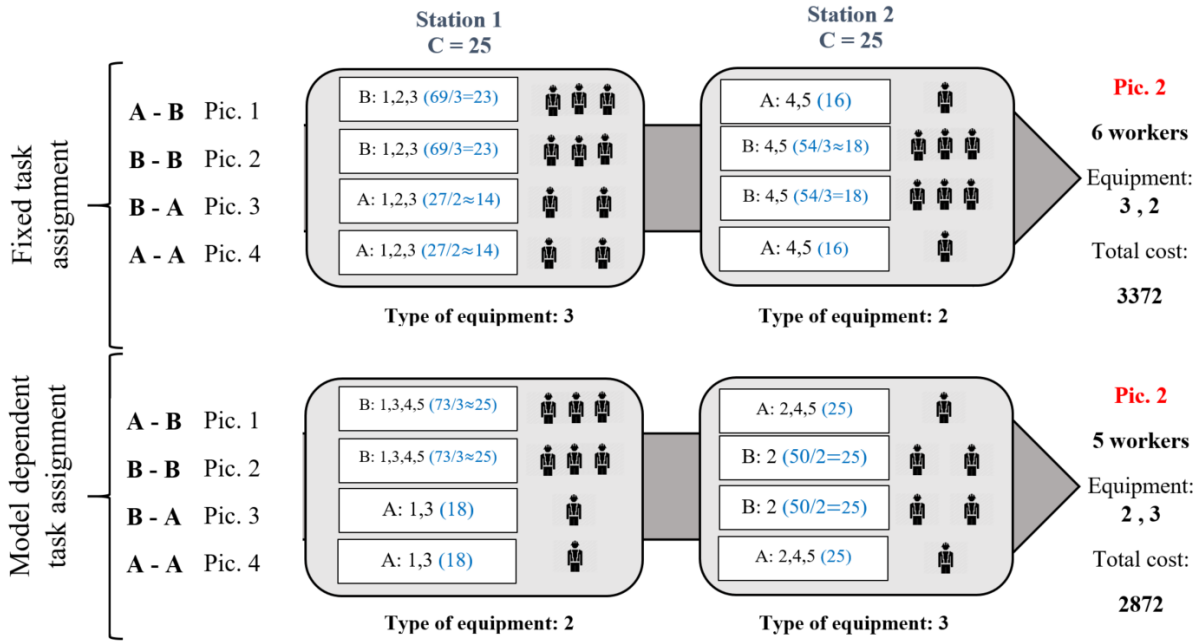


Figure 3: The optimal solution of  $MALBP - W^{Fix}$  and  $MALBP - W^{Md}$  in the simple example.

Figure 4 shows the use of the solution for a given product order. Figure 4 clarify how pictures of the line change in every takt, depending on the product order. Here, only one

order of products (B-A) is considered, where product  $B$  enter the line first, and then product  $A$  follows. Only three takts ( $\mathcal{T} = 3$ ) and therefore only three pictures are shown for the considered order:  $(1 - B, 2 - \emptyset)$ ,  $(1 - A, 2 - B)$ ,  $(1 - \emptyset, 2 - A)$ .

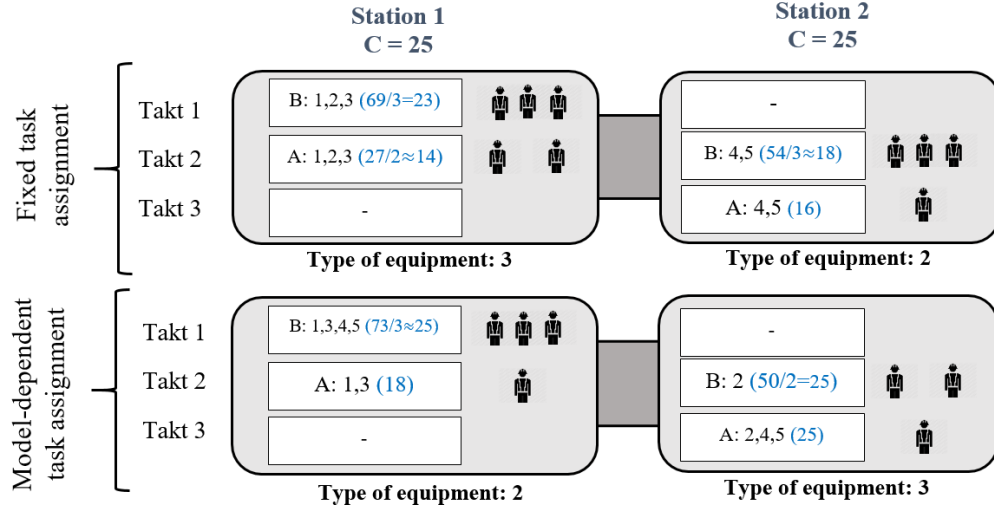


Figure 4: An example of changing pictures of the line for product order (B-A).

### 3.3. Mathematical model for fixed task assignment

This section provides the mathematical formulation  $MILP^{Fix}$  of the  $MALBP - W$  with fixed tasks. We denote  $\Omega$  as the set of all possible pictures. Note that from a given picture of the line  $\omega$ , we can determine the station  $s_i^\omega$  where product model  $i$  is processed, as well as the model  $i_s^\omega$  processed at station  $s$ . Decision variables are as follows:  $Y$  is the number of workers to hire,  $w_{se}$  is the number of equipment pieces of type  $e$  required at station  $s$ ,  $b_{sl}^\omega$  is equal to 1 if there are  $l$  workers in station  $s$  for picture  $\omega$ , and 0 otherwise;  $b_{oisl}^\omega$  is equal to 1 if there are  $l$  workers performing task  $o$  on model  $i$  at station  $s$  for picture  $\omega$ , and 0 otherwise;  $x_{soi}$  is equal to 1 if task  $o$  performed on model  $i$  is done at station  $s$ , and 0 otherwise.

The mathematical formulation of  $MILP^{Fix}$  (1)-(16) is as follows.

$$\min \quad \alpha Y + \sum_{s \in \mathcal{S}} \sum_{e \in \mathcal{E}} c_{se} w_{se} \quad (1)$$

s.t.

$$\sum_{s \in \mathcal{S}} \sum_{l=1}^{l_{max}} l b_{sl}^\omega \leq Y \quad \omega \in \Omega \quad (2)$$

$$\sum_{l=1}^{l_{max}} b_{sl}^\omega = 1 \quad \omega \in \Omega, s \in \mathcal{S} \quad (3)$$

$$\sum_{s \in \mathcal{S}} x_{soi} = 1 \quad i \in \mathcal{I}, o \in \mathcal{O}_i \quad (4)$$

$$x_{soi} = x_{soi'} \quad i, i' \in \mathcal{I}, o \in \mathcal{O}_i \cap \mathcal{O}_{i'}, s \in \mathcal{S} \quad (5)$$

$$b_{oisl}^\omega \leq x_{soi} \quad 1 \leq l \leq l_{max}, \omega \in \Omega, i \in \mathcal{I}, o \in \mathcal{O}_i, s = s_i^\omega \quad (6)$$

$$b_{oisl}^\omega \leq b_{sl}^\omega \quad 1 \leq l \leq l_{max}, \omega \in \Omega, i \in \mathcal{I}, o \in \mathcal{O}_i, s = s_i^\omega \quad (7)$$

$$b_{oisl}^\omega \geq b_{sl}^\omega + x_{soi} - 1 \quad 1 \leq l \leq l_{max}, \omega \in \Omega, i \in \mathcal{I}, o \in \mathcal{O}_i, s = s_i^\omega \quad (8)$$

$$\sum_{o \in \mathcal{O}_i} \sum_{l=1}^{l_{max}} p_{io}^l b_{oisl}^\omega \leq C \quad \omega \in \Omega, s \in \mathcal{S}, i = i_s^\omega \quad (9)$$

$$l b_{oisl}^\omega \leq \sum_{e \in \mathcal{E}} R_{oe} w_{se} \quad 1 \leq l \leq l_{max}, \omega \in \Omega, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (10)$$

$$\sum_{s \in \mathcal{S}} s x_{soi} \leq \sum_{s' \in \mathcal{S}} s' x_{s'o'i} \quad i \in \mathcal{I}, (o, o') \in A_i \quad (11)$$

$$x_{soi} \in \{0, 1\} \quad i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (12)$$

$$b_{sl}^\omega \in \{0, 1\} \quad 1 \leq l \leq l_{max}, \omega \in \Omega, s \in \mathcal{S} \quad (13)$$

$$Y \geq 0 \quad (14)$$

$$w_{se} \in \mathbb{Z}^{0+} \quad s \in \mathcal{S}, e \in \mathcal{E} \quad (15)$$

$$b_{oisl}^\omega \leq 1 \quad 1 \leq l \leq l_{max}, \omega \in \Omega, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (16)$$

