International Journal of Production Management and Engineering

https://doi.org/10.4995/ijpme.2020.12368
Received: 2019-09-22 Accepted: 2020-06-07

A mixed-integer programming model for cycle time minimization in assembly line balancing: Using rework stations for performing parallel tasks

Cavdur, F.[a1*], **Kaymaz, E.**[a2]

[a]Department of Industrial Engineering, Bursa Uludag University, Nilufer, Bursa, (Turkey).
[a1]*fatihcavdur@uludag.edu.tr,* [a2]*eliifkaymaz@gmail.com*

Abstract: In assembly lines, rework stations are generally used for reprocessing defective items. On the other hand, using rework stations for this purpose only might cause inefficient usage of the resources in this station especially in an assembly line with a low defective rate. In this study, a mixed-integer programming model for cycle time minimization is proposed by considering the use of rework stations for performing parallel tasks. By linearizing the non-linear constraint about parallel tasks using a variate transformation, the model is transformed to a linear-mixed-integer form. In addition to different defective rates, different rework station positions are also considered using the proposed model. The performance of the model is analyzed on several test problems from the related literature.

Key words: assembly line balancing, cycle time minimization, rework station, parallel tasks, mixed-integer programming.

1. Introduction

In the last station of an assembly line, it is quite frequent that some quality control procedures are also carried out in addition to the other tasks performed in the station. If a particular product does not pass the quality control check (i.e., a defective product), it is sent to the rework station for performing the necessary corrective operations to transform it into a non-defective product.

In assembly lines, rework stations are generally used for performing rework operations of defective products. On the other hand, using rework stations for this purpose only might cause inefficient usage of the resources in this station (i.e., low utilizations of operators, equipment etc.) especially in an assembly line with a low defective rate. It might be possible to increase the efficiency of the resources in a rework station by designing the station for performing standard tasks also in addition to the rework operations. In such a setting, one of the alternative utilizations of a rework station might be that it can be used for performing *parallel* tasks. In other words, some tasks might be parallelized so that they are assigned to both the rework station and a standard station where the task is performed in only one of these stations in each cycle sequentially (i.e., the task of the first product is performed in the standard station whereas the rework station is used to perform the task of the second product and so on). In this study, we propose an integer programming model considering the utilization of the rework station for parallel tasks in such a setting.

The organization of the study is as follows. In Section 2, a brief literature review on assembly line balancing and different parallelism concepts in assembly line balancing is presented. Section 3 defines the problem considered in the study. We present the details of the proposed mixed-integer programming model in Section 4 and a numerical example for illustration in Section 5. The performance of the model is tested using some sample problems from the literature as summarized in Section 6. Section 7 includes the final remarks.

To cite this article: Cavdur, F., Kaymaz, E. (2020). A mixed-integer programming model for cycle time minimization in assembly line balancing: Using rework stations for performing parallel tasks. *International Journal of Production Management and Engineering, 8*(2), 109-121. https://doi.org/10.4995/ijpme.2020.12368

2. Literature review

The concept of balancing assembly lines was introduced by Bryton in 1954 (Bryton, 1954) and the assembly line balancing (ALB) problem was defined by Salveson in 1955 (Salveson, 1955). It is possible to find different studies in the literature about the classification of ALB problems (Ghosh and Gagnon, 1989; Sivasankaran and Shahabudeen, 2014; Boysen et al., 2007). The study of Ghosh and Gagnon (1989) classifies ALB problems as single-model and multi/mixed-model ALB problems. A single type of item is produced in a single-model assembly line whereas more than different types of items are produced in a multi/mixed-model assembly line. We can make another classification of ALB problem as the problems with deterministic and stochastic task times. An assembly line balancing problem with the following assumptions is defined as the simple assembly line balancing problem (SALBP) (Baybars, 1986):

- All input parameters are known with certainty.
- A task cannot be split among two or more stations.
- The tasks cannot be processed in arbitrary sequences due to technological precedence requirements.
- All tasks must be processed.
- All stations are capable of processing any tasks.
- Task times are independent of the station at which they are performed and of the preceding or following tasks.
- Any task can be processed at any station.
- Assembly line is considered to be serial with no feeder or parallel sub-assembly lines.
- Assembly system is assumed to be designed for a unique model of a single product.
- Cycle time is given and fixed (i.e., SALBP-1).
- Number of stations is given and fixed (i.e., SALBP-2).

Some assumptions about the SALBP are rather restrictive compared to real-life assembly line systems resulting with the increasing number of studies on generalized assembly line balancing problems (GALBP) including various constraints and problem features such as parallel stations and parallel tasks. Comprehensive studies regarding the SALBP and GALBP are presented by Becker and Scholl (2006) and Scholl and Becker (2006).

In addition to the categorization of the SALP as SALBP-1 and SALBP-2 defined with the aforementioned assumptions, it can be further generalized with respect to the objective function of the problem and divided into four categories as SALBP-1, SALBP-2, SALBP-E and SALBP-F where the recently added categories (i.e., SALBP-E and SALBP-F) considers both the cycle time and the number of stations at the same time for assembly line balancing, differently from SALBP-1 where the cycle time is given and fixed and SALBP-2 where the number of stations is given and fixed (School and Becker, 2006; Wei and Chao, 2011).

