A Heuristic Method for Two-sided Assembly Line Reconfiguration in IoT Environment

Xiaofeng Hu, Chuanxun Wu, Yahui Zhang

School of Mechanical Engineering, Shanghai Jiao Tong University, China
wshxf@sjtu.edu.cn, wuchuanxun@outlook.com, zhangya0123@126.com

Abstract

This paper focuses on the two-sided assembly line reconfiguration problem when the market demands change. Firstly, the Internet of Things (IoT) based framework is proposed to support the collaborative reconfiguration of the workers, tools and parts logistic related with the reassigned task by linking the physical workstations and the cyber ones in the two-sided assembly line. Then, a heuristic method based on the position-oriented enumerative procedure is proposed to deal with the two-sided assembly line rebalancing problem to compute the reassigned tasks among the workstations. The objective is to minimize the tasks reassignment, which keeps the new lines as close as possible to the original ones to minimize the reconfiguration costs. Finally, the proposed algorithm is tested on the benchmark problems, and the results demonstrate the effectiveness of the proposed algorithm.

Keywords: Production engineering, Minimization, Internet of Things, Two-sided assembly line, Heuristic

1 Introduction

Two-sided assembly line (TAL) is typically used to assemble large-scale products, for example, trucks and buses [1]. In the literature, most researchers have addressed TAL balancing problems (TALBPs), which can be categorized into TALBP-1 and TALBP-2. The main objective of TALBP-1 [2-7] is to minimize the line length under the cycle time constraint, while the primary objective of TALBP-2 is to minimize the cycle time for a predetermined line length [8-12]. The TALBP-1 to be solved involves the new assembly line installation. In industrial practices, more attention has been paid to the reconfiguration of the existing lines, instead of the new line installation [13]. Since the workstations have been identified in the existing lines, TALBP-2 is regarded as the one that more suitable for the assembly lines reconfiguration. However, the TALBP-2 focuses on the optimization of cycle time and ignores the reconfiguration costs, which are incurred by the change of tasks assignment in the existing assembly line.

Assembly line rebalancing problem (ALRBP) was first defined and researched by Gamberini et al. [14]. Because of some modifications, including the changes of the manufacturing process, the processing time of tasks and the cycle time, tasks assignment of the previous assembly lines should be adjusted without violating the new precedence constraints among the tasks and the reset cycle time restriction to get a new efficient balance. Inevitably, some rebalance costs will be incurred in this process. Such as, workers retraining cost, switching cost of tools, storage racks and equipment, as well as, quality assurance cost. Those costs are directly affected by the amount of changes in tasks assignment. Thus, minimizing tasks re-assignment is considered as a measure of the ability to respond quickly to changes, which is the essential question of the rebalancing problem. The rebalancing problem is different from the previous balancing problem, as the cost arising from the modification to the existing TAL has to be considered. The rebalancing problem is to find a better method to achieve an efficient balance of the assembly line again with minimal rebalancing cost.

In the literature, a few attention has been paid to rebalancing existing assembly lines. Grangeon et al. [15] developed three heuristics to rebalance the existing vehicle assembly line, and the considered objectives were to minimize the number of workstations and the reassigned tasks. Yang et al. [16] developed a genetic algorithm to solve the mix-model OALrBP with the objective of minimizing the line length and the total processing time of reassigned tasks. An ant colony algorithm was developed to deal with the U-lines rebalancing problem to minimize the modification cost, considering the tasks reassignment and the workstation reconfiguration [17]. The mathematical model and exact method were proposed to solve the OALrBP with the objective of minimizing the line length and the reassigned tasks, which reduced the time and investments needed to rebuild the existing assembly line [18]. Li and Boucher [19] addressed the

*Corresponding Author: Xiaofeng Hu; E-mail: wshxf@sjtu.edu.cn
DOI: 10.3966/160792642020092105022

balancing problem of the automated one-sided assembly lines considering learning effect. The authors proposed that the learning effect of tasks led to changes of task times and the balance of assembly line was broken, therefore it needed dynamically rebalance the assembly line. Sancı and Azizoğlu developed a mixed integer linear programming-based algorithm to solve the OALrBP with the objective of minimizing the number of the reassigned tasks and cycle time [20]. All the aforementioned papers tackled the rebalancing problems of one-sided lines.

The two-sided assembly line re-balancing problem (TALrBP) was first addressed by Bartholdi [1]. He actually developed an interactive computer program to rebalance the real TAL, and the user had consistent sustained control from solution to solution in order to find the new solution close to the previous one. No cost related to the tasks reassignment was mentioned. A modified genetic algorithm was proposed to solve a real-life TALrBP with the change of market demand and assembly process [21]. But the proposed algorithm was not tested on benchmark problems. In other words, there are no published research results on TALrBP with benchmark problems in the literature so far.

However, no attention had been paid to the worker reallocation, tool (e.g. fixtures and jigs) and storage rack transfer, AGV route adjustment of parts logistic and operation instructions modification for different workstations when the tasks reassignment occurred. It is difficult to reconfigure the TALs without the current line deployment and the associated information of reassigned tasks. Fortunately, many advanced Internet of Things (IoT) technologies have been designed and adopted in recent years [22-25], especially, it has been used in assembly lines to capture the real-time status for production management [26-28]. Cohen et al. evaluate the possible evolution of the assembly system, since IoT and cloud computing technologies permit to interconnect the different parts of an assembly system [29]. Bortolini et al. discuss the assembly system design and management in the era of Industry 4.0, and investigate the application of the IoT technologies in the next generation of assembly system, which can be automatically configured by the assembly control system [30]. Thramboulidis et al. present a framework for cyber-physical assembly systems by integrating IoT technologies with the micro service architecture [31]. And they expand the above framework and utilize IoT technologies to connect with each constituent component in the assembly system [32]. Liu et al. propose the concept of IoT-enabled intelligent one-sided assembly system in order to improve the efficiency and intelligence of the assembly system [33]. IoT technology can be adopted to support the collaborative reconfiguration among the workstations in the TALs.

This paper presents an IoT-based framework for TALs reconfiguration to support the adjustment of the related resource deployment when tasks are reassigned. Then, a modified beam search algorithm is proposed to solve the TALrBPs with the objective of minimizing the tasks reassignment in the rebalanced line to reduce the costs required to rebuild the previous TAL. And the performance of the proposed algorithm is tested on all benchmark problems. The paper is organized as follows. Section 2 presents the IoT-based reconfiguration framework for TALrBPs. Section 3 proposes the heuristic algorithm for the TALrBPs with the objective of minimizing the tasks reassignment. Section 4 reports the computational results. Finally, Section 5 presents the conclusions. The major symbols are listed in Table 1.

