Jurnal
Teknologi

Full Paper

JOB SHOP SCHEDULING PROBLEMS WITH DYNAMIC BREAK TIME UNDER FATIGUE AND RECOVERY EFFECTS

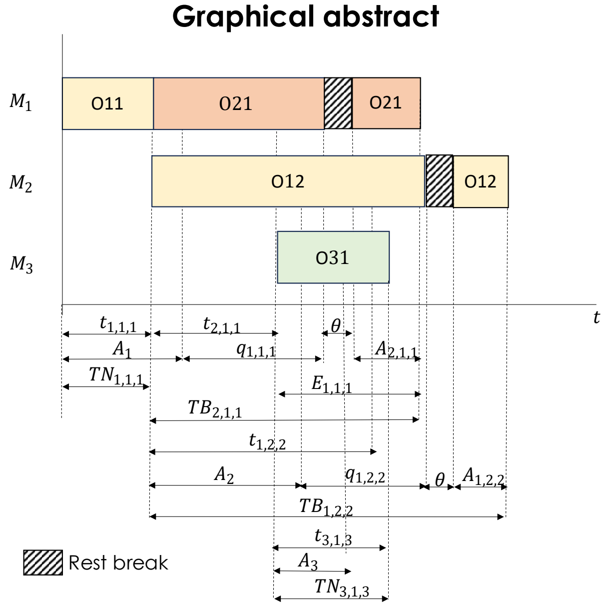
Vaniloran Elysa Andriani[a,b*], Achmad Pratama Rifai[a], Nur Mayke Eka Normasari[a], Nur Aini Masruroh[a], Wildanul Isnaini[a,c]

[a]Department of Mechanical and Industrial Engineering, Universitas Gadjah Mada, Post Box 55281, Yogyakarta, Indonesia
[b]Department of Industrial Engineering, Universitas Mahakarya Asia, Post Box 55282, Yogyakarta, Indonesia
[c]Department of Industrial Engineering, Universitas PGRI Madiun, Post Box 63117, Madiun, Indonesia

Article history
Received
30 December 2023
Received in revised form
6 February 2025
Accepted
13 June 2025
Published Online
27 February 2026

*Corresponding author
vaniloran@unmaha.ac.id

Graphical abstract



Abstract

This paper addresses the Job Shop Scheduling Problem (JSSP) with dynamic break times, focusing on the impact of worker fatigue and recovery on processing time. Traditional scheduling models assume deterministic processing times, but this research acknowledges the variability introduced by human factors, specifically fatigue, which can lead to musculoskeletal disorders, increased error frequency, safety issues, and decreased productivity. Rest breaks are identified as an effective strategy to mitigate fatigue, with various manufacturing environments demonstrating the benefits of incorporating rest breaks into scheduling processes. The paper's main contribution is the development of a Mixed Integer Linear Programming (MILP) model and an Ant Colony Optimization (ACO) algorithm to address the JSSP with dynamic break times. The results indicate a significant reduction in makespan when dynamic break times are incorporated. The average makespan reduction for problem sizes (jobs x machines) was 8.75% for 10 x 5, 13.28% for 15 x 5, 66.41% for 10 x 10, 69.18% for 15 x 10, and 74.27% for larger problems. In conclusion, this research suggests the need for advanced scheduling models that incorporate human factors, support rest breaks in work policies, and help decision-makers balance productivity with worker fatigue management.

Keywords: Job shop scheduling, human factors, work breaks, fatigue rate, recovery rate, dynamic break time

1.0 INTRODUCTION

Human labour is a vital resource in the production process, especially in the small manufacturing enterprise (SME) industry. One of the human factor constraints in the scheduling model is human fatigue during activities [1]. Fatigue is an objective condition or subjective feeling related to fatigue caused by external stimuli related to work [2]. Caldwell *et al*. (2019) describe fatigue as a complex biological phenomenon that occurs as a function of time. In mathematical modelling, some researchers describe workers' fatigue and recovery rates as exponential functions [4, 5].

Operators who experience fatigue will potentially cause musculoskeletal disorders [6–8] and increase the frequency of errors [9], resulting in increased work safety problems [10], a decrease in productivity [11], and reduced customer satisfaction [2]. In the case of traditional scheduling, the processing time is deterministic and calculable. However, when considering the level of worker fatigue, the processing time becomes uncertain. So that the completion time of a job changes and will affect the completion time of the entire order. Therefore, it is important to consider the human fatigue factor in the scheduling process to approach a more realistic production system situation.

According to some researchers, integrating rest breaks into scheduling has been identified as an effective way to reduce or minimise worker fatigue, as evidenced by research conducted in various manufacturing environments such as assembly lines [13–15], packing systems [4], order picking [16], single machine operations [17], flow shops [1, 18], and job shops [19]. In an assembly line system, Blasche *et al*. (2017) examined the effect of work rest on several workers who perfomed activities to determine their impact on reducing fatigue and stress level. Tiacci (2018) determines the rest time for the operator based on the Occupational Repetitive Action Index (OCRA) in the assembly line system to minimise ergonomic risks. Calzavara *et al*. (2019) developed an analytical model to determine rest allowance based on fatigue and recovery levels in the assembly line system and order picking.

Furthermore, Finco *et al*. (2020) investigate the influence of ergonomics on both productivity and safety in manual assembly lines, with a specific focus on mixed-model assembly lines. The authors propose a methodological framework that includes physical exhaustion and rest allowance as ergonomic factors. Subsequently, Azwir *et al*. (2020) conducted a study and discovered that the implementation of ranked positional weight (RPW) resulted in enhanced line efficiency and smoothness index within an industrial facility assembly line. Zhang *et al*. (2021) introduce a new task scheduling model for Human-Robot Collaboration Assembly Celss (HRCAC) that balances job cycle efficiency and human fatigue. The model also introduces a human fatigue recovery model and an exponential recovery model from rest breaks.

In the packing environment, Glock *et al*. (2019) propose a work break scheduling model to determine the optimal box size to minimize the total cost by considering worker fatigue and recovery level. Zhao *et al*. (2019) examine the case of an order-picking system where they develop a work-break scheduling model to minimise picking time and error picking time. Genetic algorithms are used to solve this multi-purpose problem. Subsequently, Li *et al*. (2020) developed a work break scheduling model to minimise makespan using a heuristic method. This research introduces a work break strategy in single-machine and single-worker cases and proposes placing break time in real-time to increase worker satisfaction and efficiency in production.

In flow shop scheduling, Du *et al*. (2021) developed a hybrid metaheuristic algorithm (HMA) for flexible flow shop scheduling that considers human fatigue. The model uses a genetic algorithm and novel heuristic decoding methods to assign workers to tasks based on fatigue levels. Liu *et al*. (2023) propose a simulation-based optimisation framework that minimises makespan and accommodates worker fatigue and skill levels in hybrid flow shops. Meanwhile, Hemono *et al*. (2023) discuss the importance of optimising rest breaks for human operators in FJSP. The study suggests strategies to reduce strenuousness during production planning involving cobots, emphasizing the need to complement human abilities without replacing human workers.

