International Journal of Production Research
Vol. 49, No. 21, 1 November 2011, 6387–6405

Balancing straight and U-shaped assembly lines with resource dependent task times

Yakup Kara[a]*, Cemal Özgüven[b]†, Neşe Yalçın[c] and Yakup Atasagun[a]

[a]*Industrial Engineering Department, Selçuk University, Campus 42031, Konya, Turkey;*
[b]*Department of Business Administration, Erciyes University, 38039, Kayseri, Turkey;*
[c]*Department of Business Administration, Nevşehir University, 50300, Nevşehir, Turkey*

(*Received 12 March 2010; final version received 12 October 2010*)

The basic assumption of assembly line balancing is that every task's time is fixed. However, in practice, different processing alternatives may be available to process a task with different times. The problem in this case is to assign tasks and resources to work stations that minimise total cost and so-called resource dependent assembly line balancing (RDALB) in the literature. This study proposes a new integer programming formulation for RDALB. This formulation is then modified to develop a formulation for RDULB problem in U-shaped assembly lines. To the best knowledge of the authors, this study is the first RDULB study and the developed formulation is the first RDULB formulation. The proposed formulations are illustrated and validated using several examples. An experimental analysis is also conducted to examine the percent improvement in total cost when the line layout is switched into the U-shaped from the straight line shape. Experimental results show that an improvement in total operating cost is obtained when the straight line is switched into the U-shaped line configuration. Results also show that percentage improvement in total operating cost is significantly greater for problem instances having a large number of tasks and having greater values for the strength of the precedence relationships among tasks.

Keywords: assembly line balancing; U-lines; integer programming; resource dependent task times; processing alternatives

1. Introduction

An assembly line consists of several successive work stations in which some assembly operations of a particular product are completed. A task is defined as the smallest portion of an assembly operation. The workload of a work station is the sum of the completion times of tasks assigned to this work station. The largest workload in an assembly line is defined as the cycle time of the line. Assembly lines can be classified as single-model and mixed-model assembly lines by means of different numbers of product models produced on the line. Single model assembly lines are defined as the lines on which one product model is produced while two or more product models are produced on mixed-model assembly lines. Assembly lines can also be classified as straight and U-shaped assembly lines by means of the shape of the line. In today's production systems, as a result of

---

*Corresponding author. Email: ykara@selcuk.edu.tr
†Now retired.

ISSN 0020–7543 print/ISSN 1366–588X online

DOI: 10.1080/00207543.2010.535039
http://www.informaworld.com

just-in-time (JIT) philosophy, U-shaped lines have increasingly substituted traditional straight assembly lines. The role of facility layout in JIT production systems is important to obtain the main benefits of one-piece flow manufacturing, smoothed workload, multi-skilled workforce, and other principles of JIT. Furthermore, many researchers agree that the U-shaped layout is one of the most important components for a successful implementation of JIT production (Monden 1993, Miltenburg 2001). Figure 1 shows an example U-shaped assembly line with eight tasks and three work stations.

One of the most important characteristics which make U-shaped assembly lines different from straight assembly lines is that the entrance and the exit of these lines are at the same position (Monden 1993). Product models enter the U-shaped assembly line at the front-side and exit from the back-side of the line. The lengths of front-side and back-side of the U-shaped assembly line are equal and operators work inside of the U-shaped assembly line. Existing studies on U-shaped assembly lines provide evidence for their potential to improve visibility and communication skills between operators, reduce operator requirements, increase quality, reduce work-in-process inventory, and facilitate problem-solving and efforts to adjust to changes in the external environment of the firm (Miltenburg 1998, 2001, Aase *et al.* 2004).

The literature on U-shaped assembly lines includes several experimental and empirical studies that examine the benefits of the U-shaped layout of these lines compared with the straight line shape. Cheng *et al.* (2000) examined U-shaped lines and straight lines from the point of their effect on quality, which was organised into several quantitative measures of Juran's quality planning, quality control, and quality improvement categories. They found U-shaped lines superior to straight lines in all of the aspects of product quality they examined. Miltenburg (2000) investigated the effect of the U-shape of the line on the line effectiveness when breakdowns occur on the line. Results of this research showed that effectiveness of the line increases when the line is U-shaped compared with the straight line shape. Miltenburg (2001) investigated the theory of U-shaped lines as well as their practice in a total of 114 American and Japanese U-shaped lines. Findings of this research showed that the number of tasks performed on U-shaped lines varies between 1 and 24.

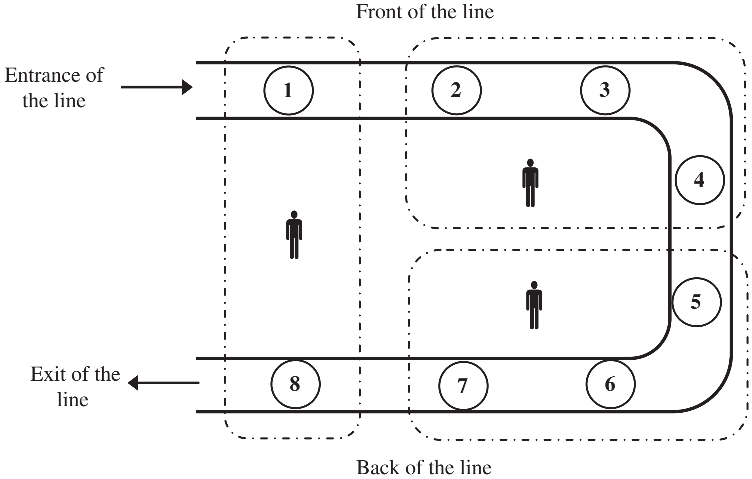

Figure 1. An example of a U-shaped assembly line.

The average number of work stations on these lines was also found to be 3.4. The reported benefits of U-shaped line implementations include productivity improvement by an average of 76%, work-in-progress drop by 86%, lead-time shrink by 75%, and defective rate drop by 83%.

Aase *et al.* (2004) focused on the effect of the U-Shape of the line on labour productivity. They examined the percent improvement in labour productivity when the line layout is switched into the U-shaped from the straight line shape with respect to three problem factors namely the number of tasks, network density, and cycle time. An experiment was conducted to compare U-shaped and straight lines by using a dataset consisting of a total randomly generated 540 assembly line balancing problems. When generating the problems, they tested each factor using three treatment levels: number of tasks (10, 30, 50), network density $(0.25, 0.50, 0.75) \pm 0.05$, and cycle time (30, 45, 60). Results of this study showed that of the 540 problems, 416 problems (i.e. 77%) experience no improvement in labour productivity. The average reduction in work stations of 540 problem instances was 0.27. Results also showed that the percent improvement in labour productivity was significantly greater for problem instances having fewer tasks, greater values of network density, and reduced values of cycle times.