Table 1. Principal symbols

Symbol	Description
CT_0	Initial cycle time
CT_R	New cycle time of a rebalancing assembly line
SI_0	Smoothing index of an initial assembly line
SI_R	Smoothing index of a rebalancing assembly line
t_i	Processing time of task i
(q,d)	A workstation of position q and its operation direction d, $d = 1$ indicates the left side and $d = 2$ indicates the right side.
i,j	Task index
S_i	Task type (L, R or E)
N	Number of tasks
NR	Number of reassigned tasks
LBT_i	Lower bound of tasks reassignment when task i is reassigned.
k,w q	Workstation index Position index, $q=[k/2]$
TIB_q	Set of tasks assigned to position q in an initial solution
TNB_q	Set of tasks assigned to position q in a new solution
UT	Set of unassigned tasks
M	Number of positions
K	Number of workstations, $K = 2M$
Pre_i	Set of direct predecessors of task i
DS_i (AS_i)	Set of direct (all) successors of task i
WS_k	Set of tasks assigned to workstation k
ST_k	Workstation time at workstation k, $ST_k = \sum_{i \in WS_k} t_i$
ST_{max}	Maximum value of ST_k
PT_q	Workstation time at position q, $PT_q = ST_{2q-1} + ST_{2q}$
$\overline{T}$	Average value of ST_k, $\overline{T} = \sum_{i=1}^{N} t_i \Big/ M$
FT_i	Finish time of task i
FTB_i^w	Finish time of the task before task i in workstation w
x_{iqd}	$x_{iqd} = 1$, if task i is assigned to workstation (q,d); $x_{iqd} = 0$, otherwise.
z_{ip}	$z_{ip}=1$, if task i is assigned earlier than task p in the same station; $z_{ip}=0$, if task p is assigned earlier than task i in the same station.

2 Problem Definition

In the TAL, the preferred operation direction constraint works together with the precedence and cycle time restrictions to make the idle time of a cycle is sometimes unavoidable. For example, the feasible solution is shown in Figure 2 for the problem P16 shown in Figure 1. So, rebalancing TAL needs to consider the sequence-dependent completion time of tasks. Then the finish time of tasks assigned to workstations in TALs can be calculated by using equation (1):

$$FT_i = \max\left(\max_{j \in \Pr e_i} \left(FT_j \right), FTB_i^w \right) + t_i \tag{1}$$

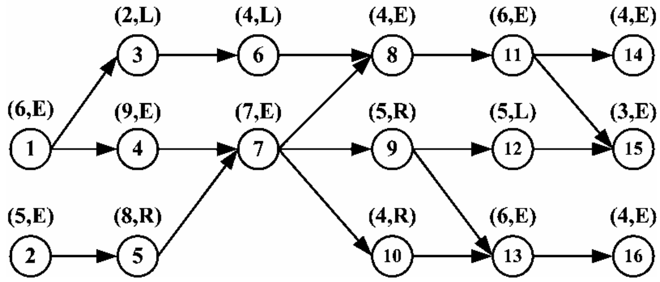


Figure 1. Example problem P16

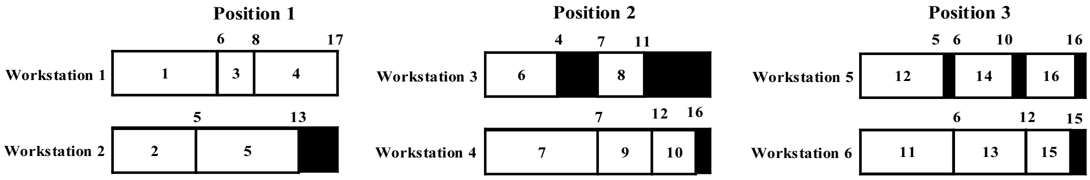


Figure 2. Feasible Solution

For simplicity, the finish time of the last task within a workstation is called the workstation completion time (WCT). The cycle time of TALs is determined by the maximum value of WCT, not the maximal workstation time because of the idle time of a workstation.

Since the WCT of each workstation is determined by the task completion time in a pair of workstations at the position, which consists of two directly facing workstations on both sides of the line (See Figure 2), the tasks reassignment in two workstations of the position should be considered simultaneously, which can reduce the WCT of the workstation. Consider the example of Figure 2, where task 14 assigned to workstation 5 and task 13 assigned to workstation 6 can be traded, and the WCT of each workstation is reduced. In such case, the costs related tasks movement are not incurred, because the workers retraining and the equipment switching are both unnecessary. Therefore, the objective of the problem we solved is to minimize the number of task reassignment (NTR) among positions, and can be represented by the following equation:

$$\text{Minimize} \quad NTR = N - \sum_{q=1}^{M} \text{cardinality}(TIB_q \cap TNB_q) \tag{2}$$

Eventually, the TALrBP solved in this paper can be described as follows. As continuous changes in volume demand, the previously balanced assembly line has been inefficient, and the assembly line reconfiguration is necessary to be carried out to make lines get a novel and high-efficiency balance in the new production environment via the lowest rebalancing cost. However, in actual applications, these cost factors are hard to evaluate for each possible tasks reassignment, but directly related to the number of tasks reassignment. Moreover, considering the particularity of TALrBP that mentioned above, minimizing the number of tasks reassignment among positions (NTR) as the objective. It is assumed that the initial feasible solution of TAL is predetermined. The main work we need do is effectively reassigning some tasks to another workstation, so that the tasks reassignment is minimized while meeting the constraints, including the precedence among tasks, cycle time, and the operation direction [34]. Constraints in the mathematical model are shown below. Constraint (3) indicates that a task can only be assigned to one workstation. Constraint (4) is the cycle time constraint, guarantees tasks are completed within the cycle time. Constraints (5) and (6) are the precedence constraints, ensure the direct predecessors of task i be assigned before it. Constraints (7) and (8) show the sequence-dependent constraints which are peculiar to two-sided assembly line and ensure that tasks without precedence relationship on the same station work in sequence. Constraints (9) ~ (12) define variables.

$$\sum_{q=1}^{M} \sum_{d=1}^{2} x_{iqd} = 1, \forall i \le N \tag{3}$$

$$FT_i \le CT_R, \forall i \le N \tag{4}$$

$$\sum_{q=1}^{M} \sum_{d=1}^{2} q x_{jqd} - \sum_{v=1}^{M} \sum_{d=1}^{2} v x_{ivd} \le 0, \forall i \le N, j \in \Pr e_i \tag{5}$$

$$FT_i - FT_j + \mu\left(1-\sum_{d=1}^{2} x_{jqd}\right) + \mu\left(1-\sum_{d=1}^{2} x_{iqd}\right) \geq t_i, \quad \forall i \leq N, j \in \mathrm{Pr}\,e_i, q \leq M \tag{6}$$

$$FT_p - FT_i + \mu\left(1-x_{pqd}\right) + \mu\left(1-x_{iqd}\right) + \mu\left(1-z_{ip}\right) \geq t_p, \quad \forall i \leq N, p \in \left\{r \mid r \in I - \left(\mathrm{Pr}\,e_i \cup DS_i\right), i < r\right\}, \quad q \leq M, d = \{1,2\} \tag{7}$$