The objective function (1) is to minimize the costs associated with the workers and equip-



ment, where  $\alpha$  represents the labor cost (salary plus other charges) of a worker. Constraints (2) compute the total number of workers. Constraints (3) state that a single number of workers must be chosen for each station in each picture of the line. Constraints (4) ensure that each operation is assigned to a single station in each picture. Equations (5) force the tasks to remain fixed at stations for all the product models  $i \in \mathcal{I}$ . Equations (6), (7), and (8) compute the value of  $b_{oisl}^\omega$  based on the values of  $b_{sl}^\omega$  and  $x_{soi}$ . Equations (9), (10), and (11) define the classical takt time, equipment, and precedence constraints, respectively. Constraints (10) ensure that the number of equipment pieces able to perform the operation is equal to the number of workers assigned to the station. Constraints (12) - (16) give the bounds on the decision variables.

### 3.4. Mathematical model for model-dependent task assignment

The mathematical formulation of  $MILP^{Md}$  is similar to  $MILP^{Fix}$ , but without constraints (5), because the assignment of tasks to stations can dynamically change from one product model to another. Therefore,  $MILP^{Md}$  corresponds to (1)- (4), (6) - (16).

## 4. Optimization approaches

As the number  $|\Omega|$  of pictures is naturally large, solving  $MILP^{Fix}$  and  $MILP^{Md}$  is time consuming. This section provides an efficient reformulation of the MILP, the constructive matheuristic (CM), and the fix-and-optimize heuristic (FOH). At the end of this section, we explain how  $MILP^{Fix}$  can serve as a heuristic for  $MILP^{Md}$ .

### 4.1. MILP reformulation

$MALBP - W$  can be decomposed in two sub-problems. The first sub-problem assigns tasks to a station, and computes the minimum number of workers ( $Y_{is}$ ) required to perform the tasks of product model  $i$  at station  $s$  within the takt time. The second sub-problem

computes the number  $f(Y_{11}, \dots, Y_{IS})$  of workers in the worst picture. In this context, we show that the linear relaxation of the sub-problem is integer. Consequently, we can use the dualization method commonly used in robust optimization (Alem and Morabito, 2012; Gorissen et al., 2015). Another way to tackle non-linearity could be the quadratic term linearization approach. However, it generates more constraints and leads to a higher CPU time consumption. In application to our problem, the dual model performs better. The linear program provides a linear function and the same optimal minimization objective value as in the primal maximization problem. Since dualization transforms the maximization sub-problem into a minimization problem, it can be inserted into the main problem. The mathematical formulation of the first sub-problem for  $MALBP - W^{Fix}$  is given below.

$$\min \sum_{s \in \mathcal{S}} \sum_{e \in \mathcal{E}} c_{se} w_{se} + f(Y_{11}, \dots, Y_{IS}) \quad (17)$$

s.t.

(11)

$$Y_{is} \geq \sum_{l=1}^{l_{max}} l b_{isl} \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (18)$$

$$\sum_{l=1}^{l_{max}} b_{isl} = 1 \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (19)$$

$$\sum_{s \in \mathcal{S}} x_{soi} = 1 \quad i \in \mathcal{I}, o \in \mathcal{O}_i \quad (20)$$

$$x_{soi} = x_{soi'} \quad i, i' \in \mathcal{I}, o \in \mathcal{O}_i \cap \mathcal{O}_{i'}, s \in \mathcal{S} \quad (21)$$

$$b_{oisl} \leq x_{soi} \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (22)$$

$$b_{oisl} \leq b_{isl} \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (23)$$

$$b_{oisl} \geq b_{isl} + x_{soi} - 1 \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (24)$$

$$\sum_{o \in \mathcal{O}_i} \sum_{l=1}^{l_{max}} p_{io}^l b_{oisl} \leq C \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (25)$$

$$lb_{oisl} \leq \sum_{e \in \mathcal{E}} R_{oe} w_{se} \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (26)$$

$$x_{soi} \in \{0, 1\} \quad i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (27)$$

$$b_{isl} \in \{0, 1\} \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, s \in \mathcal{S} \quad (28)$$

$$Y_{is} \geq 0 \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (29)$$

$$w_{se} \in \mathbb{Z}^{0+} \quad s \in \mathcal{S}, e \in \mathcal{E} \quad (30)$$

$$b_{oisl} \leq 1 \quad 1 \leq l \leq l_{max}, i \in \mathcal{I}, o \in \mathcal{O}_i, s \in \mathcal{S} \quad (31)$$

For this sub-problem of  $MALBP - W^{Md}$ , constraints (21) must be removed from the above mathematical formulation.

Compared to the mathematical formulation (1)-(16), the above  $MILP$  ignores the production orders,  $|\Omega|$  pictures of the line, and these elements are considered in the sub-problem.

In this paragraph we explain the second sub-problem for workforce assignment to the stations. Let  $u_i$  be the maximum number of units for each product model  $i$  in a picture of the line. A new binary variable  $f_{is}$  is introduced and it is equal to 1 if model  $i$  is assigned to station  $s$  in the worst picture, and 0 otherwise. This assignment requires  $Y_{is}$  workers to process model  $i$  in station  $s$ . The function  $f(Y_{11}, \dots, Y_{IS})$  corresponds to model (32) - (35). The objective (32) is to maximize the workforce cost to find the worst picture resulted from different product orders. Constraints (33) state that at most  $u_i$  model  $i$  are simultaneously present on the line. Constraints (34) state that each station is occupied by a single product model.

$$f(Y_{11}, \dots, Y_{IS}) = \max \sum_{i \in \mathcal{I}} \sum_{s \in \mathcal{S}} f_{is} \alpha Y_{is} \quad (32)$$

s.t.

$$\sum_{s \in \mathcal{S}} f_{is} \leq u_i \quad i \in \mathcal{I} \quad (33)$$

$$\sum_{i \in \mathcal{I}} f_{is} = 1 \quad s \in \mathcal{S} \quad (34)$$

$$f_{is} \in \{0, 1\} \quad i \in \mathcal{I}, s \in \mathcal{S} \quad (35)$$

**Theorem 4.1.** *The linear relaxation of sub-problem (32) - (35) yields a solution with integer values for variables  $f_{is}$ .*

*Proof of Theorem 4.1.* The solution of a linear program is integer if the matrix of constraints is totally uni-modular (Heller and Tompkins, 1956).

We assume  $A$  as the  $(m \times n)$  constraint matrix of a linear program which is able to be partitioned into two subsets  $A_1$  and  $A_2$ . Matrix  $A$  is totally uni-modular if it respects the following conditions (Heller and Tompkins, 1956): 1) every column of  $A$  contains at most two non-zero elements; 2)  $A$  contains only values 0, +1, or -1; 3) if a column of  $A$  consist of two non-zero elements with the same sign, then the row of one is in  $A_1$ , and the other in  $A_2$ ; 4) if a column of  $A$  consist of two non-zero elements with the opposite signs, then the rows of both are in  $A_1$ , or both in  $A_2$ . Herein, each variable  $f_{is}$ , for  $i \in \mathcal{I}$  and  $s \in \mathcal{S}$ , appears exactly twice in the set of constraints with a coefficient 1. The first appearance is in the set of constraints (33), and the second in the set of constraints (34). Therefore, the constraint matrix of the program (32) - (35) satisfies the four conditions mentioned in (Heller and Tompkins, 1956), and it is totally uni-modular.

□

**Dual programming for linearization:** Equation (32) is a quadratic function, since both the product model assignment and the number of workers are multiplied as binary and integer decision variables, respectively. As the integrality constraint can be relaxed in

(1)-(16), we can build a dual model of the sub-problem. The dual programming model is:

$$f(Y_{11}, \dots, Y_{IS}) = \min \sum_{i \in \mathcal{I}} u_i M_i + \sum_{s \in \mathcal{S}} N_s \quad (36)$$

s.t.

$$M_i + N_s \geq \alpha Y_{is} \quad i \in \mathcal{I} \quad s \in \mathcal{S} \quad (37)$$

$$M_i \geq 0 \quad i \in \mathcal{I} \quad (38)$$

Here,  $M_i$  and  $N_s$  are the corresponding dual variables of constraints (33) and (34), respectively.

