Jurnal Teknik Industri, Vol. 26, No. 2, December 2024 DOI: https://doi.org/10.9744/jti.26.2.87-102
ISSN 1411-2485 print / ISSN 2087-7439 online

A Dual Resource Constrained Unrelated Parallel Machine Scheduling Model Considering Tardiness and Workload Balance

Karina Amanda Larasati[1*], Sukoyo[2], Muhammad Akbar[2]
[1)] Faculty of Industrial Technology, Master Program in Industrial Engineering Department, Bandung Institute of Technology, Jl. Ganesha 10, Bandung 40132, Indonesia
[2)] Faculty of Industrial Technology, Industrial Engineering Department, Bandung Institute of Technology, Jl. Ganesha 10, Bandung 40132, Indonesia
Email: 23422008@mahasiswa.itb.ac.id*, sukoyo@itb.ac.id, muhammad@itb.ac.id
*Corresponding author

Abstract: The proposed study pertains to Multi-Task Simultaneous Supervision Dual Resource Constrained (MTSSDRC), which considers minimizing tardiness and workload balance. The workload balance is calculated using the Workload Smoothness Index (WSI). Additionally, the research concentrates on unrelated parallel machine schedules as they reflect the actual industry conditions in practice. This issue employs three methodologies: Mixed-Integer Linear Programming (MILP), Mixed-Integer Quadratic Problem (MIQP), and Mixed-Integer Quadratically Constrained Programming (MIQCP). The results in the MILP model focus on the value of total tardiness so that the results obtained have a smaller total tardiness value. However, it is still uncertain that the WSI value is better because no boundaries have been set between the two objective functions to achieve optimal values in the MILP model. The MIQP model focuses on the Workload Smoothness Index (WSI) value to give a limit to the total tardiness objective function. Limit values are obtained from MILP model values, and the resulting WSI value becomes smaller. Moreover, the MIQCP model focuses on the total tardiness value and has limits in the form of permitted WSI. This model produces a small WSI value in accordance with the specified WSI limits while adjusting the specified total tardiness.

Keywords: MTSSDRC scheduling, unrelated parallel machines, workload smoothness index, tardiness.

Introduction

Scheduling is a pivotal decision-making process commonly utilized across diverse manufacturing and service industries. This intricate procedure entails judiciously assigning resources to undertake tasks within a pre-determined timeframe while striving to optimize one or more specific objectives [1]. The scheduling of parallel machines is one of the intricate scheduling quandaries. This challenge comes from its exceptional utility, not solely in the manufacturing industry but also in the realm of computing. Numerous practical examples can be observed in diverse manufacturing systems, including circuit board printing production, technology cell groups, semiconductor manufacturing, plastic and painting industries, injection printing processes, recycling manufacturing, and more [2]. The problem of scheduling unrelated parallel machines is significant for the manufacturing industry since scheduling will save company resources, especially time management. Hence, companies can profit more quickly and precisely by solving scheduling problems [3]. In the actual manufacturing industry, unrelated parallel machines yield varying outputs due to disparities in machine performance [4]. It is attributed to the replacement of outdated machines with modern ones. Consequently, new-fangled machines function alongside their older counterparts [5]. In addition, allocated resources can include machinery, operators, and tasks performed, including operations. Each task can be given a particular priority level, such as a time limit for work. An essential aim of scheduling is to reduce the number of tasks that surpass their designated time limit [6].

The number of overdue tasks (tardy jobs) is determined by the completion duration exceeding the deadline. Tardiness is a standard metric used in the manufacturing industry to assess performance. The consideration of delay is crucial when scheduling, as it may lead to a lack of remuneration and compensation for the company [7]. Manufacturing managers in developed nations regard timely and rapid delivery as significant due to its impact on cost, profitability, and customer satisfaction. In addition, customer satisfaction also affects the company's long-term financial performance through the customer's desire to repurchase and builds loyalty among existing customers [8].

On the issue of scheduling classical machines, the number of machines involved was initially disregarded, and the number of workers in each machine was fixed. The allocation of operators to a particular job can significantly reduce completion time and enhance production efficiency. However, when the employee allocation aspect is ignored, it can lead to managerial problems in the company [9]. In addition, this leads to an unequal distribution of the workload among the operators and can provoke feelings of jealousy among them [6]. Hence, it is imperative to contemplate the equilibrium of operators' workloads while devising production schedules for them.

An illustration of the conditions of the unrelated parallel machine scheduling problem associated with the delay in completion time (tardiness) and the workload balance was found in PT. X, which is one of the aircraft manufacturing industries. Besides, the machines used have semi-automatic characteristics. The operator could leave this machine during work time. Operators were required only for internal activities such as unloading jobs and controlling or performing setups [10]. This benefit presented a chance for companies utilizing semi-automated machinery to function with a reduced workforce when faced with the task of managing multiple machines simultaneously.

Nevertheless, the implementation proved to be challenging due to the need for integrating two distinct scheduling methods: firstly, determining the work sequence on machines and secondly, establishing a task sequence that accounts for various factors such as sequence-dependent setup, unloading, moving operations, and simultaneous supervision by an operator. This intricate schedule is called Multi-Task Simultaneous Supervision Dual Resource-Constrained (MTSSDRC). MTSSDRC was first studied by Akbar and Irohara [11]. As for the literature that has continued MTSSDRC research with dual functions of objective makespan and tardiness like Pineda, *et al.* [12] with focus machine environment is a flexible job shop, but there are still a few studies that consider the state of parallel machine systems unrelated.

To the researcher's knowledge, no prior investigation has examined the scheduling of parallel, unrelated MTSSDRC engines while considering both tardiness and operator workload balance. According to the International Labour Organization (ILO) [13], Article 111 (b) of the Convention states that any distinction, exclusion, or other choice that results in the loss or reduction of equal opportunities or treatment in employment or position is referred to as discrimination. Therefore, this research is expected to contribute to better scheduling completion methodologies and address the challenges of sustainable manufacturing development by considering workload balance. In addition, the research aims to develop a mathematical model that describes problems specifically.

Methods

Literature Review

Scheduling problems are classified into three fields: machine environment (α), limitation (β), and objective function (γ) [14]. The field α contains a single machine (1), identical parallel machine (Pm), related parallel machines (Qm), unrelated parallel machinery (Rm), flow shop (Fm), flexible flow shop (FFc), job shop (Jm) and flexible job shop (FJc) [1]. The field β describes the boundaries of the scheduling problem, and the field γ explains whether a problem is a single or multi-purpose objective.

In general, parallel machine scheduling is when every $\boldsymbol{n}$ job is processed by only one m machine. A workstation cannot process up to one job at a time. Additionally, pre-emption is not allowed, i.e., any job cannot exit a particular machine's process once its processing is complete [15]. Parallel machine scheduling has become a popular research area due to its wide range of potential application areas, and this popularity has increased rapidly over the last few years due to the emergence of parallel processor computer technology [7].

An unrelated parallel machine is a condition of a series of parallel machines with no identical production capacity and no correlation that can be identified with the comparison level with other machines [1]. Factors that cause unrelated conditions on parallel machines are the use of different machines; this is because manufacturing industries rarely consider the workload of machines at the time of their operation. Therefore, research on scheduling unrelated parallel machines has become popular and has been researched by Costa *et al.* [15], Chen & Wu [16], H.G-de-Alba *et al.* [17], and A. Berthier *et al.* [18].

The field β describes the limit of the scheduling problem. Some examples of limitations used by researchers are release date, pre-emption, permutation, sequence dependent setup times, batch processing and others [1]. One of the limitations that researchers often consider is the dual resource constraint (DRC). Some resources are usually workers and machines [19]. The characteristic of DRC allows operators to move from one machine to another to carry out tasks [20]. DRC has become one of the most frequently used constraints because it focuses on scheduling and relies on workers, machines, and unique attributes of resources to provide an optimal schedule [21]. The DRC study that considered cases of more than one operator per machine to shorten processing time jobs and minimize total tardiness was conducted by Hu [9] and Chaudry & Drake [7]. Then, Costa et al. [15] developed a model for DRC that considers job sequences and limited human resources. Another consideration is that each worker has a certain skill level, which affects setup time. Besides, Bapstiste *et al.* [22] conducted DRC research by considering a series of independent and non-preemptive jobs.

