Optimizing U-Shaped Production Line Balancing Problem with Exchangeable Task Locations and Walking Times

Yiyo Kuo 1,*, Taho Yang 2 and Tzu-Lin Huang 1

1 Department of Industrial Engineering and Management, Ming Chi University of Technology, New Taipei City 243, Taiwan; m08218013@o365.mcut.edu.tw
2 Institute of Manufacturing Information and Systems, National Cheng Kung University, Tainan City 701, Taiwan; tyang@mail.ncku.edu.tw
* Correspondence: yiyo@mail.mcut.edu.tw; Tel.: +886-2-2908-9899 (ext. 3118)

Abstract: This present study proposes to solve the U-shaped production line balancing problem with two constraints. First, following the precedence relationships between any pair of tasks, the locations of tasks are exchangeable. Second, all tasks are designed to be performed in fixed locations. When two or more tasks are assigned to the same operator, the walking time between the locations has to be considered when estimating the cycle time. The proposed problem is first formulated using an integer programming formulation, which is then solved by commercial software, LINGO, to minimize the cycle time and performance of the U-shaped production line. The empirical results show that U-shaped production lines outperform traditional straight production lines. In addition, the U-shaped production line balancing problem with consideration of walking time provides an opportunity to further reduce cycle time.

Keywords: U-shaped production line; integer programming formulation; precedence relationship; walking time; line balancing problems

1. Introduction

For a single production line, two general groups depending on their configuration are commonly seen: traditional straight production lines and U-shaped production lines [1]. A traditional straight production line organizes the tasks sequentially in one direction to form stations [2]. A U-shaped production line, however, is divided into two sub-lines, namely, the entrance sub-line and the exit sub-line, and, thus, an operator may perform tasks on either one of the two sub-lines or on both sub-lines simultaneously [3]. In U-shaped production lines, if tasks assigned to a workstation are located in both sub-lines, then the corresponding operator has to move between the two sub-lines to perform combinations of tasks. In this case, the operator has to perform some production tasks on one workpiece in one sub-line sequentially and then pass the workpiece to the next adjacent operator. Then, the operator moves to another sub-line and works on another workpiece. A workstation that handles two workpieces in the same cycle is called a crossover station [4]. For example, in Figure 1, there are eight production tasks. Figure 1a presents a traditional straight production line. The workpieces flow from the left-hand side to the right-hand side; tasks A, B and C are assigned to workstation 1; tasks D, E and F are assigned to workstation 2; and tasks G and H are assigned to workstation 3. Figure 1b presents a U-shaped production line. Workpieces flow from the left-hand side to the right-hand side on the entrance sub-line and flow from the right-hand side to the left-hand side on the exit sub-line. The U-shaped production line provides more options for task assignment. For example, in Figure 1b, tasks A, B and H are assigned to workstation 1; tasks F and G are assigned to workstation 2; and tasks C, D and E are assigned to workstation 3.
U-shaped layouts have been employed in many industries [5]. The benefits associated with U-shaped production lines include better visibility of the whole production process, more communication among operators, better teamwork, lower material handling cost, fewer operator (or workstation) requirements, higher productivity, lower work-in-process inventory, less space requirements, shorter lead time, faster response to change in market demand and motivation for a higher quality level [6–10].

Many studies have confirmed that the U-shaped production line balancing problem is a very significant problem for modern production systems [11]. The aim is to assign tasks to the workstations without violating any restriction [12] so that all workstations have similar amounts of work assigned to them [13]. For example, in Figure 2, there are eight production tasks. The arrows indicate the precedence relationship between a pair of tasks. The assignment of tasks to locations in a U-shaped production line has to follow the task sequence. Based on the precedence diagram, there are many feasible assignments of tasks to locations. Figure 3 provides a feasible assignment of tasks to locations based on the precedence diagram in Figure 2.

Figure 1. An example of production types. (a): traditional straight production line; (b): U-shaped production line.

U-shaped layouts have been employed in many industries [5]. The benefits associated with U-shaped production lines include better visibility of the whole production process, more communication among operators, better teamwork, lower material handling cost, fewer operator (or workstation) requirements, higher productivity, lower work-in-process inventory, less space requirements, shorter lead time, faster response to change in market demand and motivation for a higher quality level [6–10].

Many studies have confirmed that the U-shaped production line balancing problem is a very significant problem for modern production systems [11]. The aim is to assign tasks to the workstations without violating any restriction [12] so that all workstations have similar amounts of work assigned to them [13]. For example, in Figure 2, there are eight production tasks. The arrows indicate the precedence relationship between a pair of tasks. The assignment of tasks to locations in a U-shaped production line has to follow the task sequence. Based on the precedence diagram, there are many feasible assignments of tasks to locations. Figure 3 provides a feasible assignment of tasks to locations based on the precedence diagram in Figure 2.

Figure 2. An example of precedence diagram.

Figure 3. A feasible assignment of tasks to locations.
In addition to assigning tasks to locations, the assignment of tasks to workstations is also very important. It affects the amount of work in each workstation. Moreover, when assigning tasks to workstations, the locations of the tasks should be considered at the same time or the operators will get in each other’s way. For example, in Figure 3, tasks A and H are assigned to workstation 1; tasks B, E and D assigned to workstation 2; and tasks C, F and G assigned to workstation 3. The result is shown in Figure 4. It can be seen that the walking paths of the operators in workstation 2 and 3 cross each other, an outcome that can be avoided if the task in location 7 is assigned to workstation 2. Therefore, if a location in the entrance sub-line (or exit sub-line) is close to the entrance (or exit), then the corresponding tasks cannot be assigned to the workstation that is further from the entrance (or exit). This is called “location sequence” in the present study. A task sequence and a location sequence are illustrated in Figure 5. Figure 5a illustrates a situation where the number of tasks is even, and Figure 5b illustrates a situation where the number of tasks is odd.