$$FT_i - FT_p + \mu\left(1-x_{pqd}\right) + \mu\left(1-x_{iqd}\right) + \mu z_{ip} \geq t_i, \quad \forall i \leq N, p \in \left\{r \mid r \in I - \left(\mathrm{Pr}\,e_i \cup DS_i\right), i < r\right\}, \quad q \leq M, d = \{1,2\} \tag{8}$$

$$z_{ip} = \{0,1\}, \forall i \leq N, p \in \left\{r \mid r \in I - \left(\mathrm{Pr}\,e_i \cup DS_i\right), i < r\right\} \tag{9}$$

$$x_{iq1} = \{0,1\}, \forall i \in A_L, q \leq M \tag{10}$$

$$x_{iq2} = \{0,1\}, \forall i \in A_R, q \leq M \tag{11}$$

$$x_{iqd} = \{0,1\}, \forall i \in A_E, q \leq M, d = \{1,2\} \tag{12}$$

3 IoT-based Framework For TALs Reconfiguration

An IoT-based framework for TALs reconfiguration has been developed involving the physical TAL, IoT-based monitoring system and cyber TAL shown in Figure 3.

The physical TAL is divided into several positions consisting of two directly facing workstations. Each assembly task is previously assigned to the specified workstation. Each workstation has several workplaces, which are required for the storage of tools (screw driver, drilling machine, etc.) and parts (screw, nuts, etc.). AGVs deliver the different parts associated with the tasks to the corresponding workstations according to the given logistic plans. New product items are placed in the conveyor pallets, go through the line and are progressively assembled by the workers allocated to the different workstations.

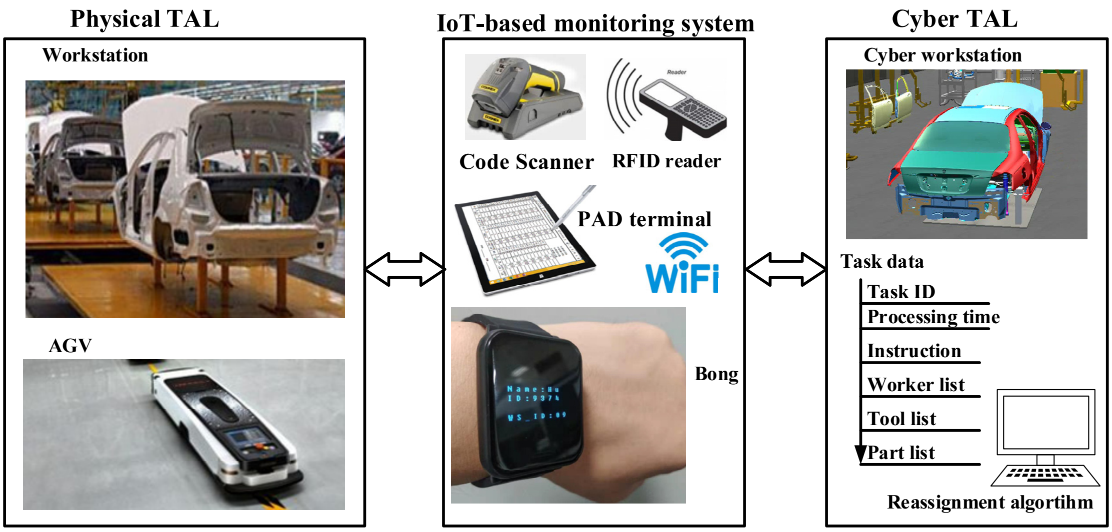


Figure 3. IoT-based framework for TALs reconfiguration

The IoT-based monitoring system mainly is composed of the code scanner, RFID reader, bong, PAD terminal and WIFI [35-38]. As every task performed in TALs involves the workers, tools, parts and operating instructions, the dynamic reconfiguration of TALs requires a much more complex coordination among them, when the tasks reassignment is carried out. The monitoring system links the physical TAL and cyber one to support the collaboration by detecting the physical objects with the unique ID in workstations and displaying the tasks reassignment-related information. First, workers wear the staff bongs which contain the RFID tag with unique worker ID. Second, the pallets for holding product items and the parts are attached RFID tag to represent the product ID and the part logistic ID. Third, two-dimension codes representing the ID are directly printed on the face of tools and parts by using the laser machine, which can be read by the code scanner. Finally, workstations have the static label ID in the associated workplace. After the tasks reassignment is determined, the PAD terminal equipped with the workstation displays the information of the resource reconfiguration, including the tools, workers and parts logistic. The message of the related worker reallocation is sent and displayed on the bong screen.

The cyber TAL contains the cyber workstation, task data and tasks reassignment algorithm. By using the IoT-based monitoring system to recognize the relevant objects with the unique ID, the physical workstation is reflected in the cyber workstation. The cyber workstation describes the existing assigned tasks, deployed tools and equipment, allocated workers and status of the parts storage of the workstation in the physical TAL, which are input into the reassignment algorithm to compute the solution of the TALrBP. The task data manage the assembly process-related information including the task ID, processing time, operational instruction, required worker list, tool list and part list, which are derived from the assembly process design. The collected information by the IoT-based monitoring system and the task data are input into the tasks reassignment algorithm, which can find the solution of the tasks reassignment among the workstations in the physical TAL. By connecting with the reassigned task data and comparing with the physical resource configuration of the existing workstations, the collaborative reconfiguration among the workers, tools, storage and logistic of the parts in the TAL can be carried out. Workers can directly move with the new location of tasks or do not move and receive retraining after building connections to new tasks. The required tools and parts to be assembled should be distributed along the sides of the line. However, in order to ensure assembly efficiency, double lines of material are not allowed. Therefore, a large number of tools and parts, especially large ones, are generally placed in a storage area with a certain distance from the assembly line. And they are delivered during the assembly process according to the logistic delivery system. The storage area and logistic delivery plan of tools and parts are revised according to the new task assignment.

4 A Heuristic Algorithm for TALrBP

In this section, a modified beam search algorithm is proposed to solve the TALrBP. It searches for a small number of the reassigned tasks by trying to find the feasible solutions for the given candidate cycle time and the initially balanced assembly line. The position-oriented enumerative procedure is developed to generate the nodes of the search tree corresponding to the partial solutions. The filtering procedure is used to eliminate some nodes by the priority rules including the proposed task priority rule and dominance rules [39], and only the remaining promising nodes are evaluated by the global evaluation function, which typically computes the minimal number of the reassigned tasks of the best solution.

4.1 Position-oriented Enumerative Procedure

The beam search algorithm extends a partial solution by assigning each available task in a forward manner station by station. As aforementioned, the tasks assignments in TAL can interfere with each other in a pair of workstations at each position. Obviously, the workstation-oriented assignment procedure developed for OALPs cannot be directly used for TALrBPs. Therefore, the proposed beam search for the TALrBP is a position-oriented tasks reassignment procedure, which reassigns tasks in a forward manner position by position.