Inserting the dual programming (36) - (38) into the main problem yields the reformulation of both  $MILP^{Fix}$  and  $MILP^{Md}$ , and they are respectively denoted  $RMILP^{Fix}$  (Equations (39)) and  $RMILP^{Md}$  (Equations (40)). While solving  $RMILP^{Fix}$  and  $RMILP^{Md}$  yield the optimal task and worker assignment as well as the worst case number of workers, it does not provide the worst picture of the line directly since it contains only the dual solution of (32) - (35). The worst picture of the line can be computed by solving (32) - (35) with the simplex method.

$$\min \sum_{s \in \mathcal{S}} \sum_{e \in \mathcal{E}} c_{se} w_{se} + \sum_{i \in \mathcal{I}} u_i M_i + \sum_{s \in \mathcal{S}} N_s \quad (39)$$

s.t.

$$(11)\&(18) - (31)\&(37) - (38)$$

$$\min \sum_{s \in \mathcal{S}} \sum_{e \in \mathcal{E}} c_{se} w_{se} + \sum_{i \in \mathcal{I}} u_i M_i + \sum_{s \in \mathcal{S}} N_s \quad (40)$$

s.t.

$$(11)\&(18) - (20)\&(22) - (31)\&(37) - (38)$$

#### 4.2. Constructive matheuristic (CM)

The preliminary experiments showed that task assignments are computationally difficult to perform, while  $RMILP^{Fix}$  and  $RMILP^{Md}$  can be solved in a few seconds if variables  $x_{soi}$  are fixed. Therefore, our solution method first focuses on the assignment of tasks to stations (variable  $x_{soi}$ ). Then,  $RMILP^{Fix}$  and  $RMILP^{Md}$  are solved assigning workers and equipment with fixed values of  $x_{soi}$ . This heuristic is called a constructive matheuristic (CM), since it involves both a heuristic algorithm and the reformulated mathematical model.

To calculate  $x_{soi}$  for each product model  $i$ , the heuristic starts with the assignment of the task without predecessors to the first station. If there is more than one task without predecessors, the following task is assigned according to one of the eight task selection rules described in Table 2. CM assigns tasks to the first station until the sum of the processing times reaches the takt time. However, CM considers the processing time of tasks performed by only one worker. The new takt time  $C_{new}$  is defined as follows:  $\max(C, \frac{\sum_{o \in O} p_{io}^1}{|S|})$ ,  $i \in \mathcal{I}$ . In  $MALBP - W$ , multiple workers can work at a station with respect to the takt time. However, the CM stops assigning when the sum of processing times of tasks performed by only one worker reaches the takt time. To obtain a feasible task assignment by CM when the summation of task processing times for a product model is higher than the production time (summation of the takt time for all stations), the takt time is calculated by the division of total processing time by the number of stations. The heuristic continues to assign tasks in a similar manner until the last station. Values of  $x_{soi}$  are then given to  $RMILPs$  to tackle the whole problem  $MALBP - W$ .

Algorithm 1 shows the basic steps of CM. A feasible solution is obtained in less than one second.

Table 2: Task selection rules considered in constructive matheuristic (CM)

	<b>Rules description</b>
<b>Rule 1</b>	Largest processing time (LPT).
<b>Rule 2</b>	Smallest processing time (SPT).
<b>Rule 3</b>	Largest number of successors (LNS).
<b>Rule 4</b>	Smallest number of successors (SNS).
<b>Rule 5</b>	Largest processing time & Largest number of successors (LPTLNS).
<b>Rule 6</b>	Smallest processing time & Largest number of successors (SPTLNS).
<b>Rule 7</b>	Largest processing time & Smallest number of successors (LPTSNS).
<b>Rule 8</b>	Smallest processing time & Smallest number of successors (SPTSNS).

---

**Algorithm 1** Constructive matheuristic (CM)

---

**Required:** Precedence graph for each model  $i$  ( $A_i$ ), task processing times for each model  $i$  performed by a single worker ( $p_{io}^1$ ), and takt time ( $C$ ).

**Step 1:** Compute the new takt time value ( $C_{new}$ ).

**Step 2:** Set  $s := 1$  first. Start assigning tasks to stations for each product model ( $x_{soi}$ ).

**Step 3:** For each model  $i \in \mathcal{I}$ , assign the task with no predecessor. If there is more than one task without predecessors, assign the task according to a selection rule proposed in Table 2.

**Step 4:** If  $C_{new}$  is exceeded, stop and pass to the next station.

**Step 5:** Repeat steps 3 and 4 with the same selection rule until all tasks are assigned to stations for all product models.

**Step 6:** Input the obtained values for  $x_{soi}$  to  $RMILP^{Fix}$  or  $RMILP^{Md}$  and solve. Obtain the feasible solution "sol".

---

#### 4.3. Fix-and-optimize heuristic (FOH)

The fix-and-optimize approach starts with the initial solution obtained by the CM, and it seeks to improve the task assignment. The initial solution is the best one among all solutions obtained using the rules from Table 2. In each iteration, FOH solves  $RMILP$  with most variables ( $x_{soi}$ ) fixed to their values in a current solution, and another part of variables

$(x_{soi})$  selected as a binary decision variable for re-optimization. To select this part of  $(x_{soi})$ , the tasks assigned to 2 or 3 adjacent stations become decision variables. These stations are selected randomly. Adjacent stations are selected in order to respect task precedence relations. Three rules are defined to select the stations randomly, see Table 3. The algorithm is stopped and the best current solution is saved if one of the following conditions is met: computational time has reached 1 hour; no improvement is observed after 20 iterations (in case of selecting 2 stations) or 10 iterations (in case of selecting 3 stations). These numbers are chosen based on several pre-computational experiments. Algorithm 2 provides the main steps of the FOH.

---

**Algorithm 2** Fix-and-optimize heuristic (FOH)

---

**Required:** Parameter  $x_{soi}$  & solution  $sol$  (from Algorithm 1).

**Step 1:** Set the limit for number of iterations without any improvement, with the iteration counter "*Count*". The limit "*COUNT*" is set to 20 iterations (in case of selecting 2 stations) and to 10 iterations (in case of selecting 3 stations). Start the first iteration (*Count* := 0).

**Step 2:** Select adjacent stations for optimization with a selection rule proposed in Table 3, as a set ( $S'$ ) of stations.

**Step 3:** Consider all tasks assigned to the selected stations  $s \in S'$  for each model  $i \in \mathcal{I}$ , as binary decision variables  $x_{soi}$  for optimization. Keep the remaining values of  $x_{soi}$  known.

**Step 4:** Solve  $RMILP^{Fix}$  or  $RMILP^{Md}$ , and get a new solution  $sol^{new}$

**Step 5:** If  $sol^{new}$  is better than solution  $sol$ , then set  $sol := sol^{new}$  and re-start the iterations from the first one (*Count* := 0). Do the steps 2, 3 and 4 with the same selection rule. Otherwise go to the next iteration (*Count*+ = 1) and do the steps 2, 3 and 4 with the same selection rule.

**Step 6:** The algorithm stops either when it reaches "*COUNT*" iterations without improvement (*Count* = *COUNT*), or when 1 hour of computational time has passed.

---



Table 3: Station selection rules applied in FOH

	<b>Rules description</b>
<b>Rule 1</b>	Randomly select 2 adjacent stations (2S).
<b>Rule 2</b>	Randomly select 3 adjacent stations (3S).
<b>Rule 3</b>	Start with 2S, and after 20 iteration without any improvement continue with 3S (2S3S).

#### 4.4. $Fix^h$ heuristic

Note that a solution to  $RMILP^{Fix}$  is a feasible but not optimal solution to  $RMILP^{Md}$ , because  $MALBP - W^{Fix}$  corresponds to  $MALBP - W^{Md}$  with the additional constraints (21). Constraints (21) state that the task assignment to stations must be identical for different product models.  $Fix^h$  uses  $RMILP^{Fix}$  as a heuristic to solve  $MALBP - W^{Md}$ . Therefore, this heuristic runs  $RMILP^{Fix}$  for the given time limit, and it provides a feasible solution for  $MALBP - W^{Md}$ .