![Figure 4](image1.png)

**Figure 4.** An example of interruption between operators.

![Figure 5](image2.png)

**Figure 5.** An illustration of task and location sequences for U-shaped production line. (a): Number of tasks is even; (b): Number of tasks is odd.

Moreover, the distances that the operators need to walk and the required travel times to stations are also important criteria. In the case of U-shaped assembly systems with workers covering several stations, these values may be lower than those in the case of straight-line systems. However, the necessary working and walking space must be available [14]. In particular, the minimization of the walking distance problem could be a significant issue when work interference occurs. In this instance, it may have a negative impact on operator productivity. Therefore, walking times should be taken into account to derive the exact cycle time [15]. For example, in Figure 6, although all tasks assigned to locations follow the
The assembly line balancing problem was first proposed by Salveson [16]. When only one single product is considered, the type of problem is classified as a simple assembly line balancing problem (SALBP), the main characteristics of which can be found in [17,18]. Scholl and Becker [19] summarized four types of SALBPs. SALBP-1 aims to minimize the number of stations for a given cycle time. SALBP-2 aims to minimize the cycle time for a given number of workstations. SALBP-E aims to maximize efficiency, thereby simultaneously minimizing cycle time and the number of workstations. Finally, Type-F aims to find a feasible design for a given cycle time and number of workstations. This research deals with SALBP-2, which is encountered frequently when the production rhythm must be adjusted in the manufacturing process. If only one operator is assigned to a workstation in a U-shaped production line, SALBP-2 involves balancing the design of a product type for a given number of operators. This means that, for every product, the cycle times can be determined when given a different number of operators. According to the balancing results, when there is demand for the corresponding products, managers can make appropriate operator assignment decisions based on the demand and the number of available operators.

This research aims to balance a U-shaped production line by minimizing the cycle time for a given number of workstations. The task sequence has to satisfy all precedence relationships, and the location sequence has to satisfy the relationship between workstations. Moreover, the walking times between all locations in the U-shaped production line are also considered.

LINGO is a simple tool that can utilize the power of linear and nonlinear optimization to concisely formulate large problems and solve them [20]. It employs branch-and-bound methods to break a model down into many convex subregions. It also has a multistart feature that restarts the nonlinear solver from a number of intelligently generated points. This allows the solver to find a number of locally optimal points and report the best one found. Finally, LINGO can automatically linearize a number of nonlinear relationships through the addition of constraints and integer variables so that the transformed linear model is mathematically equivalent to the original nonlinear model [21]. Therefore, an integer programming formulation, which can be optimized using the software LINGO 11, is developed for the proposed problem. Some cases that have been studied in the literature are modified for testing, and the results for traditional straight production lines and U-shaped production lines are compared, with and without the consideration of walking time.

The remainder of this paper is organized as follows: Section 2 reviews the relevant literature. Section 3 presents the model formulation of the proposed problem. The details...
of the empirical illustration are discussed in Section 4. Conclusions and future research opportunities are addressed in Section 5.

2. Literature Review

Miltenburg and Wijngaard [22] conducted the first study that proposed an exact methodology for UALBPs [23]. They developed a mathematical model of a UALBP, and then they proposed a dynamic programming procedure to balance small-sized problems. For larger problems, a greedy heuristic based on ranked positional weight was proposed. The UALAP then became a classic problem [24] and has been widely studied in the literature [25]. Hwang and Katayama [26] dealt with workload balancing problems in mixed-model U-shaped lines and designed an amelioration structure with a genetic algorithm (ASGA) to improve it. To implement the proposed algorithm, experiments were carried out on three test problems and a small-/medium-sized subsidiary factory of a major boiler-producing company in Japan. Rabbani et al. [25] developed a genetic algorithm (GA) to balance a mixed-model U-shaped production system. They assumed that extra time is required for crossover workstations. In their proposed mathematical model, the objective was to minimize the number of crossover stations and maximize line efficiency at the same time. Avikal et al. [2] proposed a critical path method (CPM) to assign tasks to workstations in a U-shaped assembly line. They evaluated labor productivity in U-shaped line systems and straight-line systems, and they showed that the proposed CPM can reduce the minimum number of workstations for the assignment of tasks in a U-shaped line layout below the number needed in a straight-line layout. Nourmohammadi et al. [27] proposed an imperialist competitive algorithm (ICA) to simultaneously minimize the number of workstations and the variation in workload in a UALBP. The computational results show that the proposed algorithm outperforms GA, especially in large-sized test problems. Jayaswal and Agarwal [4] proposed a simulated annealing (SA) algorithm for a resource-dependent UALBP, in which the duration of the task is dependent on equipment type and any assistance available. However, regular operators are available in a sufficient number to operate workstations but equipment and assistants are limited. The results show that the proposed SA algorithm is able to efficiently solve most small-to-moderate-sized problems to optimality or close to optimality. Hazir and Dolgui [5] dealt with a robust UALBP, in which operation times were modeled using intervals, and a robust optimization model was developed. An iterative approximate algorithm was proposed and tested with some computational experiments. They concluded that the model and solution algorithm developed could form a good base for further research on the robust design and balancing of assembly lines, and related results can be found in [28,29]. Oküz et al. [23] assumed that all workers carry out all tasks with unequal performance levels and formulated a mathematical model for worker assignment in a UALBP. Then, an Artificial Bee Colony Algorithm and a genetic algorithm were used to solve it. Li et al. [3] dealt with a UALBP and formulated an integer programming model. In the model, it is assumed that each task’s time depends on the performance of the worker who performs it and that it is different for each worker. Moreover, a worker may not be capable of performing all of the tasks. They proposed a novel heuristic approach based on multiple rules to minimize the cycle time with a given number of stations. Sahin and Kellegoz [30] developed an efficient solution method for a UALBP. They compared six different solution methods. The results show that the proposed grouping genetic algorithm (GGA) with problem-specific crossover and mutation operators produces better solutions than the mathematical formulations and other solution methods available in the literature to solve different types of assembly line balancing problems. Aydogan et al. [1] proposed a novel particle swarm optimization algorithm to solve a UALBP with stochastic task times. The results of the computational study show that the proposed approach performs quite effectively. Zhang et al. [31] dealt with task allocation and worker assignment in a UALBP, in which the processing time of each task depended on the workers. Then, an enhanced migrating birds optimization algorithm (EMBO) was used to minimize the cycle time. The numerical results demonstrate
that the proposed algorithm outperforms other algorithms. Zhang et al. [24] formulated a UALBP with worker assignment to simultaneously minimize cycle times and ergonomic risks. In the proposed problem, operators can perform all tasks with potentially different processing times, and each task is executed with a type of posture (position and movement of the upper limbs of an operator). A Restarted Iterated Pareto Greedy algorithm was designed to optimize both objectives. The proposed multi-objective algorithm outperforms existing methods on a large number of benchmark instances. In all the above literature, walking time is not considered.