Assembly line balancing (ALB) is the problem of assigning tasks to work stations in such a way that some performance measures are optimised subject to the precedence relationships among tasks (Erel *et al.* 2001). Depending on the performance measure considered, three versions of ALB can be identified (Scholl and Klein 1999):

1. Given the cycle time, the number of work stations is minimised (Type-I);
2. given the number of work stations, the cycle time is minimised (Type-II);
3. and given the cycle time and the number of work stations, the line efficiency is maximised (Type-E).

By means of the shape of the line, two groups of ALB problems can be identified as (i) single-model straight assembly line balancing problem (SALB) and (ii) single-model U-shaped assembly line balancing problem (SULB).

The SALB problem was first formulated by Salveson (1955) and has been extensively studied by many researchers to date. Interested readers are suggested to review the papers of Ghosh and Gagnon (1989), Erel and Sarin (1998), Becker and Scholl (2006) and Scholl and Becker (2006) on SALB. The literature on SULB is relatively small compared with SALB literature. The SULB literature includes dynamic programming (DP)-based, integer programming (IP)-based, branch and bound (B&B)-based, shortest route approaches, and heuristics.

The SULB problem was first discussed by Miltenburg and Wijngaard (1994). They explained the differences between SALB and SULB; developed a DP formulation, and modified ranked positional weight technique (RPWT) and Hoffman heuristics. Miltenburg and Sparling (1995) proposed three DP-based and B&B-based exact algorithms to SULB. The first IP formulation of SULB was developed by Urban (1998) that uses the phantom precedence diagram concept. A phantom precedence diagram is appended to the original precedence diagram so that assignments to the work stations can be made forward through the original diagram, backward through the phantom diagram, or simultaneously in both directions.

Scholl and Klein (1999) proposed a B&B procedure called ULINO (U-Line optimiser), an extension of their previous algorithm called SALOME. Erel *et al.* (2001) developed a simulated annealing-based approach to solve SULB with the objective of minimising the number of work stations. Aase *et al.* (2003) proposed a B&B procedure called U-OPT and found their algorithm is more effective than existing procedures when solving classical and new substantive problem sets. Gökçen *et al.* (2005) suggested a shortest route formulation for SULB. Gökçen and Ağpak (2006) developed a goal programming approach for SULB. Their goal programming approach is based on the IP formulation of Urban (1998) and it is the first multi-criteria approach for SULB. Other goal programming formulations can be found in Toklu and Özcan (2008) and Kara *et al.* (2009). A mixed-model version of U-shaped assembly line balancing can be found in Sparling and Miltenburg (1998), Hwang and Katayama (2009) and Kara and Tekin (2009).

Typically, SALB and SULB aim at minimisation of the number of work stations and hence minimisation of workers employed to staff a line while adhering to task precedence and cycle time restrictions for a given cycle time (McMullen and Tarasewich 2006). The basic assumption of SALB and SULB is that every task's time is fixed. However, this situation could not always be effective, practical and flexible especially when different resource alternatives are available to process a task with different durations. One way to adapt flexibility to assembly lines is to change task times using different processing alternatives including different resources. On the other hand, assigning only one worker to each work station is another assumption of SALB and SULB in many studies. But, some tasks in practice cannot be processed by only one worker. For example, some tasks may need to have an additional assistant or particular equipment. A worker may need assistance to perform tasks on some parts. In this case, different processing alternatives (resource combinations) have to be used in the line. Faaland *et al.* (1992) defined that the problem in this case is to assign tasks and resources to work stations that minimise total cost and so-called resource dependent assembly line balancing (RDALB). According to Faaland *et al.* (1992), the actual time required to process a task at a work station depends on the processing alternative allocated to the work station. A processing alternative consists of a set of resources namely workers, assistants and equipment. Each processing alternative has a cost and costly processing alternatives usually reduce task times. Faaland *et al.*'s (1992) objective is to minimise the total cost of processing alternatives allocated to work stations and fixed costs of work stations utilised in the line.

Most ALB problems in many industries can be treated as RDALB. A simple example can be presented here to provide evidence for this proposition. The assembly line of a solar collector producer, which is located in Konya, Turkey, was observed to obtain evidence. Figure 2 shows a flat solar thermal collector which collects solar radiation for water heating. A typical flat collector is composed of four main components: an aluminium case, a copper panel, insulation material (rockwool) and a glass.

There are currently 32 tasks and eight work stations in the line which is running with an average cycle time of 3.5 minutes. Although most of the tasks have processing alternatives, two tasks are selected to explain RDALB. The first task is called 'glass positioning' which is manually performed. Since the glass is heavy and large, this task is currently performed by a worker and an assistant in 16 seconds. However, this task can be performed by only a worker in 9 seconds if a vacuum gripper is utilised. This processing alternative provides shorter processing time and lower workforce but it requires an additional equipment cost. The second task is called 'placement of insulation material'. This task is performed by one worker in 114 seconds. However, a collector is large enough that two workers can work on the same product at the same time. Therefore, task time can be reduced to half by assigning a worker and an assistant.

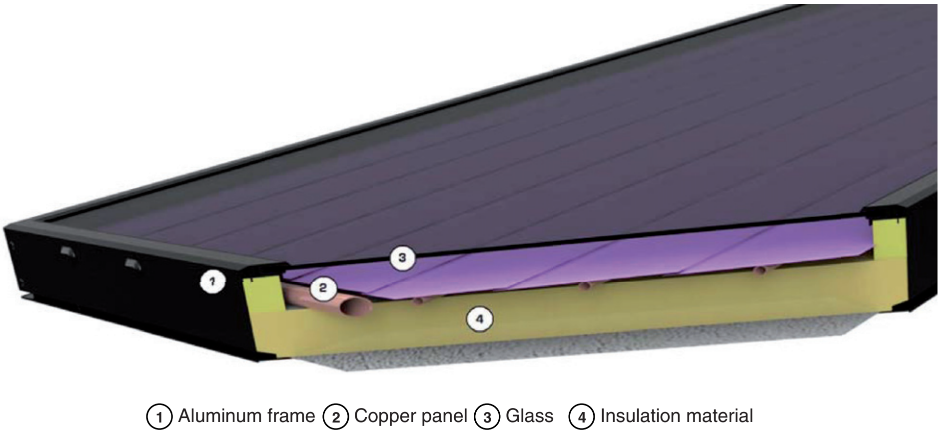

Figure 2. Flat plate solar collector.