According to the given description, to the best of our knowledge, no research has been conducted on the job shop scheduling problem (JSSP) that considers the influence of fatigue and recovery on processing time, which may require workers to take breaks. Existing research has not considered the level of fatigue and recovery that will affect processing time in job shop scheduling. Therefore, the main contribution of this paper is to study JSSP by considering dynamic break times due to worker fatigue and recovery levels. We dynamically adjust break time by analyzing accumulated operator fatigue levels to meet each operator's recovery needs. Simultaneously, the objective function employed in this study is to reduce makespan.

The Makespan criterion in job shop scheduling measures the total time required to complete a series of jobs from start to finish. Therefore, it is important to consider fatigue and recovery rates when evaluating time constraints in JSSP. This includes integrating human factors, ensuring ergonomic standards, optimizing human-robot collaboration, increasing productivity and safety, complying with labour regulations, maintaining work quality, and reducing the risk of fatigue and chronic health problems associated with overwork [19]. By combining these factors, scheduling systems can prevent overwork, improve workplace safety, and ensure long-term workforce sustainability. Incorporating these factors into the makespan criteria in JSSP will ensure a balanced and sustainable work environment that prioritizes human well-being, complies with labour standards and ultimately leads to increased productivity and quality of work.

In this paper, we developed and solved this problem with a mixed integer linear programming (MILP) mathematical model using Gurobi. Since this is a complex combinatorial optimization problem, we designed the ant colony optimization (ACO) algorithm to generate near-optimal solutions for larger problems. We utilise ACO due to its ability to handle combinatorial optimization problems effectively [26]. Its flexibility and adaptability make it suitable for a variety of scheduling scenarios [27]. Additionally, ACO can handle complex constraints [28], making it essential in modern manufacturing environments. The ACO mechanism facilitates faster convergence to the optimal solution, making it ideal for JSSP scenarios. Next, we will conduct computational experiments to compare makespan with and without break time, evaluate the performance of the MILP and ACO approaches, and perform sensitivity analysis.

2.0 METHODOLOGY

2.1 Problem Description

Job Shop Scheduling Problem (JSSP) with dynamic break time is defined as follows: there are several sets of N jobs $J_i = \{J_1, J_{2,} \dots., J_N\}$ set of machines $K_k = \{M_1, M_2, \dots, M_M\}$. Each job i has an p operation $\{O_{i1}, O_{i2}, \dots, O_{ip}\}$. Each O_{ij} must run on a specific M machine. Workers (W) must operate each M machine. In this case, the number of machines equals the number of workers on the machine. For example, on M_1 operated by W_1, as well as on M_2 operated by W_2, and so on. Each worker will experience fatigue and recovery during the production period. Worker fatigue will accumulate continuously during work periods and decrease during break periods. The level of worker fatigue is in the range of $0 < \lambda_k < 1$, and the recovery rate of workers is in the range $of\ 0 < \mu_k < 1$ [29]. Model parameters and decision variables are shown in Table 1.

The assumptions considered in this model are as follows: All materials are available, all jobs have the same priority, all jobs and machines are available at the start of planning, pre-emption cancellation is not permitted, each machine can only process one at a time, each worker can only operate one machine at a time, the machine set-up time and transport time are ignored, a breakdown machine is not allowed and the fatigue rate and recovery rate increase exponentially.

2.2 JSSP Mathematical Model with Dynamic Break Time

The developed model is a modification of Li *et al.* (2020) [17]. In contrast to scheduling on a single machine, the JSP case has unique characteristics: scheduling considers machine routing and job sequences. The following are the equations used in the model. An illustration of the developed model is shown in Figure 1.

Table 1 Model notation

No.	Notation	Description
1	i	Index of job number $i = 1, 2, 3, \dots N$
2	h	Index of job numbers preceded by job i
3	j	Index of the operation number of a job, $j = 1, 2, 3, \dots p_i$
4	k	Index of machine number, $k = 1, 2, \dots, M$
5	$O_{i,j}$	Operation j on job i
6	J	Set of jobs
7	K	Set of machines
8	W	Set of workers
Parameter		
9	N	Number of jobs
10	M	Number of machines/number of workers
11	θ	Length of break time
12	p_i	The number of operations of the job i
13	A_k	Duration of first working period on job i operation j on machine k
14	$t_{i,j,k}$	Processing time O_{ij} on the machine k
15	$t_{h,j,k}$	Processing time O_{hj} on the machine k
16	λ_k	The level of worker fatigue on the machine k $(0 < \lambda_k < 1)$
17	μ_k	The worker recovery rate on the machine k $(0 < \mu_k < 1)$
18	L	Large number
Dependent Variables		
19	$S_{i,j,k}$	Start time of operation O_{ij} on machine k
20	$S_{h,j,k}$	Start time of operation O_{hj} on machine k
21	C_i C_{ij}	Completion time job i Completion time job i operation j
22	C_{max}	Maximum completion time for all jobs (makespan)
23	$q_{i,k}$	The length of time during which workers experience a decline in work levels due to fatigue levels on the job i on machine k
24	$NT_{i,j,k}$	Maximum completion time of job i operation j on machine k
25	$NT_{h,j,k}$	Maximum completion time of job h operation j on machine k
26	$A^*_{i,j,k}$	Duration of the second working period on job i operation j on machine k
27	$A^*_{h,j,k}$	Duration of the second working period on job h operation j on machine k
28	$E_{i,j,k}$	The difference between the maximum completion time and the total completion time for job i operation j on machine k
29	$E_{h,j,k}$	The difference between the maximum completion time and the total completion time for job h operation j on machine k

No.	Notation	Description
30	$TN_{i,j,k}$	The maximum completion time of the job i operation j without a break time on machine k
31	$TN_{h,j,k}$	The maximum completion time of the job h operation j without a break time on machine k
32	$TB_{i,j,k}$	The maximum completion time of job i operation j with a break time on machine k
33	$TB_{h,j,k}$	The maximum completion time of job h operation j with a break time on machine k
Decision variable		
34	$Y_{i,h,k}$	For binary variables, the value is one if job i precedes h the job on the machine k, otherwise 0

The duration of the decrease in the work rate of workers on the machine k is adopted from Li *et al.* (2020) [17]:

$$q_{i,k} = \left(-\frac{1}{\lambda_k}\right) \times \ln(1 - \mu_k \,.\, \theta) \qquad k \in M \tag{1}$$

The length of time workers work in the second period:

$$A^*_{i,j,k} = \frac{t_{i,j,,k} - A_k + \left(\frac{1}{\lambda_k}\right)\left(e^{-\lambda_k \cdot q_{i,k}} - 1\right)}{e^{-\lambda_k \cdot q_{i,k}} + \mu_k \,.\theta} \qquad \forall i, h \in J\,; \quad j \in p_i; k \in M \tag{2}$$

$$A^*_{h,j,k} = \frac{t_{i,j,k} + t_{h,j,k} - A_k + \left(\frac{1}{\lambda_k}\right)\left(e^{-\lambda_k \cdot q_{i,k}} - 1\right)}{e^{-\lambda_k \cdot q_{i,k}} + \mu_k \,.\theta} \qquad \forall i, h \in J\,; \quad k \in M; h \neq i \tag{3}$$