## 5. Computational experiments and results

This section provides an adaptable data generation approach for  $MALBP - W$  based on benchmark data generators from the literature. It evaluates the rules used in heuristics, analyzes the performance of each optimization approach, and provides managerial insights regarding the benefits of using the model-dependent task assignment. The problems are solved using IBM ILOG CPLEX Optimization Studio V12.10. The experiments were run on an Intel(R) Core(TM) i7-8650U CPU @ 1.90GHz 2.11 GHz processor with 32 GB of RAM in MS Windows 10 Pro (64 bit) operational system. The computational time limit is set to 4 hours for each instance. The time limit for FOH is set to 1 hour.

### 5.1. Instances generation

To perform computational experiments, we extend the data generator, proposed by [Otto et al. \(2013\)](#) to the specificity of the problem in hand. Each of our instances merges  $I$  consecutive instances of [Otto et al. \(2013\)](#). For example, our first generated instance contains the

data of  $I$  first instances of [Otto et al. \(2013\)](#) and has  $I$  product models with different processing times and precedence graphs. The second instance contains the data of  $\{2 \dots I + 1\}$  first instances, and so on. There are two groups of instances, with 20 and 50 tasks, respectively. Note that the product models may have different processing times and precedence relationships between tasks. Herein, the processing time of tasks depends the number of workers, linearly, where  $p_{io}^l = \frac{p_{io}^1}{|l|}$ ,  $1 \leq l \leq l_{max}$ ,  $i \in \mathcal{I}$ , and  $o \in \mathcal{O}_i$ .

To provide an extensive analysis, eight classes of instances are considered. All these instances are solved for both fixed and model-dependent cases. [Table 4](#) shows the characteristics of each class using check-marks. Each classes of instances can be encountered in practice in different production environments. Therefore, these classes help to provide insights on the benefit of moving workers and dynamic task assignment in different production context. For example, [Battaia et al. \(2015\)](#) studied an automotive case study, where different product models have the same tasks and precedence relations but different task processing times. Some studies ([Thomopoulos, 1970](#); [Choi, 2009](#)) consider that each product model has a specific graph with specific tasks, but the graphs have a special structure allowing them to be combined into a single graph. Other studies consider separate precedence graphs for different product models ([Naderi et al., 2019](#)). Restrictions on the ratio of items in a picture are imposed at the production scheduling level in order to smooth the workload ([Respen et al., 2016](#)).

Equipment costs at each station are generated randomly with a uniform distribution in the range  $[100, 300]$ . Four different values for workers' cost are considered. Three values are chosen in comparable range to the equipment cost values which are less, within, and more than the range for equipment costs. A greatly larger workers' cost is also considered. The four values are defined as  $\alpha = \{50, 200, 500, 5000\}$ . The defined ratio between equipment and workers' cost values correspond to the data observed in ([Askin and Zhou, 1997](#)). Different number of stations and product models ( $S, I = \{3, 5, 10\}$ ) are defined. The instances' sizes

Table 4: The characteristics of different classes of instances, applied for both fix and model-dependent cases.

	The precedence graph		The set of tasks		The order of products	
	Single	Different	Single	Different	Restricted	Non-restricted
<b>Class 1</b>		✓		✓	✓	
<b>Class 2</b>		✓		✓		✓
<b>Class 3</b>	✓			✓	✓	
<b>Class 4</b>	✓			✓		✓
<b>Class 5</b>		✓	✓		✓	
<b>Class 6</b>		✓	✓			✓
<b>Class 7</b>	✓		✓		✓	
<b>Class 8</b>	✓		✓			✓

are determined by the 3-tuple  $(I, S, O)$ , where  $I$ ,  $S$ , and  $O$  represent the number of product models, stations, and tasks, respectively.

To generate the compatibility matrix,  $R_{oe}$  is set to 1 with probability  $\frac{\bar{c}_e}{\bar{c}}$  (and 0 otherwise), where  $\bar{c}_e$  is the average cost of equipment  $e$  (over all stations), and  $\bar{c}$  is the average equipment cost (over all equipment and stations).

The takt time in the instances of [Otto et al. \(2013\)](#) is set to 1000. In *MALBP – W*, several workers may perform tasks in a station. It is reasonable to consider a reduced takt time since the processing time decreases with a higher number of workers. Here, the takt time can take the following values  $\{1000, 500, 250\}$ , and we only report results for "proper", more complicated instances, i.e. feasible and with more than one worker per station. Such instances represent production cases in which the desired productivity and therefore the takt time do not allow solutions with single worker at stations. Before the determination of these instances, takt time values are tested for different sizes of instances. Three outcomes are possible: a "proper" solution with multiple workers at stations, an infeasible solution, and a solution with a single worker per station. [Table 5](#) reports the numbers of tested instances with multiple workers, infeasible and single-worker-per-station solutions for each takt time

value.

Table 5: Determination of instances with multiple workers at stations, using different takt time values.

	C = 250	C = 500	C = 1000
Solutions with multiple workers at stations	320/1600	640/1600	640/1600
Infeasible solutions	1280/1600	640/1600	0/1600
Solutions with single worker per station	0/1600	320/1600	1280/1600

**Single/different precedence graphs for product models:** [Otto et al. \(2013\)](#) considered different precedence graphs for different product models. By different graphs we mean the number of arc position changes between nodes, while the overall precedences remain fixed. We only use their precedence graph for the first product model in the cases with the same single precedence graph (see classes 3,4,7,8).

**Single/different set of tasks for product models:** having the same set of tasks for all product models as [Otto et al. \(2013\)](#) raises no issues. In order to have different sets of tasks for product models, we randomly eliminate a number of tasks, in the range  $[8, 12]$  for 20 task instances and in the range  $[20, 30]$  for 50 task instances.

**Restricted/non-restricted number of units of product models:** in the restricted case, we consider a single unit restriction of all products ( $u_i = 1$ , for all  $i \in \mathcal{I}$ ). Thus,  $\Omega$  includes all possible pictures of the line with a single unit of each product model ( $|\Omega| = I!/(I - S)!$ ). For the non-restricted case,  $u_i = S$  for all  $i \in \mathcal{I}$  ( $|\Omega| = I^S$ ).

For each size and each class, 10 instances from all instances of [Otto et al. \(2013\)](#) are randomly selected. It leads to a total number of instances equal to 1600 which are solved for both problems  $MALBP - W^{Fix}$  and  $MALBP - W^{Md}$ .

## 5.2. Analysis of the heuristics

Table 6 shows the average solution quality for problems  $MALBP - W^{Fix}$  and  $MALBP - W^{Md}$ . It provides the average gap (the percentage %) between either heuristics CM or FOH, and the best solution provided by *RMILPs* for each selection rule. The best solution

provided by *RMILPs* refers to either the optimal solution of *RMILPs* for optimally solved instances or to the best upper bound found by *RMILPs* for non-solved instances within the considered time limit. The best rule used in CM is to assign the task with Smallest Processing Time and Smallest Number of Successor (SPTSNS) (23.2% for fixed and 19.4% for model-dependent task assignment). However, the difference among the rules used in CM is not significant. The best rule used in FOH is to start by selecting 2 adjacent stations and continue by selecting 3 stations as soon as there is no improvement of the solution (2S3S), see the bold values in Table 6 (−3.2% for fixed and −7.1% for model-dependent task assignment). Note that the negative gap for FOH means that heuristics resulted a better solution than the best solution found by *RMILPs* when it reached the time limit.

Table 6: Solution quality of the heuristics.