There are several studies in the literature that are tightly related to RDALB. Pinto *et al.* (1983) proposed a method of simultaneously considering both the choice of processing alternatives and the assignment of tasks to work stations so as to minimise labour and fixed cost. Bukchin and Tzur (2000) proposed a model and solution procedures for RDALB. They assumed that the duration of a task is deterministic, but depends on the equipment selected to perform the task. They also assumed that a single piece of equipment is assigned to each work station on the line. The objective of their approach is to minimise the total equipment cost for a given cycle time. Bukchin and Rubinovitz (2002) studied assembly system design with parallel work stations including equipment selection.

Corominas *et al.* (2008) proposed a binary linear formulation based on the assumption that the task times depend on the worker performing the task. They categorised the line workers as skilled and unskilled. Their study was constituted on the process of rebalancing the line at a motorcycle assembly plant which has permanent (skilled) and temporary (unskilled) workers. They attempted to minimise the number of temporary workers employed in the line. Moon *et al.* (2009) extended the SALB problem to one that has resource restrictions. They defined multifunctional workers as a restriction assuming these workers are paid different salaries depending on their skills. They also assumed that worker's operation time cannot be longer than the predetermined upper limit. They proposed a mixed integer linear program and a genetic algorithm to simultaneously optimise the assignments of tasks to work stations and the assignments of multifunctional workers to the tasks in order to minimise the associated overall costs. They addressed their extension as integrated ALB with resource restrictions.

A common feature of the above presented studies is that the assembly line is arranged traditionally straight. To the best of our knowledge, the SULB problem has not been studied considering resource dependent task times and resource restrictions. In this study, we first propose a new integer programming formulation for RDALB. The proposed formulation is then modified to develop a formulation for SULB with resource dependent task times and resource restrictions which we denote as resource dependent U-shaped assembly line balancing (RDULB). The objective is to minimise the sum of fixed work station costs, equipment costs and labour costs. We then focus on the effect of the U-shape of the line on total cost and examine the percent improvement in total cost when the line layout is switched into the U-shaped from the straight line shape. The remainder of the paper is organised as follows: the formulations for RDALB and RDULB are developed in Section 2. Some illustrative examples are presented and formulations are validated in Section 3. An experimental study is conducted to compare straight and U-shaped assembly lines by means of total operating cost. Some concluding remarks are given in Section 5.

2. Mathematical formulations

In this section we propose integer programming formulations for RDALB and RDULB. The following assumptions have been made in order to clarify the setting in which the problem arises:

- A single product is assembled on the line.
- The precedence relationships among tasks are known.
- The duration of a task is deterministic, but depends on the resources (equipment type and assistant) allocated to perform the task. The duration of a task is independent of the work station to which the task is assigned.
- A task can be assigned to any work station as long as it does not violate precedence relationships.
- Some tasks cannot be completed by only one worker (multi-worker tasks). If any multi-worker task is assigned to a work station, an assistant should be assigned to this work station as well.
- The duration of some tasks can be reduced by performing these tasks with the assistance of an assistant.
- Some tasks should be performed using particular equipments. There may be alternate equipment types for a task. Some tasks can be performed with an equipment type or without any equipment. Each equipment type is specified with a cost.
- There is sufficient number of workers required to operate work stations. But, the amounts of other resources (equipment types and assistants) are limited.
- No work-in-process inventory is allowed between work stations.

Following the above assumptions, the decisions that have to be made are (i) the resources (equipments and assistants) that have to be assigned to tasks and (ii) the tasks that have to be assigned to work stations. These decisions have to be made such that precedence, cycle time and resource restrictions are satisfied. The objective is to minimise the sum of total cost of work station utilisation and total cost of resource utilisation. Total cost of work station utilisation is composed of fixed costs and it is proportional to the number of work stations utilised on the line. However, total cost of resource utilisation depends on which equipments and how many assistants are assigned to work stations.

The notation used to describe the proposed formulations is given below:

Indices

i, r, s	task
j	work station
e	equipment

Parameters and sets

t_{ie0}	duration of task i if it is processed with equipment e without assistant
t_{ie1}	duration of task i if it is processed with equipment e and an assistant
T	set of all tasks
W	set of work stations
E	set of all equipments
NE	available number of equipment e
NA	available number of assistants
$K_{\max}$	maximum number of work stations
CT	cycle time
E_i	set of equipments that can be used to process task i
PR	set of precedence relationships
$(r, s) \in PR$	a precedence relationships; r is an immediate predecessor of s
M	a big number
cw	annual utilisation cost of a work station (worker + fixed costs)
ca	annual employment cost of an assistant
c_e	annual operating cost of equipment e

Variables

x_{ij}	1, if task i is assigned to work station j from the original diagram; 0, otherwise
y_{ij}	1, if task i is assigned to work station j from the phantom diagram; 0, otherwise
p_{ije}	1, if task i is processed in work station j with equipment e without any assistant; 0, otherwise
q_{ije}	1, if task i is processed in work station j with equipment e and an assistant; 0, otherwise
z_{je}	1, if equipment e is allocated to work station j; 0, otherwise
u_j	1, if work station j is utilised; 0, otherwise
k_j	1, if an assistant is assigned to work station j; 0, otherwise

The proposed integer programming formulation for RDALB is presented below:

(M1)
Minimise

$$\sum_{j\in W}(cw.u_j + ca.k_j) + \sum_{e\in E}\sum_{j\in W} c_e z_{je} \tag{1}$$