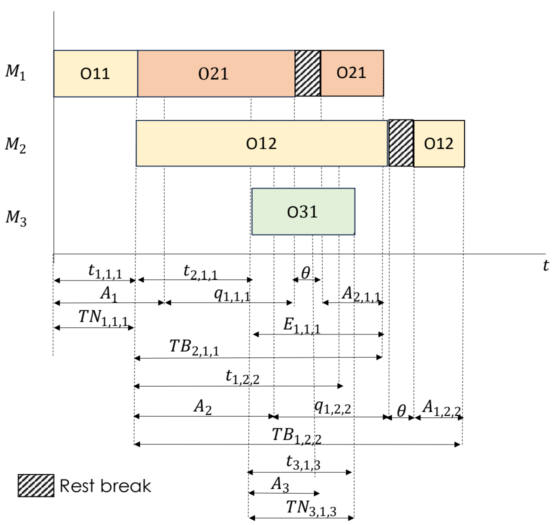


Figure 1 The illustration of the development model

Based on the notation defined in Table 1, this problem's mathematical model can be described as follows:

Objective function:

$$\text{Minimize} \qquad \max(C_{ij}) \quad \forall i \in J, j \in p_i \tag{4}$$

Constraint:

$$s_{i,j,k} + t_{i,j,k} \leq s_{h,j,k} + L \times (1 - Y_{i,h,k}) \ \forall i, h \in J\,; j \in p_i \quad k \in M; h \neq i \tag{5}$$

$$s_{h,j,k} + t_{h,j,k} \leq s_{h,j,k} + L \times (Y_{i,h,k}) \ \forall i, h \in J\,; j \in p_i \quad k \in M; h \neq i \tag{6}$$

$$\text{if} A_k > t_{i,j,k} \tag{7}$$

$$NT_{i,j,k} = t_{i,j,k} \tag{8}$$

$$E_{i,j,k} = 0 \tag{9}$$

$$TN_{h,j,k} = (A_k - \left(\frac{1}{\lambda_k}\right) .\ln\left(1 - (\lambda_k \times (t_{i,j,k} + t_{h,j,k}) + (\lambda_k \times A_k)\right)) \tag{10}$$

$$TB_{h,j,k} = A_k + q_{i,k} + \theta \quad + A^*_{h,j,k} \tag{11}$$

$$NT_{h,j,k} = \min\,(TN_{h,j,k}, TB_{h,j,k}) \tag{12}$$

$$E_{h,j,k} = NT_{h,j,k} - t_{h,j,k} - t_{i,j,k} \tag{13}$$

$$\text{If } t_{i,j,k} > A_k \tag{14}$$

$$TN_{i,j,k} = (A_k - \left(\frac{1}{\lambda_k}\right) .\ln\left(1 - (\lambda_k \times (t_{i,j,k}) + (\lambda_k \times A_k)\right)) \tag{15}$$

$$TB_{i,j,k} = A_k + q_{i,k} + \theta \quad + A^*_{i,j,k} \tag{16}$$

$$NT_{i,j,k} = \min\,(TN_{i,j,k}, TB_{i,j,k}) \tag{17}$$

$$E_{i,j,k} = NT_{i,j,k} - t_{i,j,k} \tag{18}$$

$$S_{i,j,k} \geq S_{i,(j-1),k} + t_{i,(j-1),k} + E_{i(j-1),k} \quad \forall i \in J\,; j = 1, \ldots, p_{i-1}; k \in M \tag{19}$$

$$S_{i,j,k} \geq S_{h,j,k} + t_{h,j,k} + E_{h,j,k} - L.Y_{i,h,k} \quad \forall i, h \in J\,; h \neq i; k \in M \tag{20}$$

$$S_{h,j,k} \geq S_{i,j,k} + t_{i,j,k} + E_{i,j,k} - L.(1 - Y_{i,h,k}) \quad \forall i, h \in J\,; h \neq i; k \in M \tag{21}$$

$$C_i \geq S_{i,j,k} + t_{i,j,k} + E_{i,j,k} \ \forall i \in J\,; j = 1, \ldots, p_i; k \in M \tag{22}$$

$$Y_{i,h,k} \ \in \{0,1\} \tag{23}$$

Where: Equation (4) shows the makespan objective function. Constraints (5) and (6) to ensure the processing time of successive operations on the machine. Equation (7) indicates that the duration of the first working period must be longer than the processing time. Equation (8) to determine the maximum job i completion time on the machine k. Equations (9), (13) and (18) show excess processing time due to the effect of accumulated worker fatigue. Equations (10) and (15) show the maximum job i completion time without a break time to machine k. Equations (11) and (16) lead to the maximum job i completion time with a break time to machine k. Equations (12) and (17) show the selection of the maximum completion time for work. Equations (14) show that the operating time is greater than the A_k Duration. Constraints (19) to ensure predecessor operation rules are not violated on the same job. Constraints (20) and (21) ensure that each machine cannot perform multiple operations simultaneously. Constraints (22) to ensure that makespan is the latest completion time of the entire job. Constraint (23) defines the binary variable.

2.2 Ant Colony Optimization for JSSP with Dynamic Break Time

The ant colony optimization (ACO) algorithm adopts the ant strategy by utilizing pheromone trails as communication and feedback. ACO generates a good initial solution quickly, and a pheromone trail is initialized based on this solution [30]. Next, the artificial ants iteratively construct the solution by selecting jobs based on transition rules that depend on pheromone traces. Solution quality is improved through local search procedures and pheromone trails. This algorithm has proven effective in solving complex optimization problems and can produce good solutions [31].

Algorithm 1 *ACO algorithm procedure*

1. *Initialize ACO parameters (num_ants, num_iterations, α, β, ρ, q)*
2. *Initialize pheromone matrix with initial_pheromone value*
3. *Initialize best schedule*
4. *Initialize best makespan*
5. *For each iteration from 1 to num_iterations, do:*
   - 5.1. *Initialize schedule of ants randomly for each ants*
   - 5.2. *Calculate heuritic information*
     - 5.2.1. *For each operation, do:*
       - 5.2.1.1. *Calculate next operation probability (pheromone, β)*
       - 5.2.1.2. *Next operation=select next operation (probability)*
     - 5.2.2. *Makespan of ants=calculate makespan (schedule of ants)*
     - 5.2.3. *If makespan of ants < best makespan, do:*
       - 5.2.3.1. *Best schedule=copy of(schedule of ants)*
       - 5.2.3.2. *Best makespan=makespan of ants*
   - 5.3. *Update the pheromone, best schedule, best makespan*
6. *Return best schedule, best makespan*

The Function calculates heuristic information

1. *Get schedule of ants*
2. *Heuristic information for each schedule of ants = 1/job times*
3. *Return heuristic information*

The Function calculates the probability for the next operation:

1. *Get the pheromone and heuristic information*
2. *For each operation, do:*
   - 2.1. *Calculate probability of each operation =* $pheromone(operation)^{\alpha} \times heuritic\ information(operation)^{\beta}$
   - 2.2. *For each p in probability , do:*
     - 2.2.1. *Probability = p/ total all the probability of each operation*
   - 2.3. *Return probability*