Studies that deal with walking time in UALBPs are rare. Ohno and Nakade [15] dealt with a U-shaped production line and considered processing, operation and walking times. They formulated the optimal worker allocation problem, which minimized the overall cycle time and showed the advantages of the U-shaped layout over the linear layout. Nakade and Ohno [32] considered the optimal worker allocation problem for a U-shaped production line. They proposed an algorithm to find the optimal allocation of workers to machines that minimizes the cycle time for the minimum number of workers. Shewchuk [33] addressed the worker allocation problem for lean U-shaped production lines. A mathematical model that takes into account walking time, with workers following circular paths and walking around other workers, was developed. A heuristic algorithm was developed to optimize the allocation of workers.

The above literature related to U-shaped production lines deals with walking time, but all the studies focus on the worker allocation problem in which the locations of machines (or tasks) are fixed. Therefore, they ignore the alternative locations of tasks that follow precedence relationships. The present research deals with a U-shaped production line balancing problem in which the precedence relationship between tasks and walking times between the locations of tasks are simultaneously taken into consideration. The problem that the present study addresses is not found in the literature. This is the main contribution of this research.

3. Integer Programming Formulation

The main purpose of the proposed integer programming formulation is to minimize the cycle time of a U-shaped production line for a given number of operators. A description of the proposed problem is summarized as follows:
1. Only one product (or model) is considered at a time.
2. All precedence relationships between tasks (precedence diagram) are given, and the task sequence must follow it; therefore, the number of tasks is also known.
3. Production times of all tasks are deterministic and given.
4. The number of positions is equal to the number of tasks.
5. All positions that tasks can be assigned to are fixed, given and form a U shape, such as the numbers in Figure 5.
6. Location sequences are given and must be followed to keep from interruption between operators.
7. Walking times between all pairs of positions are deterministic and given.
8. Workpieces are loaded and unloaded in the entrance and exit points of the U-shaped production line.
9. One operator is assigned to all workstations. Therefore, the cycle time of a workstation is the sum of the total production times of tasks that are assigned and the corresponding walking times in addition to the walking paths.
10. Once the operators complete a task, the operator then walks to the next position and performs the corresponding task that is assigned to the same workstation. Finally, the operator will walk back to the position of the first task. Thus, the walking paths of all operators are circular. If only one task is assigned to a workstation, the corresponding operator will perform the task without walking.
11. The cycle time of the production line is equal to the maximum cycle time of all workstations.
For the development of the integer programming formulation, the notation is defined in Table 1.

Table 1. Definition of notations.

| Indices  | Parameters                                      | Decision variables                                      |
|----------|-------------------------------------------------|---------------------------------------------------------|
| $t, r, s$ | task index                                      | $CT$ cycle time                                         |
| $l, p, q$ | location index                                 | $AT_w$ total production time of tasks assigned to workstation $w$ |
| $w$      | workstation index                              | $WT_w$ total walking time in workstation $w$           |
| $T$      | number of tasks                                 |                                                         |
| $L$      | number of locations (equal to number of tasks)  |                                                         |
| $W_{\text{max}}$ | number of workstations (equal to number of operators available) |                                                         |
| $(r, s) \in PR$ | a precedence relationship between tasks; production task $r$ must be completed before task $s$ starts |                                                         |
| $(p, q) \in LR$ | a precedence relationship between locations; location $p$ must be assigned to a workstation no further from the entrance (or exit) than that to which location $q$ is assigned | $d_{pq}$ walking time for the path between location $p$ and $q$ |
| $O_t$    | production time of task $t$                    |                                                         |