Although assembly lines are usually classified as straight and U-shaped lines with respect to their designs, some other versions such as parallel lines and two-sided lines are also considered (Gokcen et al., 2006; Ozcan and Toklu, 2010; Kara et al., 2011). A straight ALB problem is considered in this study.

Various approaches are used for solving ALB problems. Some of these are exact methods such as branch-and-bound algorithm and dynamic programming. A comprehensive review of these is presented in the study of Boysen et al. (2007). Other approximate methods include various heuristics developed for some specific ALB problems and meta-heuristics used for many different types of problems (Battaia and Dolgui, 2013). Some examples of meta-heuristic approaches frequently used in the literature are genetic algorithms (Anderson and Ferris, 1994), ant colony optimization (Sabuncuoglu et al., 2009), simulated annealing (Cercioglu et al., 2009) and tabu search (Suwannarongsri and Puangdownreong, 2008).

It is noted that one of the factors affecting the cycle time of an assembly line is about the parallelism concept that can be classified into different categories such as paralleling assembly lines (Suer 1998; Gokcen et al., 2006), workstations (Bard, 1989; Askin and Zhou, 1997, Tiacci et al., 2006, Simaria and Vilarinho., 2001), tasks (Pinto et al., 1975; Kaplan, 2004; Kazemi et al., 2011) and works (Bartholdi, 1993; Kim et al., 2000; Lee et al., 2001) considering the previous studies in the literature. Note that paralleling tasks as considered in this study is defined as the assignment of a task to more than one workstation. We also note the difference between our study and some other studies in which it is possible to divide tasks into smaller units and assign any of these units (not the task itself) to different stations whereas a task, as a whole only, can be assigned to more than one station in our study due to the assumption that tasks cannot be divided into smaller units. In other words, a task can be assigned to more than one station (a standard workstation and the rework station, in our case) where consecutive tasks are performed in the alternative workstations sequentially (i.e., 1st task in the standard station, 2nd task in the rework station, 3rd task in the standard station and so on) to balance the workloads of both the corresponding standard workstation and the rework station. We also note that paralleling workstations gains more interest from the researchers compared to paralleling tasks considering the previous studies in the literature focusing on both concepts. Although some of the previous studies consider parallel tasks (Pinto et al., 1975; Kaplan, 2004; Kazemi et al., 2011), according to the best of our knowledge, none of them conceptualized and formulated the utilization of the rework station for parallel task assignment constituting the main contribution of our study.

Balancing an assembly line considering task times only might cause extreme workload on some operators which is one of the main causes of occupational accidents (Baykasoglu and Akyol, 2014; Mutlu and Ozgormus, 2012). We note an increasing interest in the consideration of ergonomic factors in assembly line balancing is in recent years. Guner and Hasgul (2012), for instance, propose an integer programming model considering ergonomic factors. In another study, Efe et al. (2014) consider Assembly Line Worker Assignment and Balancing Problem (ALWABP) focusing on the workload differences due to the ages and genders of operators. In the study of Kara et al., (2014), a model is proposed for integrating some ergonomic factors into assembly line balancing.

We finalize this section by noting some remedial actions from the related literature as these studies include some similarities to ours in that some schema (i.e., policy) is proposed to improve some metrics (i.e., cycle time minimization) of an assembly line. Some examples of such remedial actions are stopping the line (Silverman and Carter, 1986), offline repair (Gokcen and Baykoc, 1999; Kottas and Lau, 1976), hybrid lines (Lau and Shtub, 1987) and multiple manning (Shtub, 1984). Among these remedial actions, the most commonly used are stopping the line and offline repair (Altekin et al., 2016). The first one can be defined as stopping the assembly line to complete the missing tasks if the tasks assigned to a station exceed the cycle time of the total operation duration whereas the offline repair remedial action is used for unfinished tasks at the end of the cycle. As the reader might note although the idea of the offline repair remedial action has some conceptual similarity to the one proposed in our study, the methodologies are totally different as the offline repair remedial action is used for unfinished tasks at the end of the cycle to improve the line balance whereas we consider assigning parallel tasks to the rework station for the same purpose which constitutes the main contribution of our work since it is not considered in none of the aforementioned studies as well as the other related papers accessible to us for review.

3. Problem description

The utilizations the resources in the rework station might vary according to the defective rate of the assembly line. A low defective rate causes inefficient uses of the resources in the rework station. In such cases, it might be advantageous to use the rework station not only for rework operations but also some of the other standard tasks. By using the rework station for this purpose, some of the tasks performed on other standard workstations can also be assigned to the rework station as parallel tasks to be performed on both the corresponding standard workstations as well as the rework station. In other words, a specialized version of parallel task assignment is considered in this study where a parallel task is assigned to both the rework station and a standard station where the task is performed in only one of these stations in each cycle sequentially (i.e., the task of the first product is performed in the standard station whereas the rework station is used to perform the task of the second product and so on). Utilization of the rework station for parallel tasks might contribute to the minimization of the cycle time of the assembly line. On the other hand, the defective rate of the assembly line must be considered while assigning standard tasks to the rework station since a low (high) defective rate results in more (less) assignments. In this study, we consider three different defective rates.

In addition to the defective rate of the assembly line, the position in which the rework station is located is also considered in this study. Since precedence relations of the tasks change the order in which they are performed, the number of tasks that can be assigned to the rework station can vary depending on the position of the rework station. It might be possible to improve the cycle time by changing the position of the rework station. Within the scope of the study, three different alternative designs are created where the rework station is located in the last three station positions as illustrated in Figure 1, Figure 2 and Figure 3, respectively where the flows to the rework station are colored as red for defective products and blue for the standard products (i.e., parallel tasks).