Steps of the Position-oriented Enumerative Procedure

Step 1. Select the workstation in the position with the minimal WCT value. If the WCT of two mated-workstations in the position is the same, the left workstation is assumed to be selected.

Step 2. Find the available tasks set for the selected workstation, considering the operational side restriction, the precedence, and the cycle time constraints. If no task is available, another workstation in the position is selected and generates the corresponding available tasks set again. If neither workstations can be selected, then the position-based assignment procedure is finished. The next position is open, and go to step 1.

Step 3. Compute the priority value of each available task.

Step 4. Select the available task with the highest priority and assign it to the selected workstation, update the WCT of the workstation and go to Step 1.

4.2 Filtering Procedure

The filtering procedure in the proposed algorithm employs the reassigned task priority rule to keep the new balancing as close as possible to the previous balancing.

Reassigned task priority rule. The objective of the TALrBP is to minimize the tasks reassignment. When task i assigned to position q is reassigned to another succeeding position, its successors assigned in the current position q should be reassigned because of the precedence constraints. The LBTi can be computed as follows.

$$LBT_i = \begin{cases} \text{cardinality}(TIB_q \cap AS_i) + 1, & i \in TIB_q \\ \text{cardinality}(TIB_q \cap AS_i), & i \notin TIB_q \end{cases} \quad \textbf{(13)}$$

The bigger the value of LBT_i is, the higher the task i priority is. This rule can be used to select the tasks with a higher priority, which prefers the previous position. Since the proposed algorithm is based on the beam search, the dominance rule, maximum load rule and prefixing rule developed by Hu et al. [39] can be used to eliminate some nodes, which filter partial solutions without explicitly completing them to full solutions.

Position load rule. If the total time of the unassigned tasks does not exceed the total capacity of the remaining positions, in terms of the left workstation, the right workstation and mated-workstation in position q respectively, then the tasks reassignment in position q are feasible.

$$\sum_{i \in UT, S_i = L} t_i \le C \times RP \quad (14)$$

$$\sum_{i \in UT, S_i = R} t_i \le C \times RP \quad (15)$$

$$\sum_{i \in UT} t_i \le 2 \times C \times RP \quad (16)$$

Inequations (14) and (15) mean that the remaining left and right workstations possibly contain the unassigned L-type and R-type tasks, respectively. Inequation (16) means that the remaining position can possibly contain all unassigned tasks. If any of the above inequations fail to hold, then the partial solution is pruned.

4.3 Global Evaluation

The global evaluation function used in the proposed algorithm calculates the number of the reassigned tasks. The mathematical equation (17) is derived from equation (2) and used to compute the number of the reassigned tasks in the current position (NTRPq) for the feasible partial solutions generated by the position-oriented enumeration procedure, and the promising nodes can be selected.

$$\begin{aligned} NTRP_q &= \text{cardinality}(TIB_q) \\ &- \text{cardinality}(TIB_q \cap TNB_q) \end{aligned} \quad (17)$$

4.4 Position-oriented Beam Search Procedure

In TALrBP, the node in the search tree represents a solution state represents the partial reassignment of tasks assigned in the initial solution. The leaf nodes correspond to the new solution obtained of tasks reassignment. In order to find the promising nodes, the hybrid evaluation procedure performed on the individual task reassignment and the tasks reallocation in a position are developed to estimate the promised value. The task priority rule is defined to select the possible β1 promising nodes corresponding to the individual task reassignment. After the tasks reallocation in a position is completed, the equation (17) is used to estimate the tasks reassignment of the partial solutions represented by the full tasks reassignment in one position, and the best $\beta 2$ promising nodes are selected for the further searching in the next position. The filtering procedure is employed to eliminate some nodes by the proposed local evaluation function, and only the remaining nodes are globally evaluated.

Steps of the proposed algorithm.

Input: An initial solution S0, a candidate cycle time CT, a beam width of the individual task β1, a beam width of the position β2.

Output: A valid tasks reassignment or “failed” if invalid

Step 1. Initialization

Step 2. Generate the β1 descendant nodes by using the proposed position-oriented enumerative procedure, and progress level by level. If the number of the nodes is less than the beam width β1, then expand nodes by generating further level nodes until the total number of nodes in the last level is greater than β1.

Step 3. Filter some nodes by using the dominance rules.

Step 4. After the position-based assignment is completed, some nodes are pruned off by employing the aforementioned prefixing rule and position load rule.

Step 5. Compute the promised value of the remaining nodes by using the global evaluation function, select the promising β2 nodes, and go to step 2 until all positions are fathomed. If the number of the nodes is less than the beam width β2, select all the nodes and go to step 2.

Step 6. The one with the minimum objective value is selected, if at least one feasible solution is found; else return “failed”.

5 Experimental Results

5.1 Test Problems and Parameter Setting

The proposed heuristic algorithm based on the beam search is implemented in C++ and runs on a PC with a 2.8 GHz Core i7 and 8GB of main memory. Due to the limitation of example data, a set of 32 test instances is derived from the six benchmark problems of the TALBP [40]. They are characterized by three experimental factors: the number of positions, the initial cycle time and the final cycle time. The number of the positions and the lower bound of the cycle time for each benchmark problem are both collected from the literature [8]. The Hoffman heuristic proposed by Hu et al. [41] for the TALBP-1 is utilized repeatedly to find the feasible solution by continuously increasing the value of cycle time from the lower bound until the number of positions changes. One of these feasible solutions with various cycle time is randomly selected as the initial solution of the TALrBP with the given number of positions. The final cycle time is firstly set the known cycle time provided by Kim et al. [8], Lei and Guo [10], Tang et al. [11] and Li et al. [12]. If the proposed algorithm could not find a feasible solution, then the final cycle time is increased by one until the feasible solution is obtained.

In preliminary experiments, considering the different computational effort between the small-scale and large-scale problems, the reasonable parameters for the heuristic algorithm based on the beam search are given in Table 2. As aforementioned, β1 and β2 correspond to the individual tasks reassignment and the tasks reallocation in a position respectively. In order to reduce the search space, the prefixing rule and position load rule are used to truncate some nodes that represent tasks reallocation of a position, and β2 can be set less than β1.

Table 2. Parameters of the beam search used in the computational experiments.