Subject to:

$$\sum_{j\in W} x_{ij} = 1 \quad \forall i \in T \tag{2}$$

$$\sum_{e\in E_i}(p_{ije} + q_{ije}) = x_{ij} \quad \forall i \in T; \forall j \in W \tag{3}$$

$$\sum_{j\in W}(K_{\max} - j + 1)(x_{rj} - x_{sj}) \geq 0 \quad \forall (r, s) \in PR \tag{4}$$

$$\sum_{i\in T}\sum_{e\in Ei}(t_{ie0}p_{ije}+t_{ie1}q_{ije})\leq CTu_j \quad \forall j\in W \tag{5}$$

$$\sum_{i\in T}(p_{ije}+q_{ije})-Mz_{je}\leq 0 \quad \forall j\in W; \forall e\in E_i \tag{6}$$

$$\sum_{j\in W}z_{je}\leq NE \quad \forall e\in E \tag{7}$$

$$\sum_{i\in T}\sum_{e\in E_i}q_{ije}-Mk_j\leq 0 \quad \forall j\in W \tag{8}$$

$$\sum_{j\in W}k_j\leq NA \tag{9}$$

The objective function defined in Equation (1) minimises total cost associated with work station utilisation, assistant assignment and equipment allocation. Equation (2) assures that each task is assigned to at least and at most one work station. Equation (3) determines the resources (equipment type and assistant) allocated to a work station. Precedence relationships among tasks are satisfied by the set of constraints given in Equation (4). Equation (5) ensures the workload of a work station is equal to or less than the predetermined cycle time. Equation (6) determines whether equipment e is allocated to work station j. Equation (7) restricts the number of equipment type e allocated to work stations by the available number of this type. Equation (8) determines whether an assistant is assigned to work station j. Equation (9) assures that the number of assistants assigned to work stations should not exceed the available number of assistants.

The key difference between SALB and SULB can be described as follows: in SALB, the set of assignable tasks consists of tasks whose all predecessors have already been assigned to earlier utilised work stations. However in SULB, this set of tasks consists of tasks whose all predecessors or all successors have already been assigned to earlier work stations (Miltenburg and Wijngaard 1994). It means that U-shaped assembly lines provide more alternatives for grouping tasks into a work station. The productivity improvement on U-shaped assembly lines comes from this flexibility of assembly line balancing. Urban (1998) used the 'phantom precedence diagram' concept to incorporate this difference into a SULB problem. A phantom precedence diagram is the appended copy of an original precedence diagram. Once these two diagrams are combined, tasks are assigned to work stations by beginning from the intersection of these diagrams and progressing in forward, backward or simultaneously in both directions. The integer programming formulation proposed for RDULB in the current study adopts the phantom precedence diagram concept of Urban (1998). Figure 3 shows an original precedence diagram with eight tasks and its phantom diagram.

Urban (1998) used the variable y_{ij} in his formulation to express the assignment of task i from the phantom precedence diagram. In this way, a task can be assigned to a work station either from the original or from the phantom diagram. In other words, either x_{ij} or y_{ij} variable can be 1 for the assignment of task i to work station j. The proposed RDALB formulation can be revised to develop a model for RDULB using the phantom precedence diagram concept and variable y_{ij}. Due to the use of this concept and new variable, some

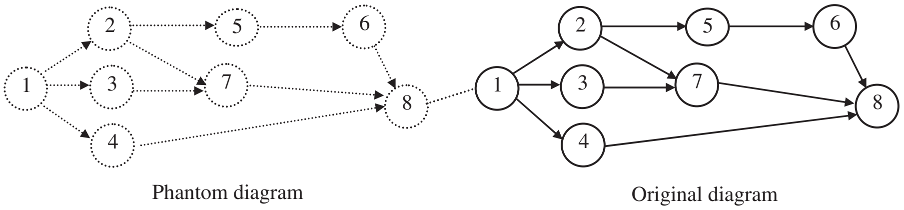

Figure 3. Original and phantom precedence diagrams with eight tasks.

modifications are needed in the sets of assignment and precedence constraints. Other sets of constraints and objective function remain unchanged.

(M2)
Minimise

Equation (1)

Subject to:

$$\sum_{j\in W}(x_{ij}+y_{ij})=1 \quad \forall i\in T \tag{10}$$

$$\sum_{e\in E_i}(p_{ije}+q_{ije})=x_{ij}+y_{ij} \quad \forall i\in T; \forall j\in W \tag{11}$$

$$\sum_{j\in W}(K_{\max}-j+1)(x_{rj}-x_{sj})\geq 0 \quad \forall (r,s)\in PR \tag{12}$$

$$\sum_{j\in W}(K_{\max}-j+1)(y_{sj}-y_{rj})\geq 0 \quad \forall (r,s)\in PR \tag{13}$$

Equations (5) to (9)

Equation (10) simply assures that each task is assigned to at least and at most one work station from either original or from phantom diagram. Equation (11) determines the resources allocated to a work station. Precedence relationships among tasks are satisfied by the set of constraints given by Equations (12) and (13). The model (M2) apparently consists of larger number of variables and constraints than model (M1).

3. Illustrative examples

In this section we apply the proposed models to an eight-task example problem. There are two different types of equipments. The available amounts of these equipments are $N_1=2$ and $N_2=1$ respectively. In addition, as other resources, we have two assistants who can be assigned to work stations. The precedence relationships among tasks and resource dependent task times (in minutes) are given in Table 1.

The data in Table 1 actually describes processing alternatives of tasks. For instance, there are two processing alternatives for task 1 which does not require any equipment. Task 1 can be completed without any assistant within 7 minutes. However, the completion

Table 1. Task data of the example problem.

Task no.	Immediate predecessors	Assistant assignment	No. equipment	Equipment 1	Equipment 2
1	–	Yes	5		
		No	7		
2	1	Yes	6		
		No			
3	1	Yes		3	
		No		6	
4	1	Yes	2		
		No	5		
5	2, 3	Yes			
		No		4	7
6	4	Yes		6	
		No			
7	6	Yes			
		No		3	5
8	5, 7	Yes	2		
		No	4		

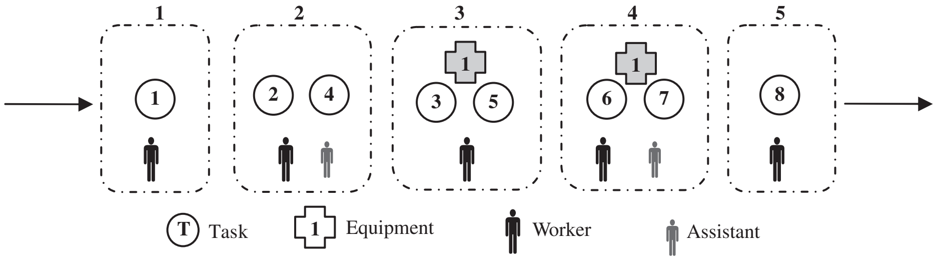

Figure 4. Optimal RDALB solution of the example obtained by (M1).

time is reduced to 5 minutes if an assistant helps the worker. There are also two processing alternatives for task 5. One of both equipments can be used to process this task with different durations. The costs defined in the models are taken as $cw = 100$, $ca = 70$, $c_1 = 25$ and $c_2 = 20$ cost units. The case 'no equipment' is also defined as equipment in the models by labelling equipment number 0. Hence, equipment set is obtained as $E = \{0, 1, 2\}$. As a matter of course, the cost of equipment 0 should be set to zero and its available amount should not be restricted in the models. Finally, we set $K_{\max} = 5$ and the example problem is solved for $CT = 10$ minutes. Figure 4 shows the optimal RDALB solution of the example problem.