The Function select next operation:

1. *Select the next operation based on probability distribution given by probability*
2. *Return selected operation*

The Function update_pheromone:

1. *Get pheromone, best schedule, best makespan*
2. *For each job, do:*
   - 2.1. *Pheromone* $= (1-\rho) \times pheromone\ from\ best\ shcedule + q/$*best makespan*

2.3 Computational Experiments

This experiment's benchmark is based on Lawrence (1984) and contains the same number of tasks, operations, and machines as the benchmark. There are two scenarios with problem sizes of 10 x 5 and 10 x 10 for small-sized problems. Meanwhile, there are two scenarios with dimensions of 15 x 5 and 15 x 10 for medium-sized difficulties. Furthermore, for large-sized use the benchmark from Taillard (1983). In addition, problem parameters are generated using the following algorithm. Fatigue rate (λ_k) and recovery rate (μ_k) are generated according to a uniform distribution (0.001, 0.003) and uniform distribution (0.005, 0.01) [32] and shown in Table 2. The break time (θ) duration is 5 minutes. A_k is generated uniformly (100, 120). While the ACO parameters include: pheromone factor (α) of 2, heuristic factor (β) of 2, and an evaporate rate (ρ) of 0.25. The quantity of ants is proportional to the amount of labour.

Table 2 Parameters used

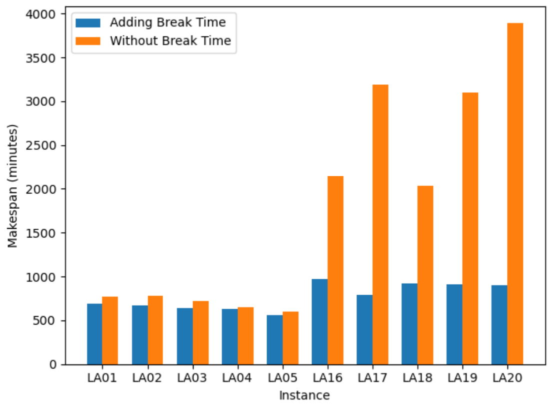
	Parameter		
	λ_k	μ_k	A_k
M1	1.8e-3	8.8e-3	120
M2	1.6e-3	7.7e-3	100
M3	2.1e-3	9.3e-3	130
M4	1.3e-3	5.5e-3	120
M5	1.5e-3	9.5e-3	110
M6	1.1e-3	9.5e-3	112
M7	1.7e-3	7.8e-3	125
M8	1.4e-3	6.4e-3	115
M9	1.8e-3	5.5e-3	112
M10	2.1e-3	8.5e-3	125

3.0 RESULTS AND DISCUSSION

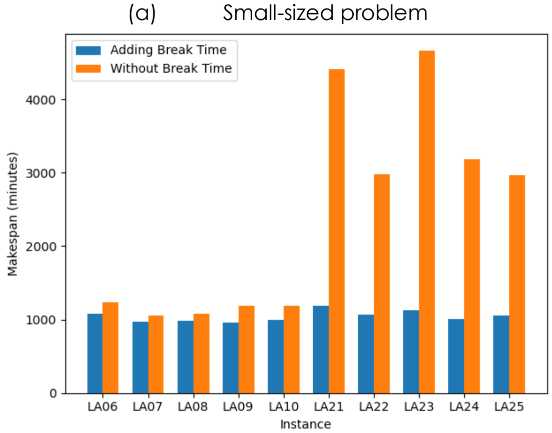
3.1 Comparative Study of the Optimal Solutions: Adding Break Time vs. Without Break Time

A comparative analysis of the impact of incorporating break time into the optimization process for various example problems is shown in Figure 2. The makespan, which refers to the overall duration needed to finish jobs, is being compared among different examples denoted as LA01 to LA25, TA21, and TA41. Small-sized problem instances exhibit a negligible disparity in makespan between scenarios with and without break time, indicating that break time has an insignificant impact on the overall time needed to finish activities. Nevertheless, there is a consistent pattern between the inclusion of break time and a decrease in the makespan, suggesting an enhancement in the optimization process.

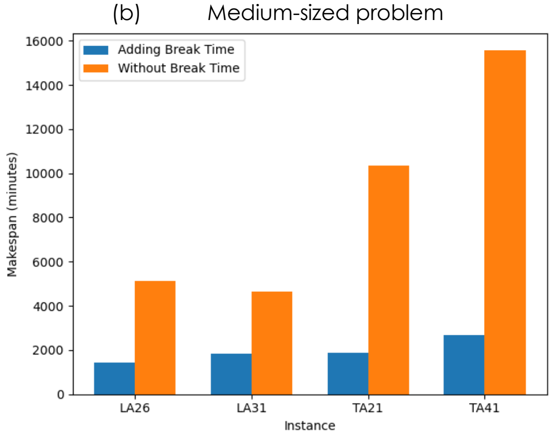
As problem instances increase in size and complexity, the effect of break time becomes more noticeable. The graphs illustrate a substantial difference in makespan, with the time required being significantly higher when break time is not included. Specifically, the average makespan reduction for problem sizes (jobs x machines) of 10x5 is 8.75%, 15x5 is 13.28%, 10x10 is 66.41%, 15x10 is 69.18%, and for the larger problems, it reaches up to 74.27%. This highlights the importance of considering break time in the optimization process for medium and large-size problems, as it can lead to significant improvement in the overall efficiency and realism of the solution.



(a) Small-sized problem



(b) Medium-sized problem



(c) Larger-sized problem

Figure 2 Comparative of makespan obtained with and without break time

3.2 Evaluation of the Effectiveness of the MILP Method and the ACO Algorithm

This study presents the outcomes of solving scheduling problems using two optimization methods: mixed-integer linear programming (MILP) and ant colony optimization (ACO). The specific result may be found in Tables 3 through 5 and Figure 3. The table provides the makespan results, which include the break time. Although MILP often yields a lower makespan value, indicating a more efficient solution, ACO demonstrates competitiveness in other aspects.

Table 3 Computational results on the small-sized problem

Problem	Size (n,m)	MILP			ACO		
		CP	C_{max}	OP	CP	C_{max}	Gap
LA01	10,5	16.1	694.8	0	44.9	713.2	18.4
LA02	10,5	1.9	670.1	0	63.7	731.1	61,0
LA03	10,5	9.7	637.6	0	67.1	727.6	90.0
LA04	10,5	4.2	629.1	0	50.2	632.8	3.7
LA05	10,5	23.1	562.6	0	66.5	756.9	194.3
LA16	10,10	5.9	968.6	0	112.9	969.7	1.1
LA17	10,10	5.8	793.8	0	153.6	815.4	21.6
LA18	10,10	12.6	920.7	0	115.3	922.3	1.6
LA19	10,10	9.3	908.4	0	131.6	921.2	12.8
LA20	10,10	4.1	901.1	0	137.1	948.2	47.1

CP: CPU duration (seconds); OP: optimality gaps (%)