Parameter	Small-scale Instances	Large-scale Instances
	P12, P16, P24	P65, P148, P205
β1	40	110
β2	24	70
Maximum search time (s)	10	60

5.2 Performance Evaluation of the Proposed Algorithm

The proposed algorithm is developed to solve the TALrBP with the aim of minimizing the tasks reassignment, given the initial solution and the candidate cycle time. Since no comparable experiments on the TALrBP exist in the literature, the comparison between the original lines and the new ones are provided in Table 3 and Table 4. Firstly, in terms of the rebalancing cost, the ratio of the number of tasks reassignment to the total number of tasks (RNT) is defined to evaluate the performance of the proposed algorithm.

$$RNT = \frac{NR}{N} \times 100\% \quad \textbf{(18)}$$

Table 3. Comparison between original lines and new ones for small-scale problems

Instance	Positions	CT_0	CT_R	RCT (%)	NR	RNT(%)	SI_0	SI_R	RSI (%)
P12	2	10	7	30.0	5	41.7	16.3	1.7	89.6
	3	6	5	16.7	3	25	7.3	3	58.9
P16	2	27	22	18.5	7	43.8	49.7	3.7	92.6
	3	17	16	5.9	4	25	20.8	7.3	64.9
P24	2	44	35	20.5	5	20.8	43.6	0	100
	3	27	24	11.1	9	37.5	44.1	3.2	92.7
	4	22	18	18.2	8	33.3	30.1	2	93.4
Average				17.3		32.4			84.6

Table 4. Comparison between original lines and new ones for large-scale problems

Instance	Positions	CT_0	CT_R	RCT (%)	NR	RNT (%)	SI_0	SI_R	RSI (%)
P65	4	660	639	3.2	8	12.3	101.9	5.7	94.4
	5	530	512	3.4	9	13.8	122.9	9.4	92.4
	6	449	428	4.7	18	27.7	186.2	15.5	91.7
	7	395	369	6.6	15	23.1	288.3	28.2	90.2
	8	348	322	7.5	15	23.1	289.3	19.9	93.1
P148	4	664	641	3.5	7	4.7	132.9	3.2	97.6
	5	529	513	3.0	8	5.4	117.1	0	100
	6	447	427	4.5	10	6.8	164.1	0	100
	7	387	366	5.4	17	11.5	206.6	0	100
	8	337	321	4.7	14	9.5	146.7	7.5	94.9
	9	303	285	5.9	22	14.9	203.6	2.4	98.8
	10	283	257	9.2	17	11.5	296.5	7.1	97.6
	11	254	235	7.5	21	14.2	255.2	14	94.5
	12	237	215	9.3	21	14.2	319.9	11.7	96.3
P205	4	3186	2940	7.7	8	3.9	1096.6	98.4	91
	5	2517	2348	6.7	13	6.3	1113.7	73.6	93.4
	6	2052	1960	4.5	13	6.3	551.1	81.7	85.2
	7	1818	1680	7.6	25	12.2	882.6	89.9	89.8
	8	1588	1473	7.2	20	9.8	1116.7	88	92.1
	9	1387	1309	5.6	19	9.3	889	98.4	88.9
	10	1241	1184	4.6	27	13.2	579.4	166.8	71.2
	11	1149	1093	4.9	23	11.2	781	232.5	70.2
	12	1000	984	1.6	25	12.2	274.4	102	62.8
	13	1000	944	5.6	21	10.2	996.2	463.4	53.5
	14	1004	944	6.0	20	9.8	1529.3	832.3	45.6
Average				5.6		11.9			87.4

Secondly, considering the fact that the final aim of our work is to find a rebalance of assembly lines with a fixed number of workstations, there is no doubt that cycle time is the key indicator of measuring the efficiency of rebalanced lines (*RLE*), because *RLE* is equal to $\sum_{i=1}^{N} t_i \Big/ K * CT_R$. Meanwhile, cycle time can also represent the production capacity of lines, because it has the reciprocal relationship with production capacity theoretically. Thus, minimized cycle time is equivalent to maximization of the assembly line efficiency and the production rate. In addition, smoothing index (SI) is a key factor for measuring the balance rate of the assembly line, which directly affects the final capacity of lines. Intuitively, in OALs, the cycle time can be improved when the SI is enhanced. However, in TALs, these two criteria are not necessarily identical, since the cycle time is determined by the maximum value of WCT, while the smoothness is affected also by the workstations with below average workloads. Therefore, it is better to utilize the differences of those two factors between the initial balanced lines and rebalanced lines to demonstrate the efficiency of the proposed algorithm. The reduction of the cycle time (RCT) and smoothing index (RSI) are defined by equation (20) and (21), respectively.

$$SI = \sqrt{\sum_{j \in J} \sum_{k=1}^{2} (ST_{\max} - ST_{jk})^2 \Big/ 2N} \quad \mathbf{(19)}$$

Where $ST_{jk} = \sum_{i \in S_{jk}} t_i$

$$RCT = \frac{CT_0 - CT_R}{CT_0} \times 100\% \quad \mathbf{(20)}$$

$$RSI = \frac{SI_0 - SI_R}{SI_0} \times 100\% \quad \mathbf{(21)}$$

For the small-scale TALrBPs displayed in Table 3, each candidate cycle time is set to the optimal value, which has been found in the literature [8]. The average RCT, RNT, and RSI are 17.3%, 32.4% and 84.6%, respectively. For the large-scale TALrBPs including P65, P148 and P205 shown in Table 4, each candidate cycle time is first set to the value of the best-known solution found in the literature [10-12]. If the feasible solution for the TALrBP can be found, then the candidate cycle time would decrease and the procedure is carried out again until it fails to find the feasible solution. If the feasible solution for the TALrBP cannot be found, then the candidate is increased until it gets the feasible one. The average RCT, RNT, and RSI are 5.6%, 11.9% and 87.4%, respectively. Eventually, we can obtain the conclusion that the cycle time and smoothness both can be significantly improved by reassigning a small number of the tasks in the previous TALs, i.e., keep the new lines as close as possible to the previous ones. As an illustrative example, the task assignment of the original line and the new one for P65 with 4 positions is shown in Figure 4(a) and Figure 4(b). It visually verifies the effectiveness of the proposed algorithm.