In Figure 1, the rework station is in last station position (i.e., the (n+1)th station position) in an assembly line including n standard workstations. In a similar setting in Figure 2, the rework station is in the nth station position and it serves as a workstation where both the corrective operations for the defective products coming from the last station of the assembly line as well as the parallel tasks

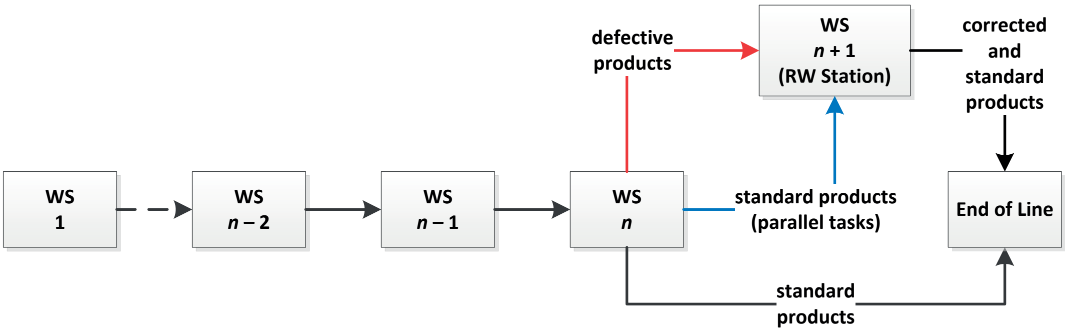

Figure 1. Positioning the rework station as the ($n+1$)th station.

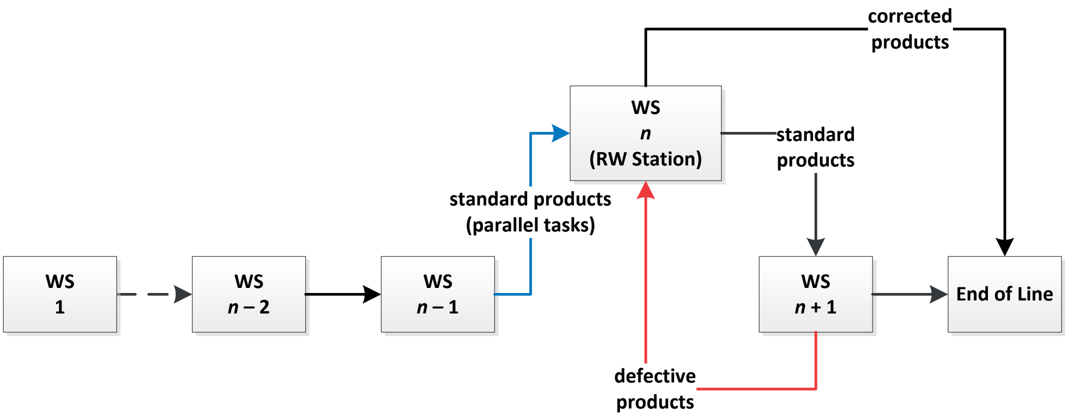

Figure 2. Positioning rework station as the nth station.

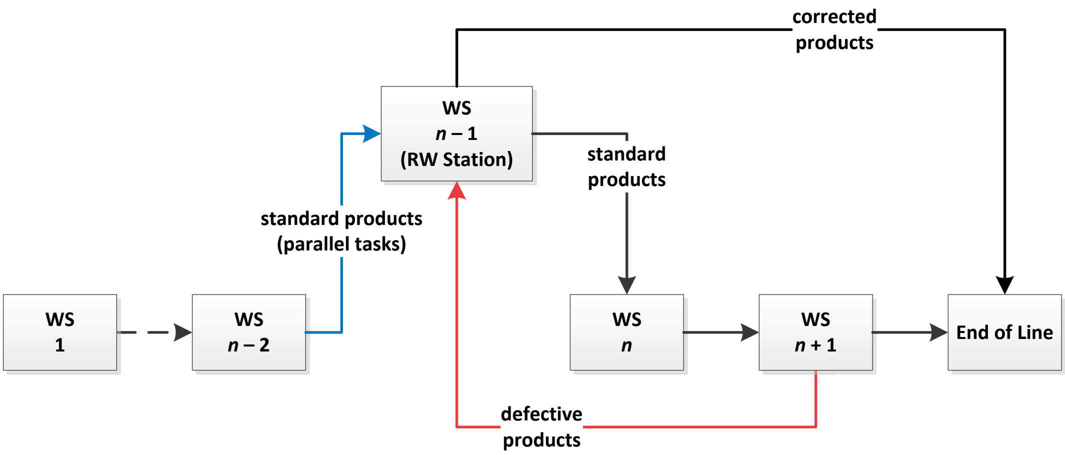

Figure 3. Positioning rework station as the ($n-1$)th station.

for the products coming from the previous station (i.e., $(n-1)$th station) are performed. In Figure 3, the rework station is located in the $(n-1)$th station position where it serves similarly at its new position.