Technological advances in the industry led to the emergence of semi-automatic machines; operators only need to perform tasks before and after the machine, such as setup and unloading [23]. Thus, the production process or machine does not need to be processed specifically by the operator [22]. However, industries that use semi-automatic machines will only be able to handle productivity problems if an operator is assigned to each machine. The operator's job becomes redundant once the machine begins processing the work. Regarding these conditions, Akbar and Irohara [11] solved the DRC problem with the MTSSDRC approach. This scheduling improves DRC's existing problems in the manufacturing industry to produce a better solution. The characteristic of MTSSDRC is that each operator can operate several machines simultaneously for different working elements, namely set up and unloading. Although strategies with fewer operators can increase labor productivity, a manufacturer must be careful when making decent schedules. Inappropriate schedules can result in long waiting times and a lot of makepan or tardiness.

Moreover, it could create an imbalance in the workload between operators that could provoke a feeling of jealousy. Variable workloads can easily be reduced when the schedule is looser, resulting in longer makepan or tardiness intervals [11]. MTSSDRCs have fewer operators capable of controlling parallel semi-automatic machines simultaneously in the face of unbalanced workloads on operators [24].

The field γ explains whether a problem is a single objective schedule or a multi-objective. The γ field minimizes total tardiness, makespan, completion time, total waiting time, and production costs [1]. Minimizing total tardiness is becoming popular in scheduling research because it is essential to complete work more quickly so that companies can avoid cost losses or penalties. As for research on the objective function of tardiness, there are H. G.-de-Alba *et al.* [17], P. Beldar *et al.* [25] and Paksi & Ma'aruf [26]. Using a solver is one way to get an optimal global solution to a scheduling problem. Although scheduled problem solving can use an exact method with the help of tools such as mathematical model-based solvers, when faced with cases with vast amounts of data, such as those in an aircraft manufacturing company, solvers can not only spend much time but also cannot produce a solution that is good enough to be analyzed, as has been proven by Akbar & Irohara [11]. Based on the scheduling characteristics that had been explained, a research roadmap for DRC scheduling problems from several papers before this research is presented in Table 1.

Table 1. Mapping characteristics DRC scheduling issues

No	Literature	Problem characteristics					Objective
		Machine type	DRC Type				
			w vs m	Simultaneous supervision	Task type	Moving	
1	Hu [9]	IPM	$w > m$	-	-	-	Total tardiness
2	Chen & Wu [16]	UPM	$w < m$	-	-	-	Total tardiness
3	Chaudry & Drake [7]	IPM	$w > m$	-	-	-	Total tardiness
4	Costa *et al.* [15]	UPM	$w < m$	√	1: (s)	-	Makespan
5	Baptiste *et al.*[22]	IPM	$w \leq m$	√	4: (l, s, c, u)	-	Makespan
6	Akbar & Irohara [24]	IPM	$w < m$	√	2: (s, u)	√	Total tardiness & WSI
7	Akbar & Irohara [6]	IPM	$w < m$	√	2: (s, u)	√	Makespan & WSI
8	H.G-de-Alba *et al.* [17]	UPM	$w < m$	-	2: (s, u)	-	Total tardiness
9	A. Berthier *et al.* [18]	UPM	$w > m$	-	2: (s, u)	-	Makespan
10	This Research (2024)	UPM	$w < m$	√	2: (s, u)	√	Total tardiness & WSI

Note:
- Machine type: Identical Parallel Machine (IPM), Unrelated Parallel Machine (UPM)
- w vs m: worker (w), machine (m)
- Task type: loading (l), setup (s), controlling (c), unloading (u)

Problem Statement

Referring to the previous research conducted by Akbar dan Irohara [24], the issue of scheduling DRC on parallel machines was addressed, along with the identification of two objective functions: total tardiness and workload smoothness index (WSI), as represented in equations (1) and (2). WSI gauges the degree of workload imbalance derived from the smoothness index (SI) utilized for assembly line balancing. Scheduling with a lower WSI value results in superior balance across workloads. Achieving a perfect balance is attained when the WSI attains zero. When balancing an assembly line, SI evaluates the duration of each station's activity without factoring in idle time. Considering this, our research scrutinizes the overall time spent by every operator engaged in production activities, disregarding any waiting periods experienced by said operators, assuming NWk as the total no-waiting time performed by operator k, which consists of setup, moving, and unloading, and NWmax as the total maximum non-waiting time between operators. Then, the operator smoothness index (WSI) is calculated using equation (2), where the lower the SI, the more balanced the workload [11].

In this study, the researchers want to solve the problem of scheduling MTSSDRC on unrelated parallel machine environments. The production system consists of a set $I = (i_1, i_2, \ldots, i_m)$ of m semiautomatic machines with a set $K = (k_1, k_2, \ldots, k_w)$ of w operators, where $w < m$, to run a set $J = (j_1, j_2, \ldots, j_n)$ of n jobs. For modeling purposes, a set of $J' = (j_0, j_1, j_2, \ldots, j_n)$, which includes the dummy job j_0, is also available to assist in order restrictions. Each job requires three sequential activities when entering the production system: setup, loading, and unloading. Because the machines used are semi-automatic, this machine can be left by the operator during work time. Operators can contribute to setting up and unloading activities in a task's set $A = (a_s, a_u)$. Thus, the production process or machine does not need to be operated specifically by the operator. After completing an activity, the operator can move to another machine to perform another activity for the duration of the movement. The time it takes to complete task b of the job l is O_{bl}. Next, the machining time of the job l on machine i is P_{il}. Last, V_{hi} describes the time of movement of the operator between machine h and machine i. Other parameters are the big numbers M required in modeling. The model formula encompasses two distinct purpose functions, specifically the total tardiness and workload smoothness index (WSI), as explicitly demonstrated in equations (1) and (2).

Figure 1 shows an example of this problem with three machines and two operators processing five jobs in Akbar & Irohara's [24] research. Any setup or unloading activity starts if at least one operator and one machine are unemployed and the operator has moved to that machine.

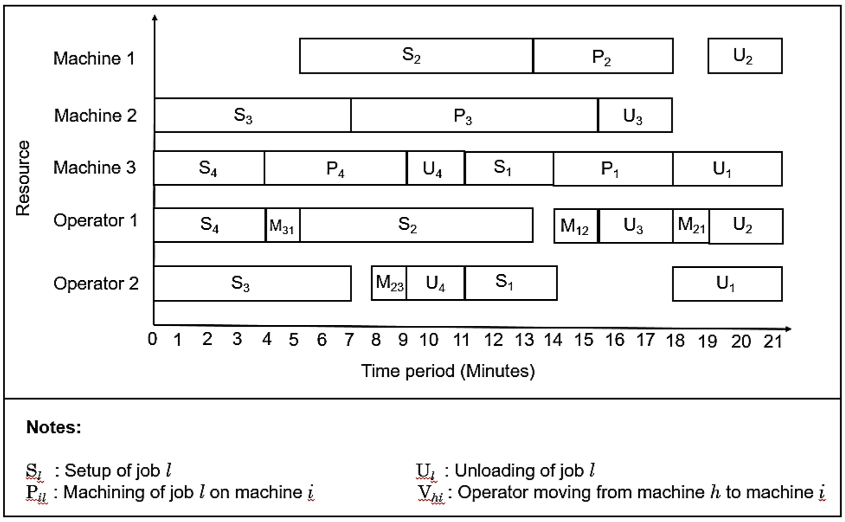


Figure 1. Illustration of DRC scheduling problems [24]

The Mathematical Model

The researchers used a mathematical model reference from previous research, namely Akbar & Irohara [24]. The difference between this research and Akbar & Irohara's is that this research focuses on parallel unrelated machine types. As a result, by changing the condition of the machines from identical parallel machines, where the processing time for jobs is the same across all machines, to unrelated parallel machines, where the processing time for jobs varies depending on the machine assigned, the researchers had to add new variables or constraints to the mathematical model. There are additional constraints on the equations (9), (10), (11), (12), and (17) as a new development of the previous model as a description of the research on this MTSSDRC scheduling problem. The mathematical formulation is described as follows.