Table 4 Computational results on the medium-sized problem

Problem	Size (n,m)	MILP		ACO		Gap
		C_{max}	OP	CP	C_{max}	
LA06	15,5	1073.6	26.9	133.3	1198.5	124.9
LA07	15,5	966.4	15.3	154.3	1099.9	133.5
LA08	15,5	978.3	27.7	155.1	981.1	2.8
LA09	15,5	956.2	4.85	129.1	1241.9	285.7
LA10	15,5	992.7	3.04	128.9	1362.5	369.8
LA21	15,10	1185.9	21.1	331.1	1461.0	275.1
LA22	15,10	1065.3	10.6	384.4	1282.8	217.5
LA23	15,10	1122.1	7.14	273.1	1413.8	291.7
LA24	15,10	1007.4	2.89	254.5	1420.7	413.3
LA25	15,10	1059.9	0	323.7	1390.3	330.4

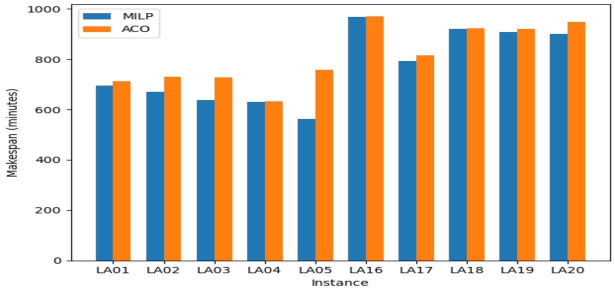
CPU duration (seconds) MILP=3600; OP: optimality gaps (%)

Table 5 Computational results on the large-sized problem

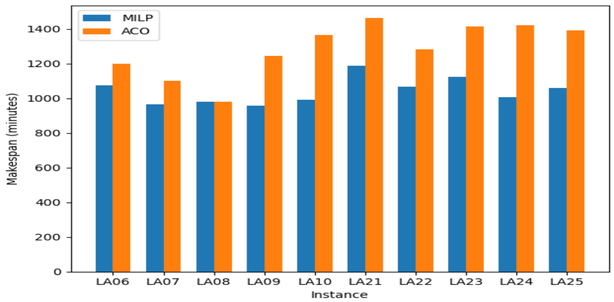
Problem	Size (n,m)	MILP		ACO		Gap
		C_{max}	OP	CP	C_{max}	
LA26	20,10	1448,9	9.9	460.5	2117.6	668.7
LA31	30,10	1827	18.3	841.2	3239.9	1412.9
TA21	20,20	1880.8	0	459.6	2226.4	345.6
TA41	30,20	2672.7	14.4	465.1	2327.7	-345
TA61	50,20	-	-	529.2	5926.1	-5926.1
TA71	100,20	-	-	1952.3	12147.8	-1952.3

CPU duration (seconds) MILP=3600; OP: optimality gaps (%)

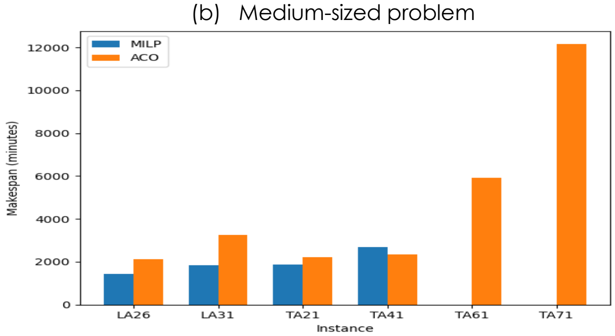
According to the computational results shown in Tables 3 to 5 and Figure 3, the optimality gap of the MILP issue increases as the problem size grows, suggesting that it becomes more difficult to discover the optimal solution within a certain time restriction. The Ant Colony Optimization (ACO) algorithm has superior scalability in terms of CPU duration. However, this advantage comes at the cost of higher makespan values. To summarize, MILP is more effective at finding optimal solutions for small problems. Nevertheless, when the problem size increases, MILP struggles to discover an optimal solution within a reasonable timeframe. While ACO may not be the most ideal solution, it provides better scalability when considering the size of the problem. Moreover, based on the experimental results, ACO discovered an improved solution within an acceptable timeframe for a large-sized problem with dimensions of 30x20, 50x20, and 100x20. This finding is depicted in Figure 3c.



(a) Small-sized problem



(b) Medium-sized problem



(c) Large-sized problem

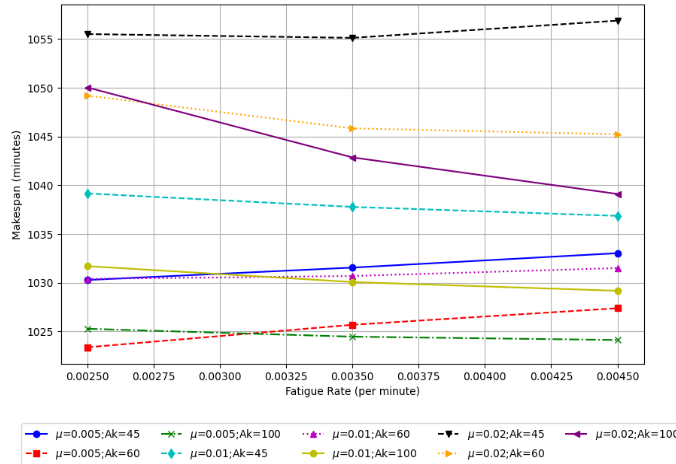
Figure 3 Comparison of makespan results

3.3 Sensitivity Analysis

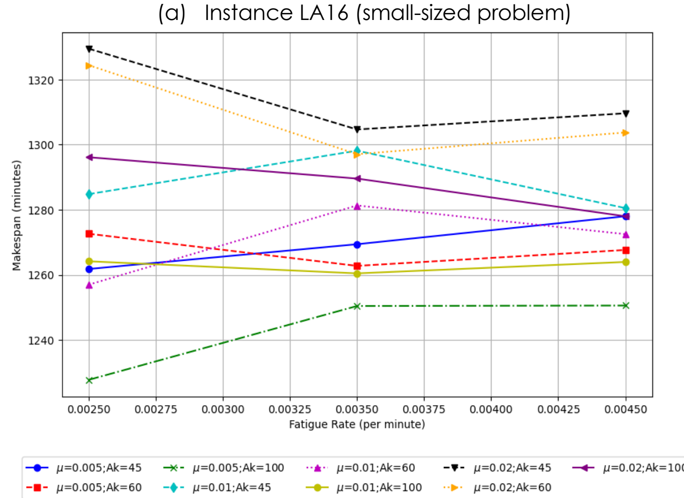
In this section, a sensitivity analysis is conducted to investigate the effect of an important parameter, namely the fatigue rate (λ), recovery rate (μ), and the duration of the first working period (A_k) on makespan with break time. We perform a 3^3 factorial design to identify the significant parameters. The values of input parameters are shown in Table 6. We have selected three levels of fatigue and recovery (low, medium, and high)[32]. To evaluate the influence of each parameter on the MILP method, problems LA16, LA21 and TA41 are selected for sensitivity analysis. Figures 4, 5, and 6 display the outcomes of the sensitivity analysis.