The optimal RDALB solution consists of five work stations with a total cost of 690 units. Task 1 is performed in work station 1 without any equipment and assistance. Assistants are assigned to work stations 2 and 4 whereas equipment 1 is allocated to work stations 3 and 4. Work station 2 consists of tasks 2 and 4 that will be processed with the cooperation of worker and assistant without any equipment. In work station 3, tasks 3 and 5 are processed by worker using equipment 1. The assistant assigned to work station 4 is

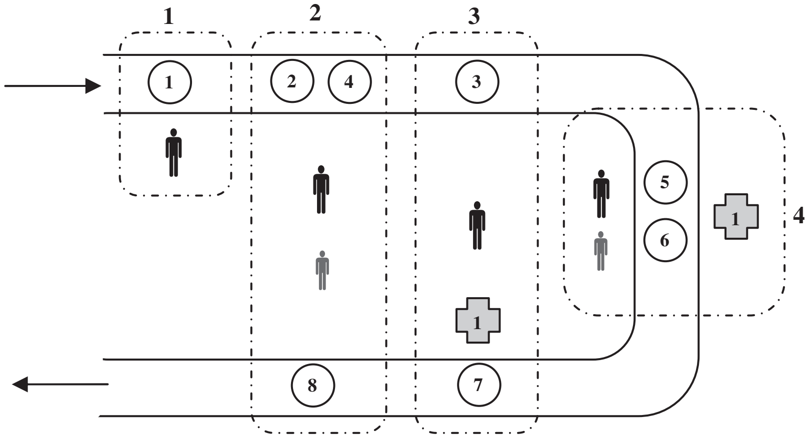

Figure 5. Optimal RDULB solution of the example obtained by (M2).

required to perform task 6. Finally, task 8 is performed in work station 5 without any equipment and assistance. We then solved the example considering the U-shaped line. The optimal RDULB solution is given in Figure 5.

The optimal RDULB solution consists of four work stations with a total cost of 590 units. That is, total cost of U-shaped layout is less than that of straight line layout. Equipment 1 is allocated to work stations 3 and 4. Assistants are assigned to work stations 2 and 4. Based on the locations of tasks throughout the line, two types of work stations appear in U-shaped assembly lines: the first type is called 'regular work stations'. All tasks that are assigned to this type of work stations are located at the same position of the line. In other words, tasks are located on one position of the line with a sequential order. The second type is called 'crossover work stations'. This type of work station includes tasks located at both front-side and back-side of the line. A work station is called crossover if tasks are assigned to this work station from both the original and the phantom diagram. In Figure 5, work stations 1 and 4 are regular while others are crossover.

According to the optimal RDULB assignments shown in Figure 5, tasks 3 and 7 are processed in crossover work station 3. In this work station, both tasks require equipment 1 to be processed but these tasks are located on different sides of the line. In this case, the equipment can be located at one side of crossover work station and parts can be carried to perform their tasks located at the other side of the line. For instance, as shown in Figure 5, equipment 1 in work station 3 can be located at the back side of the line and parts can be carried to this side to perform task 3. Since parts usually become heavier as they move from the first task to the last, it would be better to locate the equipment to the back side of the line. It should be noted here that carrying parts from one side to another is applicable as long as they are not voluminous and heavy. If the parts are voluminous and heavy, the line should be balanced considering this issue. That is, if two or more tasks, which are located at two sides of the line in the same crossover work station, require the same equipment then they cannot share this equipment. The proposed RDULB formulation does not take into account this restriction in its current form. We modified (M2) to take into account this restriction and developed a new formulation (M3). We first introduce the following additional notation:

zf_{je}	1, if equipment e is assigned to the front of work station j; 0, otherwise
zb_{je}	1, if equipment e is assigned to the back of work station j; 0, otherwise
T_e	set of tasks that can be processed with equipment e

(M3)
Minimise

$$\sum_{j\in W}(cw.u_j + ca.k_j) + \sum_{e\in E}\sum_{j\in W} c_e(zf_{je} + zb_{je}) \tag{14}$$

Subject to:

$$\sum_{j\in W}(zf_{je} + zb_{je}) \leq NE \quad \forall e \in E \tag{15}$$

$$p_{ije} + q_{ije} + x_{ij} - 2zf_{je} \leq 1 \quad \forall e \in E; \forall i \in T_e; \forall j \in W \tag{16}$$

$$p_{ije} + q_{ije} + y_{ij} - 2zb_{je} \leq 1 \quad \forall e \in E; \forall i \in T_e; \forall j \in W \tag{17}$$

Equation (5)
Equations (8) to (13)

Equation (15) restricts the number of equipment type e allocated to work stations by the available number of this type. By Equation (16), the variable zf_{je} will be 1 if any task that is processed with equipment e is assigned to work station j from the original diagram. Similarly, Equation (17) ensures that zb_{je} will be 1 if any task that is processed with equipment e is assigned to work station j from the phantom diagram. We solved the example problem using (M3). The new optimal RDULB solution is given in Figure 6.

The optimal RDULB solution is obtained with the same objective value of (M2), 590 cost units. Resource and work station utilisation in the new solution is also the same. However, no equipment is assigned to the crossover work station 2.