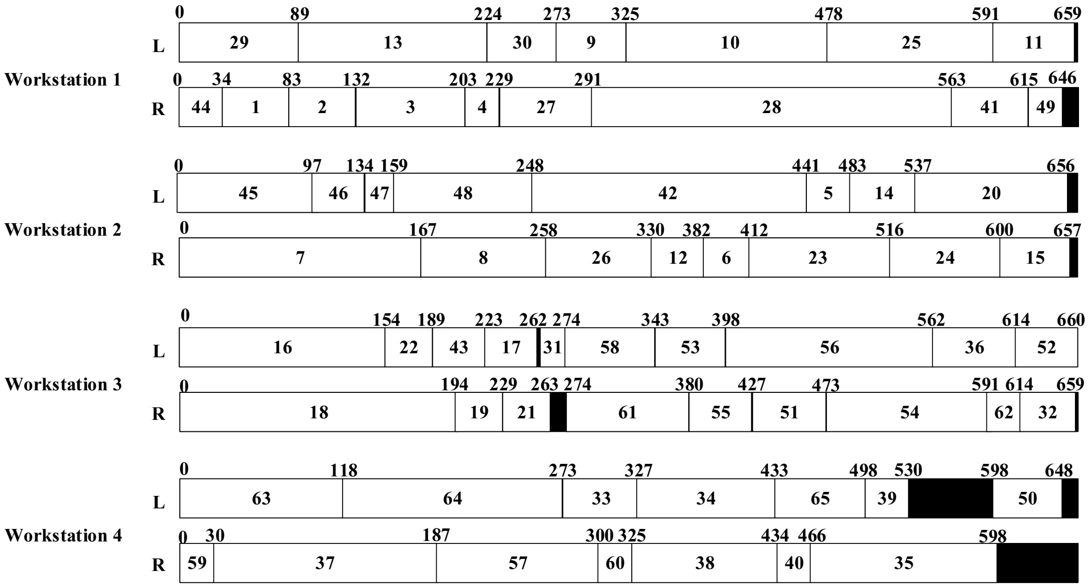
Furthermore, Table 5 shows the computed gap between the CT_R and the best cycle time collected from other algorithms, including the neighborhood genetic algorithm (n-GA) [8], the variable neighborhood search (VNS) [10], the improved discrete artificial bee colony (DABC) algorithm [11], the iterated greedy (IG1 and IG2) algorithm [12], which solved the TALBP-2 with the objective of minimizing the cycle time for a given number of positions. Among 32 instances, the proposed algorithm can find the feasible rebalancing solutions for 19 TALrBPs with the best-known cycle time, and the gaps of 6 others are not more than 2. The maximal value of the gap is 19 for the P205 problem with the best cycle time of 1074. Especially, the best cycle time of the P148 with 7 positions is found, 366, which is less 1 than the best-known one. The results show that the proposed algorithm can effectively deal with the TALrBPs.

6 Conclusions

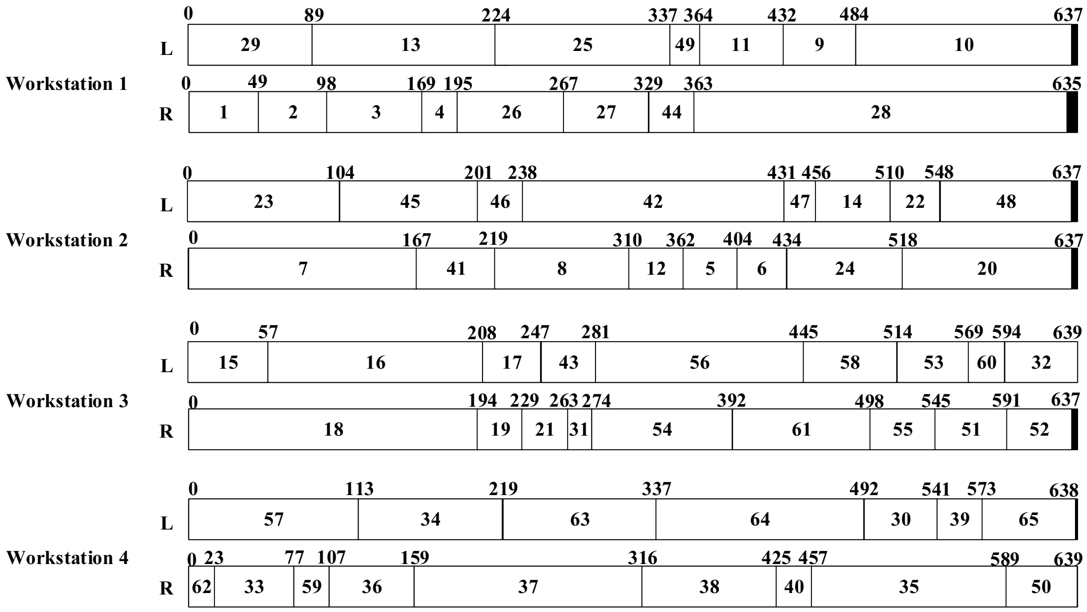
The changes of market demands highlight the necessity of the reconfiguration of existing lines. This means continuous reconfigures in the initially balanced assembly lines, involving consequent tasks reassignment and the related worker reallocation, tools movement, parts logistic adjustment among the workstations in TALs. When tasks reassignment occurs in the existing assembly lines, the production costs increase as a consequence of some factors, such as operators retraining, quality assurance, and equipment switching.

This paper proposes an IoT-based framework to support the collaborative reconfiguration of the workers, tools and parts logistic among the workstations in the TAL, when the tasks reassignment. The mathematical model and a modified beam search algorithm based on the position-oriented enumerative procedure are developed to address the two-sided assembly line rebalancing problem with the objective of minimizing the tasks reassignment, which is defined to measure the degree of similarity between the previous and the new tasks assignment, and represents the rebalancing costs.

Because of no comparable experiments on the TALrBP in the literature, the proposed algorithm is tested on the benchmark problems by using the best-known cycle time collected from the literature, and the comparison between the original lines and the new ones is performed. For the small-scale TALrBPs including P12, P16 and P24, the average RCT, RNT and RSI are 17.3%, 32.4%, and 84.6%, respectively. For the large-scale TALrBPs including P65, P148 and P205, the average RCT, RNT and RSI are 11.9%, 5.6%,



(a) Task assignment of the original line for P65 with 4 positions



(b) Task assignment of the rebalanced line for P65 with 4 positions

Figure 4. Comparison between the original line and rebalanced one for P65 with 4 positions

Table 5. Comparison of the proposed algorithm with the existing methods

Instance	Positions	$\overline{ct}$	n-GA	VNS	DABC	IG1	IG2	Proposed algorithm	Gap
P12	2	6.25	**7**	**7**	**7**	**7**	**7**	**7**	0
	3	4.17	**5**	**5**	**5**	**5**	**5**	**5**	0
P16	2	20.5	**22**	**22**	**22**	**22**	**22**	**22**	0
	3	13.67	**16**	**16**	**16**	**16**	**16**	**16**	0
P24	2	35	**35**	**35**	**35**	**35**	**35**	**35**	0
	3	23.33	**24**	**24**	**24**	**24**	**24**	**24**	0
	4	17.5	**18**	**18**	**18**	**18**	**18**	**18**	0
P65	4	637.4	641	639	**638**	639	**638**	639	1
	5	509.9	515	513	**511**	512	512	512	1
	6	424.9	432	430	427	**426**	427	428	2
	7	364.2	372	368	**367**	368	**367**	369	2
	8	318.7	327	324	**321**	322	**321**	322	1

Table 5. Comparison of the proposed algorithm with the existing methods (continue)

