

DOI: <http://doi.org/10.24867/IJIEM-2019-4-247>

Impact of work-sharing on the performance of production line with heterogeneous workers

Thanatat Pasupa

Tokyo Institute of Technology, Department of Industrial Engineering and Economics, O-okayama, Meguro, Tokyo 152-8552, Japan, pasupa.t.aa@m.titech.ac.jp

Sadami Suzuki

Tokyo Institute of Technology, Department of Industrial Engineering and Economics, O-okayama, Meguro, Tokyo 152-8552, Japan, suzuki.s.ag@m.titech.ac.jp

Received (21-MAY-2019); Revised (5-OCT-19); Accepted (20-OCT-2019)

Abstract

This study addresses the problem of Dynamic Line Balancing (DLB) in a two-stage production assembly tandem line where there are worker differences. This study examines the dynamic decision rules for workers to operate under DLB with a low level of work in process (WIP), in the constant work-in-process (CONWIP) system. We develop a new dynamic decision rule, the Smallest R No Starvation-II (SRNS-II), to maximize the throughput of the system. Results showed that DLB improves the throughput rate under a low WIP level. This study provides managerial insights for worker assignments and workload balancing.

Key words: line balancing; work-sharing; CONWIP; cross-training

1. INTRODUCTION

Traditional line balancing is a procedure in which tasks are equally assigned to workstations over the long term. A production assembly line is designed to balance the workload among workers. However, the line could be severely unbalanced due to the variability that occurs (mostly from human operators), especially for unpacked serial production lines which heavily rely on workers [1]. By having workforce flexibility under the work-sharing mechanism, workers can operate more than one task, to become cross-trained workers. Workers work together as a team to achieve the highest production rate. The flexibility of cross-trained workers can improve the line to become more balanced and can enhance line performance [2]. During the past decade, the application of work-sharing under the concept of lean manufacturing has been widespread. Initially, the work-sharing system was first implemented by Toyota Sewn Production Management System (TSS). TSS requires workers to move back and forth to pick up a task and hand it to a succeeding worker. This system is one type of moving worker module (MWM). The number of machines/workstations is usually more than the number of workers [3]. Bartholdi and Eisenstein [4] examined the TSS mechanism by an analytical model. They proposed a mechanism called the "Bucket Brigade" which is different from TSS in the way that a worker can move freely. The worker has to wait for the succeeding worker

to finish the ongoing task. Alves [5] conducted a hands-on simulation to investigate the performance of MWMs in cellular manufacturing. Results show that maximum productivity can be achieved by operating the Bucket Brigade System. In contrast, another type of work-sharing system is Dynamic Line Balancing (DLB). DLB does not require a worker to move. Workers stay at their workstation. DLB allows workers to help each other by sharing the work at their station, which requires duplicate tools/equipment to do the shared task (operation). There are two types of tasks assigned to workers: fixed task and shared task. A fixed task is a task that a designated worker is assigned, while a shared task is meant to be carried out by one worker of an adjacent pair of workers. Two types of DLB can be found in the literature: state-dependent and state-independent. For the state-dependent type, every time the first worker finishes a fixed task, there is a rule for him/her to send the shared task or keep doing it. This decision depends on the state of the system. Figure 1 shows the hierarchy of the work-sharing system. The consideration of the differences in workers' capabilities is necessary for a manufacturing system. In reality, human operators are different in capabilities. With absenteeism and turnover, the replacement of inexperienced workers is the primary cause of creating a new bottleneck and throughput loss on an assembly line [6]. There is evidence showing that worker differences cannot be eliminated. For tasks such as software coding, the best worker could perform 10

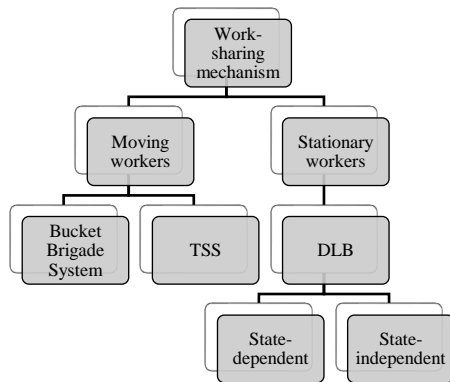


Figure 1. Hierarchy of work-sharing system

times faster than the worst worker [7].

The purposes of this paper are: (i) to develop the dynamic decision rule under the DLB mechanism to improve the production throughput rate, (ii) to explore the impact of work-sharing with skill level differences among workers, and (iii) to investigate the impact of granularity on the performance of DLB. In contrast to Askin and Chen [8], who assumed workers are identical, we are interested in DLB systems where there are capability differences among workers. We organize the paper as follows: Section 2 provides the related theory and previous works, based on the literature. In Section 3, we describe the modeling assumptions and the framework of our developed rule. In Section 4, we examine the comparative performance of the SRNS-II heuristic and the HFB heuristic. In Section 5 we investigate the effects of granularity on the performance of DLB. Lastly, we summarize the results and set the direction of future works in Section 6.

2. LITERATURE REVIEW

This section elucidates the past works in the literature, emphasizing cross-training, workforce skill level, workload balancing, and DLB mechanism. The objective of this section is to provide a statement of the problems addressed by previous research, as well as the issue of our interest.

2.1 Cross-training workforce and skill level

In recent years, several authors have investigated the impacts of cross-training and workforce skill levels on economic and human issues. Wu et al. [9] explored the trade-off between skill training cost and workload balance by proposing a swarm intelligence optimizer. Their proposed algorithm generates a better solution than the candidate algorithm. Kiassat and Safaei [10] proposed a methodology to investigate the relationship between the system profitability and the precision of skill level in a dynamic market. Results show that overestimation of profitability occurs if skill impreciseness is ignored. Niakan et al. [11] proposed an algorithm called NGS-II-MOSA, to solve the dynamic cell formation problem considering the economic, environmental, and human aspects. The objectives of their proposed model minimize the production cost, labor cost, and production waste. A recent work of Digiesi [12]

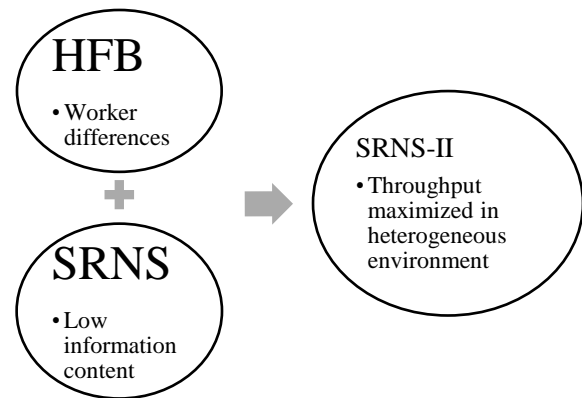


Figure 2. Foundation of the SRNS-II

proposed a job-rotation workforce scheduling model, to minimize the exposure risk of workers occurring from repetitive manual tasks. They implemented the Rapid Upper Limb Assessment (RULA) method to consider the exposure risk. In the literature review work of De Bruecker [13], they show that a successful implementation of skills in a workforce planning problem must classify the skill levels of workers correctly. They discovered two different skill classes in the literature: hierarchical class and categorical class. The hierarchical class differentiates the skill level by the production output of workers in the same task. The categorical class refers to the cross-training level of workers. These two skill classes were implemented in this study.

2.2 Workload balancing

Bukchin and Cohen [6] suggested that work-sharing should be done about 25-30% of the total workload, to achieve optimal throughput during absenteeism and turnover. Wang et al. [14] proposed the bubble allocation policy, to enhance the throughput rate of a labor-intensive production system during turnover. They found that the bubble allocation policy works best in a balanced production line. They also suggested that the work-sharing mechanism is suitable for an unbalanced production line. Lian et al. [15] proposed an optimization model to improve the inter-cell and inter-worker workload balance in a Japanese cellular manufacturing system. Results show that homogeneous workers perform better, based on the inter-worker workload balance. Moreover, the inter-cell workload balance increased when the proficiency level of workers had a higher standard deviation. Yu et al. [16] investigated the workload balancing in a Japanese cellular manufacturing system. They defined and formulated two balancing factors, Seru Balance (SB) and Seru System Balance (SSB), to evaluate the workload within the cell and the workload of the system.

2.3 Related theory

DLB makes use of cross-trained workers to provide flexibility, to maximize the throughput rate of the production line. Without a controlling policy, workers cannot obtain a task-sharing decision in real-time. Evidence from Gel et al. [17] shows that there are 1023

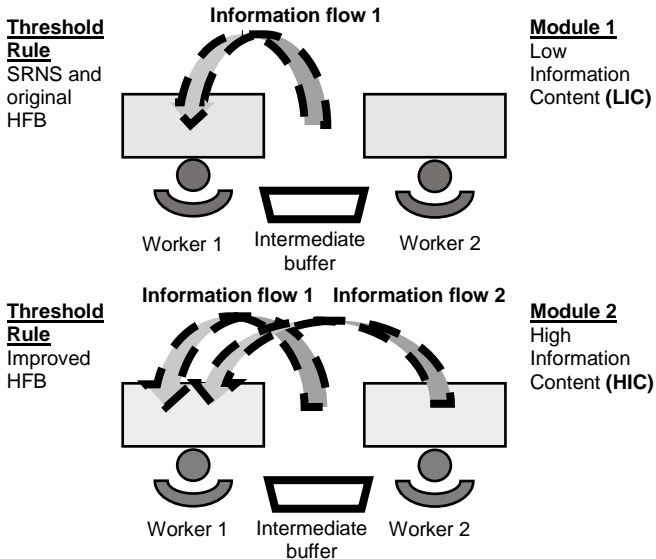


Figure 3. Level of information content under DLB

states when calculating the best threshold policy of a two-stage tandem line where the WIP level is ten jobs, which is complex. DLB was first introduced by Ostolaza et al. [18]. Unlike traditional line balancing, DLB dynamically balances the workload of workers. Workers can share their tasks under the heuristic rule. They proposed a rule called Half-Full-Buffer (HFB). The rule dictates that the preceding worker can decide to do a shared task or pass it to the succeeding worker. They examined the rule under two-stage lines with exponential processing time. Jobs in the buffer are classified into 2 types: long job (shared task not completed) and short job (shared task completed). Jobs in the buffer are picked up by succeeding workers under the SPT (Shortest Processing Time) policy. They found that the HFB rule works well in reducing the effects of blockages and starvation of workers. However, DLB did not perform well when the buffer capacity is 2 or less. The DLB mechanism was improved by McClain et al. [19]. They extended the HFB rule by increasing the information on workload available at the preceding workstation, which is measured by the total number of subtasks. This measurement also includes the workload that is being processed at the succeeding workstation. They also examined the FCFS (First Come First Serve) policy for jobs queueing at the buffer. Based on their findings, DLB improves efficiency even when the buffer capacity is 2 or less (unlike [18]). Gel et al. [17] adapted the HFB rule proposed by McClain et al. [19] to count the workload with general processing time and consider the worker differences in processing time. They also proposed a new threshold heuristic called "50-50 work content". The rule has an incentive to let succeeding workers do the shared task when the workload content of the preceding worker is more than 50% of the total workload in the system. Two serial workstations under the CONWIP control policy, invented by Spearman et al. [20], were examined under two rules. They investigated three factors that affect the performance of DLB: Preemptability, Granularity, and Variability. They found