Table 6 Input data

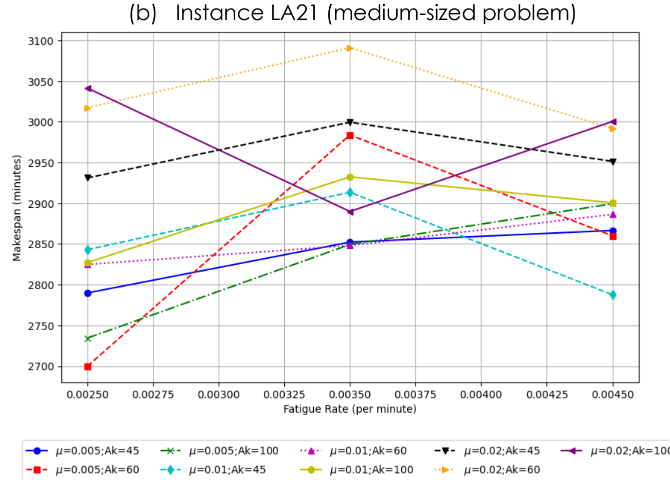
	Low (1)	Medium (2)	High (3)
λ (per minute)	2.5e-3	3.5e-3	4.5e-3
μ (per minute)	0.005	0.01	0.02
A_k (minute)	45	60	100



(a) Instance LA16 (small-sized problem)



(b) Instance LA21 (medium-sized problem)

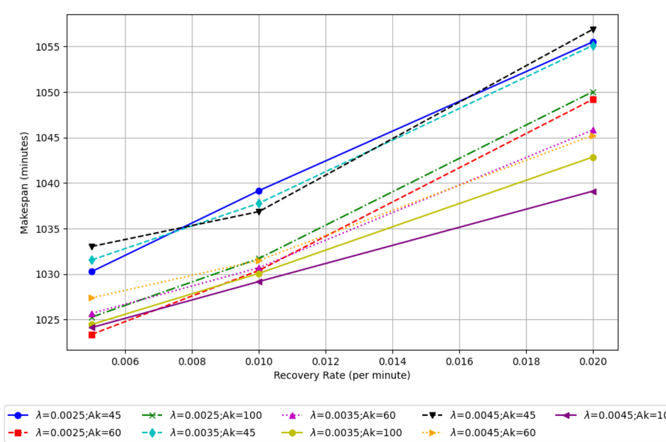


(c) Instance TA41 (large-sized problem)

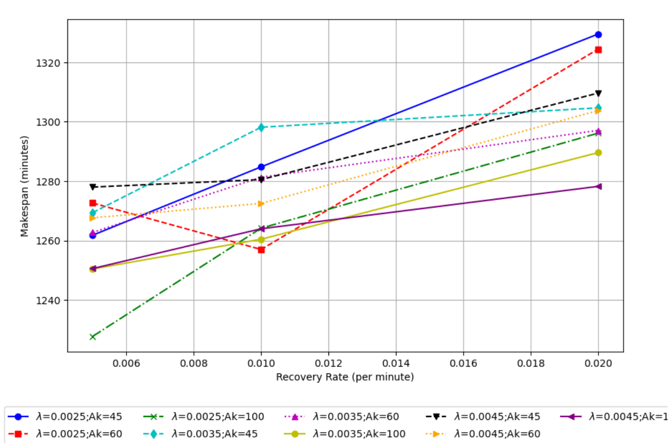
Figure 4 Sensitivity analysis for parameter fatigue rate (λ)

Figure 4 examines the relationship between makespan and fatigue rate for issues of varying sizes, including small, medium, and large-sized problems. The graphs illustrate the relationship between the fatigue rate and makespan. Each line on the graphs represents a distinct combination of the recovery rate and the duration of the first working period. The makespan for small-sized problems varies between 1025 and 1055 minutes, but for medium-sized problems, it ranges from 1240 to 1320 minutes. The effect of different combinations of μ and A_k on the makespan is more noticeable, with certain lines exhibiting a more rapid increase as the fatigue rate increases. For large-sized problems, the makespan ranges widely from 2700 to 3100 minutes, and the lines are widely spaced, indicating a high sensitivity of the makespan to changes in the fatigue rate. The makespan's response to varying fatigue rates is significantly affected by the different combinations of μ and A_k. These analyses provide valuable insights for decision-makers in optimizing schedules and workloads to manage the effects of fatigue on overall productivity, especially in contexts where the makespan is a critical performance metric.

Furthermore, Figure 5 shows the relationship between recovery rate and makespan with break time for various combinations of parameters λ and A_k. The data demonstrates that as the rate of recovery improves, the makespan typically lowers, indicating that a faster recovery rate can result in a shorter total time for completing tasks. A higher fatigue rate leads to a higher makespan, as workers become fatigued more quickly, necessitating longer recovery times and thus extending the overall makespan.

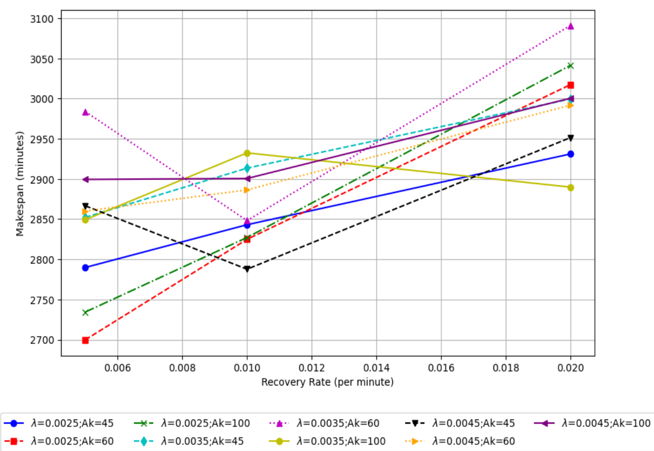


(a) Problem LA16



(b) Problem LA21

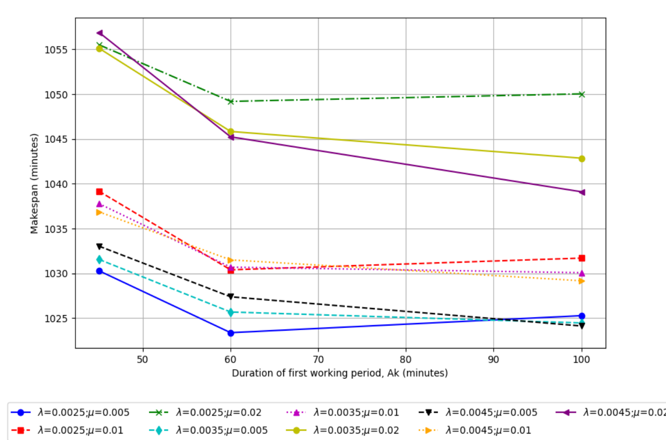
Figure 5 Sensitivity analysis for parameter recovery rate (μ)