In addition to the aforementioned positions, the rework station can be moved to the previous station positions; however, since the rework station is primarily used as a station where defective products from the last station are corrected, moving away from the last station position increases the transportation times and distances of the defective products requiring corrections. On the other hand, instead of being located at the last station position, the number of potential tasks that can be assigned to the rework station can be increased by placing the rework station in positions near the last station (such as the $(n-1)$th station, the $(n-2)$th station positions) depending on the precedence relations constraints.

It is noted from the foregoing discussion that when the position of the rework station moves towards to the first station position, we might have more flexibility in assigning tasks to the rework station depending on the precedence relations, but the transportation times/distances of the corrective operations increase. On the contrary, if the rework station position moves towards to the last station position, the flexibility might be lost in task assignments, but transportation times/distances of the corrective operations decrease in return.

4. Methodology

This section details the integer programming model developed to minimize the cycle time for the problem described in the previous pages.

Indexes:

i	tasks, $i=1,\ldots,m$
j	workstations $j=1,\ldots,n$

Parameters:

m	total number of tasks
n	total number of workstations
t_i	processing time for task i
p_{ik}	precedence relation matrix element, equals 1 if task i is predecessor of task k and 0 otherwise
r	rework station position
α	penalty for parallel task assignment
β	defective rate coefficient, $1 \le \beta \le 1.75$
λ	scaling factor for the objective function terms
γ	penalty used to limit the number of jobs that can be assigned to the rework station

Variables:

c	cycle time
x_{ij}	equals 1 if task is assigned to station j; 0 otherwise
y_i	equals 1 if task i is assigned to the rework station; 0 otherwise
z_{ij}	equals 1 if task i is assigned to both the rework station and station j; 0 otherwise

Objective Function:

$$\min z = \lambda c + (1-\lambda)\left(\sum_{i=1}^{m} \alpha y_i\right) \tag{1}$$

Constraints:

$$\le \sum_{j=1}^{n+1} x_{ij} \le 2, \ \ \forall i \tag{2}$$

$$x_{kj} \le 1 - x_{iq}, \ \ \forall i,j,k,q: p_{ik} = 1; \forall q \ge j+1 \tag{3}$$

$$\sum_{i=1}^{m}\left(\frac{t_i}{2}\right) x_{ij} y_i + \sum_{i=1}^{m} t_i (1-y_i) x_{ij} \le c, \qquad \forall j \tag{4}$$

$$\beta^{\gamma}\left(\sum_{i=1}^{m}\left(\frac{t_i}{2}\right) x_{ij} y_i\right) \le c, \ \ \forall j = r \tag{5}$$

$$x_{ij} = y_i, \ \ \forall i; j = r \tag{6}$$

$$y_i = \sum_{j=1}^{n+1} x_{ij} - 1, \ \ \forall i \tag{7}$$

$$x_{ij} \in \{0,1\}, \ \ \forall i,j \tag{8}$$

$$y_i \in \{0,1\}, \ \ \forall i \tag{9}$$

$$c \ge 0 \tag{10}$$

The objective function in Equation (1) includes the weighted sum of the cycle time and the total number of parallel tasks. In this expression, α is the penalty for each parallel task and $\lambda \in [0,1]$ is defined as the scaling factor between the objective function components. In order to emphasize the effects of defective rate and rework station position, this study has been carried out with the assumption that there is no negative effects of parallel task assignment (i.e., for $\alpha=0$ and $\lambda=1$) and the effects of these parameters similarly can be analyzed in future studies. The constraint given by Equation (2) ensures that tasks are assigned to at most two stations and at least one station. In other words, if a task is assigned to the rework station, (i.e., if it is a parallel task), it is then assigned to another standard station (i.e., the task is assigned to two stations). If it is not a parallel task, it can only be assigned to one station. The constraint given in Equation (3) indicates the precedence relations between tasks. It is ensured with Equation (4) that the total processing times of the tasks assigned to a station do not exceed the cycle time.

Since parallel tasks are assigned to two stations at the same time, half of the total processing times of the tasks is taken into consideration when calculating the cycle time. Similarly, the tasks that are assigned to the rework station are limited depending on the defective rate of the assembly line using Equation (5) where defective rate coefficient $\beta \in [1,1.75]$ in the equation reserves some time for rework operations (i.e., corrective operations) where the extreme values (1 and 2) represent the cases in which no rework operations are performed at all and the rework station only performs rework operations. The other parameter γ is defined as the penalty used to limit the number of tasks that can be assigned to the rework station for various reasons, such as ergonomic factors due the rework station position. It might be desired that the number of jobs assigned to the rework station might be limited in the case that the rework station moves towards to the first station position by defining $\gamma=n/r$ as the ratio of the total number of stations to the rework station position. It is noted that with β^{γ} factor, the number of tasks assigned to the rework station can be limited beyond the defective rate since $\gamma=n/r$ parameter takes larger values as the rework station moves towards to the first station position. In the context of this study, all computations are performed with the assumption of no effects of assigning parallel tasks or moving the rework station (i.e., $\gamma=1$). A detailed sensitivity analysis can be performed in future studies to examine the effects of these parameters.