that both rules perform well in terms of sub-optimality while suggesting that the shared task should be assigned in the right amount, that is, between 30% to 50% in a non-preemptive system. Askin and Chen [8] proposed a new threshold rule called "Smallest-R-No-Starvation" (SRNS). The motivation behind this new threshold rule is to use less information. DLB with a low information level was model A and for a high information level was model B. They examined SRNS under two-, three-, and five-stage CONWIP tandem lines with the assumption of identical workers. They compared the SRNS rule to the HFB rule extended by Gel et al. [17]. The SRNS rule shows a promising result, regarding line efficiency. To our knowledge, none of the previous studies has examined the impact of workers' skill level differences on the DLB mechanism, under a non-preemption environment. According to Figure 2, this paper explores the gap in experiments conducted by Gel et al. [17] and Askin and Chen [8]. Only Gel et al. [17] examined HFB by considering worker differences under the preemption assumption, while Askin and Chen [8] ignored the differences of workers, but focused on less information content. We develop the SRNS-II rule by taking the outstanding attributes of two previous rules, HFB by Gel et al. [17] and SRNS by Askin and Chen [8], to focus on the improvement of throughput rate. Two remarkable characteristics are: 1) low information content, and 2) workers possess different skill levels. A new rule is needed for controlling the DLB mechanism under the characteristics mentioned above.

3. DYNAMIC LINE BALANCING

3.1 Mechanism of DLB

According to Figure 4, after worker 1 finishes doing the fixed task (known as fixed-task A), he/she has to decide whether to pass on the shared task to worker 2 or to start doing the shared task (Note that shared task B can be done by either worker 1 or worker 2). The sharing decision depends on a comparison of the total workload available at workstation 2 ("T") to the threshold value (also known as cut-off value) "R", which depends on the threshold rule. The DLB mechanism in this study is operated under the CONWIP policy. The reason to implement the CONWIP policy is that the DLB focuses on the worker capacities, for which the CONWIP policy is also a capacity constraint system. The threshold rule works as follows: worker 1 will send the shared task to worker 2 only if the total workload at workstation 2 (T) is less than the threshold (cut-off) value (R), $T < R$. Otherwise, worker 1 will start doing the shared task. In a real production line, the electronic display for both the threshold value and workload status (controlled by the Manufacturing Execution System (MES)) must be prepared before production activities. According to Figure 3, information content determines the total workload at workstation 2, which has two levels: high and low [8]. In this study, low information content (LIC) is called module 1, which is implemented under the SRNS rule proposed by Askin and Chen [8] and the original HFB proposed by Ostolaza et al. [18]. The workload

content (T) available at workstation 2 considers only the buffer content in front of worker 2, which contains the total jobs waiting to be processed. The workload in the buffer includes the total workload belonging to jobs that only fixed-task C has not done (Type-C job) and also jobs that contain an undone shared task (Type-B job). However, the original HFB considers only the workload from jobs that contain undone shared tasks. In contrast, high information content (HIC), which is module 2, can be operated under the new HFB rule improved by McClain et al. [19] and Gel et al. [17]. Besides the workload content in the buffer, the workload of the job which is processed by worker two is also considered.

3.2 Model assumption and threshold heuristic

3.2.1 Environmental assumptions

The production assembly line is a two-stage serial line operating under the Dynamic Line Balancing (DLB) mechanism. There is an infinite capacity buffer in front of each workstation. The total number of jobs in the system is restricted to the WIP level of the CONWIP policy. One worker operates each workstation. In the experiment, there are three skill levels of workers, which are high, medium, and low. At each skill level, a worker can operate the task with different processing times. Similar to Askin and Chen [8] and Gel et al. [17], each job has three tasks to be processed: Task A, Task B, and Task C. Task A is the fixed task which is designated for the preceding worker. Likewise, Task C is the fixed task that is designated for the succeeding worker. Task B is the shared task which can be done by either the succeeding worker or the preceding worker.

Our model assumptions can be summarized as follows:

- (1) There is a two-stage assembly production tandem line, operated by the CONWIP policy under the DLB mechanism.
- (2) The line is a one product type dedicated line.

- (3) There is a buffer in front of each workstation and each workstation can be operated by one worker at a time.
- (4) The job queue in the buffer is served as FCFS.
- (5) Pre-emption is not allowed.
- (6) There are ample raw materials and space to place the finished good inventory. Thus, the first worker starves only by the CONWIP restriction and the second worker never is blocked.
- (7) Only the first worker can decide whether to send the shared task or keep it.
- (8) The flow of the product is unidirectional. After the preceding worker passes a shared task to the succeeding worker, he/she cannot ask the succeeding worker to send the shared task back.
- (9) Every task is assumed to be identical. The processing time depends on the worker skill level only.

3.2.2 Heuristic1: HFB (Half-Full-Buffer)

The mechanism of the HFB rule: Worker 1 will pass on the shared task when the workload in the half-full intermediate buffer and the workload of succeeding workers are more than the threshold (cut-off) value. The total workload at the succeeding station includes 2 sources: Half-full intermediate buffer workload and workload of worker 2. The threshold value of the HFB rule (R_{HFB}) is given by

$$R_{HFB} = \frac{T_{B1}}{1 + v_2} + \frac{[\left(\frac{T_{B1}}{v_2} + T_C\right) + T_C]}{2} \times \frac{WIP - 2}{2} \quad (1)$$

where T_{B1} is the shared task workload of the preceding worker, WIP is the CONWIP level, and v_2 is the ratio of workers' shared task time. Details of the HFB rule mechanism are provided in [17].

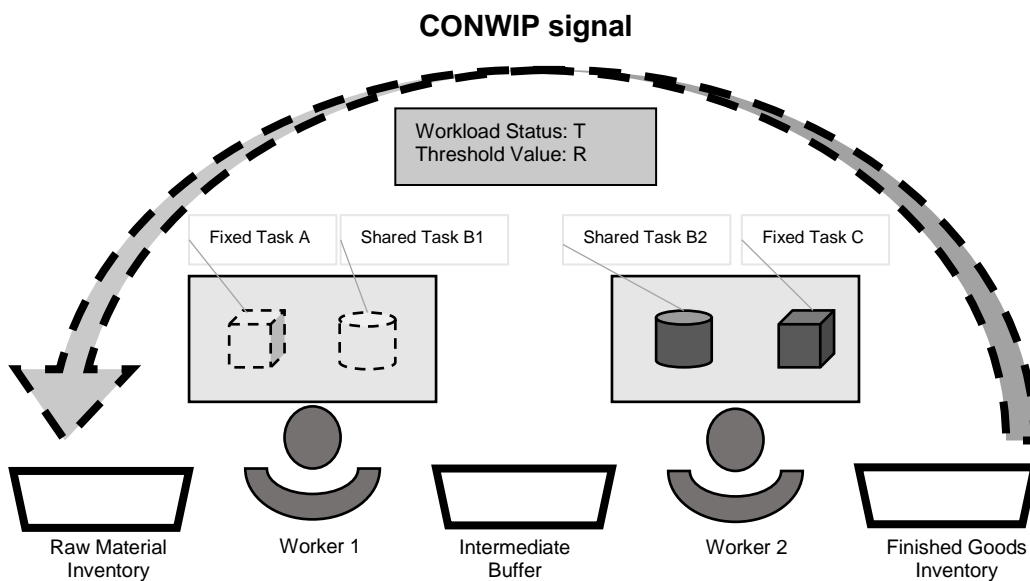


Figure 4. DLB mechanism

3.2.3 Heuristic2: SRNS (Smallest-R-No Starvation)

The mechanism of the SRNS rule: The first worker will pass on the shared task when the workload in the intermediate buffer is more than the threshold (cut-off) value. The workload of SRNS considers only the workload in the intermediate buffer. The rule seeks to avoid the starvation of workers. The threshold value of SRNS rule (R_{SRNS}) is given by

$$R_{SRNS} = \begin{cases} (WIP - 2)T_C & \text{if } T_B \geq (WIP - 2)T_C \\ [T_B, (WIP - 2)T_C] & \text{if } T_B < (WIP - 2)T_C \end{cases} \quad (2)$$

3.2.4 Heuristic3: SRNS-II (Smallest-R-No-Starvation-II)

While the HFB rule has already considered the difference of workers, the SRNS rule assumes identical workers. Under the situation where there are skill level differences among workers, the SRNS rule cannot be implemented due to the shared task workload of workers, which is identical. Therefore, we propose an SRNS-II rule to maximize the production output where workers' capabilities are different. For identical workers, the SRNS-II rule behaves similarly to the SRNS rule, but we consider the shared task workload instead of the Type-B job for the T_B value. The construction of SRNS-II is based on the following criteria:

- When workers are not identical, the shared task time of each worker is different since each worker operates a shared task with different capabilities. Thus, there are two values of shared task workload, which are T_{B1} and T_{B2} .
- The shared task workload of each worker is compared to the workload in the buffer content that contains only a Type-C job.
- There are 6 possible outcomes, as the shared task workload of each worker is compared to the workload in the buffer content.
 - (1) $T_{B1} > (WIP - 2)T_C$
 - (2) $T_{B1} = (WIP - 2)T_C$
 - (3) $T_{B1} < (WIP - 2)T_C$
 - (4) $T_{B2} > (WIP - 2)T_C$
 - (5) $T_{B2} = (WIP - 2)T_C$
 - (6) $T_{B2} < (WIP - 2)T_C$

Next, the mechanism of SRNS-II is based on the following steps:

- (1) Match each workload of workers into pairs.
 - (1) A and D (Divided into two cases since both T_{B1} and T_{B2} are higher than $(WIP - 2)T_C$)
 - (1) A and D where $T_{B1} > T_{B2}$
 - (2) A and D where $T_{B1} < T_{B2}$
 - (2) A and E
 - (3) A and F
 - (4) B and D
 - (5) B and F
 - (6) C and D

- (7) C and E
- (8) C and F (Divided into two cases since both T_{B1} and T_{B2} are lower than $(WIP - 2)T_C$)
 - (1) C and E where $T_{B1} > T_{B2}$
 - (2) C and F where $T_{B1} < T_{B2}$
- (2) Then, examine all possible workload situations to find the best R value for each of 10 cases, which are
 - (1) $(WIP - 2)T_C < T_{B2} < T_{B1}$
 - (2) $(WIP - 2)T_C = T_{B2} < T_{B1}$
 - (3) $T_{B1} < T_{B2} < (WIP - 2)T_C$
 - (4) $T_{B1} < T_{B2} = (WIP - 2)T_C$
 - (5) $T_{B2} < (WIP - 2)T_C < T_{B1}$
 - (6) $T_{B1} < (WIP - 2)T_C < T_{B2}$
 - (7) $T_{B1} = (WIP - 2)T_C < T_{B2}$
 - (8) $T_{B2} < T_{B1} < (WIP - 2)T_C$
 - (9) $(WIP - 2)T_C < T_{B1} < T_{B2}$
 - (10) $T_{B2} < T_{B1} = (WIP - 2)T_C$
 In each case, the candidate R values are examined with all possible workload situations.
- (3) After obtaining the best R value for each case, the SRNS-II rule as given by

$$R_{SRNS-II} = \begin{cases} (WIP - 2)T_C & \text{if } \begin{cases} (WIP - 2)T_C \leq T_{B2} < T_{B1} \\ T_{B2} < (WIP - 2)T_C < T_{B1} \\ T_{B2} < T_{B1} \leq (WIP - 2)T_C \\ T_{B1} = (WIP - 2)T_C < T_{B2} \end{cases} \\ [T_{B1}, (WIP - 2)T_C) & \text{if } \begin{cases} T_{B1} < T_{B2} = (WIP - 2)T_C \\ T_{B1} < (WIP - 2)T_C < T_{B2} \end{cases} \\ (0, (WIP - 2)T_C) & \text{if } (WIP - 2)T_C < T_{B1} < T_{B2} \\ [T_{B1}, T_{B2}) & \text{if } T_{B1} < T_{B2} < (WIP - 2)T_C \end{cases} \quad (3)$$

Figure 5 illustrates Step 2 of the search methodology of SRNS-II. A more detailed explanation of the SRNS-II rule mechanism is provided in Appendix A.

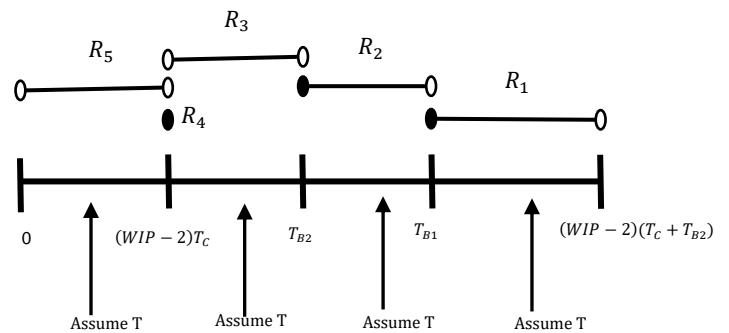


Figure 5. Intermediate buffer workload interval of SRNS-II

4. EXPERIMENTAL DESIGN

We are interested in investigating the production line behavior under the DLB mechanism when there are differences among workers. Simulations models are created and run using AnyLogic 8.0 Personal Learning Edition. Optimization experiments are conducted by using the OptQuest utility of AnyLogic®. Each simulation case is done by ten replications with a length of 43,200-time units. The warm-up length is obtained by checking

the throughput rate plots until it reaches a steady state. Table 1 shows the experimental settings of this study.

Table 1. Overview of the experiment settings

Settings	Gel et al.[17]	Askin and Chen [8]	This study
Control policy	CONWIP	CONWIP	CONWIP
WIP level	3-12	3,4,5	3,4,5
Mean	1,1.5,2	1	Depends on skill level
processing time			
Shared task ratio	0.1-0.9 (10% increment)	0.2-0.8 (20% increment)	0.25-0.75 (25% increment)
Rule tested	HFB, 50-50 work content	HFB, SRNS	HFB, SRNS-II
Optimal value	MDP, Dynamic programming	Simulation optimization	Simulation optimization
Time Distribution	Erlangen	Erlangen	Lognormal
Investigation	Sub-optimality, Throughput	Efficiency	Sub-optimality, Throughput
Task formation	Balanced, unbalanced	Balanced, unbalanced	Balanced
Stage	2	2,3,5	2
Decision Point	1,2	1,2,3,4,6	1,2,3,4,6
Total no. of Subtask	10	10,15,25	8
Workers differences	Succeeding worker = 2,4 times faster than preceding worker	N/A	Heterogeneous by skill level

Table 2. Task formation and portion of sharing workload

Task formation	Number of fixed task A	Number of shared task B	Number of fixed task C	Sharing workload
4-0-4	4	0	4	0%
3-2-3	3	2	3	2/8 = 25%
2-4-2	2	4	2	4/8 = 50%
1-6-1	1	6	1	6/8 = 75 %

Table 3. Skill level parameters and standard task

Skill level	Cap. coefficient (α_{ij})	Coeff. of variation (C_e)	Standard task time (T_{SD})	Qualification
High (H)	0.7	0.3		Experienced
Medium(M)	1	0.6		Standardized
Low (L)	1.3	0.9		Inexperienced

Table 4. Skill level configuration of a two-stage tandem line

Configuration	Pattern	Worker 1 skill level	Worker 2 skill level
	1	H	H
	2	M	M
	3	L	L
	4	H	M
	5	M	L
	6	H	L
	7	M	H
	8	L	H
	9	L	M

4.1 Setting 1: Task formation

To observe the improved opportunity of DLB, we divided each main task (Fixed Task A, Shared Task B, and Fixed Task C) into subtasks as proposed by McClain et al. [19]. Similar to Chen and Askin [3], Askin and Chen [8], McClain et al. [19], and Gel et al. [17], the task formation

illustrates the number of each type of task and the sharing-workload percentages.

4.2 Setting 2: Work time distribution

Following the settings of [8], [3], and [17]. The mean task time is a 1-time unit. The task time distribution is set to be the lognormal distribution. According to the field experiment done by Dudley [21], the operational cycle of operators on repetitive tasks is positively skewed when workers are doing the task at their own pace. The work from the field experiment of Knott and Sury [22] also observed that the work-time distribution of workers is positively skewed. Also, there is evidence supported by Powell and Pike [23], to use the lognormal distribution for modeling high variability workstations.

4.3 Setting 3: Skill level parameters

We categorize workers by the mean processing time and coefficient of variation (CV) into three levels: high skill, medium skill, and low skill. The worker that possesses the lowest skill in the line is considered a "bottleneck worker". Powell and Pyke [24] claimed that the notion of "Bottleneck" is widely used but not carefully defined. Goldratt and Cox [25] define bottleneck as a workstation/worker that has the longest processing time. The term "capability coefficient" is defined by the capability difference of workers. This coefficient determines the expected mean processing time, required by each skill level. The processing task time of each worker is calculated by

$$PT = \alpha_{ij} \times T_{SD} \tag{4}$$

For instance, if a high-skilled worker was assigned to the first station and a low-skilled worker was assigned to the second station, then $\alpha_{H1} = 0.7$ (30% faster than a medium-skilled worker) and $\alpha_{L2} = 1.3$ (30% slower than a medium-skilled worker), respectively. The work of Maggard [26] and Baudin [27] motivated this study to use this coefficient, to differentiate the processing time of each skill level. Baudin [27] stated that the predetermined time standards of a labor-intensive assembly line are typically in the range of $\pm 30\%$ of the actual times. According to the field experiment conducted by Knott and Sury [22], a CV from 0.57 to 0.22 was observed from well-motivated, skilled workers. In addition, we define the unskilled workers as low-skilled workers. The tasks of low-skilled workers could incur short adjustments or could face a quality problem. From these perspectives, we set the CV of low-skilled workers to be the highest (0.9). Even though the CV of low-skilled workers is the highest value, it is still in the range of the moderate class of variability, as stated by Hopp and Spearman [28]. The configuration of the worker skill levels in the two-stage tandem line is shown in Table 4.

5. RESULTS AND OBSERVATIONS

5.1 Optimality test

To check how well the threshold value of each heuristic attains optimality, we compare the throughput rate obtained from the threshold value (R) of both rules (HFB and SRNS-II) with an optimized threshold value (R*) through paired t-test at the 95% level of confidence. The test is then divided into two parts: optimality under identical workers and optimality under heterogeneous workers. In each case, the controlling variables are task formation, WIP level, and skill pattern.

5.1.1 HFB optimality

From Table 5, results show similar behavior to Askin and Chen [8] for identical workers. HFB shows promising optimality. Only 2 cases out of 27 cases show the non-optimal value. In contrast, the performance of HFB is worse when it is applied to heterogeneous workers. None of the cases equalize the throughput obtained from the optimized threshold value. We consider the skill level configurations 4 and 7 of task formation 3-2-3 of WIP level 3. As expected, the optimized threshold value of the HIC model (R*-HIC) tends to decrease when a non-bottleneck worker is assigned as the decision-maker (1.450<3.500). This is due to the nature of the DLB mechanism. The smaller the R value, the more likely that the decision-maker will keep doing his/her shared task. Therefore, an R value of 1.450 in the aforementioned case would force the decision-maker (high-skilled worker) to keep doing the shared task, which maximizes the throughput. From Equation (1), the threshold value calculated by the HFB rule represents the amount of workload that includes the expected idle time of two workers and the amount of workload in the half-full buffer. Under these conditions, HFB does not take the position of workers into account.