Indices

f, j, l = 0,1,2, ... n jobs
g, h, i = 1,2, ... , m machines
k, q = 1,2, ... , w workers
a, b = 1, 2 activities (1–setup; 2–unloading)

Parameters

P_{il} = processing time of job l on machine i
V_{hi} = operator moving time of machine h to machine i
O_{bl} = the duration required to finish task activity b of job l
d_l = due date of job l
M = a big number

Decision variables

$X_{khajibl}$ = a binary variable that results in 1 if operator k is designated to undertake task b of project l on machine I after completing task an of project j on machine h.
Q_{fl} = a binary variable that results in 1 if the setup activity of job l performs before the unloading activity of job f in the same machine.
O^c_{bl} = the duration required to finish activity task b of job l.
P^c_{il} = the machining completion time of job l on machine i.
V^c_{bl} = the moving completion time to perform task b of job l by the operator.
t_l = tardiness of job l.
b_k = the cumulative non-waiting time of operator k.
b_{max} = the maximum total non-waiting time from all operators.

Mathematical model

$$\text{Minimize } f_1 = \sum_{l\in J} t_l \quad (1)$$
$$\text{Minimize } f_2 = WSI = \sqrt{\sum_{k\in K}(b_k - b_{max})^2} \quad (2)$$
$$\sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}\sum_{i\in I} X_{khajibl} = 1 \quad \forall b \in A;\ l \in J \quad (3)$$
$$\sum_{k\in K}\sum_{h\in I}\sum_{i\in I}\sum_{b\in A}\sum_{l\in J} X_{khajibl} \le 1\ \forall a \in A; j \in J \quad (4)$$
$$\sum_{h\in I}\sum_{i\in I}\sum_{b\in A}\sum_{l\in J} X_{kha_u j_0 ibl} \le 1 \quad \forall k \in K \quad (5)$$
$$\sum_{k\in K}\sum_{h\in I}\sum_{i\in I}\sum_{b\in A}\sum_{l\in J} X_{kha_s j_0 ibl} = 0 \quad (6)$$
$$\sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'} X_{khajia_s l} - \sum_{q\in K}\sum_{g\in I}\sum_{x\in A}\sum_{f\in J'} X_{qgcfia_u l} = 0 \quad \forall i \in I;\ l \in J \quad (7)$$
$$\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'} X_{khajibl} \ge \sum_{g\in I}\sum_{b\in A}\sum_{f\in J} X_{kiblgcf} \quad \forall k \in K;\ i \in I;\ b \in A;\ l \in J \quad (8)$$
$$O^c_{bl} - V^c_{bl} \ge \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}\sum_{i\in I} O_{bl} . X_{khajibl}\ \forall b = 1,2;\ l = 1,2,\dots,n \quad (9)$$
$$V^c_{bl} - O^c_{al} \ge \sum_{h\in I}\sum_{i\in I}\sum_{k\in K} V_{hi} \times X_{khajibl} - M \times \left(1 - \sum_{h\in I}\sum_{i\in I}\sum_{k\in K} X_{khajibl}\right) \quad \forall b \in A; l\, J; a \in A;\ j \in J' \quad (10)$$
$$O^c_{a_u l} - P^c_{il} \ge \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}\sum_{i\in I} O_{kia_u l} . X_{khajia_u l}\ \forall l = 1,2,\dots,n \quad (11)$$
$$P^c_{il} - O^c_{a_s l} \ge \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}\sum_{i\in I} O_{pil} . X_{khajia_s l}\ \forall l = 1,2,\dots,n \quad (12)$$
$$\begin{cases} O^c_{a_s l} - O^c_{a_u f} \ge \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'} O_{a_s l} \times X_{khajia_s l} \\ - M \times \left(2 - \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}(X_{khajia_s l} + X_{khajia_s f}\right) + Q_{fl}) \\ O^c_{a_s f} - O^c_{a_u l} \ge \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'} O_{a_s f} \times X_{khajia_s f} \\ -M \times \left(2 - \sum_{k\in K}\sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}(X_{khajia_s l} + X_{khajia_s f}\right) + 1 - Q_{fl}) \end{cases} \forall i = 1,2,\dots,m;\ f = 1,2,\dots,n; l + 1, f + 2, \dots, n \quad (13)$$
$$O^c_{a_u j0} = 0 \quad (14)$$
$$t_l \ge O^c_{a_u l} - d_l \quad \forall l \in J \quad (15)$$
$$t_l \ge 0 \quad \forall l \in J \quad (16)$$
$$b_k = \sum_{h\in I}\sum_{a\in A}\sum_{j\in J'}\sum_{i\in I}\sum_{b\in A}\sum_{l\in J} X_{khajibl} \times (O_{bl} + V_{hi})\ \forall k \in K \quad (17)$$
$$b_{max} \ge b_k \quad \forall k \in K \quad (18)$$
$$X_{khajibl} \in \{0; 1\}\ \forall k \in K; h \in I;\ a \in A; j \in J';\ i \in I;\ b \in A; l \in J \quad (19)$$
$$Q_{fl} \in \{0; 1\}\ \forall f \in J; l > f \quad (20)$$

Equations (1) and (2) represent each of the dual-purpose functions considered in scheduling issues, i.e., minimizing tardiness and WSI. Equations (3) and (4) require each task to be dedicated to only one machine and a unique operator. These equations also ensure that each task has only one predecessor and a maximum of one substitute. Equations (5) and (6)) ensure that only unloading tasks are performed as the first tasks of each operator for the task j_0. Equation (7) requires each job to use only one machine. Equation (8) sets a restriction on the sequence of tasks each operator performs. Equation (9) indicates the difference between the minimum completion time of operation activity b job l and the completion time of the transition to activity b work l is equal to the time operator k completed operation b job l on machine i. Equation (10) indicates the minimum gap between the time after moving to the b job I and the time before completing activity a, job j is the same as the time of transition of the operator towards activity b, job l. Equation (11) ensures that operator k performs the unloading activity on job l after performing the job machine process is completed on the computer i. Equation (12) ensures that operator k performs the setup activity on job l after performing the job machine completion process on machine i. Equation (13) is a double constraint that accommodates priority constraints between job l and each machine i. The preceding task's unloading procedure must be completed before initiating the subsequent job's setup procedure. Equation (14) ensures the unloading time on dummy job j_0 is zero. Equations (15) and (16) set the tardiness value for each job. Equations (17) and (18) count the busy and busiest times. Equations (19) and (20) are binary constraints.