Equation (6) and Equation (7) show the relationships between the corresponding variables by ensuring that if a task is assigned to the rework station, it must be a parallel task assigned to another standard workstation. Variable definitions are given by Equation (8), Equation (9), and Equation (10). Note that the constraints in Equation (4) and Equation (5) are nonlinear constraints due to the term of $x_{ij}y_i$, for all i and j. We however note that it can be easily linearized since both variables are binary and a linear model can be obtained by introducing the new variables defined as $z_{ij}=x_{ij}y_i$, for all i and j, as shown in Equation (11), Equation (12) and Equation (13).

$$z_{ij} \geq x_{ij}+y_i-1,\ \forall\ i,j \tag{11}$$

$$z_{ij} \leq x_{ij},\ \forall\ i,j \tag{12}$$

$$z_{ij} \leq y_i,\ \forall\ i,j \tag{13}$$

5. Numerical example

We illustrate the proposed model using the Jackson test problem with 11 tasks and three and four stations from the literature. The precedence relations the Jackson sample together with the processing times of the tasks is shown in Figure 4. We solve the problem for three different defective rates and rework station positions as detailed in the previous section. Note that the rework station is added as an additional station to the original problem. In other words, in the three-station version of the Jackson sample, the rework station added as the fourth station, and thus, three setting considered are the ones where the rework station is located in the second, third and fourth station positions. Similarly, the model is solved for three different defective rates (i.e., β=1.25, β=1.5, and β=1.75) in addition to the case of zero defective production (i.e., β=1.25) in order to show the effect of the defective rate. The results are given in Table 1.

In addition, the results obtained for different rework station positions (for $r=4$, $r=3$ and, $r=2$ respectively) are shown in Figure 5, Figure 6 and Figure 7 for a defective rate coefficient of β=1.25 where we represent the standard tasks in white, parallel tasks in light gray and rework operations in dark gray. The effect of changing the position of the rework station in the figures is clearly observed. When the rework station is in the last station position, the number of potential tasks that can be assigned to the rework station is more limited depending on precedence relations and only a parallel task (with task number 11) can be assigned as shown in Figure 5. Accordingly, it is seen that the total time of the tasks performed in the rework station is considerably less than the cycle time. On the other hand, by changing the position of the rework station by locating it in the third and second positions, it becomes possible to assign more tasks to the rework station, which makes it possible to achieve more improvements in the cycle time as seen in Figure 6 and Figure 7, respectively.

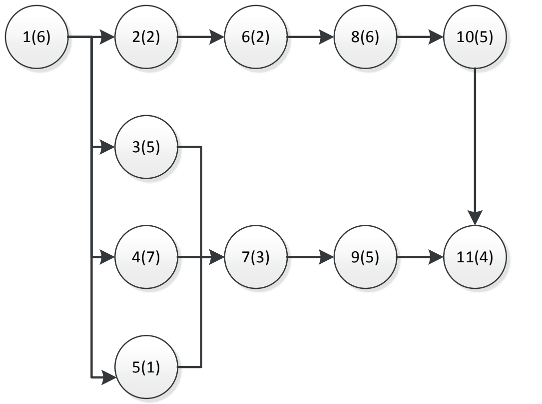

Figure 4. Precedence diagram and process times for Jackson data set.

The results in Table 1 show how the optimal solution changes depending on the position of the rework station and the defective rate of the assembly line. It is noted, for a 25% of defective rate (i.e.,) for instance, that when the rework station is in the second and third workstation positions, the cycle time is 12.5 time units whereas it is 15 time units when the rework station is in the fourth station position at the end of the assembly line. Similarly, increases in defective rate, also increases the cycle time due to more rework operations performed in the station. In addition to the Jackson sample, the test problems of Mitchells, Sawyer and Kilbrid are also solved for different number of stations to show the performance of the model and presented in the next section.

6. Computational results

In this section, the performance of the model is tested using various ALB problem test samples (i.e., *β*=1.00 the test problems of Jackson, Mitchells, Heskiaoff, Sawyer, Kilbrid) from the literature. The results are summarized in Table 2. The data about the test problems used in the study can be accessed at http://assembly-line-balancing.mansci.de

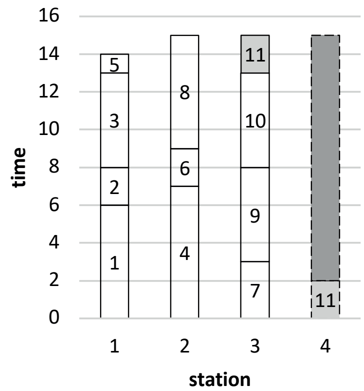

Figure 5. Assignments for *β*=1.25 and *r*=4.

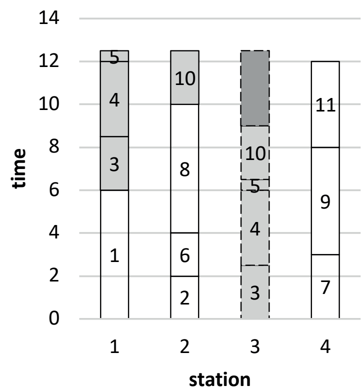

Figure 6. Assignments for *β*=1.25 and *r*=3.

Table 1. Results for the Jackson sample.

NS	CT	ST	RW Station Position	*β*=1.00		*β*=1.25		*β*=1.50		*β*=1.75	
				CT	ST	CT	ST	CT	ST	CT	ST
3+1	16.00	0.23	2	12.00	0.27	12.50	0.03	13.00	0.30	13.00	0.10
			3	12.50	0.22	12.50	0.06	13.00	0.27	13.00	0.18
			4	15.00	0.17	15.00	0.03	15.00	0.20	15.00	0.03
4+1	12.00	0.11	3	9.50	0.25	10.00	0.13	10.50	0.28	10.50	0.10
			4	9.50	0.26	10.00	0.10	10.50	0.28	10.50	0.08
			5	11.00	0.11	11.00	0.03	11.00	0.11	11.00	0.02

NS (Number of Stations), CT (Cycle Time-time unit), ST (Solution Time-second).