Instance	Positions	$\overline{ct}$	n-GA	VNS	DABC	IG1	IG2	Proposed algorithm	Gap
P148	4	640.5	**641**	**641**	**641**	**641**	**641**	**641**	0
	5	512.4	514	**513**	**513**	**513**	**513**	**513**	0
	6	427	428	428	**427**	428	428	**427**	0
	7	366	368	368	367	367	367	***366***	**-1**
	8	320.3	323	323	**321**	322	**321**	**321**	0
	9	284.7	287	287	**285**	286	286	**285**	0
	10	256.2	259	258	**257**	258	258	**257**	0
	11	232.9	237	236	**234**	235	**234**	235	1
	12	213.5	218	216	**215**	216	216	**215**	0
P205	4	2918.1	2946	**2927**	2931	2947	**2927**	2940	13
	5	2334.5	2364	2348	**2342**	2359	2345	2348	6
	6	1945.4	1984	1957	**1954**	1968	1956	1960	6
	7	1667.5	1709	**1676**	1681	1692	1682	1680	4
	8	1459.1	1507	1472	**1469**	1486	1474	1473	4
	9	1296.9	1337	**1309**	1310	1328	1311	**1309**	0
	10	1167.3	1189	**1180**	1182	1198	1181	1184	4
	11	1061.1	1095	**1074**	1077	1082	1078	1093	19
	12	972.7	1039	995	992	1000	**984**	**984**	0
	13	897.9	**944**	**944**	**944**	**944**	**944**	**944**	0
	14	833.8	**944**	**944**	**944**	**944**	**944**	**944**	0

and 87.4%, respectively. The results show that the rebalanced solutions obtained by the proposed algorithm are significantly better than the original lines in terms of cycle time and smoothing index with less rebalancing cost. However, in this paper, it is assumed that the number of stations is fixed, the tasks redefinition and the space constraints in the real application are ignored, thus, the proposed algorithm can only solve the rebalancing problem caused by small changes in demand. The future research will consider more variable factors to solve the complicated and realistic rebalancing problems.

Acknowledgments

This research work is supported by the National Natural Science Foundation of China (Grant No. 51975373, 51475303).

References

[1] J. Bartholdi, Balancing Two-sided Assembly Lines: A Case Study, *International Journal of Production Research*, Vol. 31, No. 10, pp. 2447-2461, October, 1993.

[2] A. Baykasoglu, T. Dereli, Two-sided Assembly Line Balancing Using An Ant-colony-based Heuristic, *The International Journal of Advanced Manufacturing Technology*, Vol. 36, No. 5-6, pp. 582-588, March, 2008.

[3] W.-C. Chiang, T. L. Urban, C. Luo, Balancing Stochastic Two-sided Assembly Lines, *International Journal of Production Research*, Vol. 54, No. 20, pp. 6232-6250, October, 2016.

[4] Y. Delice, E. K. Aydoğan, U. Özcan, M. S. İlkay, A Modified Particle Swarm Optimization Algorithm to Mixed-model Two-sided Assembly Line Balancing, *Journal of Intelligent Manufacturing*, Vol. 28, No. 1, pp. 23-36, January, 2017.

[5] B. Yuan, C. Zhang, X. Shao, Z. Jiang, An Effective Hybrid Honey Bee Mating Optimization Algorithm for Balancing Mixed-model Two-sided Assembly Lines, *Computers & Operations Research*, Vol. 53, pp. 32-41, January, 2015.

[6] Q. Tang, Z. Li, L. Zhang, C. Zhang, Balancing Stochastic Two-sided Assembly Line with Multiple Constraints Using Hybrid Teaching-learning-based Optimization Algorithm, *Computers & Operations Research*, Vol. 82, pp. 102-113, June, 2017.

[7] Y. Li, The Type-II Assembly Line Rebalancing Problem Considering Stochastic Task Learning, *International Journal of Production Research*, Vol. 55, No. 24, pp. 7334-7355, July, 2017.

[8] Y. K. Kim, W. S. Song, J. H. Kim, A Mathematical Model and a Genetic Algorithm for Two-sided Assembly Line Balancing, *Computers & Operations Research*, Vol. 36, No. 3, pp. 853-865, March, 2009.

[9] H. D. Purnomo, H.-M. Wee, Maximizing Production Rate and Workload Balancing in a Two-sided Assembly Line Using Harmony Search, *Computers & Industrial Engineering*, Vol. 76, pp. 222-230, October, 2014.

[10] D. Lei, X. Guo, Variable Neighborhood Search for the Second Type of Two-sided Assembly Line Balancing Problem, *Computers & Operations Research*, Vol. 72, pp. 183-188, August, 2016.

[11] Q. Tang, Z. Li, L. Zhang, An Effective Discrete Artificial Bee Colony Algorithm with Idle Time Reduction Techniques for Two-sided Assembly Line Balancing Problem of Type-II, *Computers & Industrial Engineering*, Vol. 97, pp. 146-156, July, 2016.

[12] Z. Li, Q. Tang, L. Zhang, Minimizing the Cycle Time in Two-sided Assembly Lines with Assignment Restrictions: Improvements and a Simple Algorithm, *Mathematical*

Problems in Engineering, Vol. 2016, pp. 1-15, May, 2016.

[13] E. Falkenauer, Line Balancing in the Real World, *The International Conference on Product Lifecycle Management PLM*, 2005, Vol. 5, pp. 360-370.

[14] R. Gamberini, A. Grassi, B. Rimini, A New Multi-objective Heuristic Algorithm for Solving the Stochastic Assembly Line Re-balancing Problem, *International Journal of Production Economics*, Vol. 102, No. 2, pp. 226-243, August, 2006.

[15] N. Grangeon, P. Leclaire, S. Norre, Heuristics for the Re-balancing of a Vehicle Assembly Line, *International Journal of Production Research*, Vol. 49, No. 22, pp. 6609-6628, November, 2011.

[16] C. Yang, J. Gao, L. Sun, A Multi-objective Genetic Algorithm for Mixed-model Assembly Line Rebalancing, *Computers & Industrial Engineering*, Vol. 65, No. 1, pp. 109-116, May, 2013.

[17] E. Celik, Y. Kara, Y. Atasagun, A New Approach for Rebalancing of U-lines with stochastic Task Times Using Ant Colony Optimisation Algorithm, *International Journal of Production Research*, Vol. 52, No. 24, pp. 7262-7275, December, 2014.

[18] F. Makssoud, O. Battaïa, A. Dolgui, K. Mpofu, O. Olabanji, Re-balancing Problem for Assembly Lines: New Mathematical Model and Exact Solution Method, *Assembly Automation*, Vol. 35, No. 1, pp. 16-21, February, 2015.

[19] Y. Li, T. O. Boucher, Assembly Line Balancing Problem with Task Learning and Dynamic Task Reassignment, *The International Journal of Advanced Manufacturing Technology*, Vol. 88, No. 9-12, pp. 3089-3097, February, 2017.

[20] E. Sancı, M. Azizoğlu, Rebalancing the Assembly Lines: Exact Solution Approaches, *International Journal of Production Research*s, Vol. 55, No. 20, pp. 5991-6010, October, 2017.