Results and Discussions

The proposed model's development is an extension of the research conducted by Akbar & Irohara [6], which uses three reference models: mixed-integer linear programming (MILP), mixed-integer quadratic problem (MIQP), and mixed-integer quadratically constrained programming (MIQCP) on the type of scheduling of identical parallel machines with multi-function objective minimization makespan and WSI. This research used the same model structure as MILP, MIQP, and MIQCP but had a different scope of research, namely the type of scheduling that focuses on parallel machines unrelated to the multi-functional purpose of minimizing tardiness and WSI. MILP is used to solve the case with a single objective of minimizing tardiness, MIQCP is used to solve the square function derived from the MILP solution, and MICQCP is used to convert squares to the function of the goal. This research uses uniform distribution. There are 11 cases for comparison experiments, which are identified by $J \times M \times O$ (jobs number × machines number × operators number). We refer to our previous paper [24] in generating all parameters using the uniform distribution, i.e., $U[1, 79]$, $U[1, 99]$, $U[1, 20]$, and $U[3,10]$, respectively, for setup, machining, unloading, and moving time. The researchers also refer to [27] in generating the due dates from the uniform distribution $[P(1 - T - R/2), P(1 - T + R/2)\ ]$, where P is computed using equation (21), T is the mean tardiness factor with 0.8 for the tight due dates, and R is the relative range of due dates with 0.4 to make it tight.

$$P = \frac{\sum_{l \in J} o_{asl} + p_{il} + o_{aul}}{V^2} \tag{21}$$

The researchers use one example from three data conditions in the 4 x 3 x 2 case, allocating 4 jobs, 3 machines, and 2 operators to illustrate the mathematical model that has been developed using actual data. The determination of setup, unloading, machining, and moving parameters can be found in the Appendix.

Table 2 shows the solution obtained using Gurobi 7.5.2 software with Intel® Core™ i3-6006U CPU @ 2.00GHz. In MIQP and MIQCP models, Gurobi utilizes the PreMIQCPForm and PreQLinearize parameters to control the transformations applied when generating the model to produce a solver. The quadratic linearization resolution in PreQLinearize is done by controlling the linearization of the Q matrix during the pre-solve stage. When binary variables are used in quadratic equations, there is flexibility in expressing the same equation in various ways. Options 1 and 2 aim to simplify quadratic constraints or objectives by replacing quadratic terms with linear ones, introducing extra variables and linear constraints. This could potentially convert an MIQP or MIQCP model into a MILP. Option 1 concentrates on creating a MILP version with a robust LP relaxation, aiming to restrict the size of the MIP search tree. Option 2 seeks a concise reformulation to decrease the cost per node. Option 0 strives to maintain the Q matrices unchanged; it refrains from adding variables or constraints, though it might still tweak quadratic objective functions to ensure they are positive semi-definite (PSD). Then, for the presolved MIQCP model using, the PreMIQCPForm parameter is determined by defining the format of the presolved MIQCP model. Option 0 keeps the model in MIQCP form, allowing the branch-and-cut algorithm to handle a model with arbitrary quadratic constraints. Option 1 uniformly converts the model into MISOCP form, where quadratic constraints are translated into second-order cone constraints. Option 2 uniformly transforms the model into disaggregated MISOCP form, translating quadratic constraints into rotated cone constraints, where each rotated cone consists of two terms and involves only three variables.

The resulting %Gap is the percentage gap between the lower and upper bound objective function values of the currently found best solution and the expected optimal value. In the optimization context, this gap describes how close the currently found solution is to the actual optimal solution.

Table 2. Comparison of solutions between three condition data

$J \times M \times O$	Problem	Setting		Solution		%Gap	Run time (s)
		Tardiness	WSI[2]	Tardiness	WSI		
4 x 3 x 2	MILP	-	-	336	134.01	0	5
		-	-	315	243.53	0	1
		-	-	460	107.17	0	2
	MIQP	336	-	336	8	0	30
		315	-	315	11	0	15
		460	-	460	20	0	10
	MIQCP	-	4	345	2	0	21
		-	9	224	3	0	5
		-	16	464	4	0	2
4 x 4 x 2	MILP	-	-	310	39.01	0	5
		-	-	300	188.26	0	45
		-	-	420	344.27	0	16
	MIQP	310	-	310	0	0	8
		300	-	300	23	0	65
		420	-	420	36	0	30
	MIQCP	-	9	310	3	0	6
		-	9	307	3	0	25
		-	9	437	3	0	20
4 x 4 x 3	MILP	-	-	239	1	0	1
		-	-	247	279.27	0	2
		-	-	333	641.13	0	0
	MIQP	239	-	239	1	0	0
		247	-	247	0	0	2
		333	-	333	2	0	0
	MIQCP	-	16	239	4	0	6
		-	16	251	4	0	3
		-	4	333	2	0	1
5 x 3 x 2	MILP	-	-	523	188.67	0	32
		-	-	515	88.35	0	140
		-	-	552	125.48	0	30
	MIQP	523	-	-	-	Infeasible	600
		515	-	-	-	Infeasible	600
		552	-	-	-	Infeasible	600
	MIQCP	-	9	537	3	0	20
		-	4	515	2	0	190
		-	16	497	4	0	25
5 x 4 x 2	MILP	-	-	514	87.50	0	330
		-	-	494	89.48	0	110
		-	-	526	93.69	0	145
	MIQP	514	-	-	-	Infeasible	600
		494	-	-	-	Infeasible	600
		526	-	-	-	Infeasible	600
	MIQCP	-	9	514	3	0	415
		-	4	509	2	0	165
		-	9	540	3	18.33	600
5 x 4 x 3	MILP	-	-	409	35.70	0	30
		-	-	413	135.96	0	20
		-	-	409	69.89	0	5
	MIQP	409	-	409	10	100	600
		413	-	413	3	100	600
		409	-	409	1	100	600
	MIQCP	-	1	415	1	0	55
		-	4	414	2	0	30
		-	4	409	2	0	5
7 x 4 x 2	MILP	-	-	842	858.16	37.41	600
		-	-	766	425.79	26.95	600
		-	-	868	787.96	30.97	600
	MIQP	842	-	-	-	Infeasible	600
		766	-	-	-	Infeasible	600
		868	-	-	-	Infeasible	600
	MIQCP	-	9	846	3	41.29	600
		-	4	829	2	38.85	600
		-	9	891	3	33.95	600

	MILP	-	-	1477	79.01	54.32	600
		-	-	1335	175.98	52.47	600
		-	-	1514	571.29	51.95	600
	MIQP	1477	-	-	-	Infeasible	600
9 x 4 x 2		1335	-	-	-	Infeasible	600
		1514	-	-	-	Infeasible	600
	MIQCP	-	16	1474	4	53.15	600
		-	25	1534	5	59.09	600
		-	16	1682	4	61.26	600
	MILP	-	-	977	135.65	40.02	600
		-	-	953	436.03	40.10	600
		-	-	1027	364.68	39.40	600
	MIQP	977	-	-	-	Infeasible	600
9 x 5 x 3		953	-	-	-	Infeasible	600
		1027	-	-	-	Infeasible	600
	MIQCP	-	9	1058	3	46.16	600
		-	4	1137	2	51.72	600
		-	4	1196	2	50.33	600
	MILP	-	-	1428	112.30	53.00	600
		-	-	1248	889.32	45.70	600
		-	-	1381	211.30	51.78	600
	MIQP	1428	-	-	-	Infeasible	600
10 x 4 x 3		1248	-	-	-	Infeasible	600
		1381	-	-	-	Infeasible	600
	MIQCP	-	4	1461	2	50.10	600
		-	4	1574	2	60.86	600
		-	16	1471	4	48.06	600
	MILP	-	-	1313	184.50	48.84	600
		-	-	1212	148.61	48.58	600
		-	-	1393	316.10	48.44	600
	MIQP	1313	-	-	-	Infeasible	600
10 x 5 x 3		1212	-	-	-	Infeasible	600
		1393	-	-	-	Infeasible	600
	MIQCP	-	16	1392	4	54.14	600
		-	4	1219	2	48.16	600
		-	16	1510	4	62.05	600

Note:
- J: *Job*, M: *Machine*, O: *Operator*

The MILP model approach begins with researchers running the single objective MTSSDRC function to minimize the tardiness in the overall case presented in Table 2. Based on the problem model described in this case, mixed-integer linear programming (MILP) is an operator alongside an additional constraint that determines the total busy time for each operator, as demonstrated by Akbar and Irohara [6]. This shows that MILP will be equal to mixed integer non-linear programming (MINLP) by removing equations (2) and (18). Therefore, MILP can be formulated as follows:

$$\text{Minimize } f_1 = \sum_{l \in J} t_l \quad (22)$$

Subject to:
Eqs. (4) – (17), (19) – (20)

Through this experiment, researchers can assess the efficacy of MILP in producing optimal WSI values. To further improve tardiness and WSI values, it is imperative to incorporate them into the model generation process. Notably, MILP has demonstrated remarkable success in identifying optimal solutions for even highly constrained problems. Nevertheless, the outcome does not ensure optimal WSI. As illustrated in Table 2, the result consistently surpasses the best-found value, implying that the MILP model generates an extensive WSI value discrepancy among operators. Based on the results of Table 2, MILP in small cases 4 x 3 x 2 to 5 x 4 X 3 yields a %gap of 0, which indicates that the result is the optimal global solution. However, in medium cases 7 x 4 x 2 up to significant cases 10 x 5 x 3, it results in %gaps above 0, which suggests that the results have not reached the optimum global solution due to the presence of a set running time limit set at 600 s, based on research of Akbar & Irohara [6] which requires a longer running time to obtain the optimal overall solution.