5.1.2 SRNS-II optimality

Under the identical worker settings, SRNS-II performs slightly worse than HFB. SRNS-II shows promising optimality even though the rule implements less information than HFB. Five cases out of 27 cases show non-optimality. From Table 7, the threshold value of SRNS-II in some cases can be more than one value, as long as it is in the specified range. For instance, formation 3-2-3 with WIP level 4 requires a threshold value from 1.4 to 4.2. As a result of the flexibility of the SRNS-II threshold value, the optimal threshold value of the above case (R*-LIC = 3.127) lies within the specified range. Likewise, the threshold value of SRNS-II from Table 8, such as task formation 3-2-3, patterns 4,5, and 6, could be any values as long as it is in the specified range. Overall, SRNS-II yields throughput that equalizes the optimal value more than HFB in a heterogeneous environment. As expected, the threshold value of the optimized LIC module becomes zero at low WIP with a large gap of skill level differences. From Table 8, pattern 6 with task formation 3-2-3 clarifies the situation. The reason for this is because the optimized model will force

Table 5. Optimality test of HFB heuristic: Identical workers

Formation	Pattern	WIP	TH*-HIC	TH-HFB	R*-HIC	R-HFB	p-value
		3	0.356	0.356	2.020	2.100	0.456
		4	0.357	0.357	2.770	3.500	0.392
		5	0.357	0.357	3.170	4.900	0.657
		3	0.244	0.244	2.960	3.000	0.779
		4	0.249	0.249	4.770	5.000	0.879
		5	0.250	0.250	7.240	7.000	0.791
		3	0.180	0.181	3.820	3.900	0.128
		4	0.187	0.187	6.260	6.500	0.264
		5	0.190	0.190	8.770	9.100	0.947
		3	0.349	0.349	2.430	2.800	0.905
		4	0.357	0.357	4.060	4.200	0.784
		5	0.357	0.357	5.430	5.600	0.860
		3	0.236	0.236	3.910	4.000	0.624
		4	0.247	0.247	5.600	6.000	0.022
		5	0.249	0.250	6.710	8.000	0.235
		3	0.175	0.176	3.810	5.200	0.127
		4	0.186	0.186	7.370	7.800	0.457
		5	0.190	0.190	9.260	10.400	0.996
		3	0.344	0.344	3.270	3.500	0.097
		4	0.356	0.356	4.970	4.900	0.411
		5	0.357	0.357	5.630	6.300	0.633
		3	0.223	0.228	3.340	5.000	0.023
		4	0.244	0.245	6.950	7.000	0.482
		5	0.249	0.249	8.420	9.000	0.527
		3	0.168	0.169	7.020	6.500	0.755
		4	0.183	0.183	8.810	9.100	0.982
		5	0.189	0.189	10.220	11.700	0.684

Bold numbers stand for optimality, p<0.05

Table 6. Optimality test of HFB heuristic: Heterogeneous workers

Formation	Pattern	WIP	TH*-HIC	TH-HFB	R*-HIC	R-HFB	p-value
		3	0.295	0.290	1.450	2.820	0.000
		4	0.320	0.297	2.560	4.820	0.000
		5	0.303	0.300	3.080	6.820	0.000
		3	0.208	0.207	2.140	3.730	0.000
		4	0.216	0.214	3.400	6.330	0.000
		5	0.219	0.217	4.650	8.930	0.000
		3	0.247	0.236	1.180	3.510	0.000
		4	0.254	0.245	1.270	6.110	0.000
		5	0.254	0.249	2.970	8.710	0.000
		3	0.297	0.290	3.500	2.220	0.000
		4	0.302	0.299	5.890	3.620	0.000
		5	0.303	0.301	7.890	5.020	0.000
		3	0.248	0.237	13.230	2.310	0.000
		4	0.253	0.246	9.530	3.710	0.000
		5	0.255	0.250	14.660	5.110	0.000
		3	0.207	0.207	5.990	3.130	0.014
		4	0.216	0.215	7.570	5.130	0.000
		5	0.220	0.217	10.400	7.130	0.000
		3	0.294	0.282	1.980	3.650	0.000
		4	0.302	0.292	3.600	5.650	0.000
		5	0.303	0.299	4.870	7.650	0.000
		3	0.208	0.200	2.440	4.860	0.000
		4	0.220	0.211	4.880	7.460	0.000
		5	0.219	0.217	6.570	10.060	0.000
		3	0.250	0.229	1.870	4.420	0.000
		4	0.270	0.244	2.560	7.020	0.000
		5	0.271	0.252	3.080	9.620	0.000
		3	0.292	0.284	4.330	3.050	0.000
		4	0.301	0.297	5.870	4.450	0.000
		5	0.303	0.302	7.980	5.850	0.000
		3	0.255	0.240	5.670	3.220	0.000
		4	0.266	0.254	7.970	4.620	0.000
		5	0.272	0.263	10.070	6.020	0.000
		3	0.206	0.202	5.900	4.260	0.000
		4	0.216	0.214	7.740	6.260	0.000
		5	0.219	0.218	10.260	8.260	0.044
		3	0.285	0.257	1.890	4.470	0.000
		4	0.294	0.284	3.780	6.470	0.000
		5	0.302	0.297	5.890	8.470	0.000
		3	0.201	0.191	2.190	5.990	0.000
		4	0.211	0.207	5.690	8.590	0.000
		5	0.218	0.215	8.300	11.190	0.000
		3	0.246	0.232	1.290	5.330	0.000
		4	0.264	0.235	2.510	7.930	0.000
		5	0.268	0.246	3.470	10.530	0.000
		3	0.281	0.266	5.180	3.870	0.000
		4	0.298	0.294	6.250	5.270	0.000
		5	0.302	0.300	8.450	6.670	0.000
		3	0.248	0.224	7.060	4.130	0.000
		4	0.259	0.248	8.580	5.530	0.000
		5	0.270	0.257	11.290	6.930	0.000
		3	0.199	0.193	6.810	5.390	0.000
		4	0.213	0.212	8.320	7.390	0.014
		5	0.218	0.216	11.460	9.390	0.004

Bold numbers stand for optimality, p<0.05

Table 7. Optimality test of SRNS-II heuristic: Identical workers

Formation	Pattern	WIP	TH-LIC	TH-SRNS-II	R*-LIC	R-SRNS-II	p-value
3	3	3	0.354	0.354	2.059	1.400	0.058
		4	0.357	0.357	3.127	1.400-3.400	0.631
	5	3	0.357	0.357	4.338	1.400-5.400	0.962
		4	0.248	0.247	1.636	2.000-6.000	0.049
	3	3	0.242	0.242	2.457	2.000-3.000	0.811
		4	0.248	0.247	1.636	2.000-6.000	0.049
	5	3	0.250	0.249	4.039	3.000-9.000	0.090
		4	0.179	0.180	2.650	2.600-3.600	0.104
	3	3	0.187	0.186	6.040	2.600-6.600	0.043
		4	0.190	0.189	6.507	3.600-10.600	0.013
	3	3	0.348	0.348	0.974	1.400	0.053
		4	0.356	0.357	2.128	2.800	0.281
	5	3	0.357	0.357	3.482	2.800-3.800	0.675
		4	0.237	0.237	1.969	2.000	0.672
	3	3	0.247	0.247	3.415	4.000	0.325
		4	0.249	0.249	4.765	4.000-6.000	0.228
	3	3	0.177	0.176	1.708	2.600	0.023
		4	0.186	0.186	4.997	5.200	0.566
	5	3	0.190	0.189	7.267	5.200-7.200	0.057
		4	0.337	0.337	0.449	0.700	0.696
	3	3	0.356	0.354	2.800	1.400	0.000
		4	0.357	0.357	3.284	2.100	0.005
	3	3	0.227	0.227	0.560	1.000	0.663
		4	0.244	0.244	6.087	2.000	0.463
	5	3	0.249	0.248	2.632	3.000	0.507
4		0.169	0.170	0.478	1.300	0.078	
3	3	0.183	0.183	5.336	2.600	0.187	
	4	0.188	0.188	5.293	3.900	0.761	

Bold numbers stand for optimality, p<0.05

the preceding worker to do all of the shared tasks when the threshold value becomes zero. Thus, formation 3-2-3 automatically becomes a 5-0-3 formation.

5.2 Throughput improvement opportunity

Work-sharing under the DLB mechanism can improve the throughput rate of a production line even under a low level of WIP. To guarantee this, we observe the improvement opportunity of threshold heuristic rules by using one-way ANOVA at the 95% level of confidence as a statistical tool for all 81 test cases of the identical worker environment and the heterogeneous worker environment. The no-sharing system (task formation 4-0-4) was set as the benchmark.

5.2.1 Improvement opportunity in identical worker environment

Based on Figure 6, overall, HFB improves the throughput rate by 2%, 1% more than SRNS-II in an identical worker environment. The improvement opportunity of both rules seems to be lost in formation1-6-1 with a low level of WIP since worker 1 may make a wrong decision to send the shared task to worker 2 and face immediate starvation afterward. Some cases even deteriorate the line performance rather than improve it. The benefits of DLB can be clearly seen in formation 3-2-3 of pattern 2 under the HFB rule. DLB can achieve higher throughput while keeping WIP at a minimum. The no-sharing system

