INTERNATIONAL JOURNAL OF INNOVATION IN ENTERPRISE SYSTEM

Identical Parallel Machine Scheduling to Minimize Makespan Using Suggested Algorithm Method at XYZ Company

Naura Maisazahra[1*], Murni Dwi Astuti[2], Murman Dwi Prasetio[3]

[1] School of Industrial and System Engineering
Telkom University, Bandung, Indonesia
nauramaisazahra@student.telkomuniversity.ac.id

[2] School of Industrial and System Engineering
Telkom University, Bandung, Indonesia
murnidwiastuti@telkomuniversity.ac.id

[3] School of Industrial and System Engineering
Telkom University, Bandung, Indonesia
murmandwi@telkomuniversity.ac.id

*nauramaisazahra@student.telkomuniversity.ac.id

ARTICLE INFO	ABSTRACT
	XYZ company produce the various shape of motor spare parts product. The company has three identical parallel spot welding machines that use a random method of production scheduling, based on machine capacity without any sequence of jobs, and only use daily production targets given to operators. Based on the data, the actual scheduling of the machines has a very large completion time difference between each machine, or the machine loading is uneven. As a result, the makespan becomes longer with a value of 440000 seconds (26 days). This research aims to minimize the existing makespan by giving proposed scheduling, using the suggested algorithm method, which has a small number of iterations and has an optimal result. The method begins with the longest processing time sequence rule which is used as the upper bound for the first iteration, then continued to calculate the lower bound and machine workload. The calculation stops at the 15th iteration because the completion time value exceeds the lower and upper bound so that the optimal scheduling taken is scheduled in the 14th iteration with a makespan value of 914412 seconds (16 days). The proposed scheduling can minimize the makespan from the actual schedule by 38%.

1. INTRODUCTION

Scheduling is the allocation of resources needed or used from time to time to support implementation in completing a specific activity or job [2]. If the scheduling system is not good, it can make the company profit loss. Scheduling is deciding activity compliance and resource to solve a group of jobs just in time and have their quality [4]. If the quality of the company is good, the company can win from the competitiveness and can fulfill the consumer's needed [5]. XYZ company is an example of a manufacturing company that produces motorcycle spare parts made of iron or steel. XYZ company has several types of machines used to produce consumer demand, one of them is spot welding machines.

In actual scheduling production by comparing the number of jobs and machines, they have quite a far gap makespan. It is a needed total time to finish all jobs or a completion time [3]. Spot welding machines are in the category of identical parallel machines because they have the same characteristics. An identical parallel machine is an m-parallel machine with identical characteristics that can complete every n job with the same speed so that the processing time of each job on any machine is the same [1]. In universal, this case can be expressed for m-parallel machines, where n jobs arrive in several periods (horizon) at dissimilar time points for processing. The spot welding machine has 16 jobs and 3 identical machines. The amount of demand that must be finished by spot welding machines always fluctuates every month. At the beginning of the month, XYZ company always receives orders from customers and must complete them in approximately 1 month, the due dates of a job are different but fall on the 4th week at the month customers give the amount of order. There are gaps in the number of demand and production output.

In carrying out the production process, there are some latenesses, which are with negative value (earliness) and positive value (tardiness). On spot welding machine, there are 16 types of jobs are done, with total tardiness 8 jobs, and 8 jobs are earliness. This delay also occurred due to external factors, such as the manpower factor, engine conditions, and environmental conditions due to the Covid-19 virus. From the due date that has been set by customers, several jobs are late. Apart from many other factors, the main problem of this delay occurs by the cumulative load of the spot-welding machine is uneven, so the makespan value becomes long. Fig. 1 shows the Gantt chart from the actual scheduling in XYZ company.
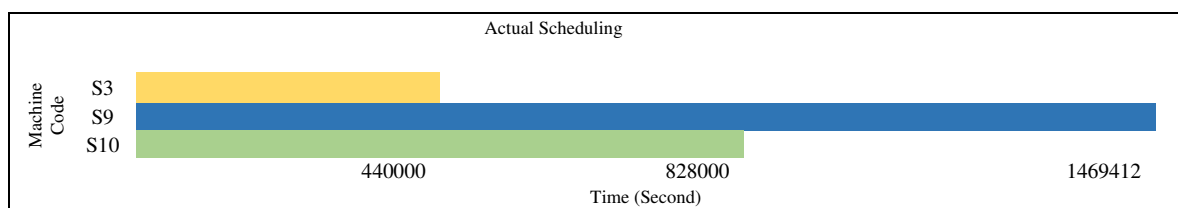