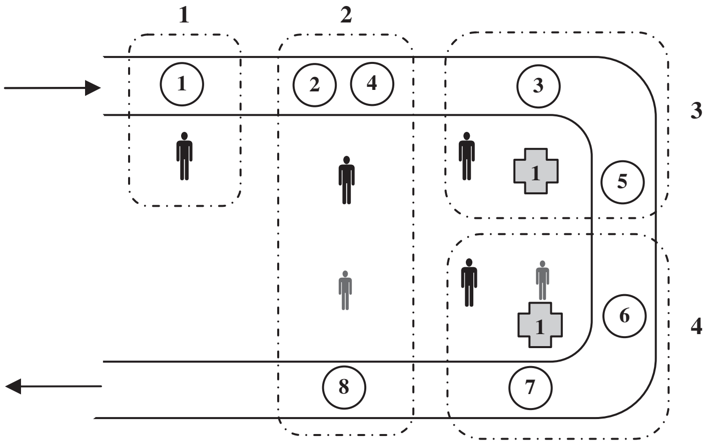

Figure 6. Optimal RDULB solution of the example obtained by (M3).

4. Experimental Results

Two cost-oriented models for balancing straight and U-shaped assembly lines were developed in previous sections. The models aim at minimising total operating costs of assembly lines by assigning resources and grouping assembly tasks. These models can now be used as tools to compare straight and U-shaped assembly lines in terms of their total operating costs. Existing literature includes several studies that compare straight and U-shaped assembly lines by means of labour productivity, quality, inventory levels and lead times. However, to the best of our knowledge, there is no study that compares straight and U-shaped lines by means of total operating costs.

In this section, an experiment is conducted to compare U-shaped assembly lines with straight assembly lines in terms of improvement in the total operating cost. Assembly systems are generally characterised by three factors: problem size, strength of the precedence ordering relations among tasks, and the expected number of tasks per work station (Aase *et al.* 2004). These factors can be defined by the number of tasks (*NT*), flexibility ratio (*FR*), and cycle time (*CT*) measures, respectively. The data set used in the experiment consists of a total of 270 problems that are generated by considering the above factors and corresponding measures. The design of experiment is given in Table 2.

The number of tasks performed on U-shaped lines used in practice varies between 1 and 24 with an average value of 10.2 (Miltenburg 2001). According to Aase *et al.* (2004) and Miltenburg (2001), the problem instances having less than 20 tasks represent the U-shaped assembly line balancing problems in practice. Five levels of *NT* (10, 20, 30, 40 and 50) and three levels of *FR* (0.25 ± 0.05, 0.50 ± 0.05, 0.75 ± 0.05) are considered in 15 precedence diagrams. For example, there are three different 10-task diagrams with three different levels of *FR*. Nine precedence diagrams generated by Kara (2004) are adopted for 10, 20 and 30-task problems in the experiment. The remaining six precedence diagrams for 40 and 50-task problems are generated by the authors.

The numbers of different equipment types are selected 3, 6, 9, 12 and 15 for 10, 20, 30, 40 and 50-task problems, respectively. Available numbers of equipments (*NE*) for all problems are randomly restricted by 1 or 2. Available numbers of assistants (*NA*) are restricted by 2, 4, 6, 8 and 10 for 10, 20, 30, 40 and 50-task problems, respectively.

Resource dependent task times are randomly assigned following a discrete uniform distribution with $U(1, 15)$. It should be noted here that, a task's time with assistant should not be greater than that of without an assistant. If such a random assignment was generated, we re-generated lesser times for these tasks. In all test problems, the fixed work station and assistant costs are selected 10 and 6 cost units respectively. Equipment costs are randomly generated following a continuous uniform distribution with $U(1, 6)$.

Table 2. Design of the experiment.

NT	*FR*	*CT*	Number of different equipment types	Available number of assistances
10	0.25; 0.50; 0.75	30, 45, 60	3	2
20	0.25; 0.50; 0.75	30, 45, 60	6	4
30	0.25; 0.50; 0.75	30, 45, 60	9	6
40	0.25; 0.50; 0.75	30, 45, 60	12	8
50	0.25; 0.50; 0.75	30, 45, 60	15	10

Assignments of task times are repeated three times and hence 45 problem instances are obtained. Each problem instance is solved considering three levels of cycle time CT (30, 45, 60) and considering two different line shapes. That is, 135 RDALB and 135 RDULB problems are attempted to be optimally solved. The model (M1) is used to obtain optimal solutions to RDALB problems. On the other hand, (M2) is used to obtain optimal solutions to RDULB problems assuming parts are not voluminous and heavy. All problems are solved using Gurobi LP/MIP Solver Engine V10.0 on an IntelCore2 Duo 2.00 GHz and 2GB RAM computer. The CPU time required to obtain optimal solutions is limited to three hours.

In the experiments, two of 135 RDALB and 15 of 135 RDULB problem instances could not be optimally solved within 3 hours. In addition, eight problem instances could not be optimally solved for both RDALB and RDULB within 3 hours. Therefore, both RDALB and RDULB solutions of these 25 instances are excluded from the experimental data and the remaining 220 solutions are used in all analyses. The maximum CPU times for 10-task and 20-task problems are observed 1 second and 51 seconds, respectively. The average of CPU times for 30, 40 and 50-task problems are 10 minutes and 29 seconds, 37 minutes and 32 seconds and 1 hour and 30 minutes and 18 seconds, respectively. The summary of CPU times is given in Table 3.

Table 3 shows that 9 of 27 RDALB and 18 of 27 RDULB instances of 50-task problems could not be optimally solved within 3 hours. That is, more than half of 50-task RDULB problems could not be optimally solved using Gurobi LP/MIP which is quite an efficient solver.

The ‘percent improvement in total operating cost’ (PI) is selected as the dependent variable for the experiment. For a problem instance, given the optimal RDALB and RDULB solutions, the PI value is calculated by the following equation:

$$PI = [[\mathrm{Cost(RDALB)} - \mathrm{Cost(RDULB)}]/\mathrm{Cost(RDALB)}] \times 100 \tag{18}$$

An analysis of variance (ANOVA) is performed for PI and variabilities are partitioned under the main effects of NT, FR, and CT to test differences between PI of various levels of experimental factors. Duncan grouping tests are also performed to further analyse the significant effects. The SPSS 15.0 statistical package is used to analyse experimental data.

Table 3. The summary of CPU times.