Test problems are solved using GUROBI Optimizer in Mathematical Programming Language (MPL) on a personal computer (Intel (R) Core i7-7500 CPU 2.70 GHz 2.90 GHz). As in the previous section, all computations are performed for $\alpha=0$, $\lambda=1$ and $\gamma=1$ in order to emphasize the effects of defective rate and rework station position. In future studies, the effects of these parameters can be similarly investigated. The results are summarized in Table 2 where two different number-of-stations combinations from the study of Ugurdag et al., (1997) are taken into consideration.

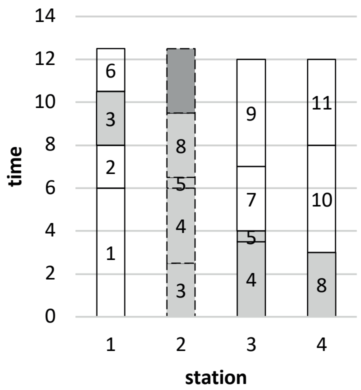

Figure 7. Assignments for β=1.25 and r=2.

Table 2. Computational results.

Problem	NS	CT	ST	RW Station Position	β=1.00 CT	β=1.00 ST	β=1.25 CT	β=1.25 ST	β=1.50 CT	β=1.50 ST	β=1.75 CT	β=1.75 ST
Jackson (11 task)	3+1	16.00	0.23	2	12.00	0.27	12.50	0.03	13.00	0.30	13.00	0.10
				3	12.50	0.22	12.50	0.06	13.00	0.27	13.00	0.18
				4	15.00	0.17	15.00	0.03	15.00	0.20	15.00	0.03
	4+1	12.00	0.11	3	9.50	0.25	10.00	0.13	10.50	0.28	10.50	0.10
				4	9.50	0.26	10.00	0.10	10.50	0.28	10.50	0.08
				5	11.00	0.11	11.00	0.03	11.00	0.11	11.00	0.02
Mitchell (21 task)	3+1	35.00	≈ 0.00	2	31.00	0.05	31.00	0.11	31.00	0.03	31.00	0.09
				3	31.00	0.06	31.00	0.10	31.00	0.05	31.00	0.10
				4	33.50	0.01	33.50	0.04	33.50	0.02	33.50	0.05
	5+1	21.00	0.02	4	18.50	0.17	18.50	0.37	19.00	0.17	19.50	0.32
				5	18.00	0.13	18.50	0.18	19.00	0.22	19.50	0.26
				6	20.50	0.05	20.50	0.09	20.50	0.03	20.50	0.07
Heskiaoff (28 task)	4+1	256.00	0.01	3	205.00	0.42	213.50	2.57	219.50	0.98	224.00	1.28
				4	205.00	0.21	213.50	1.82	219.50	1.78	224.00	0.26
				5	247.00	0.03	247.00	0.07	247.00	0.36	247.00	0.05
	5+1	205.00	0.06	4	171.00	0.28	176.88	3.10	181.00	1,080.00	184.00	2.79
				5	171.00	0.34	176.88	3.73	181.00	5.47	184.00	1.65
				6	198.00	0.05	198.00	0.08	198.00	0.23	198.00	0.12
Sawyer (30 task)	5+1	65.00	0.08	4	55.00	2.27	56.00	4.71	57.50	2.98	58.50	4.05
				5	57.00	1.47	57.50	2.97	57.50	2.65	58.63	3.57
				6	61.00	0.09	61.00	0.10	61.00	0.16	61.00	0.10
	8+1	41.00	0.08	7	37.00	8.59	37.50	7.19	38.00	4.67	38.00	7.67
				8	36.50	2.84	37.00	6.50	38.00	8.88	38.00	8.52
				9	39.00	2.01	39.00	2.12	39.00	3.10	39.00	3.66
Kilbrid (45 task)	3+1	184.00	0.33	2	141.00	0.58	145.50	0.85	150.75	0.60	154.88	1.17
				3	138.00	0.55	145.50	1.30	150.75	0.94	154.88	3.77
				4	182.00	0.25	182.00	0.08	182.00	0.05	182.00	0.09
	6+1	92.00	0.20	5	79.00	1.48	81.50	4.31	83.00	3.80	84.00	5.04
				6	79.00	2.03	81.50	2.38	83.00	4.16	84.00	2.58
				7	91.00	0.15	91.00	0.55	91.00	0.41	91.00	0.41

NS (Number of Stations), CT (Cycle Time-time unit), ST (Solution Time-second).

The computational results in this section are presented in a similar form to the results of the numerical example given for the Jackson sample in the previous section. In addition to modified versions of the problems with the added rework stations, the original test problems (the ones without the rework stations) are also solved.