Table 8. Optimality test of SRNS-II heuristic: Heterogeneous workers

Formation	Pattern	WIP	TH-LIC	TH-SRNS-II	R*-LIC	R-SRNS-II	p-value
3	3	3	0.284	0.281	0.000	1.640-1.980	0.000
		4	0.300	0.300	1.629	1.510-1.970	0.343
	5	3	0.303	0.303	0.899	1.490-1.980	0.121
		4	0.201	0.201	1.599	2.054-2.553	0.985
	3	3	0.216	0.215	3.070	2.053-2.499	0.226
		4	0.219	0.219	1.303	2.187-2.561	0.323
	3	3	0.248	0.227	0.000	1.428-2.574	0.000
		4	0.253	0.249	0.000	1.480-2.414	0.000
	3	3	0.254	0.253	0.615	1.468-2.501	0.237
		4	0.297	0.296	3.389	2.100	0.223
	3	3	0.302	0.302	5.029	4.200	0.254
		4	0.303	0.303	6.913	6.300	0.923
	3	3	0.248	0.244	10.373	2.100	0.000
		4	0.252	0.252	8.626	4.200	0.826
	3	3	0.255	0.253	13.34	6.300	0.000
		4	0.210	0.210	2.334	3.000	0.157
	3	3	0.217	0.217	5.454	6.000	0.675
		4	0.219	0.219	9.792	9.000	0.235
	3	3	0.276	0.277	0.905	0.065-1.056	0.229
		4	0.298	0.290	0.862	2.831-3.979	0.000
	3	3	0.303	0.303	2.068	2.819-3.929	0.380
		4	0.200	0.200	2.051	0.688-2.465	0.715
	3	3	0.214	0.210	1.120	4.043-5.185	0.000
		4	0.219	0.219	3.747	4.003-5.196	0.230
	3	3	0.236	0.230	0.000	0.214-2.398	0.000
		4	0.258	0.242	2.150	2.959-5.171	0.000
	3	3	0.269	0.264	1.299	3.122-5.162	0.000
		4	0.291	0.286	2.723	1.400	0.000
	3	3	0.301	0.297	4.768	2.800	0.000
		4	0.303	0.297	6.468	4.200	0.000
	3	3	0.250	0.245	3.146	1.400	0.000
		4	0.267	0.251	5.373	2.800	0.000
	3	3	0.270	0.252	7.916	4.200	0.000
		4	0.205	0.205	0.205	2.000	0.322
	3	3	0.214	0.214	2.825	4.000	0.650
		4	0.219	0.215	9.072	6.000	0.000
	3	3	0.261	0.261	0.636	0.119-0.918	0.916
		4	0.288	0.288	0.909	0.097-1.872	0.710
	3	3	0.299	0.297	2.442	0.723-2.899	0.074
		4	0.192	0.192	0.180	0.108-1.200	0.646
	3	3	0.207	0.207	1.984	0.279-2.526	0.118
		4	0.217	0.215	2.851	0.153-3.810	0.104
	3	3	0.217	0.216	0.613	0.334-1.188	0.335
		4	0.248	0.244	1.152	0.219-2.267	0.035
	3	3	0.264	0.258	1.774	0.286-3.549	0.019
4		0.280	0.266	2.021	0.700	0.000	
3	3	0.296	0.292	5.150	1.400	0.000	
	4	0.303	0.293	6.067	2.100	0.000	
3	3	0.241	0.227	4.013	0.700	0.000	
	4	0.255	0.244	7.892	1.400	0.000	
3	3	0.269	0.244	9.335	2.100	0.000	
	4	0.196	0.194	4.692	1.000	0.001	
3	3	0.212	0.211	3.331	2.000	0.469	
	4	0.218	0.212	8.435	3.000	0.000	

Bold numbers stand for optimality, p<0.05

requires WIP level 5, to achieve throughput equalizing the DLB under HFB of WIP level 3. The performance of DLB is good, notably when workers' variability increases. Overall, approximately a 2% improvement opportunity is achieved under low-skilled workers. Starvation tends to occur more frequently in a skilled-worker environment than a low-skilled worker environment. As a result, DLB yields only a 0.5% improvement, on average.

5.2.2 Improvement opportunity in heterogeneous worker environment

Unlike the identical worker environment, with the DLB control mechanism, bottleneck workers can borrow the capacity from the higher-skilled workers, as mentioned by Hopp et al. [29]. The workload of the bottleneck worker can be transferred to the adjacent station. When the bottleneck worker is located at workstation2, the T value increases. The decision maker will keep doing the shared task until the T value is automatically less than the R value. In contrast, if a bottleneck worker is assigned to workstation 1, the T value decreases since most jobs accumulate in front of worker 1. Therefore, worker 1 tends to send the shared task to worker 2. As a result, Figure 7 shows the throughput improvement made by almost all the cases in a heterogeneous worker environment. mechanism significantly improved the line performance.

SRNS-II outperforms HFB in formation 3-2-3. Both rules improve the throughput rate of the line even when increasing the WIP level. This shows that DLB can improve an unbalanced line with a high WIP level, for instance, in the case of task formation 1-6-1 with skill configuration 6. Both rules improve throughput rates from 0% to 12%. Overall, assigning high-skilled workers and low-skilled workers under the DLB

5.3 Granularity of shared tasks

We are interested in a system in which pre-emption is not allowed. Unlike a task that can be pre-empted, the succeeding worker cannot take the job unless the preceding worker passes on the job. The granularity allows the preceding worker to make multiple sharing decisions. Figure 8 illustrates the mechanism of granularity of formation 2-4-2 in DLB. Granularity splits the shared tasks into shared subtasks by setting the DP value, which indicates how many times the preceding worker can decide to keep/send the shared task. For instance, 4DP stands for 4 decision points. The preceding workers can decide 4 times to share or keep doing the shared task after finishing each shared subtask. Therefore, the shared task becomes more agile. The granularity can increase the production output of the line.

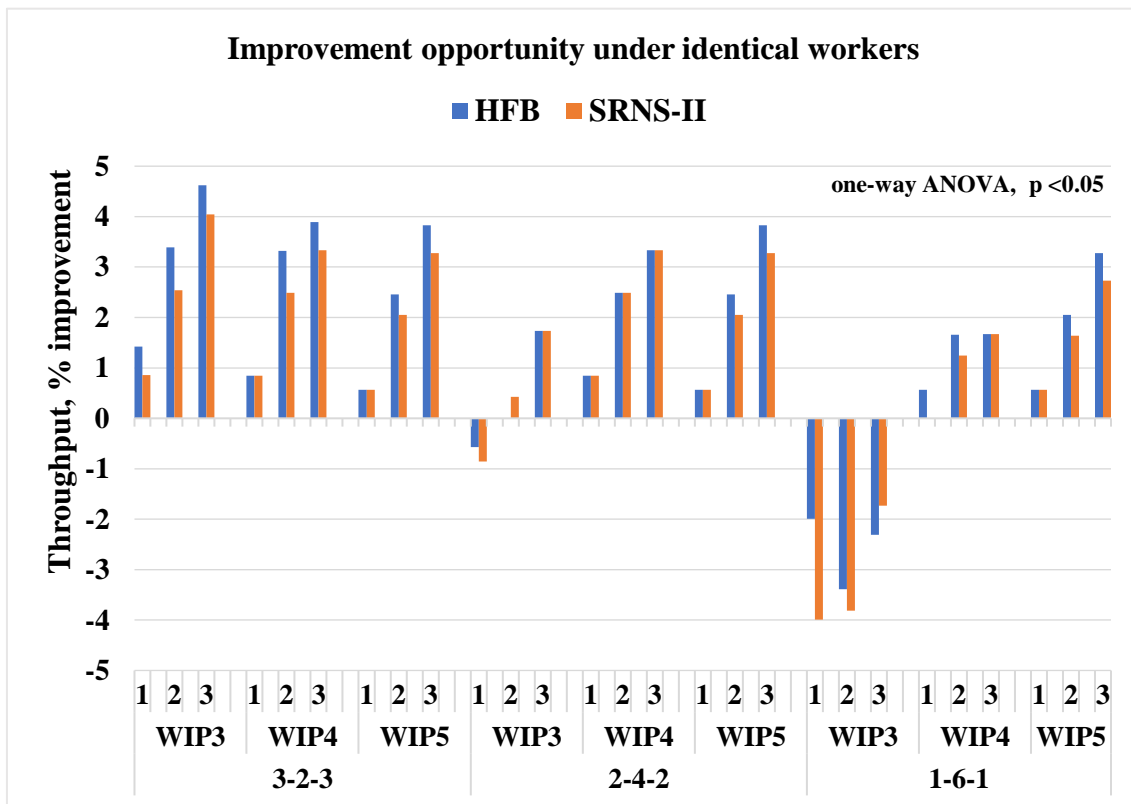


Figure 6. HFB outperforms SRNS-II under identical workers

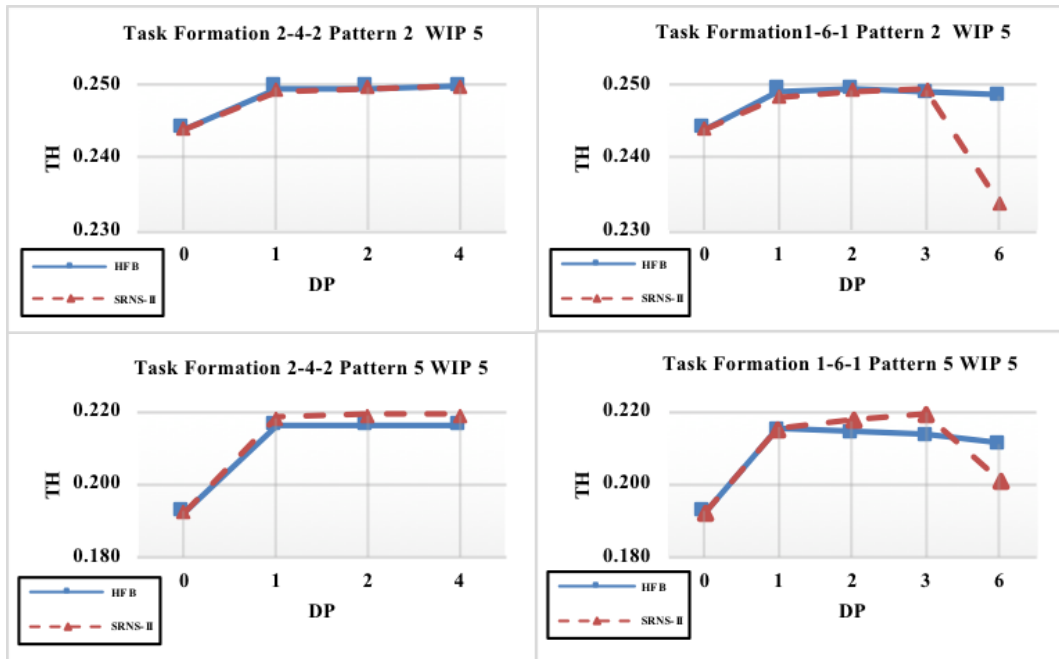


Figure 9. SRNS-II exhibits an inverted bowl-shaped pattern as the shared task becomes more granular under the pair of identical workers and non-bottleneck decision maker

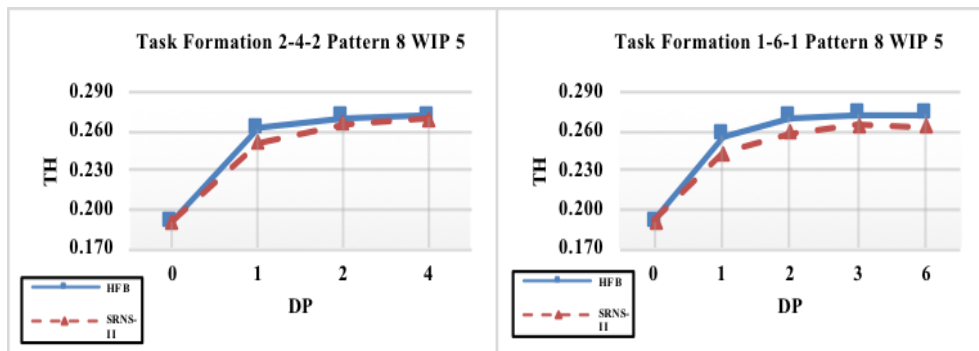


Figure 10. Throughput rate reaches the upper bound as the shared task becomes more granular with the bottleneck decision-maker

When there is more WIP in the system, with a higher DP, the decision-maker decides more frequently to pass the shared task to the next worker. If the decision-maker passes the job to the next worker without the information of the next worker (module 1), jobs may accumulate in front of the next worker. Our result from Figure 9 shows that when a shared task becomes larger with more decision points under the SRNS-II heuristic, DLB decreases the throughput rate of the line.