Minimize $CT$  

\[
\sum_{l=1}^{L} x_{tl} = 1 \quad \forall \, t 
\]  

\[
\sum_{t=1}^{T} x_{tl} = 1 \quad \forall \, l 
\]  

\[
\sum_{t=1}^{T} \sum_{l=1}^{L} l \times x_{tl} \leq \sum_{l=1}^{L} l \times x_{sl} \quad \forall \, (r, s) \in PR 
\]  

\[
\sum_{w=1}^{W_{\text{max}}} y_{lw} = 1 \quad \forall \, l 
\]  

\[
\sum_{w=1}^{W_{\text{max}}} w \times y_{pw} \leq \sum_{w=1}^{W_{\text{max}}} w \times y_{qw} \quad \forall \, (p, q) \in LR 
\]  

\[
\sum_{t=1}^{T} \sum_{l=1}^{L} O_t \times x_{tl} \times y_{lw} = AT_w \quad \forall \, w 
\]  

\[
y_{pw} \times y_{qw} = z_{pqw} \quad \forall \, w, \quad q - p = 1 
\]  

\[
y_{pw} \times \prod_{l=p+1}^{q-1} (1 - y_{lw}) \times y_{qw} = z_{pqw} \quad \forall \, w, \quad q - p \geq 2 
\]  

\[
y_{pw} \times \prod_{l=p+1}^{q-1} (1 - y_{lw}) \times \prod_{l=1}^{p-1} (1 - y_{lw}) \times y_{qw} = z_{pqw} \quad \forall \, w, \quad p - q \geq 1, \quad p = L, \quad q \neq 1 
\]  

\[
y_{pw} \times \prod_{l=1}^{q-1} (1 - y_{lw}) \times y_{qw} = z_{pqw} \quad \forall \, w, \quad p - q \geq 1, \quad p = L, \quad q \neq 1 
\]
\[ y_{pw} \times \prod_{l=p+1}^{L}(1-y_{lw}) \times y_{qw} = z_{pqw} \quad \forall \ w, \ p-q \geq 1, \ p \neq L, \ q = 1 \]  
\[ y_{pw} \times y_{qw} = y_{pqw} \quad \forall \ w, \ p = L, \ q = 1 \]  
\[ \sum_{p=1}^{L} \sum_{q=1}^{L} z_{pqw} \times d_{pq} = W_{T_w} \quad \forall \ w \]  
\[ AT_{w} + W_{T_w} \leq CT \quad \forall \ w \]  
\[ x_{wl}, y_{lw}, z_{pqw} \in \{0, 1\} \]

The integer programming formulation aims to minimize the cycle time as given in Equation (1). The cycle time is the maximum cycle time of all workstations, which is presented in Equation (15). Equation (2) ensures that each assembly task is only assigned to one location, and Equation (3) ensures that only one task is assigned to each location. Based on the restriction of Equations (1) and (2), \( \sum_{l=1}^{L} l \times x_{tl} \) can calculate the location that task \( t \) is assigned to. If assembly task \( r \) has to be carried out before assembly task \( s \) based on the assembly sequence, Equation (4) avoids assigning assembly task \( r \) to the location after assembly task \( s \) based on task sequence as illustrated in Figures 1 and 5. Equation (5) ensures that each location is assigned to only one workstation. As with the logic of Equation (4), \( \sum_{w=1}^{W_{max}} w \times x_{lw} \) can calculate the workstation that location \( l \) is assigned to. In Equation (6), based on the location sequence, if location \( p \) is closer to the entrance (or exit) than location \( q \) in the U-shaped assembly line, location \( q \) will not be assigned to the workstation closer to the entrance (or exit) than location \( p \). If task \( t \) is assigned to location \( l \) and location \( l \) is assigned to workstation \( w \), task \( t \) is assigned to workstation \( w \). Equation (7) is used to calculate the assembly times for all workstations.

Equations (8)–(13) determine the walking paths of all operators in their corresponding workstations. Equation (8) adjusts the walking path between adjacent locations \( p \) and \( q \), and \( q - p = 1 \). Equation (9) adjusts the walking path between nonadjacent locations \( p \) and \( q \), and \( q - p \geq 2 \). When an operator completes all the tasks assigned to their workstation, they have to walk back to the first location of the workstation. Equations (10)–(13) adjust the walking back path between locations \( p \) and \( q \), and \( p - q \geq 1 \). Equation (10) adjusts the path when \( p \neq L \) and \( q \neq 1 \). Equation (11) adjusts the path when \( p = L \) and \( q \neq 1 \). Equation (12) adjusts the path when \( p \neq L \) and \( q = 1 \). Equation (13) adjusts the path when \( p = L \) and \( q = 1 \). For example, in Figure 1b, there are three workstations, of which workstations 1 and 3 are crossover workstations. Equation (8) can determine \( Z_{121}, Z_{672}, Z_{343} \) and \( Z_{453} \); Equation (9) can determine \( Z_{281} \); Equation (10) can determine \( Z_{262} \) and \( Z_{333} \); and Equation (13) can determine \( Z_{831} \). Equations (11) and (12) are not relevant to the walking paths in Figure 2. For a better understanding, Figure 7 illustrates the paths that correspond to each of the equations from (8) to (13). The numbers in parentheses indicate the corresponding equation numbers.

![Figure 7. Illustration of walking paths.](image)

Equation (14) calculates the walking times of operators in all workstations. The total production time and walking time in a workstation is the cycle time of the workstation and cannot exceed the cycle time given in Equation (15). Equation (16) restates the definitions of binary variables.
4. Experimental Results

The proposed integer programming formulation was validated on 13 problems in the literature. The corresponding information about the problems is summarized in Table 2. Because the walking times and layouts were ignored in previous studies, the present research adopted the ratios of average processing time as the walking time of one unit. All U-shaped assembly lines are laid out in a grid system, as illustrated in Figure 8a.