MALBP-W	CM (%)								FOH (%)		
	LPT	SPT	LNS	SNS	LPTLNS	SPTLNS	LPTSNS	SPTSNS	2S	3S	2S3S
Fixed	25.6	23.8	23.6	24.4	23.8	23.5	23.9	<b>23.2</b>	0.8	-2.9	<b>-3.2</b>
Model-dependent	22.3	19.5	20.4	21.9	19.9	19.6	<b>19.4</b>	<b>19.4</b>	-0.9	-6.5	<b>-7.1</b>

### 5.3. Performance of the optimization approaches

This subsection evaluates the performance of optimization approaches in terms of solution quality and computational time. Table 7 shows the number of instances solved by *RMILPs*, the average optimality/integrality gap provided by CPLEX, the average gap between the solutions obtained by heuristics CM, and FOH and the best solution found by *RMILPs*. Values  $Fix^h$  show the average relative gap between the model-dependent case solution value and the fixed one. It is computed using the formula (41). A positive value of  $Fix^h$  means that a better cost was obtained in the model-dependent case. For (5,5,50) and (10,10,50)-size instances fixed case solutions provided better values than model-dependent case solutions. In fact,  $RMILP^{Md}$  reached the time limit, while  $RMILP^{Fix}$  provided better solutions. The negative gap for (5,5,50)-size instances solved by FOH and (10,10,50)-size instances solved by

CM and FOH means that *RMILP* reached the time limit, while heuristics provided better solutions. Both *RMILPs* solve small instances with 3 stations and 3 product models to optimality. The *RMILPs*, for both problems, provide the same solution as *MILPs*. For larger instances with the number of tasks equal to 50 and the number of product models and stations equal to 10, CPLEX starts running out of memory.

$$Gap = \frac{Cost(MALBP - W^{Fix}) - Cost(MALBP - W^{Md})}{Cost(MALBP - W^{Fix})} 100\% \quad (41)$$

Table 7: Solution quality of optimization approaches depending on the instances' size.  $(I, S, O)$  stands for the number of product models, stations, and tasks, respectively.

Size (I,S,O)	MALBP-W <sup>Fix</sup>				MALBP-W <sup>Md</sup>				
	RMILP		CM	FOH	RMILP		CM	FOH	Fix <sup>h</sup>
	N° solved	Opt. gap (%)	(%)	(%)	N° solved	Opt. gap (%)	(%)	(%)	(%)
(3,3,20)	320/320	0.00	38.5	0.0	320/320	0.00	40.0	0.0	5.2
(3,3,50)	320/320	0.00	29.6	0.0	320/320	0.00	28.6	0.0	2.1
(5,5,20)	320/320	0.00	37.2	1.5	305/320	0.07	36.2	1.9	10.7
(5,5,50)	298/320	0.23	25.3	1.0	112/320	8.45	15.8	-2.6	2.3
(10,10,50)	0/320	54.64	-15.2	-18.5	0/320	65.33	-24.1	-34.8	-25.9

Table 8 shows the average computational times of MILPs, RMILPs, CM, FOH,  $Fix^h$  for different instances' sizes. Applying *RMILP* significantly improves computational times compared to *MILP*. CM provides a feasible and close to optimal solution in less than 1 second, whereas FOH provides a closer to optimal solution within few minutes.  $MILP^{Fix} / RMILP^{Fix}$  are solved significantly faster than  $MILP^{Md} / RMILP^{Md}$  because the fixed case requires many fewer task assignment decisions. Moreover, it is seen that the problem is harder to solve when  $\alpha$  is less than the equipment cost ( $\alpha = 50$ ) because a lower cost of workers leads to solutions more workers in the line and thus to a higher flexibility of task scheduling.

#### 5.4. Managerial insights

This section successively evaluates the benefit of the model-dependent task assignment compared to the fixed one, the interdependence between model-dependent task assignment

Table 8: Average computational time of the optimization approaches.

Size ( <b>I,S,O</b> )	MALBP- $W^{Fix}$				MALBP- $W^{Md}$			
	MILP (s)	RMILP (s)	CM (s)	FOH (s)	MILP (s)	RMILP (s)	CM (s)	FOH (s)
<b>(3,3,20)</b>	108.8	1.2	<1	14.1	310.3	2.3	<1	33.1
<b>(3,3,50)</b>	1924.3	15.5	<1	124.4	4487.1	22.2	<1	115.7
<b>(5,5,20)</b>	-	215.8	<1	107.9	-	2720.3	<1	112.5
<b>(5,5,50)</b>	-	3680.2	<1	485.0	-	11337.8	<1	721.9
<b>(10,10,50)</b>	-	14400.0	<1	602.8	-	14400.0	<1	1590.4

and walking workers, the performance of the proposed model-dependent scenario when compared to the dynamic task assignment studied in Hashemi-Petroodi et al. (2020). Finally, we provide a methodology to compute the number of stations needed.

#### 5.4.1. Comparison of MALBP – $W^{Md}$ and MALBP – $W^{Fix}$

This sub-section compares the benefit of model-dependent task assignment over fixed task assignment. We evaluate such benefit in terms of the equipment cost, the number of workers required, and the number of equipment duplication, and we compare situations with different costs of workers and different classes of instances. The values reported in this section are either optimal solution values obtained by  $RMILP^{Fix}$  and  $RMILP^{Md}$  or approximate solution values returned by  $FOH^{Fix}$  and  $FOH^{Md}$  when the instances are not optimally solved. All instances are taken into account in this sub-section.

Table 9 shows the impact of different workers’ costs on the equipment costs, the number of duplications, and the number of workers in the line. The number of equipment duplications is calculated as  $\sum_{e \in \mathcal{E}} ND_e$ , where  $ND_e$  is the number of duplications of equipment type  $e$ , and  $ND_e = (\sum_{s \in \mathcal{S}} w_{se}) - 1$ , if  $\sum_{s \in \mathcal{S}} w_{se} \geq 2$  for each equipment type  $e$ , otherwise  $ND_e = 0$ . It can be seen that increasing the cost of workers increases the cost of equipment, but not the number of duplications, while the number of required workers decreases. As each worker requires an equipment piece in the station to perform tasks, there are more duplications when the number of workers increases, and instances with smaller worker costs  $\alpha$  lead to

more workers. There are also more duplications when the workers' cost is much larger than the equipment cost, because the total cost of equipment does not significantly impact the total cost compared to the total workers' cost.

Table 9 also shows that increasing workers' cost makes the model-dependent task assignment less profitable than the fixed one (from 8.8% to 7.5%, and 6.9% then to 6%). A model-dependent scenario requires less workers, and it leads to a lower number of equipment duplications. It provides less equipment cost when the workers' cost is comparable to the equipment cost. However, a much higher cost of workers ( $\alpha = 5000$ ) increases equipment cost in a model-dependent task assignment since it uses a more sophisticated and therefore a more expensive equipment. Although it is still much smaller than  $\alpha$ . In fact, application of fixed or model-dependent task assignments in different instances may require equipment pieces with more capabilities. Moreover, increasing the workers' cost when it is comparable with the equipment costs, results in a less number of workers in both fixed and model-dependent scenarios because of the balancing the both costs of equipment and workers. When the workers' cost is much larger ( $\alpha = 5000$ ), the model acts the same as in the case when the workers' cost is less than the cost of equipment ( $\alpha = 50$ ). The same is true for the results given in Table 10.

Table 9: The impact of the worker cost on the number of workers, equipment cost and duplication, and cost saving via  $MALBP - W^{Md}$ .

$\alpha$	MALBP- $W^{Fix}$			MALBP- $W^{Md}$			Fixed/Model-dependent
	Eq. Cost	N° Worker	N° Dup.	Eq. Cost	N° Worker	N° Dup.	Gap (%)
<b>50</b>	1585.5	14.6	8.3	1405.6	13.9	7.9	8.8
<b>200</b>	1731.9	14.1	8.3	1513.1	13.6	7.8	7.5
<b>500</b>	1718.1	13.8	8.0	1481.0	13.5	7.6	6.9
<b>5000</b>	1991.1	14.4	8.2	2077.9	13.8	7.8	6.0

Table 10 shows the influence of different classes of instances on the equipment cost, the number of workers and equipment duplications. According to these results, manufacturing companies are able to reduce the workers' cost, in both model-dependent and fixed cases,



when they: 1) consider a restricted number of product model units in the line 2) assembly products with different sets of tasks 3) assembly products with different precedence graphs between the tasks. Companies can also hire less workers using a model-dependent task assignment.

Table 10: The influence of different classes of instances on the equipment cost, the number of workers and equipment duplications.