5.3.2 Granularity part 2: Non-bottleneck worker as a decision-maker

Jobs accumulate in front of the bottleneck worker, which increase the workload in the buffer more than the other two cases (identical worker case and bottleneck worker as a decision-maker case). The bottleneck worker sets the production rate. Thus, the throughput rate of the line should remain constant even if the shared tasks become more granular. Figure 9 shows this for medium-skilled workers and low-skilled workers. Workload information on the low-skilled worker affects the throughput rate when increasing DP. Consider task formation 1-6-1 in which both workers are almost completely cross-trained. Under SRNS-II, worker 1 could pass on the shared task

while neglecting the workload of bottleneck workers. As DP increases, worker 1 could pass on the shared job more frequently to the bottleneck worker (since the T value will decrease with time). As a result, Figure 9 shows the inverted bowl-shape under task formation 1-6-1 of the SRNS-II heuristic. This is unlike the traditional bowl-phenomenon where the optimal throughput can be achieved when assigning the fastest workstation in the middle. Our results show that the optimal throughput can be found in the middle range of granularity under the SRNS-II heuristic.

5.3.3 Granularity part 3: Bottleneck worker as a decision-maker

A pair of low-skilled and high-skilled workers at the line operated with a WIP level of 5 represents the tendency of granularity on the throughput rate of all cases in this configuration. Under task formation 2-4-2 and 1-6-1, both of the rules show a similar tendency of the throughput rate. The throughput rates of the line tend to reach the upper bound when increasing the number of decision points. If the bottleneck worker was assigned as a decision-maker, he or she will have more chances to pass the shared task to the adjacent worker (who works faster) when the shared task becomes more granular. In this case, the bottleneck worker can borrow more capacity from a higher-skilled worker. Our results from Figure 10 confirm that when the shared task becomes more granular, the throughput rate of both rules keeps increasing until the upper bound is reached.

6. CONCLUSION

This study investigates the impact of work-sharing on a two-stage tandem production line under the DLB mechanism among partially cross-trained workers whose capabilities differ. We believe this is the first study that considers the workers' skill-level differences under the DLB mechanism as the main interest. The threshold heuristic is proposed, namely SRNS-II, which is developed from the SRNS heuristic of Askin and Chen [8]. A numerical simulation experiment is performed to improve the HFB rule by Gel et al. [17]. The SRNS-II heuristic yields promising results for both identical and heterogeneous worker configurations even though the rule itself requires only a single information content from the intermediate buffer. Overall, HFB and SRNS-II could improve the throughput rate of the production line by 12%.

The production line manager can implement the concept of DLB in a situation with turnover and absenteeism. DLB can be used for an unbalanced line, under heterogeneous workers. The throughput rate could be maximized and also the WIP level (which is low). However, DLB requires equipment such as MES for controlling the WIP level and tracking a worker's workload, and tools for each workstation. The investigation of a trade-off between tooling costs and throughput improvement remains as future work. Moreover, the impact of the social loafing effect as investigated by Cantor and Jin [30] also needs to be examined. Our model is limited to only a two-stage tandem dedicated production line. To investigate the effectiveness of the SRNS-II rule, the line can be extended with multiple sharing zones. The variability from other sources rather than workers is also an attractive area for research, for instance, a line that produces multiple product types. Lastly, to verify the benefits of DLB, it is necessary to conduct a field experiment to guarantee the feasibility of this theoretical concept.

7. ACKNOWLEDGMENTS

The authors would like to thank the editorial team, the anonymous reviewers, and Dr. Sun Olapiriyakul for their insightful suggestions. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

8. REFERENCES

- [1] Hudson, S., McNamara, T., Shaaban, S., "Unbalanced lines: Where are we now?," *Int. J. Prod. Res.*, vol. 53, no. 6, pp. 1895-1911, Sep. 2014, doi: 10.1080/00207543.2014.965357.
- [2] Gel, E. S., Hopp, W. J., Van Oyen, M. P., "Hierarchical cross-training in work-in-process-constrained systems," *IIE Transactions*, vol. 39, no. 2, pp. 125-143, Feb. 2007, doi: 10.1080/07408170600729184.
- [3] Chen, J., Askin, R. G., "Throughput maximization in serial production lines with worksharing," *Int. J. Prod. Econ.*, vol. 99, no. 1, pp. 88-101, Feb. 2006 doi: 10.1016/j.ijpe.2004.12.009.
- [4] Bartholdi, J. J., Eisenstein, D. D., "A production line that balances itself," *Operations Research*, vol. 44, no.1, pp. 21-34, May 1999, doi: 10.1287/opre.44.1.21.
- [5] Alves, A. C., "U-shaped cells operating modes: A review and hands-on simulation comparison," *Int. J. Ind. Eng. Manag.*, vol. 9 no. 2, pp. 87-97, Jun. 2018.
- [6] Bukchin, Y., Cohen, Y., "Minimising throughput loss in assembly lines due to absenteeism and turnover via work-sharing," *Int. J. Prod. Res.*, vol. 51, no. 20, pp. 6140-6151, Oct. 2013, doi: 10.1080/00207543.2013.807374.
- [7] Buzacott, J. A., "The impact of worker differences on production system output," *Int. J. Prod. Econ.*, vol. 78, no. 1, pp. 37-44, Feb. 2002, doi: 10.1016/S0925-5273(00)00086-4.
- [8] Askin, R. G., Chen, J., "Dynamic task assignment for throughput maximization with worksharing," *Eur. J. Oper. Res.*, vol. 168, no. 3, pp. 853-869, Feb. 2006, doi: 10.1016/j.ejor.2004.07.033.
- [9] Wu, L., Cai, F., Li, L., Chu, X., "Cross-trained worker assignment problem in cellular manufacturing system using swarm intelligence metaheuristics," *Math. Prob. Eng.*, vol. 2018, no.1, pp. 1-15, Nov.2018, doi: 10.1155/2018/4302062.
- [10] Kiassat, C., Safaei, N., "Effect of imprecise skill level on workforce rotation in a dynamic market," *Com. and Ind. Eng.*, vol. 131, pp. 464-476, Apr. 2018, doi: 10.1016/j.cie.2018.04.019.
- [11] Niakan, F., Baboli A., Moyaux, T., Botta-Genoulaz, V., "A bi-objective model in sustainable dynamic cell formation problem with skill-based worker assignment," *J. Manu. Sys.*, vol. 38, pp. 46-62, Jan. 2016, doi: 10.1016/j.jmsy.2015.11.001.
- [12] Digiesi, S., Facchini, F., Mossa, G., Mummolo, G., "Minimizing and balancing ergonomic risk of workers of an assembly line by job rotation: A MINLP model," *Int. J. Ind. Eng. Manag.*, vol. 9 no. 3, pp. 129-138, Sep. 2018, doi: 10.24867/IJIEM-2018-3-129.
- [13] De Bruecker, P., Van den Bergh, J., Beliën, J., Demeulemeester E., "Workforce planning incorporating skills: State of the art," *Eur. J. Oper. Res.*, vol. 243, no. 1, pp. 1-16, May 2015, doi: 10.1016/j.ejor.2014.10.038.
- [14] Wang, C., Kang, N., Zheng L., "A factory-level dynamic operator allocation policy: The bubble allocation" *Int. J. Prod. Res.*, vol. 54, no. 14, pp. 4240-4255, Jan. 2016, doi: 10.1080/00207543.2014.965357.
- [15] Lian, J., Liu, C., Li, W., Yin, Y., "A multi-skilled worker assignment problem in seru production systems considering the worker heterogeneity," *Com. and Ind. Eng.*, vol. 118, pp. 366-382, Feb. 2018, doi: 10.1016/j.cie.2018.02.035.
- [16] Yu, Y., Wang, J., Ma, K., Sun, W., "Seru system balancing: Definition, formulation, and exact solution," *Com. and Ind. Eng.*, vol.122, pp. 318-325, May 2018, doi: 10.1016/j.cie.2018.05.048.
- [17] Gel, E. S., Hopp, W. J., Van Oyen, M. P., "Factors affecting opportunity of worksharing as a dynamic line balancing mechanism," *IIE Transactions*, vol. 34, no.10, pp. 847-863, Oct. 2002, doi: 10.1080/07408170208928917.
- [18] Ostolaza, J., McClain, J. O., Thomas, J., "The use of dynamic state dependent assembly-line balancing to improve throughput," *J. Manu. and Oper. Manag.*, vol. 3, no. 2, pp. 105-133, Jun. 1990.
- [19] McClain, J. O., Thomas J. L., Sox, C., "'On-the-fly' line balancing with very little WIP," *Int. J. Prod. Econ.*, vol. 27, no. 3, pp. 283-289, Feb. 1992.
- [20] Spearman, M. L., Woodruff, D. L., Hopp, W. J., "CONWIP: A pull alternative to Kanban," *Int. J. Prod. Res.*, vol. 28, no. 5, pp. 879-894, May. 1990, doi: 10.1080/00207549008942761.
- [21] Dudley, N. A., "The effect of pacing on worker performance," *Int. J. Prod. Res.*, vol. 1, no.2, pp. 60-72, Jan. 1962, doi: 10.1080/00207546108943082.
- [22] Knott, K., Sury R. J., "A study of work-time distributions on unpaced tasks," *IIE Transactions*, vol. 19, no. 1, pp. 50-55, Mar. 1987, doi: 10.1080/07408178708975369.
- [23] Powell, S. G., Pyke, D. F., "Buffering unbalanced assembly systems," *IIE Transactions*, vol. 30, no. 1, pp. 55-65, Jan. 1997, doi: 10.1023/A:1007493512502.
- [24] Powell, S. G., Pyke, D. F., "Allocation of buffers to serial production lines with bottlenecks," *IIE Transactions*, vol. 28, no. 1, pp. 18-29, Jan. 1996, doi: 10.1080/07408179608966249.
- [25] Goldratt, E., and Cox, J., *The Goal*, Second Revised Edition, Great Barrington, MA, USA: North River Press, 1992.
- [26] Maggard, M. J., Hogg, G. L., Phillips, D. T., "The efficiency and economics of small heterogeneous labour limited queueing systems," *Int. J. Prod. Res.*, vol. 18, no. 5, pp. 619-636, Sep. 1980, doi: 10.1080/00207548008919694.
- [27] Baudin, M., *Lean Assembly*. Boca Raton, FL, USA: CRC Press, 2002.
- [28] Hopp, W. J., Spearman, M. L., *Factory Physics*, Third Edition, Long Grove, IL, USA: Waveland Press Inc., 2011.
- [29] Hopp, W. J., Tekin, E., Van Oyen, M. P., "Benefits of skill chaining in serial production lines with cross-trained workers," *Management Science*, vol. 50, no. 1, pp. 83-98, Jan. 2004, doi: 10.1287/mnsc.50.1.83.27051.
- [30] Cantor, D. E., Jin Y., "Theoretical and empirical evidence of behavioural and production line factors that influencing helping behavior," *J. Oper. Manage.*, vol. 65, pp. 312-332, Mar. 2019, doi: 10.1002/joom.1019.