Then, the MIQP model approach converts the function of minimizing tardiness (Equation 1) into a constraint. This approach begins with the value of tardiness limited to the maximum value obtained from the MILP. The second objective function, the workload smoothness index (WSI), developed by Akbar & Irohara [11], serves to measure workload unbalance, which is adapted from the smoothness index (SI) from the assembly line balancing problem. WSI compares each operator's busy (non-waiting) time, including setup, unloading, and moving activities [11]. The optimal solution obtained not only minimizes one objective function but also tries to achieve a balance of two objective functions.

$$\text{WSI}=\sqrt{\sum_{k\in K}(b_k - b_{max})^2} \tag{23}$$

The WSI model, which was initially in root form, was then squared to adjust the MIQP model and the modules contained in the solver. The WSI objective function is then squared to align with the solver's decision-making requirements. Subsequently, the mixed-integer quadratic programming (MIQP) model is formulated in the following manner:

$$\text{Minimize} = WSI^2 = \sum_{k\in K}(b_k - b_{max})^2 \tag{24}$$

Subject to:
Eqs. (4) – (20)

The tardiness value on the MIQP model is expected to have a smaller value equal to the maximum value of tardiness that has been set in the following model:

$$t_l \le t_l^{lim} \tag{25}$$

Table 2 reveals that the MIQP of cases ranging from 4 x 3 x 2 to 4 x 4 x 3 displays a %gap of zero, thus signifying the attainment of an optimal global solution. However, for cases spanning from 5 x 3 x 2 to 10 x 5 x 3, the MIQP yields infeasible %gaps indicating that the result failed to reach its optimal overall solution. The run time needs to be longer for the decomposer to obtain a suitable solution. Case 5 x 4 x 3 produces a %Gap of 100. This suggests that the achieved result does not reach the optimal general solution because the optimal solution point has an objective value of 0 or is very close to constraint, so it requires a longer running time to get the optimum global solution result. The WSI value depends on the working-hour balance between the operators, so, logically, the MIQP value can change. The WSI value progressively improves with the objective of MIQP, which is to identify an optimal solution for the quadratic mathematical model. Although Gurobi can execute a multi-purpose function, it was not utilized in this study because the MILP model is linear and MIQP is non-linear.

Furthermore, the MIQCP model approach begins with the WSI object function squared (Equation 21) becoming a barrier when working cases on this problem so that the value of WSI^2 has a value smaller than the limit value. Then, the model of mixed-integer quadratically constrained programming is formulated as follows:

$$\text{Minimize } f_1 = \sum_{l\in J} t_l \tag{26}$$

Subject to:
Eqs. (4) – (20)

The WSI value on the MIQCP model is expected to have a smaller value equal to that of the specified WSI as in the following model:

$$\text{WSI}^2 \ge \sum_{k\in K}(b_k - b_{max})^2 \tag{27}$$

$$\text{WSI}^2 \le WSI_{lim}^2 \tag{28}$$

WSI values are determined by trial and error, starting with the square values of 0, 1, 4, 9, etc. This is because when the researchers entered the number 0 as the optimal number, it turned out that the resulting output could not have been optimal or had an infeasible value. Hence, the researchers tried another quadratic value. Determining the quadratic value does not have to be a whole number but can use other random numbers. Using quadratic values with round numbers only makes it easier to determine quadratic values. Ideally, the optimum solution is expected when the WSI value is = 0 or close to 0, but infeasible values are produced in some cases when generating models. This is because the model could not meet the function of the purpose of tardiness, so the researchers increased the quadratic value gradually so that an ideal value was obtained. In Table 2, MIQCP in cases 4 x 3 x 2 to 5 x 4 x 3 yields a %gap of 0. This indicates that the result obtained is the optimal global solution, but in cases 5 x 4 x 2 of data 3, cases 7 x 4 x 2 to 10 x 5 x 3 yielding %gaps above 0. This suggests that the results achieved have not reached the optimum global solution due to a set running time limit, so a longer running time is required to produce an optimal overall solution. The overall solution produced on the case 4 x 3 x 2 to 10 x 5 x 3 produces different run times in generating models; it depends on the case's complexity to get the optimal solution. In addition, the complexity of a case can also be seen in the iteration of the gurobi output on the resolved rows and columns section. The higher the number of rows and columns, the higher the iteration value; this indicates the more complex the case.