MALBP-W	MALBP-W <sup>Fix</sup>			MALBP-W <sup>Md</sup>		
	Eq. Cost	N° Worker	N° Dup.	Eq. Cost	N° Worker	N° Dup.
<b>Restricted</b>	1723.3	12.3	8.0	1554.2	11.7	7.4
<b>Non-restricted</b>	1790.0	16.1	8.4	1684.7	17.7	8.1
<b>Same set of tasks</b>	2194.0	15.1	9.7	1903.6	14.5	9.1
<b>Different sets of tasks</b>	1319.3	13.7	6.7	1335.3	12.9	6.7
<b>Same precedence graph</b>	1765.9	14.5	8.4	1734.3	14.3	8.1
<b>Different precedence graphs</b>	1747.4	13.9	8.0	1504.4	13.1	7.4

Table 11 shows cost saving advantages of  $MALBP - W^{Md}$  over  $MALBP - W^{Fix}$  for different classes of instances and instance sizes. Model-dependent task assignment performs better than the fixed case, especially when: 1) the number of product models increases (5.2% to 10.5% for 20 tasks, and 2.1% to 8.8% and 10% for 50 tasks); 2) the user considers restrictions on the number of product model units in the line (8.1% rather than 6.5%); 3) all product models require all tasks from the same set (7.8% rather than 6.9%), and 4) products have different precedence graphs (8.6% rather than 6.1%).

Table 11: Cost saving (%) via  $MALBP - W^{Md}$  as compared to  $MALBP - W^{Fix}$ .

MALBP-W	Size (I,S,O)					Average
	(3,3,20)	(3,3,50)	(5,5,20)	(5,5,50)	(10,10,50)	
<b>Restricted</b>	5.8	2.5	12.6	9.5	10.4	8.1
<b>Non-restricted</b>	4.7	1.9	8.4	8.1	9.6	6.5
<b>Same set of tasks</b>	9.5	2.1	11.3	7.5	8.6	7.8
<b>Different sets of tasks</b>	1.0	2.3	9.7	10.1	11.4	6.9
<b>Same precedence graph</b>	3.7	1.0	8.6	8.0	9.1	6.1
<b>Different precedence graphs</b>	6.8	3.4	12.4	9.6	10.9	8.6
<b>Average</b>	5.2	2.1	10.5	8.8	10.0	

#### 5.4.2. Evaluation of the flexibility measures

This section analyzes the interdependence of the flexibility measures, such as the model-dependent task assignment scenario and walking workers. To provide further managerial insights over flexibility measures, the impact of having walking workers at stations is compared to the case with fixed workers. The comparison is made over different classes of instances as well as different cost values of workers.

The computational experiments of this sub-section are performed only for some size of instances  $((3, 3, 50), (5, 5, 20))$ , but for all classes of instances and costs of workers. We take into account only the instances solved to optimality using *RMILPs*. In order to fix workers to stations, constraints  $b_{isl} = b_{i'sl}$  for all  $i, i' \in \mathcal{I}$ ,  $s \in \mathcal{S}$ ,  $1 \leq l \leq l_{max}$ , are added to both *RMILP<sup>Fix</sup>* and *RMILP<sup>Md</sup>*. Table 12 reports the cost savings of considering each flexibility measure (the model-dependent task assignment and walking workers, as well as both of these measures) compared to the case where there is no flexibility by considering fixed task assignment and fixed workers at stations. Therefore, the three columns of Table 12 show the gap between the cost of each scenario (model-dependent task assignment with fixed workers, fixed task assignment with walking workers, and model-dependent task assignment with walking workers), respectively versus using fixed task assignment with fixed workers. Table 12 shows that model-dependent task assignment has more impact on the cost savings than walking workers (in average 5.3% rather than 4.8%). Moreover, considering both flexibility measures provides significantly large cost saving compared to considering them, separately (in average 11.3% rather than 5.3% and 4.8%).

Figure 5 also shows the impact of fixed/walking workers and fixed/model-dependent task assignment scenarios for different workers' costs. The left part of Figure 5 shows the cost savings of model-dependent task assignment compared to the fixed one as a function of the workers' cost for the cases with fixed and walking workers. The right part of Figure 5 shows

Table 12: The gap (%) between model-dependent task assignment or/and walking workers versus considering fixed task assignment and fixed workers.

MALBP-W	Model-dependent scenario	Fixed scenario	Model-dependent scenario
	& fixed workers (%)	& walking workers (%)	& walking workers (%)
<b>Restricted</b>	5.5	9.7	17.5
<b>Non-restricted</b>	5.1	0.0	5.1
<b>Same set of tasks</b>	5.8	5.0	12.1
<b>Different sets of tasks</b>	4.8	4.7	10.5
<b>Same precedence graph</b>	3.7	5.5	9.8
<b>Different precedence graphs</b>	6.9	4.2	12.8
<b>Average</b>	<b>5.3</b>	<b>4.8</b>	<b>11.3</b>

the cost savings of walking workers compared to fixed workers as a function of the workers' cost for the cases with fixed and model-dependent task assignments. Increasing the workers' cost decreases the gap between fixed and model-dependent scenarios, whereas walking workers provide more cost savings compared to the case of fixed workers. In addition, the right part of Figure 5 shows that there is a correlation between the workers' cost and the gap between the cases of fixed and walking workers. Overall, a more flexible line with walking workers and model-dependent task assignment results in larger cost saving.

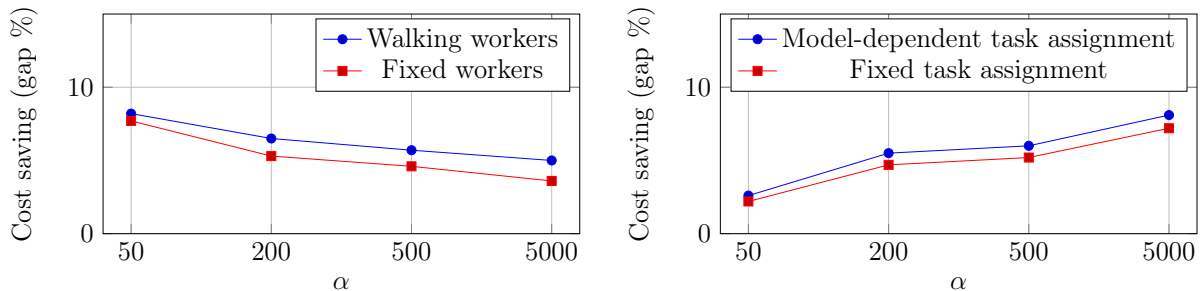


Figure 5: Evaluation of the impact of fixed/walking workers (left-side) and fixed/model-dependent task assignment scenarios (right-side) for different workers' costs.

Flexibility measures (e.g. walking workers and model-dependent task assignment) improve the workers' efficiency, but they may cause ergonomic issues for workers. [Otto and Battaia \(2017\)](#) discuss the ergonomics aspect at the planning stage of the assembly lines. The authors highlighted several interesting perspectives which require a stronger interaction between production managers, ergonomists and operations researchers. For example, increasing a flexible task assignment and frequent changes in product orders may create ergonomic risks related to an accumulated stress and fatigue. This factor should be considered by decision-makers. Next, in addition to the cost, productivity optimization for a line using walking workers represents an attractive topic. As an example, [Bischak \(1996\)](#) discovered several advantages of using walking workers related to the improvement in motivation, accountability, and responsibility of workers. In addition, our study can be integrated with the concept of open-station assembly line ([Sarker and Pan, 2001](#); [Lopes et al., 2020](#); [Ruppert et al., 2020](#)). An open-station line increases the workers' flexibility because they can work ahead of schedule and manage possible backlogs. Most of industries, especially the automotive industry, use open-station lines in which workers follow the product unit to another station if the task was not finished on time. However, in this study we do not consider specific worker's movement. Extending the current study to individual workers' movements, possible backlogs and task executions before schedule represents an interesting research prospective. For example, the flexibility caused by an open-station line can benefit from the model-dependent task assignment scenario where the takt time can be restricted by re-assigning tasks when the worker move to next station to finish his/her work.

#### *5.4.3. Evaluation of the dynamic (item-dependent) task assignment*

This sub-section compares the impacts of fixed and model-dependent assignments against the dynamic task assignment studied in ([Hashemi-Petroodi et al., 2020](#)).

In [Hashemi-Petroodi et al. \(2020\)](#), task assignment can change at each takt depending

on the product units located at stations. It appeared that the model with model-dependent task assignment provides the same results as the model with dynamic tasks assignment for the worst case/takt of the line. The cost savings (in percentage %) comparison for each pair of task assignment scenarios is given in Table 13. Only some instances (of size (3,3,20) and all classes) are given in this Table. It is worth mentioning that fixed and model-dependent assignments are more realistic and easier to implement in practice. In a dynamic task assignment, where tasks can be re-assigned in each takt (for each product unit), the problem becomes very complicated and may not be attractive for practitioners.

Table 13: Cost savings comparison for fixed, model-dependent, and dynamic (unit-dependent) (Hashemi-Petroodi et al., 2020) task assignment scenarios.