9. APPENDIX A: VALIDATION OF SRNS-II

9.1 Workers idle and starvation conditions

To prevent all possible immediate starvation and minimize myopic starvation of workers, we examine all conditions that affect the starvation of workers. The starvation of workers occurs when there are no available jobs for the workers. Two types of starvation are considered in our validity, immediate starvation and myopic starvation, as mentioned by Askin and Chen [6].

9.1.1 Immediate starvation of worker 1

We define immediate starvation as starvation of worker 1 when there is no available job for worker 1 after worker 1 has just passed on the job to worker 2.

$$\text{Condition 1: } T \geq (WIP - 2)T_C$$

At least (WIP-2) jobs are available at stage 2. Thus, if $T < R$, then worker 2 takes the shared task. Consequently, worker 1 may starve immediately.

$$\text{Condition 2: } T < (WIP - 2)T_C$$

The workload at stage 2 is less than (WIP-2) jobs. If $T < R$, then worker 2 takes the shared task. Then, the total number of jobs at stage 2 becomes less than (WIP-1). Hence, worker 1 will have at least one job waiting at stage 1.

9.1.2 Immediate starvation of worker 2

We define myopic starvation as starvation of worker 2 when worker 2 completes the current job before worker 1 finishes the shared task of the current job.

$$\text{Condition 3: } T \geq T_{B1}$$

The workload of stage 2 is at least equal to the shared task workload of worker 1. Even when $T \geq R$, worker 1 decides to continue doing the shared task. Worker 2 still has jobs to do and will not face myopic starvation.

$$\text{Condition 4: } T < T_{B1}$$

The workload at stage 2 is less than the shared task workload of worker 1. When $T \geq R$, worker 1 does the shared task. Consequently, worker 2 will face myopic starvation.

9.2 Threshold value search methodology

The best threshold (cut-off) value can be found by search methodology. Similar to the SRNS rule searching methodology proposed by Askin and Chen [6], candidate R values are defined in each of 10 cases. Consequently, every possible T value is assumed in each R value setting to find the best R value that does not let workers starve or face myopic starvation. For brevity, the conclusion of each setting is provided below.

$$\text{Case 1: } (WIP - 2)T_C < T_{B2} < T_{B1}$$

$$\text{Setting1 } (R_1): R \geq T_{B1}$$

$$\text{Setting2 } (R_2): T_{B2} < R < T_{B1}$$

$$\text{Setting3 } (R_3): (WIP - 2)T_C < R \leq T_{B2}$$

$$\text{Setting4 } (R_4): R = (WIP - 2)T_C$$

$$\text{Setting5 } (R_5): R < (WIP - 2)T_C$$

Both setting 4 and setting 5 only let worker 2 face myopic starvation. Setting 4 is chosen because of $R_4 > R_5$. It is more likely that worker 1 will send the shared task to worker 2 who possesses a higher skill level than worker 1.

$$\text{Case 2: } (WIP - 2)T_C = T_{B2} < T_{B1}$$

$$\text{Setting1 } (R_1): R \geq T_{B1}$$

$$\text{Setting2 } (R_2): T_{B2} < R < T_{B1}$$

$$\text{Setting3 } (R_3): R = (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): R < (WIP - 2)T_C$$

Both setting 3 and setting 4 only let worker 2 face myopic starvation. Setting 3 is chosen because of $R_3 > R_4$. It is more likely that worker 1 will send the shared task to worker 2 who possesses a higher skill level than worker 1.

$$\text{Case 3: } T_{B1} < T_{B2} < (WIP - 2)T_C$$

$$\text{Setting1 } (R_1): R > (WIP - 2)T_C$$

$$\text{Setting2 } (R_2): R = (WIP - 2)T_C$$

$$\text{Setting3 } (R_3): T_{B2} \leq R < (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): T_{B1} \leq R < T_{B2}$$

$$\text{Setting5 } (R_5): R < T_{B1}$$

Both setting 3 and setting 4 will prevent both workers from facing starvation. R_4 is chosen because of $R_4 < R_3$, providing more opportunity to worker 1, to keep doing the shared task.

$$\text{Case 4: } T_{B1} < T_{B2} = (WIP - 2)T_C$$

$$\text{Setting1 } (R_1): R > (WIP - 2)T_C$$

$$\text{Setting2 } (R_2): R = (WIP - 2)T_C$$

$$\text{Setting3 } (R_3): T_{B1} \leq R < (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): R < T_{B1}$$

Both setting 2 and setting 3 do not let workers face starvation. Setting 3 is chosen because of $R_3 < R_2$, which tends to let worker 1 (higher skilled worker) do more of the shared task than worker 2.

$$\text{Case 5: } T_{B2} < (WIP - 2)T_C < T_{B1}$$

$$\text{Setting1 } (R_1): R \geq T_{B1}$$

$$\text{Setting2 } (R_2): (WIP - 2)T_C < R < T_{B1}$$

$$\text{Setting3 } (R_3): R = (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): T_{B2} \leq R < (WIP - 2)T_C$$

$$\text{Setting5 } (R_5): R < T_{B2}$$

Both setting 3 and setting 4 only let worker 2 face myopic starvation. Setting 3 is chosen because of $R_3 > R_4$. It is more likely that worker 1 will send a shared task to worker 2 who possesses a higher skill level than worker 1.

$$\text{Case 6: } T_{B1} < (WIP - 2)T_C < T_{B2}$$

$$\text{Setting1 } (R_1): R \geq T_{B2}$$

$$\text{Setting2 } (R_2): (WIP - 2)T_C < R < T_{B2}$$

$$\text{Setting3 } (R_3): R = (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): T_{B1} \leq R < (WIP - 2)T_C$$

$$\text{Setting5 } (R_5): R < T_{B1}$$

Both setting 3 and setting 4 do not let workers face starvation. Setting 4 is chosen because of $R_4 < R_3$, providing more opportunity to worker 1, to keep doing the shared task.

$$\text{Case 7: } T_{B1} = (WIP - 2)T_C < T_{B2}$$

$$\text{Setting1 } (R_1): R \geq T_{B2}$$

$$\text{Setting2 } (R_2): (WIP - 2)T_C < R < T_{B2}$$

$$\text{Setting3 } (R_3): R = (WIP - 2)T_C$$

$$\text{Setting4 } (R_5): R < T_{B1}$$

Setting 3 is chosen because both workers will not face starvation.

$$\text{Case 8: } T_{B2} < T_{B1} < (WIP - 2)T_C$$

$$\text{Setting1 } (R_1): R > (WIP - 2)T_C$$

$$\text{Setting2 } (R_2): R = (WIP - 2)T_C$$

$$\text{Setting3 } (R_3): T_{B1} \leq R < (WIP - 2)T_C$$

$$\text{Setting4 } (R_4): T_{B2} \leq R < T_{B1}$$

$$\text{Setting5 } (R_5): R < T_{B2}$$

Both setting 2 and setting 3 will not let workers face starvation. Setting 2 is chosen because of $R_2 > R_3$. It is more likely that worker 1 will send the shared task to worker 2 who possesses a higher skill level than worker 1.

$$\text{Case 9: } (WIP - 2)T_C < T_{B1} < T_{B2}$$

$$\text{Setting1 } (R_1): R \geq T_{B2}$$

$$\text{Setting2 } (R_2): T_{B1} \leq R < T_{B2}$$

$$\text{Setting3 } (R_3): (WIP - 2)T_C < R < T_{B1}$$

$$\text{Setting4 } (R_4): R = (WIP - 2)T_C$$

$$\text{Setting5 } (R_5): R < (WIP - 2)T_C$$

Both setting 4 and setting 5 only let worker 2 face myopic starvation. Setting 5 is chosen because worker 1 possesses a higher skill level than worker 2. Moreover, $R_5 < R_4$ tends to allow worker 1 to do more of the shared task than worker 2. The threshold value has to be a positive number since $R = 0$ will force worker 1 to do the shared task (all the time). Thus, we set $R_5 = (0, (WIP - 2)T_C)$.

Case 10: $T_{B2} < T_{B1} = (WIP - 2)T_C$

Setting1 (R_1): $R > (WIP - 2)T_C$

Setting2 (R_2): $R = (WIP - 2)T_C$

Setting3 (R_3): $T_{B2} \leq R < (WIP - 2)T_C$

Setting4 (R_4): $R < T_{B2}$

Only setting 2 does not let workers face starvation. Thus, R_2 is selected. To sum up, SRNS-II can be expressed as Equation (3) provided in section 3.2.4.