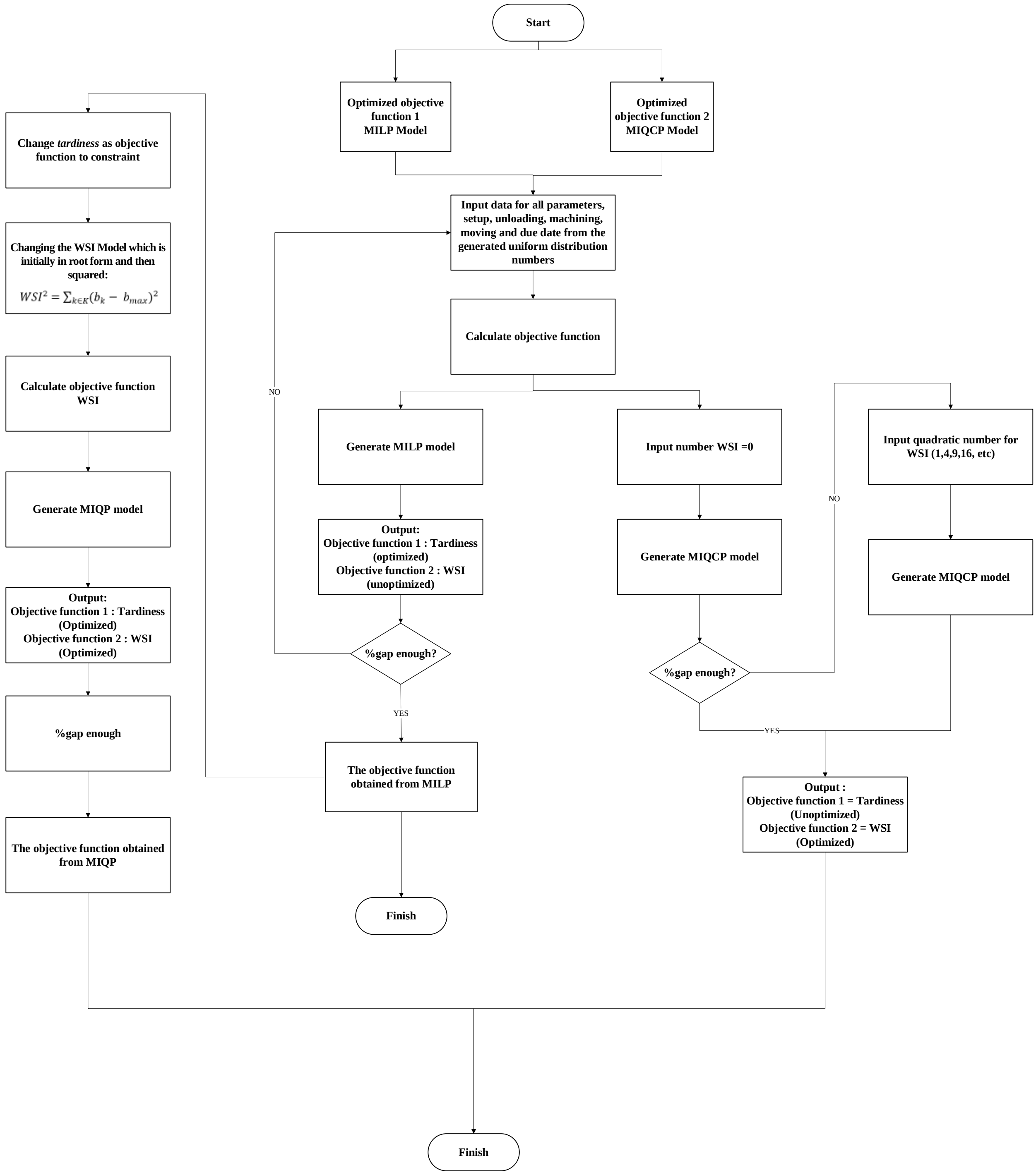


Figure 2. MILP, MIQP, MIQCP flowchart

The optimal solution of a multi-objective problem can be described in a Pareto front. However, the optimal solution that results has a large number, so it is challenging to search for overall. Pareto front has an infinite number, so the researchers only selected several solutions for this case in the hope that the researcher will reach the extreme point. This extreme point is obtained by completing or optimizing objective function 1 (Tardiness) first, then looking for a solution in objective function 2 (WSI), or it can also be done vice versa. The extreme tardiness points are obtained from the MILP, while the WSI excretion points are from the MIQCP model. These excretions have different characteristics, such as the extreme point of tardiness having the lowest objective value characteristic, whereas, at the trim point, WSI has the adjustable characteristic. It can be entered from MILP without considering MIQP, and to guarantee a better WSI value, consider the MIQCP. However, in the MIQCP model, there are weaknesses; we find it quite challenging to find the starting point of the WSI, so there is an adjustment using quadratic numbers (0,1,4,9, etc.).

Ultimately, the selection method depends on specific problems and preferences in decision-making. Based on the model characteristics explained, the following is a flowchart of the MILP, MIQP, and MICP models in obtaining solver values, which can be seen in Figure 2.

Figure 3 is one of the Gantt chart results that illustrates MTSSDRC scheduling based on Table 2. This schedule allocates 4 jobs, 3 machines, and 2 operators or 4 x 3 x 2 cases. The researchers compared different scheduling results from the MILP, MIQP, and MIQCP models. On the MILP model, this scheduling only makes tardiness a function of purpose. As a result, operator 2 is very busy compared to operator 1, leading to a higher WSI rating of 134.01 with a total busy schedule time on the MILP model of 165 units. On the MIQP model, this schedule limits the tardiness value to the maximum value obtained from MILP. WSI's target functions are squared to meet the needs of the decoder on the solver. As a result, although operator 2 looks very busy compared to operator 1, the resulting WSI value is much lower, 8, with a total busy time scheduling on the MIQP model of 129 units of time. It shows that the workload between operators is beginning to approach balance. Then, on the MIQCP model, this scheduling changes the squared WSI target function to a barrier so that the WSI2 value has a value smaller than the limit value. As a result, operator 2 becomes no busier than operator 1 with a tardiness value that changes to 345 and WSI to 2 with a total busy time scheduling on the MIQCP model that is 127 units of time.

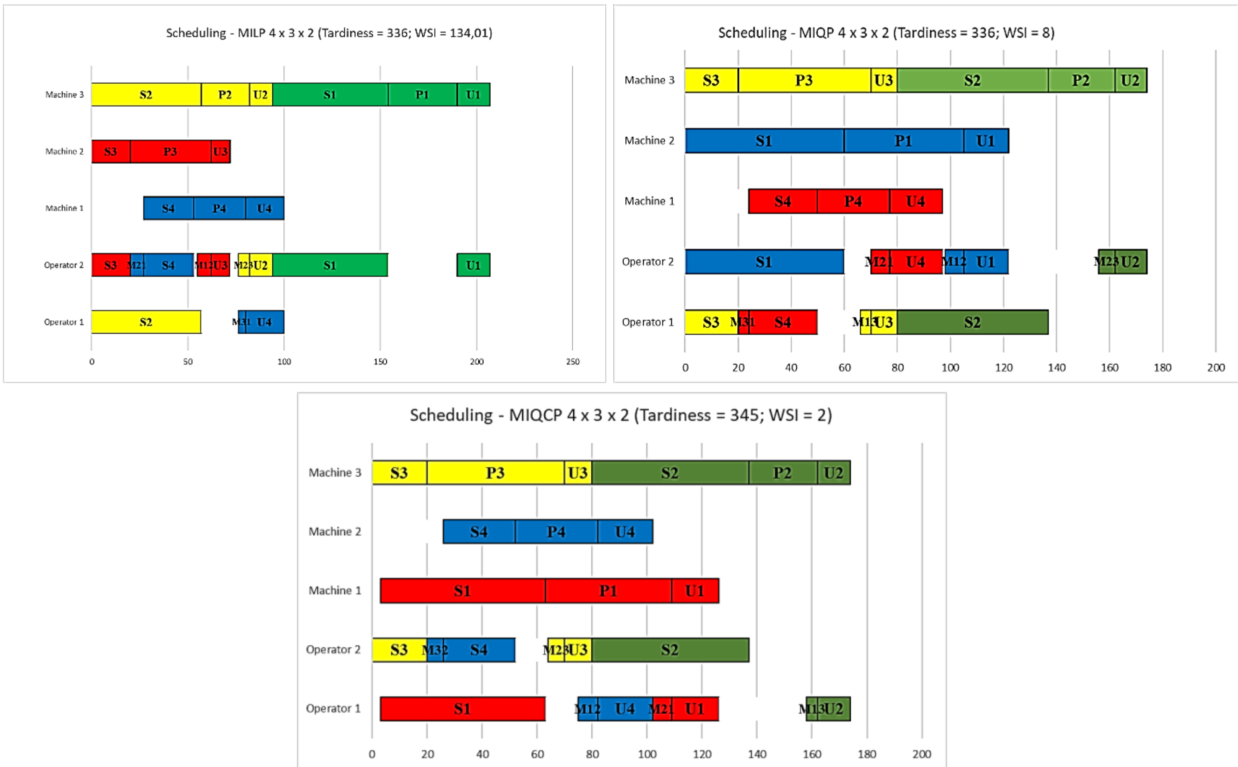


Figure 3. *Gantt chart case* 4 x 3 x 2 (MILP, MIQP, MIQCP)

Conclusions

This research is proposed as a development and addition to contributions from previous research on MTSSDRC scheduling with unrelated parallel machine focus, unlike Akbar & Irohara's [28], which focuses on the type of scheduling of identical parallel machines with multi-functional purposes such as minimizing tardiness and WSI. However, the research uses only the MINLP model approach; as a consequence, the researchers adopted the model approach used by Akbar and Irohara [5]. This research used the same model structure as MILP, MIQP, and MIQCP but developed the model due to its different scope of research, namely a type of scheduling focused on parallel machines unrelated to the multi-function purpose of minimizing tardiness and WSI. The results showed that efforts to improve the workload balance would increase the total tardiness, as seen in Table 2 compared with data with similar model approaches. In addition, the WSI can be significantly improved in small proportions.