(c) Problem TA41

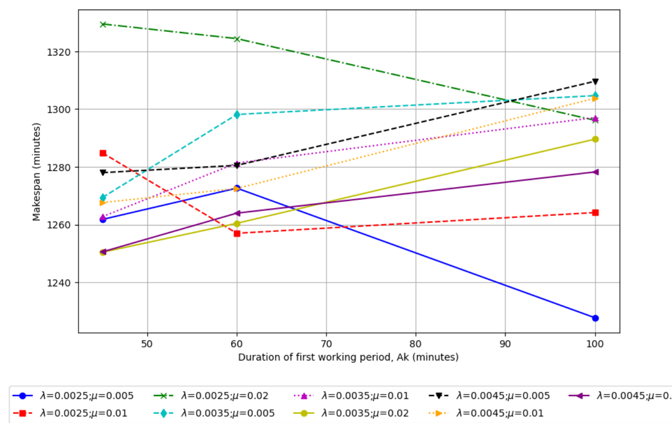
Figure 5 Sensitivity analysis for parameter recovery rate (μ)

The study examines the relationship between makespan and the duration of the first working period (A_k), as the period depicted in Figure 6. The results show that longer A_k before taking a break can reduce the total time required to complete the job. The decrease in makespan depends on the specific fatigue rate and recovery rate, where higher values will result in higher makespan. The sensitivity of time limits to changes in the first term of employment is influenced by a combination of these factors. The study also identifies the optimal working period that minimizes the time limit for a given set of λ and μ values. A comparative investigation indicates that makespan is highly responsive to variations in these characteristics. This analysis can assist in determining the most efficient work rest plan to minimise the overall time required to complete jobs while considering the influence of fatigue and recovery on worker productivity.

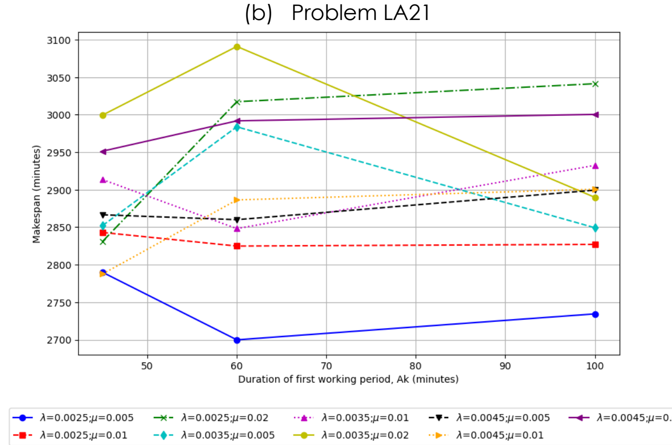


(a) Problem LA16

Figure 6 Sensitivity analysis for parameter duration of first working period (A_k)



(b) Problem LA21



(c) Problem TA41

Figure 6 Sensitivity analysis for parameter duration of first working period (A_k) *(Continued)*

4.0 CONCLUSION

This research discusses the issue of the job shop scheduling problem (JSSP) with dynamic break time, considering the effects of worker fatigue and recovery. The research integrates human factors, such as fatigue and recovery rate, into the JSSP, providing a more realistic representation of the production environment. Rest breaks are confirmed as an effective strategy to mitigate fatigue effects, leading to a reduction in makespan. The study employs mixed integer linear programming (MILP) and ant colony optimization (ACO) to solve the JSSP. Computational investigations demonstrate that the influence of break time on makespan is particularly noticeable in medium and large-sized problems. This suggests that taking the break periods into account can enhance efficiency and realism in complex and larger scheduling challenges. The sensitivity analysis highlights the significant influence of fatigue rate, recovery rate, and the duration of the first working period on makespan, providing insights into how these parameters interact and affect overall scheduling performance.

This study has implications for both theoretical and practical aspects. It suggests the need for advanced scheduling models that incorporate human factors, promoting more effective and humane practices. This research also supports rest breaks in work policies and workplace design, enhancing productivity while ensuring worker health and safety. The MILP and ACO methods are valuable methods for decision-makers in manufacturing, evaluating scheduling scenarios and developing strategies to balance productivity with worker fatigue management. In addition, future research could explore other human factors affecting scheduling, machine learning techniques for fatigue prediction, and real-time scheduling systems. The findings have direct applications in industries where manual labour is prevalent, enabling more resilient and adaptive scheduling systems.

Acknowledgement

The authors would like to acknowledge the Department of Mechanical and Industrial Engineering, Gadjah Mada University for their support.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] Du, H., Qiao, F., Wang, J., & Lu, H. 2021. A Hybrid Metaheuristic Algorithm with Novel Decoding Methods for Flexible Flow Shop Scheduling Considering Human Fatigue*. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*. 2328–2333. https://doi.org/10.1109/SMC52423.2021.9658692.

[2] Xu, S., & Hall, N. G. 2021. Fatigue, Personnel Scheduling and Operations: Review and Research Opportunities. *European Journal of Operational Research*. 295(3): 807–822. https://doi.org/10.1016/j.ejor.2021.03.036.

[3] Caldwell, J. A., Caldwell, J. L., Thompson, L. A., & Lieberman, H. R. 2019. Fatigue and Its Management in the Workplace. *Neuroscience and Biobehavioral Reviews*. 96(October 2018): 272–289. https://doi.org/10.1016/j.neubiorev.2018.10.024.

[4] Glock, C. H., Grosse, E. H., Kim, T., Neumann, W. P., & Sobhani, A. 2019. An Integrated Cost and Worker Fatigue Evaluation Model of a Packaging Process. *International Journal of Production Economics*. 207(September 2018): 107–124. https://doi.org/10.1016/j.ijpe.2018.09.022.

[5] Tan, W., Yuan, X., Wang, J., & Zhang, X. 2021. A Fatigue-Conscious Dual Resource Constrained Flexible Job Shop Scheduling Problem by Enhanced NSGA-II: An application From Casting Workshop. *Computers and Industrial Engineering*. 160(May): 107557. https://doi.org/10.1016/j.cie.2021.107557.

[6] Panahi, A. K., Cho, S., & Gordon, C. 2021. Muscle Fatigue Analysis During Welding Tasks Using sEMG and Recurrence Quantification Analysis. *International Journal of Applied Industrial Engineering*. 8(1): 1–16. https://doi.org/10.4018/ijaie.287609.

[7] Susihono, W., & Adiatmika, I. P. G. 2021. The Effects of Ergonomic Intervention on the Musculoskeletal Complaints and Fatigue Experienced by Workers in the Traditional Metal Casting Industry. *Heliyon*. 7(2): e06171.

https://doi.org/10.1016/j.heliyon.2021.e06171.

[8] Salleh, K. F., Fadzil, S. M., & Daud, M. Y. M. 2022. Cmdq, a Tool for Pain Sensation Solution for Ergonomic Postural Assessment During Practical Laboratory Work. *Jurnal Teknologi*. 84(6–2): 105–111. https://doi.org/10.11113/jurnalteknologi.v84.19357.

[9] Pang, L., Li, P., Guo, L., Wang, X., & Qu, H. 2022. Prediction Method of Human Fatigue in an Artificial Atmospheric Environment Based on Dynamic Bayesian Network. *Mathematics*. 10(15). https://doi.org/10.3390/math10152778.