NT	Line shape	CPU < 1 h	1 h < CPU < 2 h	2 h < CPU < 3 h	3 h < CPU
10	Straight	27	–	–	–
	U-shaped	27	–	–	–
20	Straight	27	–	–	–
	U-shaped	27	–	–	–
30	Straight	27	–	–	–
	U-shaped	24	1	1	1
40	Straight	23	3	–	1
	U-shaped	20	–	–	7
50	Straight	18	–	–	9
	U-shaped	10	–	2	15

Experimental results indicated that of the optimally solved 110 problems, 33 problems experience no reduction in total operating cost. This result shows that the U-shaped layout provides no improvement in total operating cost for 30% of all problem instances. Twenty of these instances are 10-task, five of them are 20-task, four of them are 30-task, two of them are 40-task and two of them are 50-task instances. On the other hand, seven of these instances have 0.25 *FR*, eight of them have 0.50 *FR,* and 18 of them have 0.75 *FR*. However, the U-shaped layout provided an improvement in total operating cost for 70% of test problems. Table 4 summaries the mean values of *PI* obtained in the experiment.

Table 4 shows that when the line is switched into the U-shaped configuration, an average improvement of 5.24% in total operating cost is obtained. That is, a U-shaped assembly line provides a 5.24% reduction in operating costs compared with an equivalent straight assembly line. This is a significant improvement. To further analyse the results, a three-way ANOVA is performed to examine the effects of three factors namely *NT*, *FR*, and *CT* on *PI* and the results are given in Table 5.

ANOVA results given in Table 5 show that the effects of *NT* and *FR* are significant at 0.01 level while the effects of *CT*, two-way interactions, and the three way interaction are not significant. The results indicate that percent improvement in total cost is influenced by the number of tasks and the strength of the precedence relationships among tasks. To clearly indicate the effects of *NT*, *FR* and *CT* on *PI*, Duncan groupings were also performed. The results of Duncan groupings with the significance level of 0.05 are given in Table 6.

Duncan grouping results suggest that, mean *PI* values for 20, 30, 40 and 50-task problems are significantly greater than that of 10-task problems. There is no significant difference between mean *PI* values for 20, 30, 40 and 50-task problems. The grouping

Table 4. Summary of experimental results (mean values of *PI*).

	$NT=10$		$NT=20$		$NT=30$		$NT=40$		$NT=50$		Total	
CT	*FR*	Mean	*FR*	Mean	*FR*	Mean	*FR*	Mean	*FR*	Mean	*FR*	Mean
30	0.25	4.054	0.25	9.204	0.25	12.066	0.25	6.539	0.25	–	0.25	7.827
	0.50	4.784	0.50	6.185	0.50	6.655	0.50	–	0.50	–	0.50	5.875
	0.75	1.041	0.75	2.852	0.75	2.806	0.75	1.864	0.75	–	0.75	2.196
	Mean	3.293		6.080		6.564		4.202				5.189
45	0.25	3.653	0.25	10.125	0.25	9.130	0.25	6.571	0.25	–	0.25	7.370
	0.50	5.158	0.50	5.279	0.50	8.805	0.50	6.654	0.50	–	0.50	6.474
	0.75	0.000	0.75	4.847	0.75	0.926	0.75	0.830	0.75	6.093	0.75	2.285
	Mean	2.937		6.750		6.287		4.685		6.093		5.214
60	0.25	0.000	0.25	8.929	0.25	10.320	0.25	9.386	0.25	7.558	0.25	7.239
	0.50	0.000	0.50	8.947	0.50	12.156	0.50	8.131	0.50	3.850	0.50	6.815
	0.75	0.000	0.75	3.115	0.75	3.810	0.75	0.638	0.75	2.108	0.75	1.934
	Mean	0.000		6.997		8.762		6.052		4.587		5.295
Total	0.25	2.569	0.25	9.419	0.25	10.310	0.25	7.773	0.25	7.558	0.25	7.429
	0.50	3.314	0.50	6.804	0.50	9.205	0.50	7.393	0.50	3.850	0.50	6.456
	0.75	0.347	0.75	3.605	0.75	2.514	0.75	0.895	0.75	3.702	0.75	2.127
	Mean	2.077		6.609		7.229		5.252		4.888		5.240

Table 5. ANOVA results of the experiment.

Source of variance	Sum of squares	*df*	Mean Square	*F*	Significance of *F*
NT	444.854	4	111.213	8.574	0.000
CT	1.433	2	0.717	0.055	0.946
FR	527.150	2	263.575	20.320	0.000
NT CT*	116.128	7	16.590	1.279	0.273
*NT*FR*	115.950	8	14.494	1.117	0.362
*CT*FR*	13.794	4	3.448	0.266	0.899
*NT*CT*FR*	85.858	11	7.805	0.602	0.821
Error	920.955	71	12.971		
Total	5327.762	110			

Table 6. Duncan groupings for experimental factors.

Duncan grouping for *NT*				Duncan grouping for *CT*			Duncan grouping for *FR*			
NT	*N*	Subset 1	Subset 2	*CT*	*N*	Subset 1	*FR*	*N*	Subset 1	Subset 2
10	27	2.077		30	28	5.189	0.75	39	2.127	
50	10		4.888	45	38	5.214	0.50	35		6.456
40	20		5.252	60	44	5.295	0.25	36		7.429
20	27		6.609							
30	26		7.229							
Sig.		1.000	0.070			0.908			1.000	0.252

Table 7. The *t*-test results for comparison of operating costs.

Line shape	*N*	Mean	Standard deviation	*t*	*p*
Straight	110	60.276	25.374	11.609	<0.01
U-shaped	110	56.848	23.790		

results for *FR* is similar to the results for *NT*. The mean *PI* values for the problems with $FR = 0.50$ and $FR = 0.25$ are significantly greater than that of problems with $FR = 0.75$. Duncan test results for *CT* are also given in Table 6.

Total operating costs for straight and U-shaped assembly lines are compared with a paired samples *t*-test. The results of the *t*-test are given in Table 7.

Table 7 shows that the difference between mean operating costs is significant at $\alpha = 0.01$ significance level. That is, mean operating cost for U-shaped assembly lines is significantly lower than that of straight assembly lines.

The results showed that in five 10-task, six 20-task, nine 30-task, six 40-task and two 50-task RDULB problems, two or more tasks, which are located at two sides of the line in the same crossover work station, required the same equipment. These problems are resolved using (M3) and results are compared with the results obtained by (M2). The same objective values are obtained in 17 of these problems. We obtained higher objective function values in the remaining 11 problems. The average increase in total operating cost for 28 RDULB problems is 0.64% that can be considered as an ignorable difference.

5. Conclusions

This study focused on balancing single model straight and U-shaped assembly lines with resource dependent task times. Integer programming formulations are proposed to solve RDALB and RDULB problem under resource restrictions. It should be emphasised that this is the first study that deals with U-shaped assembly lines with resource dependent task times. As the decision problems dealt with are essentially to assign tasks and resources to work stations simultaneously, it should also be emphasised that designing and balancing are integrated into assembly line problems.