9. APPENDIX B: RESULTS OF SECTION 5.2 AND 5.3

Table 9. Improvement opportunity of DLB: Identical workers

Formation	Pattern	WIP	TH- No share	TH-HFB	TH-SRNS-II	%Imp HFB	%Imp SRNS-II
		3	0.351	0.356*	0.354	1%	1%
		4	0.354	0.357*	0.357	1%	1%
		5	0.355	0.357*	0.357	1%	1%
		3	0.236	0.244*	0.242	3%	3%
		4	0.241	0.249*	0.247	3%	2%
		5	0.244	0.250*	0.249	2%	2%
		3	0.173	0.181*	0.180	5%	4%
		4	0.180	0.187*	0.186	4%	3%
		5	0.183	0.190	0.189	4%	3%
		3	0.351*	0.349	0.348	-1%	-1%
		4	0.354	0.357	0.357	1%	1%
		5	0.355	0.357	0.357	1%	1%
		3	0.236	0.236	0.237*	0%	0%
		4	0.241	0.247	0.247	2%	2%
		5	0.244	0.250	0.249	2%	2%
		3	0.173	0.176	0.176	2%	2%
		4	0.180	0.186	0.186	3%	3%
		5	0.183	0.190	0.189	4%	3%
		3	0.351*	0.344	0.337	-2%	-4%
		4	0.354	0.356*	0.354	1%	0%
		5	0.355	0.357	0.357	1%	1%
		3	0.236*	0.228	0.227	-3%	-4%
		4	0.241	0.245	0.244	2%	1%
		5	0.244	0.249*	0.248	2%	2%
		3	0.173*	0.169	0.170	-2%	-2%
		4	0.180	0.183	0.183	2%	2%
		5	0.183	0.189	0.188	3%	3%

* Optimal value, $p < 0.05$

Bold numbers stand for second optimal value or shared optimality, $p < 0.05$.

Table 10. Improvement opportunity of DLB: Heterogeneous workers

Formation	Pattern	WIP	TH- No share	TH-HFB	TH-SRNS-II	%Imp HFB	%Imp SRNS-II
		3	0.250	0.290*	0.281	16%	12%
		4	0.251	0.297	0.300*	19%	20%
		5	0.250	0.300	0.303*	20%	21%
		3	0.191	0.207*	0.201	8%	5%
		4	0.192	0.214	0.215*	11%	12%
		5	0.193	0.217	0.219*	13%	14%
		3	0.193	0.236*	0.227	22%	18%
		4	0.192	0.245	0.249*	28%	30%
		5	0.192	0.249	0.253*	29%	32%
		3	0.250	0.291	0.296*	16%	19%
		4	0.250	0.299	0.302*	20%	21%
		5	0.250	0.301	0.303*	21%	21%
		3	0.192	0.237	0.244*	24%	27%
		4	0.192	0.246	0.252*	28%	31%
		5	0.192	0.250	0.253*	30%	32%
		3	0.191	0.207	0.210*	8%	10%
		4	0.192	0.215	0.217*	12%	13%
		5	0.192	0.217	0.219*	13%	14%
		3	0.250	0.282*	0.277	13%	11%
		4	0.251	0.292*	0.290	16%	16%
		5	0.250	0.299	0.303*	19%	21%
		3	0.191	0.201	0.200	5%	5%
		4	0.192	0.211	0.210	10%	9%
		5	0.193	0.217	0.219*	12%	13%
		3	0.193	0.229	0.230	18%	19%
		4	0.192	0.244*	0.242	27%	26%
		5	0.192	0.252	0.264*	31%	37%
		3	0.250	0.284	0.286*	14%	14%
		4	0.250	0.297	0.297	19%	19%
		5	0.250	0.302*	0.297	21%	19%
		3	0.192	0.240	0.245*	25%	27%
		4	0.192	0.254*	0.251	32%	31%
		5	0.192	0.263*	0.252	37%	31%
		3	0.191	0.202	0.205*	6%	7%
		4	0.192	0.214*	0.214	12%	12%
		5	0.192	0.218*	0.215	14%	12%
		3	0.250	0.257	0.261*	3%	4%
		4	0.251	0.284	0.288*	13%	15%
		5	0.250	0.297	0.297	19%	19%
		3	0.191	0.191	0.192	0%	1%
		4	0.192	0.207	0.207	8%	8%
		5	0.193	0.215	0.215	12%	12%
		3	0.193	0.232*	0.216	20%	12%
		4	0.192	0.235	0.244*	22%	27%
		5	0.192	0.246	0.258*	28%	34%
		3	0.250	0.266	0.266	6%	6%
		4	0.250	0.294*	0.293	18%	17%
		5	0.250	0.300*	0.293	20%	17%
		3	0.192	0.224	0.227*	16%	18%
		4	0.192	0.248*	0.244	29%	27%
		5	0.192	0.257*	0.244	34%	27%
		3	0.191	0.193	0.194*	1%	2%
		4	0.192	0.212	0.211	10%	10%
		5	0.192	0.216*	0.212	13%	10%

* Optimal value, p<0.05

Bold numbers stand for second optimal value or shared optimality, p<0.05.

Table 11. Granularity of task formation 2-4-2

Formation 2-4-2								
	TH-HFB				TH-SRNS-II			
	DP0	DP1	DP2	DP4	DP0	DP1	DP2	DP4
1	0.351**	0.349	0.349*	0.347	0.351	0.348	0.354*	0.355**
2	0.236	0.236	0.240*	0.239	0.236	0.237	0.242*	0.243*
3	0.173	0.176	0.179*	0.179*	0.173	0.176	0.181*	0.181*
4	0.250	0.282**	0.279*	0.277	0.250	0.277	0.282**	0.279*
5	0.191	0.201	0.202*	0.200	0.191	0.200	0.202*	0.202*
6	0.193	0.229	0.229	0.228	0.193	0.230*	0.231*	0.228
7	0.250	0.284	0.298*	0.300**	0.250	0.286	0.300*	0.302**
8	0.192	0.240	0.260*	0.264**	0.192	0.245	0.261*	0.266**
9	0.191	0.202	0.211*	0.212**	0.191	0.205	0.213*	0.215**
1	0.354	0.357	0.357	0.357	0.354	0.357*	0.357*	0.356
2	0.241	0.247	0.249*	0.248*	0.241	0.247	0.248*	0.247
3	0.180	0.186	0.188*	0.188*	0.180	0.186	0.188	0.187
4	0.251	0.292	0.293*	0.293*	0.251	0.290**	0.288*	0.283
5	0.192	0.211	0.212	0.211	0.192	0.210*	0.209*	0.207
6	0.192	0.244	0.245	0.245	0.192	0.242**	0.239*	0.234
7	0.250	0.297	0.303*	0.303*	0.250	0.297	0.302*	0.303**
8	0.192	0.254	0.267*	0.270**	0.192	0.251	0.263*	0.268**
9	0.192	0.214	0.218*	0.219*	0.192	0.214	0.218*	0.219*
1	0.355	0.357	0.357	0.357	0.355	0.357	0.357	0.357
2	0.244	0.250	0.250	0.250	0.244	0.249	0.250	0.250
3	0.183	0.190	0.191	0.190	0.183	0.189	0.191*	0.191*
4	0.250	0.299	0.300*	0.300*	0.250	0.303	0.303	0.303
5	0.193	0.217	0.217	0.216	0.193	0.219	0.219	0.219
6	0.192	0.252	0.255*	0.254*	0.192	0.264	0.265	0.264
7	0.250	0.302	0.303*	0.303*	0.250	0.297	0.303*	0.304**
8	0.192	0.263	0.271*	0.272**	0.192	0.252	0.266*	0.270**
9	0.192	0.218	0.220*	0.220*	0.192	0.215	0.219*	0.220*

** Optimal value, $p < 0.05$

* Second optimal value or shared optimality, $p < 0.05$

Bold numbers stand for third optimal value or shared optimality, $p < 0.05$

Table 12. Granularity of task formation 1-6-1

Formation 1-6-1										
	TH-HFB					TH-SRNS-II				
	DP0	DP1	DP2	DP3	DP6	DP0	DP1	DP2	DP3	DP6
1	0.351***	0.344**	0.342*	0.335	0.334	0.351**	0.337	0.347*	0.352***	0.340
2	0.236**	0.228	0.232*	0.231	0.229	0.236*	0.227	0.238**	0.241***	0.233
3	0.173*	0.169	0.173*	0.173*	0.171	0.173	0.170	0.178*	0.180**	0.174
4	0.250	0.257	0.257	0.261**	0.259*	0.250	0.261	0.275**	0.275**	0.266*
5	0.191	0.191	0.191	0.192	0.189	0.191	0.192	0.200*	0.201**	0.194
6	0.193	0.232***	0.223**	0.219*	0.213	0.193	0.216	0.224*	0.223*	0.216
7	0.250	0.266	0.293**	0.293**	0.292*	0.250	0.266	0.293*	0.300***	0.295**
8	0.192	0.224	0.258*	0.263**	0.265***	0.192	0.227	0.255*	0.263**	0.263**
9	0.191	0.193	0.207*	0.207*	0.207*	0.191	0.194	0.209*	0.213**	0.208*
1	0.354	0.356	0.356	0.356	0.354	0.354	0.354	0.357*	0.357*	0.340
2	0.241	0.245*	0.245**	0.245**	0.244	0.241	0.244*	0.248**	0.248**	0.234
3	0.180	0.183	0.185*	0.185*	0.182	0.180	0.183*	0.187**	0.188***	0.178
4	0.251	0.284*	0.286**	0.284*	0.280	0.251	0.288*	0.294*	0.291*	0.273
5	0.192	0.207*	0.207*	0.206*	0.203	0.192	0.207*	0.212**	0.213**	0.200
6	0.192	0.235**	0.233*	0.234**	0.230	0.192	0.244*	0.248*	0.245*	0.227
7	0.250	0.294	0.303*	0.303*	0.303*	0.250	0.293	0.299**	0.301***	0.295*
8	0.192	0.248	0.268*	0.271**	0.272**	0.192	0.244	0.259*	0.265**	0.264**
9	0.192	0.212	0.217*	0.218*	0.217*	0.192	0.211*	0.216**	0.218***	0.209
1	0.355	0.357	0.357	0.357	0.357	0.355	0.357*	0.357*	0.357*	0.338
2	0.244	0.249	0.249	0.249	0.249	0.244	0.248*	0.249**	0.249**	0.234
3	0.183	0.189	0.190	0.190	0.188	0.183	0.188*	0.190**	0.190**	0.178
4	0.250	0.297**	0.295*	0.294*	0.292	0.250	0.297*	0.301**	0.302**	0.274
5	0.193	0.215**	0.214*	0.214*	0.211	0.193	0.215*	0.217*	0.220**	0.201
6	0.192	0.246*	0.247*	0.246*	0.241	0.192	0.258*	0.264**	0.268**	0.240
7	0.250	0.300	0.303*	0.304*	0.304*	0.250	0.293	0.299**	0.301***	0.295*
8	0.192	0.257	0.271*	0.273**	0.273**	0.192	0.244	0.259*	0.265**	0.264**
9	0.192	0.216	0.220*	0.220*	0.220*	0.192	0.212*	0.216**	0.218***	0.209

** Optimal value, p<0.05

* Second optimal value or shared optimality, p<0.05

Bold numbers stand for third optimal value or shared optimality, p<0.05