Figure 1 - Gantt Chart Actual Scheduling Spot Welding Machines

From that figure, there are 3 machines with the code are S3, S9, and S10. S3 has a completion time of 440000 seconds, then S9 is 1469412 seconds, and S10 is 828000 seconds. From that, so the maximum completion time (makespan) of the spot-welding machines is 1469412 seconds or 26 days. The actual scheduling condition of the company is still using the random method, where XYZ company scheduling only estimated the capacity of the machine, and when scheduling is given to the field it is only a daily production target that must be produced from each job, and there is no priority sequence. The characteristics of the three machines are identical so that all jobs actually can be scheduled anywhere from the three machines, therefore scheduling should be able to balance the workload of each machine, but the actual scheduling used by the company scheduling does not balance the workload of each machine, so the maximum completion time (makespan) becomes long.

When the production is less, the company must make additional work shifts or do sub-contracts which will incur additional costs. If doing additional work shifts, the company must pay operator overtime costs and then additional electricity financing for machines. In this study, we will discuss the proposed production scheduling for spot welding machines of XYZ company to minimize the makespan with ideal conditions and use the demand at the calculation. Scheduling has criteria based on time, cost, and a combination of the two, namely the first to minimize the makespan, and the second is to fulfill the due date. One of the goals of scheduling is to reduce production completion and to make arrangements for scheduling work so that can be completed on time [6]. There are several methods to determine operation priority sequence, one of them is the longest processing time [6]. This research will schedule 16 jobs on 3 identical machines with priority rules using the longest processing time (LPT) which prioritizes jobs that have the longest to shortest processing time, and LPT can be used to be a scheduling sequence rule that has the goal of minimizing makespan [7]. Minimization of the number of makespans is a base problem and a studied problem in scheduling. One of the proposed to achieve optimal scheduling is to minimize the total completion time (makespan) [8]. LPT method can be used for parallel machine scheduling [9]. LPT is a job that has a longer completion time will be processed first [10]. LPT rule can be useful for allocating jobs between m parallel machines to minimize the makespan (Cmax) [11]. The result of the LPT calculation will be the upper bound in the suggested algorithm method. The suggested algorithm is improved by the LPT algorithm with two types of operations, namely construction and backtracking [12]. The suggested algorithm is an improved method of LPT-algorithm, which is used as the main algorithm [13]. The formulation of the mathematical model in this method is used the integer linear programming (ILP) model and the longest processing time algorithm to find the initial solution. This algorithm is used to improve the initial solution in the problem-solving approach, the results show the suggested algorithm modeling is an effective algorithm compared to traditional methods[14]. In the application of the suggested algorithm method, there is a lexicographic model that uses to count residuals and sort data. In sorting jobs based on LPT rules, lexicographic order plays a role in sorting data from large to small. The loading machine will be determined sequentially, one by one, the potential load machine (i) depends on the previous load machine, whit

notation 1< i ≤ m [12]. Then the loading of the machine is continued until the last job. In this study, we will discuss the proposed production scheduling in the spot-welding area of XYZ company to minimize the makespan using the suggested algorithm method.