The scheduling problem, MTSSDRC, pertaining to unrelated parallel machines, can be applied in industries that utilize semi-automatic machines. Various industries rely on this type of equipment, including those involved in aircraft, electronics, furniture production, and agro-industry, where supervision is required for the setup and loading/unloading processes carried out by operators. The importance of making an MTSSDRC scheduling model with an unrelated parallel machine is that because identical machine speeds and processing time are the same as other machines (identical parallel machines). Moreover, this condition is deemed ideal for the fact that such conditions are almost rarely found in the real world.

This research has limitations because the solvers used can only generate small cases and are impractical. The conditions in the industry, especially PT.X, which operates in the aircraft industry, have more machines, jobs, and operators and more complex problems. A suggestion for further research is that MTSSDRC scheduling problems with multi-purpose functions can be carried out in advanced research using the suitable metaheuristic algorithm so that significant cases can be solved to obtain optimal solution values quickly.

Acknowledgment

The authors wish to express their gratitude to the Department of Industrial Engineering at the Faculty of Industrial Technology, Bandung Institute of Technology, for providing unwavering support throughout this research project and publication.

References

[1] Pinedo. Michael L, *Scheduling Theory, Algorithms, and Systems*, 2016. doi: 10.1007/978-3-319-26580-3.
[2] A. J. Ruiz-Torres, F. J. López, P. J. Wojciechowski, and J. C. Ho, "Parallel machine scheduling problems considering regular measures of performance and machine cost," *Journal of the Operational Research Society*, vol. 61, no. 5, pp. 849–857, 2010, doi: 10.1057/jors.2009.38.
[3] F. Pulansari and T. D. R. M., "The unrelated parallel machine scheduling with a dependent time setup using ant colony optimization algorithm," *Jurnal Teknik Industri*: *Jurnal Keilmuan dan Aplikasi Teknik Industri*, vol. 23, no. 1, pp. 65–74, May 2021, doi: 10.9744/jti.23.1.65-74.
[4] G. R. Ith Ba Di, R. J. Mo Rag A, and A. Al-salem, "Heuristics for the unrelated parallel machine scheduling problem with setup times," 2006. doi: 10.1007/s10845-005-5514-0.
[5] J. Jungwattanakit, M. Reodecha, P. Chaovalitwongse, and F. Werner, "A comparison of scheduling algorithms for flexible flow shop problems with unrelated parallel machines, setup times, and dual criteria," *Comput Oper Res*, vol. 36, no. 2, pp. 358–378, Feb. 2009, doi: 10.1016/j.cor.2007.10.004.
[6] M. Akbar and T. Irohara, "NSGA-II variants for solving a social-conscious dual resource-constrained scheduling problem," *Expert Syst Appl*, vol. 162, Dec. 2020, doi: 10.1016/j.eswa.2020.113754.
[7] I. A. Chaudhry and P. R. Drake, "Minimizing total tardiness for the machine scheduling and worker assignment problems in identical parallel machines using genetic algorithms," *International Journal of Advanced Manufacturing Technology*, vol. 42, no. 5–6, pp. 581–594, May 2009, doi: 10.1007/s00170-008-1617-z.
[8] D. K. Seo, C. M. Klein, and W. Jang, "Single machine stochastic scheduling to minimize the expected number of tardy jobs using mathematical programming models," *Comput Ind Eng*, vol. 48, no. 2, pp. 153–161, Mar. 2005, doi: 10.1016/j.cie.2005.01.002.
[9] P. C. Hu, "Minimising total tardiness for the worker assignment scheduling problem in identical parallel-machine models," *International Journal of Advanced Manufacturing Technology*, vol. 23, no. 5–6, pp. 383–388, 2004, doi: 10.1007/s00170-003-1716-9.
[10] M. Zouba, P. Baptiste, and D. Rebaine, "Scheduling identical parallel machines and operators within a period based changing mode," *Comput Oper Res*, vol. 36, no. 12, pp. 3231–3239, Dec. 2009, doi: 10.1016/j.cor. 2009.02.022.
[11] M. Akbar and T. Irohara, "A social-conscious scheduling model of dual resources constrained identical parallel machine to minimize makespan and operator workload balance," 2018. [Online]. Available: https://www.researchgate.net/publication/329915849
[12] J. L. Andrade-Pineda, D. Canca, P. L. Gonzalez-R, and M. Calle, "Scheduling a dual-resource flexible job shop with makespan and due date-related criteria," *Ann Oper Res*, vol. 291, no. 1–2, pp. 5–35, Aug. 2020, doi: 10.1007/s10479-019-03196-0.
[13] International Labour Organization (ILO), "Konvensi-konvensi ILO tentang kesetaraan gender di dunia kerja," 2006. Accessed: Mar. 03, 2023. [Online]. Available: https://www.ilo.org/wcmsp5/groups/public/---asia/---ro-bangkok/---ilo-jakarta/documents/publication/wcms_122045.pdf

[14] R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. R. Kan, "Optimization and approximation in deterministic sequencing and scheduling: A survey." *Annals of Discrete Mathematics*, vol. 5, pp. 287-326, 1979. https://doi.org/10.1016/S0167-5060(08)70356-X

[15] A. Costa, F. A. Cappadonna, and S. Fichera, "A hybrid genetic algorithm for job sequencing and worker allocation in parallel unrelated machines with sequence-dependent setup times," *International Journal of Advanced Manufacturing Technology*, vol. 69, no. 9–12, pp. 2799–2817, Dec. 2013, doi: 10.1007/s00170-013-5221-5.

[16] J. F. Chen and T. H. Wu, "Total tardiness minimization on unrelated parallel machine scheduling with auxiliary equipment constraints," *Omega (Westport)*, vol. 34, no. 1, pp. 81–89, 2006, doi: 10.1016/j.omega.2004.07.023.

[17] H. G.-de-Alba, S. Nucamendi-Guillén, and O. Avalos-Rosales, "A mixed integer formulation and an efficient metaheuristic for the unrelated parallel machine scheduling problem: Total tardiness minimization," *EURO Journal on Computational Optimization*, vol. 10, Jan. 2022, doi: 10.1016/j.ejco.2022.100034.

[18] A. Berthier, A. Yalaoui, H. Chehade, F. Yalaoui, L. Amodeo, and C. Bouillot, "Unrelated parallel machines scheduling with dependent setup times in textile industry," *Comput Ind Eng*, vol. 174, Dec. 2022, doi: 10.1016/j.cie.2022.108736.

[19] M. M. Cunha *et al.*, "Dual resource constrained scheduling for quality control laboratories," in *IFAC-PapersOnLine*, Elsevier B.V., Sep. 2019, pp. 1421–1426. doi: 10.1016/j.ifacol.2019.11.398.

[20] J. Xu, X. Xu, and S. Q. Xie, "Recent developments in Dual Resource Constrained (DRC) system research," *Eur J Oper Res*, vol. 215, no. 2, pp. 309–318, Dec. 2011, doi: 10.1016/j.ejor.2011.03.004.

[21] M. Treleven, "A review of the dual resource constrained system research," *IIE Transactions (Institute of Industrial Engineers)*, vol. 21, no. 3, pp. 279–287, 1989, doi: 10.1080/07408178908966233.

[22] P. Baptiste, D. Rebaine, and M. Zouba, "FPTAS for the two identical parallel machine problem with a single operator under the free changing mode," *Eur J Oper Res*, vol. 256, no. 1, pp. 55–61, Jan. 2017, doi: 10.1016/j.ejor.2016.05.062.

[23] I. Karabegović, *The Role of Industrial Robots in the Development of Automotive Industry in China*, 2016. [Online]. Available: www.kwpublisher.com

[24] M. Akbar and T. Irohara, "A DRC scheduling for social sustainability: trade-off between tardiness and workload balance," 2019. Proceeding of APMS 2019. IFIP *Advances in Information and Communication Technology*, vol 566. Springer, Cham. doi: 10.1007/978-3-030-30000-5_27.