MALBP-W	Fixed/Model-dependent Gap (%)	Fixed/Dynamic Gap (%)	Model-dependent/Dynamic Gap (%)
<b>Restricted</b>	5.83	5.83	0.00
<b>Non-restricted</b>	4.76	4.76	0.00
<b>Same set of tasks</b>	9.58	9.58	0.00
<b>Different sets of tasks</b>	1.00	1.00	0.00
<b>Same precedence graph</b>	3.78	3.78	0.00
<b>Different precedence graphs</b>	6.80	6.80	0.00

#### 5.4.4. Selection of the number of station

Finally, this sub-section presents the methodology used to determine the number of stations in practice. The proposed methodology helps to analyze the relationship between the takt time, the number of stations, and the maximum number of workers required on the line.

The company must first determine the takt time value based on the estimated demand, and the maximum number of workers based on the production constraints. Then, it is possible to calculate the lower bound for the number of stations as  $LB = \min_{i \in I} \lceil \frac{\sum_{o \in O_i} p_{io}^{l_{max}}}{C} \rceil$ . By increasing the number of stations one by one, an upper bound ( $UB$ ) for the number of stations is obtained in which more than a single worker is assigned to each station. Table

14 shows the results for 10 small instances considering 3 product models and the takt time  $C = 500$ . The lower bound is 2 stations, and there are 3 infeasible instances with 2 stations, whereas all instances are feasible with 3 stations. Table 14 provides the equipment costs, and the number of workers for each value of the number of stations. Note that the equipment cost and the number of workers decrease when the number of stations increases. However, opening a station is very expensive compared to the cost of equipment and workers. Based on this information, the user can select the most suitable line design.

Table 14: Checking the feasibility for determination of the number of stations depending on a set of given input data.

S	MALBP-W <sup>Fix</sup>				MALBP-W <sup>Md</sup>			
	N° Feasible instances	Eq. Cost	N° Worker	CPU time (s)	N° Feasible instances	Eq. Cost	N° Worker	CPU time (s)
<b>LB = 2</b>	7/10	455.4	6.4	1.4	7/10	412	6	1.19
<b>3</b>	<b>10/10</b>	<b>1177.6</b>	<b>8.7</b>	<b>3.3</b>	<b>10/10</b>	<b>1123.5</b>	<b>7.5</b>	<b>8.7</b>
<b>4</b>	10/10	983.8	7.2	14.1	10/10	814.4	6.2	119.8
<b>5</b>	10/10	825.2	6.3	60.8	10/10	707.9	5.5	560.8
<b>UB = 6</b>	10/10	804.1	5.5	153.8	10/10	599.2	4.9	4974.9

## 6. Conclusion

This work focuses on a multi-manned manual mixed-model assembly line design and balancing problem with walking workers (MALBP-W) in which products can enter the line in an arbitrary order. A new scenario based mixed-integer linear programming (MILP) model is built with the criterion of minimizing the total cost of workers and equipment. The proposed MILP is robust. It minimizes the total cost for the worst sequence of pairs station-product model in a takt. The equipment pieces can be duplicated depending on tasks assigned to stations. Two scenarios for task assignment are considered: fixed and model-dependent. In a model-dependent scenario, task assignment can change depending on the product model entering the line. Various states of the line and product models are taken into account, such as having restricted/non-restricted numbers of items of the

same product model, considering same/different set(s) of tasks required for the products, and considering the same/different precedence graph(s) among tasks. The generic problem concerns a manned mixed-model assembly line, where products require different sets of tasks with different processing times, and different precedence relationships among tasks. The main challenge is to evaluate and compare the impact of fixed and model-dependent task assignments on solutions. A reformulated version of the proposed MILP is developed. A dual programming approach is used to linearize the reformulated MILP. The reformulation significantly improves the computational time. Two heuristics, a constructive matheuristic (CM) and a fix-and-optimize heuristic (FOH) show their high efficiency both in terms of solution quality and computational time. Since a fixed scenario is a special case of the model-dependent scenario, it may serve as a heuristic for model-dependent task assignment. A set of instances is generated based on the benchmark data for the simple assembly line balancing problem. The computational results, including solution quality and computational time comparisons are shown and analyzed. Several managerial insights are highlighted. In general, model-dependent task assignment proves to be less costly than fixed task assignment.

Several research perspectives can be mentioned. The present study only focuses on the line's design and balancing for the worst case and it does not go into details of workers movement. One of the promising future research directions is to account for such movements in the model. In addition, while worker movements and task re-assignments lead to significant cost savings, they may also cause a loss in productivity and a loss of workers' accountability. Therefore, the model could be extended to include the objective of minimizing the workers' movement and tasks re-assignment. A hybrid manual-automated line with humans and (collaborative-)robots with a wide range of task variety represents an interesting research direction since such production systems are likely to form the future of mass production. Considering the heterogeneity of the workforce and equipment is also a challenging aspect for a potential future study. Finally, we consider here that several workers works on the same

task in multi-manned stations, future works should consider the case where workers work on different tasks.

## Acknowledgement

The authors of this paper would like to thank the Region Pays de la Loire, France ([www.paysdelaloire.fr](http://www.paysdelaloire.fr)) for the financial support. We also would like to thank the project ASSISTANT (<https://assistant-project.eu/>) that is funded by the European Commission, under grant agreement number 101000165,H2020-ICT -38-2020, artificial intelligence for manufacturing.

## References

- Al-Zuheri, A., Luong, L., Xing, K., 2016. Developing a multi-objective genetic optimisation approach for an operational design of a manual mixed-model assembly line with walking workers. *Journal of Intelligent Manufacturing* 27, 1049–1065.
- Alem, D.J., Morabito, R., 2012. Production planning in furniture settings via robust optimization. *Computers & Operations Research* 39, 139–150.
- Alghazi, A., Kurz, M.E., 2018. Mixed model line balancing with parallel stations, zoning constraints, and ergonomics. *Constraints* 23, 123–153.
- Askin, R., Zhou, M., 1997. A parallel station heuristic for the mixed-model production line balancing problem. *International Journal of Production Research* 35, 3095–3106.
- Battaïa, O., Delorme, X., Dolgui, A., Hagemann, J., Horlemann, A., Kovalev, S., Malyutin, S., 2015. Workforce minimization for a mixed-model assembly line in the automotive industry. *International Journal of Production Economics* 170, 489–500.



- Battaia, O., Dolgui, A., 2013. A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics* 142, 259–277.
- Battaia, O., Otto, A., Sgarbossa, F., Pesch, E., 2018. Future trends in management and operation of assembly systems: from customized assembly systems to cyber-physical systems. *Omega* 78, 1–4.
- Baybars, I., 1986. A survey of exact algorithms for the simple assembly line balancing problem. *Management Science* 32, 909–932.
- Becker, C., Scholl, A., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research* 168, 694–715. doi:[10.1016/j.ejor.2004.07.023](https://doi.org/10.1016/j.ejor.2004.07.023).
- Becker, C., Scholl, A., 2009. Balancing assembly lines with variable parallel workplaces: Problem definition and effective solution procedure. *European Journal of Operational Research* 199, 359–374.
- Biele, A., Mönch, L., 2018. Hybrid approaches to optimize mixed-model assembly lines in low-volume manufacturing. *Journal of Heuristics* 24, 49–81.
- Bischak, D.P., 1996. Performance of a manufacturing module with moving workers. *IIE transactions* 28, 723–733.
- Boysen, N., Fliedner, M., Scholl, A., 2008. Assembly line balancing: Which model to use when? *International Journal of Production Economics* 111, 509–528.
- Boysen, N., Fliedner, M., Scholl, A., 2009. Production planning of mixed-model assembly lines: overview and extensions. *Production Planning and Control* 20, 455–471.