Similar to the numerical example computations given in the previous section, the problems are solved for three different defective rates (i.e., β=1.25, β=1.50 and β=1.75) in addition to the case of zero defective production (i.e., β=1.00). Similarly, three rework station positions are considered by positioning it as the last three stations of the assembly line. All problems are solved to optimality with the corresponding solution times varying between fractions of a second (for various problems such as the all versions of the Jackson sample with 11 tasks and Mitchells sample with 21 tasks) and 1,080 seconds (for the Heskiaoff sample with 28 tasks and 6 stations where the rework station is located in the 4th station position). It is also noted however that the maximum solution time of 1,080 seconds seems to be an outlier since the optimal solutions for all other cases (even for larger samples of Sawyer with 30 tasks and Kilbrid with 45 tasks) are obtained in significantly smaller durations (i.e., less than 10 seconds). We can observe the effects of different defective rates and rework station positions in Table 2.

7. Conclusions

In this study, we propose a mixed-integer programming model for minimizing the cycle time of an assembly line considering the use of the rework stations for parallel tasks in addition to the rework operations. Using the proposed model, the tasks are assigned to the rework station and a standard workstation in parallel to utilize the resources in the rework station. By linearizing the nonlinear constraints of the proposed model, it is transformed into a linear-mixed-integer program. We test the model using some test problems from the literature where we analyze the effects of different defective rates and the rework station position.

On the other hand, it is noted that the applicability of the proposed model might be limited in some real-life environments due to the difficulties about parallel task implementation in the assembly lines without a particular level of automation. In such situations, it becomes even more important to consider human factors especially for the workers responsible for performing parallel tasks and rework operations. Nevertheless, it might be easier to deal with such ergonomics difficulties in the near future with the technological developments yielding highly automated smart production systems.

Since the proposed model is obtained by variable transformation for linearization, model size is significantly increased compared to the original nonlinear model. As a result of this increase, applying the model on larger-size problems is expected to result in longer solution times. Developing some heuristic and meta-heuristic approaches to deal larger-size problems is important in future studies. In this study, in order to emphasize the effects of defective rate and rework station position, all computations all computations are performed with the corresponding parameter combination setting (i.e., $\alpha=0$, $\lambda=1$ and $\gamma=1$) ignoring the potential negative effects of parallel task assignment to be considered in future studies. In addition, more comprehensive experiments can be considered in the future to analyze the effects of the defective rate and rework station position. Finally, the validation of the solutions obtained by the proposed approach using simulation might also be useful to analyze the bottleneck effects of parallel task assignment especially for the problems involving uncertainty.

References

Altekin, F.T., Bayindir, Z.P., Gumuskaya, V. (2016). Remedial actions for disassembly lines with stochastic task times. *Computers & Industrial Engineering*, *99*, 78-96. https://doi.org/10.1016/j.cie.2016.06.027

Anderson, E.J., Ferris, M.C. (1994). Genetic algorithms for combinatorial optimization: the assemble line balancing problem. *ORSA Journal on Computing*, *6*(2), 161-173. https://doi.org/10.1287/ijoc.6.2.161

Askin, R.G., Zhou, M. (1997). A parallel station heuristic for the mixed-model production line balancing problem. *International Journal of Production Research*, *35*(11), 3095-3106. https://doi.org/10.1080/002075497194309

Bard, J.F. (1989). Assembly line balancing with parallel workstations and dead time. *The International Journal of Production Research*, *27*(6), 1005-1018. https://doi.org/10.1080/00207548908942604

Bartholdi, J.J. (1993). Balancing two-sided assembly lines: a case study. *International Journal of Production Research*, *31*(10), 2447-2461. https://doi.org/10.1080/00207549308956868

Battaia, O., Dolgui, A. (2013). A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics*, *142*(2), 259-277. https://doi.org/10.1016/j.ijpe.2012.10.020

Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management science*, *32*(8), 909-932. https://doi.org/10.1287/mnsc.32.8.909

Baykasoglu, A., Demirkol Akyol, S. (2014). Ergonomic assembly line balancing. *Journal of the Faculty of Engineering and Architecture of Gazi University*, *29*(4), 785-792. https://doi.org/10.17341/gummfd.00296

Becker, C., Scholl, A. (2006). A survey on problems and methods in generalized assembly line balancing. *European journal of operational research*, *168*(3), 694-715. https://doi.org/10.1016/j.ejor.2004.07.023

Boysen, N., Fliedner, M., Scholl, A. (2007). A classification of assembly line balancing problems. *European journal of operational research*, *183*(2), 674-693. https://doi.org/10.1016/j.ejor.2006.10.010

Bryton, B. (1954). *Balancing of a continuous production line.* Master's Thesis, Northwestern University, Evanston.

Cercioglu, H., Ozcan, U., Gokcen, H., Toklu, B. (2009). A simulated annealing approach for parallel assembly line balancing problem. *Journal of the Faculty of Engineering and Architecture of Gazi University*, *24*(2), 331-341.

Efe, B., Kremer, G.E.O., Kurt, M. (2018). Age and gender-based workload constraint for assembly line worker assignment and balancing problem in a textile firm. *International Journal of Industrial Engineering*, *25*(1), 1-17.

Ghosh, S., Gagnon, R.J. (1989). A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *The International Journal of Production Research*, *27*(4), 637-670. https://doi.org/10.1080/00207548908942574

Gokcen, H., Baykoc, Ö.F. (1999). A new line remedial policy for the paced lines with stochastic task times. *International Journal of Production Economics*, *58*(2), 191-197. https://doi.org/10.1016/S0925-5273(98)00123-6

Gokcen, H., Agpak, K., Benzer, R. (2006). Balancing of parallel assembly lines. *International Journal of Production Economics*, *103*(2), 600-609. https://doi.org/10.1016/j.ijpe.2005.12.001

Guner, B., Hasgul, S. (2012). U-Type assembly line balancing with ergonomic factors for balance stability. *Journal of the Faculty of Engineering and Architecture of Gazi University*, *27*(2), 407-415.