The proposed formulations are illustrated and validated by using example problems. The advantage of U-shaped assembly lines over traditional straight assembly lines with respect to the total operating cost is also experimentally reported in this study. The average reduction in percentage improvement in the total cost of the problem instances is found as 5.24%. This in our opinion may serve as a guide for the practitioners in shaping their assembly lines. The results also showed that the percentage improvement in the total operating cost becomes significantly greater as the number of tasks increases and the flexibility ratio of the precedence relationships among the tasks gets higher.

Since assembly lines are well-known NP-Hard problems, in addition to the medium sized ones we also solved the large-sized resource dependent problems to show the efficiency of the proposed models for obtaining the optimal solutions. In terms of the solutions of the large-sized problems, both of the resource dependent models (RDALB and RDULB) are also found to be NP-Hard.

As the proposed models usable by the practitioners to some extent lose their capability of obtaining optimal solutions when the problem size reaches 50 tasks, it becomes necessary to develop effective heuristics for the larger sized problems. Developing effective heuristics for the larger sized problems is one of the future works. Developing mathematical formulations for the mixed-model straight and U-line balancing problems with resource dependent tasks times should also be considered among future works.

References

Aase, G.R., Schniederjans, M.J., and Olson, J.R., 2003. U-OPT: an analysis of exact U-shaped line balancing procedures. *International Journal of Production Research*, 41 (17), 4185–4210.

Aase, G.R., Olson, J.R., and Schniederjans, M.J., 2004. U-shaped assembly line layouts and their impact on labor productivity: an experimental study. *European Journal of Operational Research*, 156 (3), 698–711.

Becker, C. and Scholl, A., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research*, 168 (3), 694–715.

Bukchin, J. and Rubinovitz, J., 2003. A weighted approach for assembly line design with station paralleling and equipment selection. *IIE Transactions*, 35 (1) 573–585.

Bukchin, J. and Tzur, M., 2000. Design of flexible assembly line to minimize equipment cost. *IIE Transactions*, 32 (7), 585–598.

Cheng, C., Miltenburg, J., and Motwani, J., 2000. The effect of straight- and U-shaped lines on quality. *IEEE Transactions on Engineering Management*, 47 (3), 321–334.

Corominas, A., Pastor, R., and Plans, J., 2008. Balancing assembly line with skilled and unskilled workers. *Omega*, 36 (6), 1126–1132.

Erel, E. and Sarin, S.C., 1998. A survey of the assembly line balancing procedures. *Production Planning and Control*, 9 (5), 414–434.

Erel, E., Sabuncuoglu, I., and Aksu, B.A., 2001. Balancing of U-type assembly systems using simulated annealing. *International Journal of Production Research*, 39 (13), 3003–3015.

Faaland, B.H., *et al.*, 1992. Assembly line balancing with resource dependent task times. *Decision Sciences*, 23 (2), 343–364.

Ghosh, S. and Gagnon, J., 1989. A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research*, 27 (4), 637–670.

Gökçen, H. and Ağpak, K., 2006. A goal programming approach to simple U-line balancing problem. *European Journal of Operational Research*, 171 (2), 577–585.

Gökçen, H., *et al.*, 2005. A shortest route formulation of simple U-type assembly line balancing problem. *Applied Mathematical Modelling*, 29 (4), 373–380.

Hwang, R.K. and Katayama, H., 2009. A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems. *International Journal of Production Research*, 47 (14), 3797–3822.

Kara, Y., 2004. *New models for U-shaped assembly line balancing problems and an application in an automotive spare parts industry*. Turkey: Unpublished thesis (PhD). Selçuk University, Konya.

Kara, Y., Paksoy, T., and Chang, C.-T., 2009. Binary fuzzy goal programming approach to single model straight and U-shaped assembly line balancing. *European Journal of Operational Research*, 195 (2), 335–347.

Kara, Y. and Tekin, M., 2009. A mixed integer linear programming formulation for optimal balancing of mixed-model U-lines. *International Journal of Production Research*, 47 (15), 4201–4233.

McMullen, P.R. and Tarasewich, P., 2006. Multi-objective assembly line balancing via a modified ant colony optimisation technique. *International Journal of Production Research*, 44 (1), 27–42.

Miltenburg, J., 2000. The effect of breakdowns on U-shaped production lines. *International Journal of Production Research*, 38 (2), 353–364.

Miltenburg, J. and Sparling, D., 1995. *Optimal solution algorithms for the U-line balancing problem. Working paper*. Canada: McMaster University, Hamilton, Ontario.

Miltenburg, J., 1998. Balancing U-lines in a multiple U-line facility. *European Journal of Operational Research*, 109 (1), 1–23.

Miltenburg, J., 2001. U-shaped production lines: A review of theory and practice. *International Journal of Production Economics*, 70 (3), 201–214.

Miltenburg, J. and Wijngaard, J., 1994. The U-line balancing problem. *Management Science*, 40 (10), 1378–1388.

Monden, Y., 1993, *Toyota production system*. 2nd edn. Norcross, GA: Industrial Engineering Press, Institute of Industrial Engineers.

Pinto, P.A., Dannenbring, D.G., and Khumawala, B.M., 1983. Assembly line balancing with processing alternatives: an application. *Management Science*, 29 (7), 817–830.

Salveson, M.E., 1955. The assembly line balancing problem. *Journal of Industrial Engineering*, 6 (3), 18–25.

Scholl, A. and 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.

Scholl, A. and Klein, R., 1999. ULINO: Optimally balancing U-shaped JIT assembly lines. *International Journal of Production Research*, 37 (4), 721–736.

Scholl, A. and Klein, R., 1997. SALOME: A bidirectional branch-and-bound procedure for assembly line balancing. *INFORMS Journal on Computing*, 9 (4), 319–334.

Sparling, D. and Miltenburg, J., 1998. The mixed-model U-line balancing problem. *International Journal of Production Research*, 36 (2), 485–501.

Toklu, B. and Özcan, U., 2008. A fuzzy goal programming model for the simple U-line balancing problem with multiple objectives. *Engineering Optimisation*, 40 (3), 19–204.

Urban, T.L., 1998. Optimal balancing of U-shaped assembly lines. *Management Science*, 44 (5), 738–741.