2. METHOD

The methodology is useful for describing the stages or steps taken to solve problems. Figure 2 explains the flow of the processes from the first step until the purpose of this research is achieved.
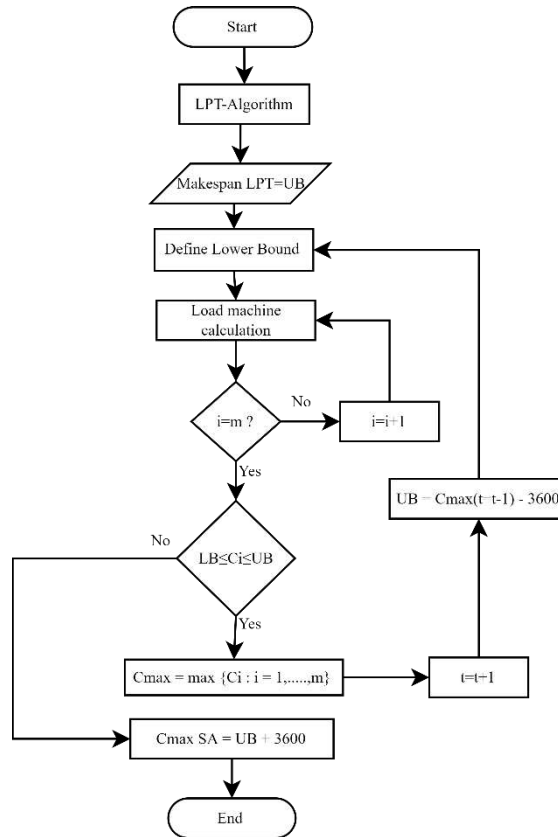

Figure 2 - Research Methodology

There are two types of operations in the suggested algorithm method, namely:
1. Construction phase, this phase is sorted in lexicographical order or commonly called dictionary order or alphabetical order. In the construction phase, it loads the machines one by one, so the machines did not have the same workload as the previous one. If the construction phase is fortunate (all machines and all conditions are fulfilled), the upper and lower bounds will be updated.
2. Backtracking phase, for researching the algorithm, it will always look for the optimal solution, therefore backtracking or backward phase is put it whenever one of the following situations is taken into calculating:
   a. When the load machine is not feasible (the value is not between the upper bound and lower bound)
   b. When a new feasible solution is found for all machines. When the engine no longer has a decent load. So, there is no feasible solution for the upper bound, and the optimal makespan is the same as the upper bound.

Step of suggested algorithm method:
1. Set i=1, and t=1.
2. Determine the upper bound, it took from the LPT rule, by finding the makespan value of LPT.
3. Calculate the lower bound with the formula:

$$LB = \max\left\{0, \sum_{j=1}^{N} p_j - (m-1)UB\right\} \qquad (1)$$

4. Calculate load machine to determine job allocation (j) on the machine (i=1,...,m), by iterating with the formula

$$k_1 = \min\left\{\left|\frac{UB}{p_1}\right|, r_1\right\} \tag{2}$$

$$k_j = \min\left\{\left|\frac{UB - \sum_{h=1}^{j-1} p_h k_h}{p_j}\right|, r_j\right\}, j = 1, \ldots, n \tag{3}$$

5. Calculate the completion time for each machine, if one of the completion times (C_i) machine is between LB≤C_i≤UB, then proceed continuously to step 6, but if the C_i value is outside the bounds, the iteration is declared complete or finished.
6. Calculate Cmax
7. Continue to the next iteration, with the value Cmax$_{t=t-1}$ as UB. Then, repeat to step 3.
   With description:
   i   = machine to-i
   m   = amount of machine
   t   = iteration to-t
   LB   = lower bound
   UB   = upper bound
   j   = job
   N   = amount of job
   p   = processing time
   r_j = residual on job
   k   = number of jobs scheduled

First, it starts with inputting data, which consists of four input data, which are demand, cycle time, number of jobs, and number of machines. In the processing data step, the data that becomes the reference for calculations for comparison to minimize makespan is the data in Table 2. There is a mathematical model to minimize makespan [12]:

$$C_{max} = \sum_{i=1}^{n} t_j \tag{4}$$

Where:

C_{max} = Makespan on n job (maximum completion time)
t_j = Processing time
n   = Number of jobs
m   = Number of machines
X_{ij} = The assignment (decision) variable

$$\text{Min } C_{max} = \sum_{j=1}^{n} t_j X_{ij} \leq C_{max} \quad i = 1, \ldots, m \tag{5}$$

$$\sum_{i=1}^{m} x_{ij}, j = 1, \ldots, n$$

$$x_{ij} \in \{0,1\} \; i = 1, \ldots, m, \qquad j = 1, \ldots, n$$

$$\text{variabel } (X_{ij}):$$

$$X_{ij} = \begin{cases} 1 & \text{if the job j is assigned to machine i} \\ 0 & \text{if the job j is not assigned to machine i} \end{cases}$$

The number of machines will be a boundary in scheduling, there are 3 machines and will finish 16 jobs to be scheduled. The demand and cycle time of each job has different numbers, the multiplication of demand and cycle time produces the processing time of each existing job. All of that data will be calculated by the suggested algorithm method. This calculation used "second", it is used to avoid decimal numbers to minimize the error rate on the program. Then, sort the processing time for each job from the longest to the shortest time, this rule is also called the longest processing time, from that sequence then determine the residual number (r), residual is the number of job multiples that have the same processing time, if two jobs have the same processing time, then the initial job becomes one and the residual value of the

new initial job is two, but if the job itself is different and has nothing in common with other jobs, then the residual job is one. To get the output sequence of processing time and residual numbers, the process is carried out in a lexicographic programming model.

After that, do the scheduling by allocating jobs on each machine by prioritizing the jobs that have the largest processing time, then calculate the makespan (C_{max}^{LPT}). There are several steps in the longest processing time (LPT):

1. Sort N-jobs by longest to shortest processing time.
2. Set (j=1)
3. Enter or schedule job (j) on the machine (i).
4. If all jobs have been scheduled, proceed to the next stage, if not, return to step 3 (j = j + 1).
5. Calculate C_{max}^{LPT} with formula,

$$C_{max} = max\{C_i : i = 1, ..., m\} \quad (6)$$

Next is do the calculations for the first machine (i=1), and first iteration (t=1), then determine the upper bound for the first iteration, where upper bound$_{t=1}$=C_{max}^{LPT}. Next, determine the lower bound with the formula (1). After the Lower Bound (LB) and Upper Bound (UB) are obtained, then the load machine calculation is carried out with (2) and (3) formulas, in the process do the calculations for i = 1 until getting the proposed job allocation, then do the calculation for i = i + 1, if i = m finds the maximum completion time (makespan) form the three machines.

Make sure the completion time (C_i) of each iteration is still between the lower bound and upper bound (LB≤Ci≤UB), if the completion times did not pass the boundaries, then it is continued by looking for the value of makespan (Cmax) in that iteration, then the iteration is continued (t=t+1), with the upper bound value using Cmax$_{t=1}$ minus 3600 (because in seconds, if in hours then minus one), from the new upper bound value, do the calculation again to find the lower bound value, and repeat as the previous process, until we find the result of Cmax$_{t=t}$ is outside the lower bound and upper bound, so the iteration is declared complete or finished.

If in the iteration-t there is C_i with the value are outside the lower bound and upper bound, then the selected scheduling is from Cmax$_{t=last iteration-1}$, this is obtained from UB in the last iteration plus 3600 which is the same value as the Cmax of the previous iteration, so the scheduling of each job is taken from the scheduling of n-iteration minus one. After getting the optimal proposed scheduling, it will be visualized using a Gantt chart. Gantt chart is a planning diagram that has a function to describe resource scheduling and time allocation [15]. Gantt chart also has a function to make it easier to see the start time and end time of a sub-task in the project [16]. This is an example of a Gantt chart.
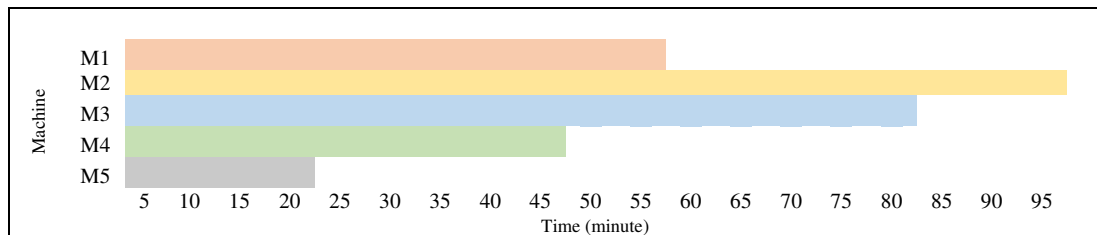