- Bukchin, J., Dar-El, E.M., Rubinovitz, J., 2002. Mixed model assembly line design in a make-to-order environment. *Computers & Industrial Engineering* 41, 405–421.
- Bukchin, Y., Raviv, T., 2018. Constraint programming for solving various assembly line balancing problems. *Omega* 78, 57–68.
- Chen, H., 2015. Fix-and-optimize and variable neighborhood search approaches for multi-level capacitated lot sizing problems. *Omega* 56, 25–36.
- Choi, G., 2009. A goal programming mixed-model line balancing for processing time and physical workload. *Computers & Industrial Engineering* 57, 395–400.
- Cortez, P.M., Costa, A.M., 2015. Sequencing mixed-model assembly lines operating with a heterogeneous workforce. *International Journal of Production Research* 53, 3419–3432.
- Dang, Q.V., van Diessen, T., Martagan, T., Adan, I., 2021. A matheuristic for parallel machine scheduling with tool replacements. *European Journal of Operational Research* 291, 640–660.
- Delorme, X., Dolgui, A., Kovalev, S., Kovalyov, M.Y., 2019. Minimizing the number of workers in a paced mixed-model assembly line. *European Journal of Operational Research* 272, 188–194.
- Dimitriadis, S.G., 2006. Assembly line balancing and group working: A heuristic procedure for workers' groups operating on the same product and workstation. *Computers & Operations Research* 33, 2757–2774.
- Dolgui, A., Kovalev, S., Kovalyov, M.Y., Malyutin, S., Soukhal, A., 2018. Optimal workforce assignment to operations of a paced assembly line. *European Journal of Operational Research* 264, 200–211.

- Dolgui, A., Proth, J.M., 2010. Supply chain engineering: useful methods and techniques. volume 539. Springer.
- Gebennini, E., Zeppetella, L., Grassi, A., Rimini, B., 2018. Optimal job assignment considering operators' walking costs and ergonomic aspects. *International Journal of Production Research* 56, 1249–1268.
- Gorissen, B.L., Yanıkoğlu, İ., den Hertog, D., 2015. A practical guide to robust optimization. *Omega* 53, 124–137.
- Hallgren, M., Olhager, J., 2009. Flexibility configurations: Empirical analysis of volume and product mix flexibility. *Omega* 37, 746–756.
- Hashemi-Petroodi, S.E., Dolgui, A., Kovalev, S., Kovalyov, M.Y., Thevenin, S., 2021. Workforce reconfiguration strategies in manufacturing systems: a state of the art. *International Journal of Production Research* 59, 6721–6744.
- Hashemi-Petroodi, S.E., Thevenin, S., Kovalev, S., Dolgui, A., 2020. The impact of dynamic tasks assignment in paced mixed-model assembly line with moving workers, in: *IFIP International Conference on Advances in Production Management Systems*, Springer. pp. 509–517.
- Helber, S., Sahling, F., 2010. A fix-and-optimize approach for the multi-level capacitated lot sizing problem. *International Journal of Production Economics* 123, 247–256.
- Heller, I., Tompkins, C.B., 1956. Linear inequalities and related systems, in: *Linear Inequalities and Related Systems.(AM-38)*, Volume 38. Princeton, NJ: *Annals of Mathematics Studies*, p. 247–254.

- Hwang, R., Katayama, H., 2010. Integrated procedure of balancing and sequencing for mixed-model assembly lines: a multi-objective evolutionary approach. *International Journal of Production Research* 48, 6417–6441.
- Kellegöz, T., 2017. Assembly line balancing problems with multi-manned stations: a new mathematical formulation and gantt based heuristic method. *Annals of Operations Research* 253, 377–404.
- Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., Van Brussel, H., 1999. Reconfigurable manufacturing systems. *Annals of the CIRP* 48, 2.
- Kucukkoc, I., Zhang, D.Z., 2014. Simultaneous balancing and sequencing of mixed-model parallel two-sided assembly lines. *International Journal of Production Research* 52, 3665–3687.
- Lang, J.C., Shen, Z.J.M., 2011. Fix-and-optimize heuristics for capacitated lot-sizing with sequence-dependent setups and substitutions. *European Journal of Operational Research* 214, 595–605.
- Lee, C.Y., Vairaktarakis, G.L., 1997. Workforce planning in mixed model assembly systems. *Operations Research* 45, 553–567.
- Li, J., Gao, J., 2014. Balancing manual mixed-model assembly lines using overtime work in a demand variation environment. *International Journal of Production Research* 52, 3552–3567.
- Lin, S.W., Ying, K.C., 2016. Optimization of makespan for no-wait flowshop scheduling problems using efficient matheuristics. *Omega* 64, 115–125.

- Lopes, T.C., Pastre, G.V., Michels, A.S., Magatão, L., 2020. Flexible multi-manned assembly line balancing problem: Model, heuristic procedure, and lower bounds for line length minimization. *Omega* 95, 102063.
- Manzini, M., Unglert, J., Gyulai, D., Colledani, M., Jauregui-Becker, J.M., Monostori, L., Urgo, M., 2018. An integrated framework for design, management and operation of reconfigurable assembly systems. *Omega* 78, 69–84.
- Michels, A.S., Lopes, T.C., Sikora, C.G.S., Magatão, L., 2019. A benders' decomposition algorithm with combinatorial cuts for the multi-manned assembly line balancing problem. *European Journal of Operational Research* 278, 796–808.
- Moreira, M.C.O., Cordeau, J.F., Costa, A.M., Laporte, G., 2015. Robust assembly line balancing with heterogeneous workers. *Computers & Industrial Engineering* 88, 254–263.
- Naderi, B., Azab, A., Borooshan, K., 2019. A realistic multi-manned five-sided mixed-model assembly line balancing and scheduling problem with moving workers and limited workspace. *International Journal of Production Research* 57, 643–661.
- Otto, A., Battaïa, O., 2017. Reducing physical ergonomic risks at assembly lines by line balancing and job rotation: A survey. *Computers & Industrial Engineering* 111, 467–480.
- Otto, A., Otto, C., Scholl, A., 2013. Systematic data generation and test design for solution algorithms on the example of salbp<sub>gen</sub> for assembly line balancing. *European Journal of Operational Research* 228, 33–45.
- Özcan, U., Çerçioğlu, H., Gökçen, H., Toklu, B., 2010. Balancing and sequencing of parallel mixed-model assembly lines. *International Journal of Production Research* 48, 5089–5113.
- Pereira, J., Álvarez-Miranda, E., 2018. An exact approach for the robust assembly line balancing problem. *Omega* 78, 85–98.

- Respen, J., Zufferey, N., Amaldi, E., 2016. Metaheuristics for a job scheduling problem with smoothing costs relevant for the car industry. *Networks* 67, 246–261.
- Ruppert, T., Dorgo, G., Abonyi, J., 2020. Fuzzy activity time-based model predictive control of open-station assembly lines. *Journal of Manufacturing Systems* 54, 12–23.
- Şahin, M., Kellegöz, T., 2019. Balancing multi-manned assembly lines with walking workers: problem definition, mathematical formulation, and an electromagnetic field optimisation algorithm. *International Journal of Production Research* 57, 6487–6505.
- Sahling, F., Buschkühl, L., Tempelmeier, H., Helber, S., 2009. Solving a multi-level capacitated lot sizing problem with multi-period setup carry-over via a fix-and-optimize heuristic. *Computers & Operations Research* 36, 2546–2553.
- Saif, U., Guan, Z., Zhang, L., Zhang, F., Wang, B., Mirza, J., 2019. Multi-objective artificial bee colony algorithm for order oriented simultaneous sequencing and balancing of multi-mixed model assembly line. *Journal of Intelligent Manufacturing* 30, 1195–1220.
- Sarker, B.R., Pan, H., 2001. Designing a mixed-model, open-station assembly line using mixed-integer programming. *Journal of the Operational Research Society* 52, 545–558.
- Schmid, N.A., Limère, V., Raa, B., 2021. Mixed model assembly line feeding with discrete location assignments and variable station space. *Omega* 102, 102286.
- Scholl, A., Becker, C., 2006. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research* 168, 666–693.
- Sikora, C.G.S., Lopes, T.C., Magatão, L., 2017. Traveling worker assembly line (re) balancing problem: Model, reduction techniques, and real case studies. *European Journal of Operational Research* 259, 949–971.

- Sun, B.q., Wang, L., 2019. A decomposition-based matheuristic for supply chain network design with assembly line balancing. *Computers & Industrial Engineering* 131, 408–417.
- Taube, F., Minner, S., 2018. Resequencing mixed-model assembly lines with restoration to customer orders. *Omega* 78, 99–111.
- Thomopoulos, N.T., 1970. Mixed model line balancing with smoothed station assignments. *Management Science* 16, 593–603.
- Tiacci, L., Mimmi, M., 2018. Integrating ergonomic risks evaluation through ocr index and balancing/sequencing decisions for mixed model stochastic asynchronous assembly lines. *Omega* 78, 112–138.