Table 2. Summary of experimental problems.

| No. | Source | Number of Tasks | Average Processing Time |
|-----|--------|----------------|------------------------|
| 1   | Bowman [34] | Figure 1 | 8 | 9.38 |
| 2   | Chen and Plebani [35] | Figure 1 | 12 | 5 |
| 3   | Hwang and Katayama [26] | Figure 1 | 9 | 4.11 |
| 4   | Hwang and Katayama [26] | Figure 2 | 10 | 4.1 |
| 5   | Hwang and Katayama [26] | Figure 3 | 7 | 3.29 |
| 6   | Hwang and Katayama [26] | Figure 4 | 11 | 4.18 |
| 7   | Avikal et al. [2] | Figure 3 | 7 | 5.29 |
| 8   | Li et al. [3] | Figure 1 | 10 | 3.6 |
| 9   | Oksuz et al. [23] | Figure 1 | 9 | 6.11 |
| 10  | Kuo and Liu [36] 1 | Figure 3 | 8 | 0.38 |
| 11  | Kuo and Liu [36] 2 | Figure 4 | 7 | 0.58 |
| 12  | Kuo et al. [37] | Figure 1 | 8 | 0.28 |
| 13  | Kuo et al. [37] | Figure 2 | 8 | 0.48 |

Figure 8. Layout of the U-shaped assembly cells. (a): Number of tasks is odd; (b): Number of tasks is even.

The walking time between any adjacent locations is one unit, and the minimal walking time between both sides of a cell is two units. In this research, the walking time of one unit is the ratio of the average processing time, ranging between 0% and 20%. For example, in problem 1, proposed by Bowman [34], there are eight tasks, and the average processing time is 9.38. If one unit of walking time is 5% of the average processing time, the walking time between any adjacent locations is 0.469 (9.38 × 5%). To ensure that the walking time between adjacent locations is one unit as in traditional straight production lines, when the number of tasks is even, the right-hand side of the cell is laid out as in Figure 8b. As all tasks are laid out in a grid system, the walking times between all pairs of locations can be easily calculated using Pythagoras’ theorem.

The purpose of layout design is to systematically generate problems that can be tested. A smaller ratio of the average processing time indicates a shorter walking time between locations. When the ratio of the average processing time is equal to zero, it means that walking times are not taken into consideration. Any real U-shaped production layout can be tested if the position of all locations (location sequence) and all walking times between any pairs of locations (\(d_{pq}\)) are given.
For the 13 problems, two, three and four workstations are tested using the software LINGO on an Intel Core i7 2.40 GHz notebook computer. Fortunately, the proposed integer programming formulation can be adopted to balance traditional straight production lines, because the position of all locations (location sequence) and all walking times between any pairs of locations are known. The location sequence and task sequence are illustrated in Figure 9. Therefore, this research also adopts the proposed integer programming formulation to balance traditional straight production lines with the consideration of walking times. A comparison between traditional straight production lines and U-shaped production lines is shown in Table 3.

![Figure 8. Layout of the U-shaped assembly cells. (a): number of tasks; (b): number of workstations](image-url)

Figure 8. Layout of the U-shaped assembly cells. (a): number of tasks; (b): number of workstations

![Figure 9. An illustration of task and location sequences for traditional straight assembly lines.](image-url)

Figure 9. An illustration of task and location sequences for traditional straight assembly lines.

| # Problems | 0%  | 5%  | 10% | 15%  | 20%  |
|------------|-----|-----|-----|------|------|
| 1          | 2.44| 4.47| 6.2 | 7.68 |
| 2          | 1.54| 2.86| 4   | 5    |
| 3          | 0.78| -   | 1.02| 2.62 |
| 4          | 2.36| 2.98| 4.19| 9.89 |
| 5          | -   | -   | -   | 2.1  |
| 6          | 1.48| 2.76| 3.08| 4.94 |
| 7          | 3.32| 4.51| 4.21| 3.95 |
| 8          | 0.71| 1.32| 2.78| 4.21 |
| 9          | 1.09| 1.56| 2.17| 3.82 |
| 10         | 2.34| 2.81| 2.64| 2.49 |
| 11         | 1   | 2.17| 3.29| 3.65 |
| 12         | 17.73| 0.2 | 0.28| 0.35 |
| 13         | 1.04| 4.13| 5.84| 7.3  |
| average    | 1.49| 1.63| 2.42| 3.17 |

Table 3. Improvement in U-shaped production lines (percentage).
In Table 3, the % indicates the ratio of the average processing time for one unit. Higher ratios indicate longer walking times between locations. In Table 3, from a total of 195 (13 × 5 × 3) cases, the U-shaped production line results in shorter cycle times in 131 cases. For different numbers of workstations and walking times, the average improvements range between 1.02% and 4.53%. Therefore, the comparison results confirm some of the arguments for the advantage of U-shaped production lines found in the literature.

Moreover, the results of 0% for both traditional straight and U-shaped production lines are based on walking times that are 0% of the average processing time. This means that these results correspond to calculations ignoring walking time. Therefore, if the walking time is added onto these results, the real performance without taking walking time into consideration can be evaluated. Table 4 shows the improvements when taking walking time into consideration.

In Table 4, there are 156 (13 × 4 × 3) cases that take walking time into consideration for both traditional straight and U-shaped production lines. For traditional straight production lines, there are 55 cases where performance improves when walking time is considered. For U-shaped production lines, 121 cases result in a shorter cycle time when walking time is considered. This means that the probability of improvement produced by taking walking time into account for U-shaped production lines is higher than that for traditional straight production lines. Moreover, longer walking times give more improvement on average for both traditional straight and U-shaped production lines.

For individual cases, some unexpected results were found. For example, when there were two workstations, the improvements in problems 1, 5, 10 and 12 over a traditional straight production line were very obvious. The improvement is even better than for a U-shaped production line. Similar results can also be found in problem 13 with three workstations and problem 10 with four workstations. These results should encourage the managers of traditional straight production lines to try to balance the line by considering walking times if it is difficult to transfer the production lines into U-shaped production lines temporarily. Moreover, for problem 8 with four workstations, the improvements in U-shaped production lines when considering walking times are most obvious when the walking time of one unit is 5% of the average processing time. This case should also encourage managers to try to balance U-shaped production lines by taking walking times into consideration, even if the scale of the production line is small and the walking time is very short.