Figure 3 - Gantt Chart

This Gantt chart will show where the jobs will be scheduled, and then show the completion time of every machine. In this research, a Gantt chart is used to visualize the scheduling, because it's easier to see and understand by the reader, so the output from the program is visualized by the Gantt chart. The horizontal side will give the number of the time, and the vertical is the code of machines. This research data processing used PyCharm software with python programming language.

2.1. Data Collection

The list of machines in this research is shown in Table 1. There are 3 identical parallel machines with the same characteristics and speed.

Table 1 - List of Machines

No	Machine Name	Machine Code	Machine Initial
1	Spot Welding Machine 1	S3	C_1
2	Spot Welding Machine 2	S9	C_2
3	Spot Welding Machine 3	S10	C_3

Table 2 explains the detailed data for 16 jobs with each job code from J01S until J16S, which must be finished by the spot welding machine, and it will be input data for scheduling.

Table 2 - Detailed Data of Jobs

Job	Demand	Cycle Time	Actual Scheduling	Processing Time (second)
J01S	10000	12	S3	120000
J02S	10000	16	S3	160000
J03S	5000	8	S3	40000
J04S	5000	8	S3	40000
J05S	5000	8	S3	40000
J06S	5000	8	S3	40000
J07S	52000	9	S10	468000
J08S	50000	5	S10	250000
J09S	3000	15	S9	45000
J10S	5000	16	S9	80000
J11S	5000	16	S10	80000
J12S	33701	12	S9	404412
J13S	47000	10	S9	470000
J14S	47000	10	S9	470000
J15S	1000	12	S10	12000
J16S	1500	12	S10	18000

2.2. Data Processing
Data processing used the python programming language with PyCharm software. The calculations below are an example of the calculation stages.

Step 1: Sequencing processing time used the longest processing time rule and determines the residual number. In data processing time calculation is used the formula:

$$p_j = D \times Ws$$

Table 3 - Processing Time Sequence with LPT Rules and Residual Numbers

Initial job (J_h)	Job	Processing Time	Residual (r)
J1	J13S	470000	2
	J14S	470000	
J2	J07S	468000	1
J3	J12S	404412	1
J4	J08S	250000	1
J5	J02S	160000	1
J6	J01S	120000	1
J7	J10S	80000	2
	J11S	80000	
J8	J09S	45000	1
J9	J03S	40000	4
	J04S	40000	
	J05S	40000	
	J06S	40000	
J10	J16S	18000	1
J11	J15S	12000	1

Step 2: Calculate the makespan value of the proposed scheduling using the LPT sequencing rule.
Set j=1 (for first machine), then
Set j=j+1(for the next machine)
j   =1 (S3)
C_1= J14S+J08S+J10S+J03S+J06S
   =470000+250000+80000+40000+40000 = 880000

j   =2 (S9)

C_2= J13S+J012S+J012S+J09S+J05S+J15S
=470000+160000+120000+45000+40000+12000 = 847000

j =3 (S10)

C_3=J07S+J12S+J11S+J04S+J16S
=468000+404412+80000+40000+18000 = 1010412

Step 3: Calculate makespan LPT with (6) formula

$C_{max}^{LPT} = max\{880000,847000,1010412\}$

$C_{max}^{LPT} = 1010412$

Then the makespan value obtained with the Longest Processing Time (LPT) rule is 1010412 seconds.