Kaplan, O. (2004). *Assembly line balancing with task paralleling*. Master's Thesis, METU, Ankara.

Kara, Y., Ozguven, C., Yalcın, N., Atasagun, Y. (2011). Balancing straight and U-shaped assembly lines with resource dependent task times. *International Journal of Production Research*, *49*(21), 6387-6405. https://doi.org/10.1080/00207543.2010.535039

Kara, Y., Atasagun, Y., Gokcen, H., Hezer, S., Demirel, N. (2014). An integrated model to incorporate ergonomics and resource restrictions into assembly line balancing. *International Journal of Computer Integrated Manufacturing*, *27*(11), 997-1007. https://doi.org/10.1080/0951192X.2013.874575

Kazemi, S.M., Ghodsi, R., Rabbani, M., Tavakkoli-Moghaddam, R. (2011). A novel two-stage genetic algorithm for a mixed-model U-line balancing problem with duplicated tasks. *The International Journal of Advanced Manufacturing Technology*, *55*(9-12), 1111-1122. https://doi.org/10.1007/s00170-010-3120-6

Kim, Y.K., Kim, Y., Kim, Y.J. (2000). Two-sided assembly line balancing: a genetic algorithm approach. *Production Planning & Control*, *11*(1), 44-53. https://doi.org/10.1080/095372800232478

Kottas, J.F., Lau, H.S. (1976). A total operating cost model for paced lines with stochastic task times. *AIIE Transactions*, *8*(2), 234-240. https://doi.org/10.1080/05695557608975072

Lau, H.S., Shtub, A. (1987). An exploratory study on stopping a paced line when incompletions occur. *IIE transactions*, *19*(4), 463-467. https://doi.org/10.1080/07408178708975421

Lee, T.O., Kim, Y., Kim, Y.K. (2001). Two-sided assembly line balancing to maximize work relatedness and slackness. *Computers & Industrial Engineering*, *40*(3), 273-292. https://doi.org/10.1016/S0360-8352(01)00029-8

Mutlu, O., Ozgormus, E. (2012). A fuzzy assembly line balancing problem with physical workload constraints. *International Journal of Production Research*, *50*(18), 5281-5291. https://doi.org/10.1080/00207543.2012.709647

Ozcan, U., Toklu, B. (2010). Balancing two-sided assembly lines with sequence-dependent setup times. *International Journal of Production Research*, *48*(18), 5363-5383. https://doi.org/10.1080/00207540903140750

Pinto, P., Dannenbring, D.G., Khumawala, B.M. (1975). A branch and bound algorithm for assembly line balancing with paralleling. *The International Journal of Production Research*, *13*(2), 183-196. https://doi.org/10.1080/00207547508942985

Sabuncuoglu, I., Erel, E., Alp, A. (2009). Ant colony optimization for the single model U-type assembly line balancing problem. *International Journal of Production Economics*, *120*(2), 287-300. https://doi.org/10.1016/j.ijpe.2008.11.017

Salveson, M.E. (1955). The assembly line balancing problem. *The Journal of Industrial Engineering*, 18-25.

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*(3), 666-693. https://doi.org/10.1016/j.ejor.2004.07.022

Shtub, A. (1984). The effect of incompletion cost on line balancing with multiple manning of work stations. *The International Journal of Production Research*, *22*(2), 235-245. https://doi.org/10.1080/00207548408942450

Silverman, F.N., Carter, J.C. (1986). A cost-based methodology for stochastic line balancing with intermittent line stoppages. *Management Science*, *32*(4), 455-463. https://doi.org/10.1287/mnsc.32.4.455

Simaria, A.S., Vilarinho, P.M. (2001). The simple assembly line balancing problem with parallel workstations-a simulated annealing approach. *Int J Ind Eng-Theory*, *8*(3), 230-240.

Sivasankaran, P., Shahabudeen, P. (2014). Literature review of assembly line balancing problems. *The International Journal of Advanced Manufacturing Technology*, *73*(9-12), 1665-1694. https://doi.org/10.1007/s00170-014-5944-y

Suer, G.A. (1998). Designing parallel assembly lines. *Computers & industrial engineering*, *35*(3-4), 467-470.

Suwannarongsri, S., Puangdownreong, D. (2008). Optimal assembly line balancing using tabu search with partial random permutation technique. *International Journal of Management Science and Engineering Management*, *3*(1), 3-18. https://doi.org/10.1080/17509653.2008.10671032

Tiacci, L., Saetta, S., Martini, A. (2006). Balancing mixed-model assembly lines with parallel workstations through a genetic algorithm approach. *International Journal of Industrial Engineering*, *13*(4), 402.

Ugurdag, H.F., Rachamadugu, R., Papachristou, C.A. (1997). Designing paced assembly lines with fixed number of stations. *European Journal of Operational Research*, *102*(3), 488-501. https://doi.org/10.1016/S0377-2217(96)00248-2

Wei, N.C., Chao, I.M. (2011). A solution procedure for type E simple assembly line balancing problem. *Computers & Industrial Engineering*, *61*(3), 824-830. https://doi.org/10.1016/j.cie.2011.05.015