The average computation times for different problems and a different number of workstations are illustrated in Figure 10. The computation times are highly dependent on the number of workstations. In every problem, more workstations led to longer computation times. As shown in Table 2, the number of tasks in problems 2, 4, 6 and 8 are 12, 10, 11.
and 10, respectively. The computation times for these four problems are longer than those for other problems. In particular, when there are four workstations, computation times exceed 1000 s. Even though longer computation times are required for larger problems, using LINGO to optimize the proposed integer programming formulation is still practical for products of mass production or long-term demand. Once the optimal line balance configurations of a product with a different number of workstations are completed, these configurations will become useful information. Every time that there is demand for a corresponding product, managers can select a configuration for it based on the demand and the number of operators available without optimizing the line balance problem again.

Table 4. Improvement when considering walking times (percentage).

| # Problems | 5% Traditional Straight Production Line | 10% Traditional Straight Production Line | 15% Traditional Straight Production Line | 20% Traditional Straight Production Line | 5% U-Shaped Production Line | 10% U-Shaped Production Line | 15% U-Shaped Production Line | 20% U-Shaped Production Line |
|------------|----------------------------------------|----------------------------------------|----------------------------------------|----------------------------------------|-----------------------------|-------------------------------|-------------------------------|-----------------------------|
| 1          | 8.74                                   | 8.08                                   | 7.51                                   | 7.02                                   | -                           | -                            | -                            | -                           |
| 2          | -                                      | -                                      | -                                      | -                                      | 3.33                        | 4.75                          | 6.61                          | 8.22                        |
| 3          | -                                      | -                                      | -                                      | -                                      | 2.36                        | 2.98                          | 4.19                          | 9.89                        |
| 4          | 2.53                                   | 4.7                                    | 6.59                                   | 6.27                                   | 1.58                        | 2.96                          | 4.18                          | 5.26                        |
| 5          | -                                      | -                                      | -                                      | -                                      | 1.58                        | 2.94                          | 3.34                          | 5.26                        |
| 6          | -                                      | -                                      | -                                      | -                                      | 2.03                        | 2.61                          | 3.47                          | -                           |
| 7          | -                                      | -                                      | -                                      | -                                      | 0.95                        | 1.91                          | -                            | -                           |
| 8          | -                                      | -                                      | -                                      | -                                      | 0.25                        | 0.94                          | -                            | 1.14                        |
| 9          | 5.56                                   | 7.1                                    | 8.41                                   | 9.54                                   | 5.09                        | 5.29                          | 4.9                           | 4.55                        |
| 10         | -                                      | -                                      | -                                      | -                                      | 1                           | 1.4                           | 1.97                          | -                           |
| 11         | 22.2                                   | 21.78                                  | 21.99                                  | 22.17                                  | 5.7                         | 5.39                          | 6.1                           | 6.56                        |
| 12         | -                                      | -                                      | -                                      | -                                      | 0.37                        | -                             | -                            | -                           |
| average    | 3                                      | 3.2                                    | 3.42                                   | 3.49                                   | 1.53                        | 2.07                          | 2.6                           | 3.59                        |
| 1          | -                                      | -                                      | -                                      | -                                      | 5.33                        | 6.47                          | 7.65                          | 9.26                        |
| 2          | 2.08                                   | 3.85                                   | 5.36                                   | 6.67                                   | 7.14                        | 11.02                         | 11.77                         | 12.41                       |
| 3          | 2.89                                   | 5.32                                   | 7.39                                   | 9.17                                   | 0.87                        | -                             | -                            | -                           |
| 4          | -                                      | -                                      | 1.3                                    | 3.38                                   | -                           | 1                             | 1.4                           | 1.97                        |
| 5          | -                                      | -                                      | -                                      | 2.96                                   | 3.8                         | 3.45                          | -                            | 2.96                        |
| 6          | -                                      | -                                      | -                                      | -                                      | 5.5                         | 6.95                          | 8.43                          | 10.4                        |
| 7          | 3.76                                   | 6.99                                   | 9.81                                   | 11.61                                  | 5.59                        | 2.27                          | 14.96                         | 17.84                       |
| 8          | -                                      | -                                      | -                                      | -                                      | 0.21                        | 0.39                          | 1.7                           | 3.64                        |
| 9          | -                                      | 0.98                                   | 3.4                                    | 5.49                                   | -                           | 0.51                          | 0.86                          | 2.01                        |
| 10         | -                                      | -                                      | -                                      | -                                      | 0.01                        | 0.42                          | 0.77                          | -                           |
| 11         | -                                      | -                                      | -                                      | -                                      | -                           | -                             | -                            | -                           |
| 12         | -                                      | -                                      | -                                      | -                                      | 2.6                         | 6.77                          | 9.41                          | -                           |
| 13         | 30                                     | 30.45                                  | 30.56                                  | 30.65                                  | 2.15                        | 4.57                          | 6.09                          | 7.91                        |
| average    | 2.98                                   | 3.66                                   | 4.45                                   | 5.38                                   | 2.58                        | 3.34                          | 4.8                           | 6.63                        |
| 1          | -                                      | -                                      | -                                      | -                                      | -                           | 6.01                          | 13.25                         | 19.03                       |
| 2          | 2.86                                   | 5.26                                   | 7.32                                   | 9.09                                   | 6.46                        | 6.48                          | 9.27                          | 8.79                        |
| 3          | -                                      | -                                      | -                                      | -                                      | -                           | 1.87                          | -                             | 4.85                        |
| 4          | -                                      | -                                      | 1.71                                   | 4.48                                   | 3.35                        | -                             | 1.4                           | 6.64                        |
| 5          | -                                      | -                                      | 3.64                                   | 2.58                                   | 4.65                        | 6.36                          | 7.78                          | -                           |
| 6          | -                                      | -                                      | -                                      | 6.81                                   | 6.4                         | 12.21                         | 12.8                          | -                           |
| 7          | -                                      | -                                      | -                                      | 4.78                                   | 1.03                        | 4.45                          | 22.94                         | -                           |
| 8          | -                                      | -                                      | 0.66                                   | -                                      | 10.4                        | 0.47                          | 2.73                          | 3.42                        |
| 9          | -                                      | -                                      | 1.79                                   | -                                      | 4.39                        | 6.03                          | 7.41                          | -                           |
| 10         | 13.06                                  | 15.53                                  | 16.57                                  | 17.26                                  | 2.95                        | 6.44                          | 9.11                          | 11.29                       |
| 11         | -                                      | -                                      | -                                      | -                                      | 6.17                        | 13.61                         | 19.43                         | 24.1                        |
| 12         | -                                      | -                                      | -                                      | 0.07                                   | 3.22                        | 3.81                          | 7.42                          | 11.55                       |
| 13         | -                                      | -                                      | 2.51                                   | 5.76                                   | -                           | 2.44                          | 4.72                          | 7.75                        |
| average    | 1.22                                   | 1.6                                    | 2.35                                   | 3.1                                    | 3.59                        | 4.29                          | 7.56                          | 11.41                       |