[10] Okumus, D., Fariya, S., Tamer, S., Gunbeyaz, S. A., Yildiz, G., Kurt, R. E., & Barlas, B. 2023. The Impact of Fatigue on Shipyard Welding Workers' Occupational Health and Safety and Performance. *Ocean Engineering*. 285(P1): 115296. https://doi.org/10.1016/j.oceaneng.2023.115296.

[11] Darwish, M. A. 2023. Optimal Workday Length Considering Worker Fatigue and Employer Profit. *Computers and Industrial Engineering*. 179(March): 109162. https://doi.org/10.1016/j.cie.2023.109162.

[12] Sarker, P., Norasi, H., Koenig, J., Hallbeck, M. S., & Mirka, G. 2021. Effects of Break Scheduling Strategies on subjective and Objective Measures of Neck and Shoulder Muscle Fatigue in Asymptomatic Adults Performing a Standing Task Requiring Static Neck Flexion. *Applied Ergonomics*. 92(November 2020): 103311. https://doi.org/10.1016/j.apergo.2020.103311.

[13] Tiacci, L. 2018. The Problem of Assigning Rest Times to Reduce Physical Ergonomic Risk at Assembly Lines. *IFAC-PapersOnLine*. 51(11): 692–697. https://doi.org/10.1016/j.ifacol.2018.08.399

[14] Finco, S., Calzavara, M., Sgarbossa, F., & Zennaro, I. 2021. Including Rest Allowance in Mixed-model Assembly Lines. *International Journal of Production Research*. 59(24): 7468–7490. https://doi.org/10.1080/00207543.2020.1843731.

[15] Blasche, G., Pasalic, S., Bauböck, V. M., Haluza, D., & Schoberberger, R. 2017. Effects of Rest-break Intention on Rest-break Frequency and Work-related Fatigue. *Human Factors*. 59(2): 289–298. https://doi.org/10.1177/0018720816671605.

[16] Zhao, X., Liu, N., Zhao, S., Wu, J., Zhang, K., & Zhang, R. 2019. Research on the Work-rest Scheduling in the Manual Order Picking Systems to Consider Human Factors. *Journal of Systems Science and Systems Engineering*. 28(3): 344–355. https://doi.org/10.1007/s11518-019-5407-y.

[17] Li, K., Xu, S., & Fu, H. 2020. Work-break Scheduling with Real-time Fatigue Effect and Recovery. *International Journal of Production Research*. 58(3): 689–702. https://doi.org/10.1080/00207543.2019.1598600

[18] Liu, Y., Shen, W., Zhang, C., & Sun, X. 2023. Agent-based Simulation and Optimization of Hybrid Flow Shop Considering Multi-skilled Workers and Fatigue Factors. *Robotics and Computer-Integrated Manufacturing*. *80*(October 2022): 102478. https://doi.org/10.1016/j.rcim.2022.102478.

[19] Hemono, P., Sahnoun, M., & Chabane, A. N. 2023. Optimizing Resource Allocation in the Flexible Job Shop Problem: Assessing the Impact of Rest Breaks on Task Strenuousness Reduction. *2023 International Conference on Decision Aid Sciences and Applications, DASA 2023*. 320–325. https://doi.org/10.1109/DASA59624.2023.10286783.

[20] Di Pasquale, V., Fruggiero, F., Iannone, R., & Miranda, S. 2017. A Model for Break Scheduling Assessment in Manufacturing Systems. *Computers and Industrial Engineering*. 11*:* 563–580. https://doi.org/10.1016/j.cie.2017.05.017.

[21] Calzavara, M., Persona, A., Sgarbossa, F., & Visentin, V. 2019. A Model for Rest Allowance Estimation to Improve Tasks Assignment to Operators. *International Journal of Production Research*. 57(3): 948–962. https://doi.org/10.1080/00207543.2018.1497816.

[22] Finco, S., Battini, D., Delorme, X., Persona, A., & Sgarbossa, F. 2020. Workers' Rest Allowance and Smoothing of the Workload in Assembly Lines. *International Journal of Production Research*. 58(4): 1255–1270. https://doi.org/10.1080/00207543.2019.1616847.

[23] Azwir, H. H., & Fadjriawan Nugraha, A. 2020. Redesigning Assembly Line By Applying Ranked Positional Weight At Heavy-Industrial Facility. *Spektrum Industri*. 18(2): 133. https://doi.org/10.12928/si.v18i2.17801.

[24] Zhang, M., Li, C., Shang, Y., Huang, H., Zhu, W., & Liu, Y. 2022. A Task Scheduling Model Integrating Micro-breaks for Optimisation of Job-cycle Time in Human-robot Collaborative Assembly Cells. *International Journal of Production Research*. 60(15): 4766–4777. https://doi.org/10.1080/00207543.2021.1937746.

[25] Liu, Y., Shen, W., Zhang, C., & Sun, X. 2023. Agent-based Simulation and Optimization of Hybrid Flow Shop Considering Multi-skilled Workers and Fatigue Factors. *Robotics and Computer-Integrated Manufacturing*. *80*(December 2021): 102478. https://doi.org/10.1016/j.rcim.2022.102478.

[26] 26. Ahmid, A., Dao, T., & Le, N. Van. 2021. Enhanced Hyper-Cube Framework Aco for Com- Binatorial Optimization Problems. (August). https://doi.org/10.20944/preprints202108.0573.v1.

[27] Bouzidi, A., & Riffi, M. E. 2017. A Comparative Study of Three Population-Based Metaheuristics for Solving the JSSP. *Europe and MENA Cooperation Advances in Information and Communication Technologies.* 235–243, 235–243. https://doi.org/10.1007/978-3-319-46568-5.

[28] Ganesan, N., Goel, N., Agarwal, V., & Thangaraju, B. 2023. SFC-ACO: A Robust Path Failure Handling Method for Service Function Chaining in Kubernetes on OpenStack Magnum. *2023 International Conference on Computer, Electronics and Electrical Engineering and their Applications, IC2E3 2023*. 1–7. https://doi.org/10.1109/IC2E357697.2023.10262703.

[29] Jaber, M. Y., Givi, Z. S., & Neumann, W. P. 2013. Incorporating Human Fatigue and Recovery Into the Learning-forgetting Process. *Applied Mathematical Modelling*. https://doi.org/10.1016/j.apm.2013.02.028.

[30] 30. Jiang, X., Lin, Z., He, T., Ma, X., Ma, S., & Li, S. 2020. Optimal Path Finding with Beetle Antennae Search Algorithm by Using Ant Colony Optimization Initialization and Different Searching Strategies. *IEEE Access*. 8: 15459–15471. https://doi.org/10.1109/ACCESS.2020.2965579

[31] Nazif, H. 2015. Solving Job Shop Scheduling Problem Using an Ant Colony Algorithm. *Journal of Asian Scientific Research*. 5(5): 261–268. https://doi.org/10.18488/journal.2/2015.5.5/2.5.261.268.

[32] 32. Bechtold, S. E., Janaro, R. E., & Sumners, D. W. L. 1984. Maximization of Labor Productivity Through Optimal Rest-Break Schedules. *Management Science*. 30(12): 1442–1458. https://doi.org/10.1287/mnsc.30.12.1442.