[21] Y. Zhang, X. Hu, C. Wu, A Modified Multi-objective Genetic Algorithm for Two-sided Assembly Line Re-balancing Problem of a Shovel Loader, *International Journal of Production Research*, Vol. 56, No. 9, pp. 3043-3063, May, 2018.

[22] W. Zhou, J. Xu, B. Wang, A Smart Home Foundation Scheme Based on Open Source Hardware and Cloud Computing, *International Journal of Internet Protocol Technology*, Vol. 10, No. 1, pp. 13-21, March, 2017.

[23] N. Ramachandran, V. Perumal, Proficient Communication between Sensor Devices Using Heuristic Approaches in IoT Environment, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 29, No. 1-2, pp. 15-27, August, 2018.

[24] M. S. Ali, M. K. H Jewel, Y. Li, F. Lin, Frequency-domain Channel Equalisation for LTE-based Uplink Narrowband Internet of Things Systems, *IET Communications*, Vol. 13, No. 3, pp. 281-288, February, 2019.

[25] A. Kumari, S. Tanwar, S. Tyagi, N. Kumar, Verification and Validation Techniques for Streaming Big Data Analytics in Internet of Things Environment, *IET Networks*, Vol. 8, No. 3, pp. 155-163, May, 2019.

[26] W. N. Liu, L. J. Zheng, D. H. Sun, X. Y. Liao, M. Zhao, J. M. Su, Y. X. Liu, RFID-enabled Real-time Production Management System for Loncin Motorcycle Assembly Line, *International Journal of Computer Integrated Manufacturing*, Vol. 25, No. 1, pp. 86-99, January, 2012.

[27] R. Krishnamurthy, J. Cecil, A Next-generation IoT-based Collaborative Framework for Electronics Assembly, *The International Journal of Advanced Manufacturing Technology*, Vol. 96, No. 1-4, pp. 39-52, April, 2018.

[28] J. Cecil, S. Albuhamood, A. Cecil-Xavier, P. Ramanathan, An Advanced Cyber Physical Framework for Micro Devices Assembly, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 49, No. 1, pp. 92-106, January, 2019.

[29] Y. Cohen, M. Faccio, F. G. Galizia, C. Mora, F. Pilati, Assembly System Configuration through Industry 4.0 Principles: The Expected Change in the Actual Paradigms, *IFAC-PapersOnLine*, Vol. 50, No. 1, pp. 14958-14963, July, 2017.

[30] M. Bortolini, E. Ferrari, M. Gamberi, F. Pilati, M. Faccio, Assembly System Design in the Industry 4.0 Era A General Framework, *IFAC-PapersOnline*, Vol. 50, No. 1, pp. 5700-5705, July, 2017.

[31] K. Thramboulidis, I. Kontou, and . Vachtsevanou, Towards an IoT-based Framework for Evolvable Assembly Systems, *IFAC-PapersOnLine*, Vol. 51, No. 11, pp. 182-187, January, 2018.

[32] K. Thramboulidis, D. Vachtsevanou, I. Kontou, CPuS-IoT: A Cyber-physical Microservice and IoT-based Framework for Manufacturing Assembly Systems, *Annual Reviews in Control*, Vol. 47, pp. 237-248, March, 2019.

[33] M. Liu, J. Ma, L. Lin, M. Ge, Q. Wang, C. Liu, Intelligent Assembly System for Mechanical Products and KEY TECHNOLOGY BASED on Internet of Things, *Journal of Intelligent Manufacturing*, Vol. 28, No. 2, pp. 271-299, February, 2017.

[34] Y. H. Zhang, X. F. Hu, C. X. Wu, Heuristic Algorithm for Type II Two-sided Assembly Line Rebalancing Problem with Multi-objective, *MATEC Web of Conferences*, vol 175, July, 2018.

[35] J. Wan, H. Yan, Q. Liu, K. Zhou, R. Lu, D. Li, Enabling Cyber-physical Systems with Machine-to-machine Technologies, International *Journal of Ad Hoc and Ubiquitous Computing*, Vol. 13, No. 3-4, pp. 187-196, July, 2013.

[36] H. Huang, J. Zhou, W. Li, J. Zhang, X. Zhang, G. Hou, Wearable Indoor Localisation Approach in Internet of Things, *IET Networks*, Vol. 5, No. 5, pp. 122-126, September, 2016.

[37] D. J. Chiang, Real-Time Data Delivering Based on Prediction Scheme over Internet of Things, *Journal of Internet Technology*, Vol. 18, No. 2, pp. 395-405, March, 2017.

[38] Y. W. Ma, C. H. Chen, J. L. Chen, S. S. Lin, Operation, Administration and Maintenance (OA&M) Architecture Design for Internet of Things, *Journal of Internet Technology*, Vol. 18, No. 6, pp. 1223-1229, January, 2017.

[39] X. F. Hu, E. F. Wu, J. S. Bao, Y. Jin, A Branch-and-bound Algorithm to Minimize the Line Length of a Two-sided Assembly Line, *European Journal of Operational Research*,

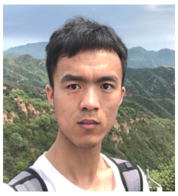
Vol. 206, No. 3, pp. 703-707, November, 2010.

[40] T. O. Lee, Y. Kim, Y. K. Kim, Two-sided Assembly Line Balancing to Maximize Work Relatedness and Slackness, *Computers & Industrial Engineering*, Vol. 40, No. 3, pp. 273-292, July, 2001.

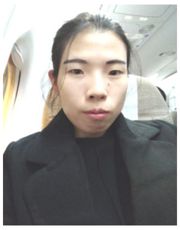
[41] X. F. Hu, E. F. Wu, Y. Jin A Station-oriented enumerative Algorithm for Two-sided Assembly Line Balancing, *European Journal of Operational Research*, Vol. 186, No. 1, pp. 435-440, April, 2008.

Biographies

Xiaofeng Hu received the Ph.D. degree in mechanical design and theory from Shanghai Jiao Tong University, Shanghai, China in 2005, received the B. Tech. and M.S. degrees in mechanical design and theory from Xi'an University of Architecture and Technology, Xi'an, China, in 1999 and 2002, respectively. Now, he is an associate professor in the School of Mechanical Engineering, Shanghai Jiao Tong University.



Chuanxun Wu received the B. Tech. degree in Mechanical engineering and automation from Shanghai Jiao Tong University, Shanghai, China in 2017. He is currently pursuing M.S. degree in the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China.



Yahui Zhang received the B.Admin. degree in Logistics Management from Beijing University of Chemical Technology, Beijing, China in 2010 and the M.S. degree in Mechanical Engineering (Logistics Engineering) from Shanghai Maritime University, Shanghai, China in 2012. She is currently pursuing Ph.D. degree in the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China.