#: number of workstation.
Figure 10. Computation time.

5. Discussion

The benefit of using U-shaped production lines has been demonstrated both in industry and in research. This research deals with the problem of U-shaped production line design and aims to minimize cycle times when given the number of operators. An integer programming formulation was developed to optimize the assignment of tasks to locations and locations to workstations. Unlike other related studies, the proposed integer programming formulation takes both precedence relationships of tasks and walking times between locations into consideration. The proposed integer programming formulation was then validated on 13 problems that have been reported in the literature. As the information about walking times was not provided in these problems, it was assumed that the walking times were a proportion of the average production times. The experimental results confirm that U-shaped production lines can reduce cycle times when compared with traditional straight production lines. The results also show that when walking times are taken into consideration for balancing production lines, the advantages of U-shaped production lines are more obvious, especially when walking times are longer.

Computation time increases when the number of workstations or tasks increases. According to the experimental results, the computation times increase seriously for problems with more than 10 tasks and 4 workstations. When the demand has a small volume and a large variety, new products are always being requested by customers. Then, the proposed integer programming formulation needs to be optimized frequently. Therefore, an efficient heuristic, such as genetic algorithms, tabu search or simulated annealing, might be appropriate, and this may be an opportunity for future research. Then, more results can be compared and analyzed.