[25] P. Beldar, M. Moghtader, A. Giret, and A. H. Ansaripoor, "Non-identical parallel machines batch processing problem with release dates, due dates and variable maintenance activity to minimize total tardiness," *Comput Ind Eng*, vol. 168, Jun. 2022, doi: 10.1016/j.cie.2022.108135.

[26] A. B. N. Paksi and A. Ma'Ruf, "Flexible job-shop scheduling with dual-resource constraints to minimize tardiness using genetic algorithm," *IOP Conf Ser Mater Sci Eng*, vol. 114, no. 1, 2016, doi: 10.1088/1757-899X/114/1/012060.

[27] Y. K. Lin, H. T. Hsieh, and F. Y. Hsieh, "Unrelated parallel machines scheduling problem using an ant colony optimization approach," *International Journal of Industrial and Manufacturing Engineering*, vol. 6, no. 8, pp. 1727–1732, 2012.

Appendix

Table A.1. Setup and unloading time (O_{bl}) parameters of the case 4 x 3 x 2

Activity (b)	Job (l)			
	$l_1 = 1$	$l_2 = 2$	$l_3 = 3$	$l_4 = 4$
a_s = s (setup)	60	57	20	26
a_u = u (unloading)	17	12	10	20

Table A.2. Machining/Processing Time (P_{il}) Parameters of the Case 4 x 3 x 2

From Machine (i)	To Job (l)			
	$l_1 = 1$	$l_2 = 2$	$l_3 = 3$	$l_4 = 4$
$i_1 = 1$	46	60	95	27
$i_2 = 2$	45	49	42	30
$i_3 = 3$	36	25	50	48

Table A.3. Moving time (V_{hi}) parameters of the case 4 x 3 x 2

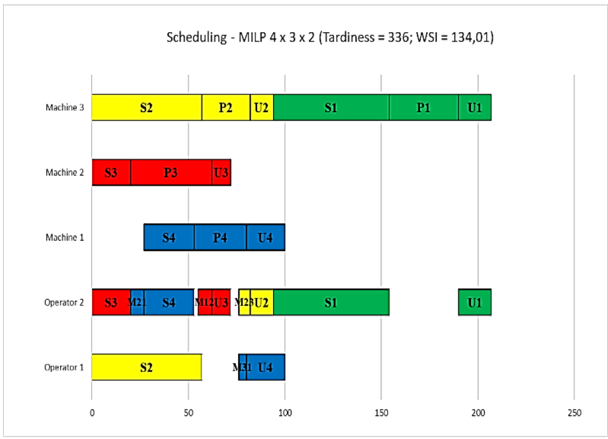
From Machine (h)	To Machine (i)		
	$i1 = 1$	$i2 = 2$	$i3 = 3$
$h1 = 1$	0	7	4
$h2 = 2$	7	0	6
$h3 = 3$	4	6	0

Table A.4. Due date time (d_l) parameters of the case 4 x 3 x 2

Due Date of Job (*dl*)			
$l_1 = 1$	$l_2 = 2$	$l_3 = 3$	$l_4 = 4$
30	48	29	30

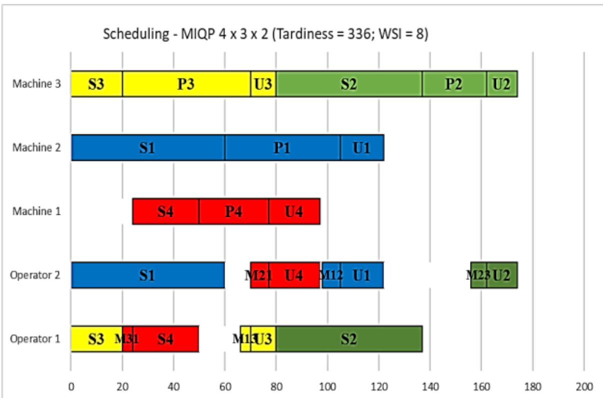
The output obtained is as follows

```
Variable                x
-------------------------
x_1_3_s_2_1_u_4             1
x_1_3_u_0_3_s_2             1
x_2_1_s_4_2_u_3             1
x_2_2_s_3_1_s_4             1
x_2_2_u_0_2_s_3             1
x_2_2_u_3_3_u_2             1
x_2_3_s_1_3_u_1             1
x_2_3_u_2_3_s_1             1
       q_1_2             1
       q_1_3             1
operationcompletion_s_1          154
operationcompletion_s_2           57
operationcompletion_s_3           20
operationcompletion_s_4           53
operationcompletion_u_1          207
operationcompletion_u_2           94
operationcompletion_u_3           72
operationcompletion_u_4          100
jobprocesscompletion_1          190
jobprocesscompletion_2           82
jobprocesscompletion_3           62
jobprocesscompletion_4           80
movingcompletion_s_1            94
movingcompletion_s_4            27
movingcompletion_u_1           190
movingcompletion_u_2            82
movingcompletion_u_3            62
movingcompletion_u_4            80
 Tardiness_1           177
 Tardiness_2            46
 Tardiness_3            43
 Tardiness_4            70
   tardiness           336
  busytime_1            81
  busytime_2           165
 busiesttime       168.697
         WSI       17959.8
```



When considering Tardiness as a constraint, the Tardiness value is set to 336 time units.
Thus, we obtain:

```
Variable                x
-------------------------
x_1_1_s_4_3_u_3             1
x_1_3_s_3_1_s_4             1
x_1_3_u_0_3_s_3             1
x_1_3_u_3_3_s_2             1
x_2_1_u_4_2_u_1             1
x_2_2_s_1_1_u_4             1
x_2_2_u_0_2_s_1             1
x_2_2_u_1_3_u_2             1
       q_1_3             1
       q_2_3             1
operationcompletion_s_1           60
operationcompletion_s_2          137
operationcompletion_s_3           20
operationcompletion_s_4           50
operationcompletion_u_1          122
operationcompletion_u_2          174
operationcompletion_u_3           80
operationcompletion_u_4           97
jobprocesscompletion_1          105
jobprocesscompletion_2          162
jobprocesscompletion_3           70
jobprocesscompletion_4           77
movingcompletion_s_2            80
movingcompletion_s_4            24
movingcompletion_u_1           105
movingcompletion_u_2           162
movingcompletion_u_3            70
movingcompletion_u_4            77
 Tardiness_1            92
 Tardiness_2           126
 Tardiness_3            51
 Tardiness_4            67
   tardiness           336
  busytime_1           121
  busytime_2           129
 busiesttime           129
         WSI            64
```



If considering WSI as a constraint:

$$WSI = \sqrt{\sum_{k \in K} (b_k - b_{max})^2}$$

With the WSI value set to 4 time units. Thus, we obtain:

```
Variable              x
-------------------------
x_1_1_s_1_2_u_4            1
x_1_1_u_0_1_s_1            1
x_1_1_u_1_3_u_2            1
x_1_2_u_4_1_u_1            1
x_2_2_s_4_3_u_3            1
x_2_3_s_3_2_s_4            1
x_2_3_u_0_3_s_3            1
x_2_3_u_3_3_s_2            1
       q_1_3            1
       q_2_3            1
operationcompletion_s_1          63
operationcompletion_s_2         137
operationcompletion_s_3          20
operationcompletion_s_4          52
operationcompletion_u_1         126
operationcompletion_u_2         174
operationcompletion_u_3          80
operationcompletion_u_4         102
jobprocesscompletion_1         109
jobprocesscompletion_2         162
jobprocesscompletion_3          70
jobprocesscompletion_4             82
movingcompletion_s_1             3
movingcompletion_s_2            80
movingcompletion_s_4            26
movingcompletion_u_1           109
movingcompletion_u_2           162
movingcompletion_u_3            70
movingcompletion_u_4            82
 Tardiness_1             96
 Tardiness_2            126
 Tardiness_3             51
 Tardiness_4             72
   tardiness            345
  busytime_1            127
  busytime_2            125
 busiesttime            127
         WSI              4
```