The proposed integer programming formulation was validated on 13 problems in the literature. The 13 problems were not all originally proposed for U-shaped production lines. Therefore, only precedence relationships of tasks and task times were used for validation. In this regard, the information concerning layout and walking times was designed for this research. Therefore, it is difficult to compare the results with similar problems in the literature. Applying the proposed integer programming formulation to a real case and analyzing the results of a practical application would be future work of this research.
Author Contributions: Conceptualization, Y.K. and T.Y.; methodology, Y.K.; software, Y.K.; validation, Y.K., T.Y. and T.-L.H.; formal analysis, Y.K. and T.-L.H.; investigation, Y.K.; resources, Y.K.; data curation, Y.K. and T.-L.H.; writing—original draft preparation, Y.K. and T.Y.; writing—review and editing, Y.K. and T.Y.; visualization, Y.K.; supervision, Y.K. and T.Y.; project administration, Y.K.; funding acquisition, Y.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: This work was supported in part by the Ministry of Science and Technology of Taiwan, Republic of China, under grant MOST-109-2221-E-006 -152 -MY3.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Aydogan, E.K.; Delice, Y.; Ozean, U.; Gencer, C.; Ozkan, B. Balancing stochastic U-lines using particle swarm optimization. J. Intell. Manuf. 2019, 30, 97–111. [CrossRef]
2. Avikal, S.; Jain, R.; Mishra, P.K.; Yadav, H.C. A heuristic approach for U-shaped assembly line balancing to improve labor productivity. Comput. Ind. Eng. 2013, 64, 895–901. [CrossRef]
3. Li, M.; Tang, Q.; Zheng, Q.; Xia, X.; Floudas, C.A. Rules-based heuristic approach for the U-shaped assembly line balancing problem. Appl. Math. Model. 2017, 48, 423–439. [CrossRef]
4. Jayaswal, S.; Agarwal, P. Balancing U-shaped assembly lines with resource dependent task times: A Simulated Annealing approach. J. Manuf. Syst. 2014, 33, 522–534. [CrossRef]
5. Hazir, O.; Dolgui, A. A decomposition based solution algorithm for U-type assembly line balancing with interval data. Comput. Oper. Res. 2015, 59, 123–131. [CrossRef]
6. Scholl, A.; Klein, R. ULINO: Optimally balancing U-shaped JIT assembly lines. Int. J. Prod. Res. 1999, 37, 721–736. [CrossRef]
7. Miltenburg, J. The effect of breakdowns on U-shaped production lines. Int. J. Prod. Res. 2000, 38, 353–364. [CrossRef]
8. Miltenburg, J. U-shaped production lines: A review of theory and practice. Int. J. Prod. Econ. 2001, 70, 201–214. [CrossRef]
9. Cheng, C.; Miltenburg, J.; Motvani, J. The effect of straight- and U-shaped lines on quality. IEEE Trans. Eng. Manag. 2000, 47, 321–334. [CrossRef]
10. Aase, G.R.; Olson, J.R.; Schniederjans, M.J. U-shaped assembly line layouts and their impact on labor productivity: An experimental study. Eur. J. Oper. Res. 2004, 156, 698–711. [CrossRef]
11. Simaria, A.S.; Zanella de Sá, M.; Vilarinho, P.M. Meeting demand variation using flexible U-shaped assembly lines. Int. J. Prod. Res. 2009, 47, 3937–3955. [CrossRef]
12. Pilati, F.; Ferrari, E.; Gamberi, M.; Margelli, S. Multi-manned assembly line balancing: Workforce synchronization for big data sets through simulated annealing. Appl. Sci. 2021, 11, 2523. [CrossRef]
13. Mukund Nilakantan, J.; Ponnambalam, S.G. Robotic U-shaped assembly line balancing using particle swarm optimization. Eng. Optim. 2016, 48, 231–252. [CrossRef]
14. Zulch, M.; Zulch, G. Production logistics and ergonomic evaluation of U-shaped assembly systems. Int. J. Prod. Econ. 2017, 190, 37–44. [CrossRef]
15. Ohno, K.; Nakade, K. Analysis and optimization of a U-shaped production line. J. Oper. Res. Soc. Jpn. 1997, 40, 90–104. [CrossRef]
16. Salveson, M.E. The assembly line balancing problem. J. Ind. Eng. 1955, 6, 18–25.
17. Baybars, I. A survey of exact algorithms for the simple assembly line balancing problem. Manag. Sci. 1986, 32, 909–932. [CrossRef]
18. Scholl, A.; Becker, C. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. Eur. J. Oper. Res. 2006, 168, 666–693. [CrossRef]
19. Krishnaraj, C.; Anand Jayakumar, A.; Deepa Shri, S. Solving supply chain network optimization models using LINGO. Int. J. Appl. Eng. Res. 2015, 10, 14715–14718.
20. LINGO User’s Guide; LINDO Systems Inc.: Chicago, IL, USA, 2004.
21. Miltenburg, G.J.; Wingaard, J. The U-Line line balancing problem. Manag. Sci. 1994, 40, 1378–1388. [CrossRef]
22. Oksuz, M.K.; Buyukozkan, K.; Satoglu, S.I. U-shaped assembly line worker assignment and balancing problem: A integer and two meta-heuristics. Comput. Ind. Eng. 2017, 112, 246–263. [CrossRef]
23. Zhang, Z.; Tang, Q.; Ruiz, R.; Zhang, L. Ergonomic risk and cycle time minimization for the U-shaped worker assignment assembly line balancing problem: A multi-objective approach. Comput. Oper. Res. 2020, 118, 104905. [CrossRef]
24. Rabbani, M.; Kazemi, S.M.; Manavizadeh, N. Mixed model U-line balancing type-1 problem: A new approach. J. Manuf. Syst. 2012, 31, 131–138. [CrossRef]
26. Hwang, R.; Katayama, H. A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems. *Int. J. Prod. Res.* **2009**, *47*, 3797–3822. [CrossRef]
27. Nourmohammadi, A.; Zandieh, M.; Tavakkoli-Moghaddam, R. An imperialist competitive algorithm for multi-objective U-type assembly line design. *J. Comput. Sci.* **2013**, *4*, 393–400. [CrossRef]
28. Sotskov, Y.N.; Dolgui, A.; Lai, T.C.; Zatsiupa, A. Enumerations and stability analysis of feasible and optimal line balances for simple assembly lines. *Comput. Ind. Eng.* **2015**, *90*, 241–258. [CrossRef]
29. Lai, T.C.; Sotskov, Y.N.; Dolgui, A.; Zatsiupa, A. Stability radii of optimal assembly line balances with a fixed workstation set. *Int. J. Prod. Econ.* **2016**, *182*, 356–371. [CrossRef]
30. Sahin, M.; Kellegoz, T. An efficient grouping genetic algorithm for U-shaped assembly line balancing problems with maximizing production rate. *Memetic Comput.* **2017**, *9*, 213–229. [CrossRef]
31. Zhang, Z.; Tang, Q.; Han, D.; Li, Z. Enhanced migrating birds optimization algorithm for U-shaped assembly line balancing problems with workers assignment. *Neural Comput. Appl.* **2019**, *31*, 7501–7515. [CrossRef]
32. Nakade, K.; Ohno, K. An optimal walker allocation problem for a U-shaped production line. *Int. J. Prod. Econ.* **1999**, *60–61*, 353–358. [CrossRef]
33. Shewchuk, J.P. Worker allocation in lean U-shaped production lines. *Int. J. Prod. Res.* **2008**, *46*, 3485–3502. [CrossRef]
34. Bowman, E.H. Assembly line balancing by linear programming. *Oper. Res.* **1960**, *8*, 385–389. [CrossRef]
35. Chen, S.; Plebani, L. Heuristic for balancing U-shaped assembly lines with parallel stations. *J. Oper. Res. Soc. Jpn.* **2008**, *51*, 1–14. [CrossRef]
36. Kuo, Y.; Liu, C.C. Operator assignment in a labor-intensive manufacturing cell considering inter-cell manpower transfer. *Comput. Ind. Eng.* **2017**, *110*, 83–91. [CrossRef]
37. Kuo, Y.; Chen, Y.P.; Wang, Y.C. Operator assignment with cell loading and product sequencing in labour-intensive assembly cells—A case study of a bicycle assembly company. *Int. J. Prod. Res.* **2018**, *56*, 5495–5510. [CrossRef]