Step 4: set i=1 (first machine) and t=1 (first iteration).

Step 5: Determine upper bound for the first iteration

Upper Bound$_{t=1}$ = C_{max}^{LPT}

UB$_{t=1}$ = 1010412

Step 6: Calculate lower bound

$$LB = max\left\{0, \sum_{j=1}^{N} p_j - (m-1)UB\right\}$$

$$LB = max\left\{0, \sum_{j=1}^{16} p_j - (3-1)1010412\right\}$$

$LB = max\{0, 2737412 - (2 \times 1010412)\}$

$LB = max\{0, 2737412 - (2020824)\}$

$LB = max\{0, 716588\}$

$LB = 716588$

Step 7: Calculate load machine

$$k_1 = min\left\{\left|\frac{UB}{p_1}\right|, r_1\right\}$$

$$k_j = min\left\{\left|\frac{UB - \sum_{h=1}^{j-1} p_h k_h}{p_j}\right|, r_j\right\}, j = 1, \ldots, n$$

1stIteration, 1st machine

$$k_1 = min\left\{\left|\frac{UB}{p_1}\right|, r_1\right\}$$

$$k_1 = min\left\{\left|\frac{1010412}{470000}\right|, 2\right\}$$

$k_1 = min\{|2.149|, 2\}$

$\mathbf{k_1 = 2}$

After k_1 then next to k_2

$$k_j = \min\left\{\left|\frac{1010412 - \sum_{h=1}^{j-1} p_h k_h}{p_j}\right|, r_j\right\}, j = 1, \ldots, n$$

$$k_2 = \min\left\{\left|\frac{1010412 - p_1 k_1}{p_2}\right|, 1\right\}, j = 2$$

$$k_2 = \min\left\{\left|\frac{1010412 - 470000 \times 2}{468000}\right|, 1\right\}, j = 2$$

$$k_2 = \min\left\{\left|\frac{1010412 - 940000}{468000}\right|, 1\right\}, j = 2$$

$$k_2 = \min\left\{\left|\frac{70412}{468000}\right|, 1\right\}, j = 2$$

$$k_2 = \min\{0.150, 1\}, j = 2$$

$$\mathbf{k_2 = 0}$$

Then, continue for all initial jobs, repeat step 4 with set i = 2 (2nd machine), t = 1 then do steps 5-7, until the 3rd machine is completed. Here are the results:

C_1 = J13S, J14S, J09S, J16S = 1003000

C_2 = J07S, J12S, J01S, J15S = 1004412

C_3 = J08S, J02S, J10S, J11S, J03S, J04S, J05S, J06S = 730000

Then, next to step-8

Step 8: Look $C_i^{t=1}$ there were on boundary or not

$$LB \leq C_i^{t=1} \leq UB$$

$$716588 \leq 1003000, 1004412, 1004412 \leq 1010412$$

Because $C_i^{t=1}$ is still between the boundary, so the iteration is continued (t=t+1). The next step is to repeat step 4 and the next step until there was the value of C_i is outside the bounds of LB and UB, so the iteration is finished.

3. RESULT AND DISCUSSION

The result from PyCharm software with the suggested algorithm method are the job sequence of scheduling, completion time number of each machine, and the makespan. This method stopped in the 15th iteration, with the makespan being 914412 seconds. The detailed scheduling is shown in Table 4.

Table 4 - The Output of Proposed Scheduling with Suggested Algorithm

Proposal Scheduling with Suggested Algorithm
Amount of iteration: 15
Cmax SA: 914412.0
C1: 914412.0   CJob1: ['J13S', 'J12S', 'J03S']
C2: 910000.0   CJob2: ['J14S', 'J08S', 'J02S', 'J16S', 'J15S']
C3: 913000.0   CJob3: ['J07S', 'J01S', 'J10S', 'J11S', 'J09S', 'J04S', 'J05S', 'J06S']

Except for that data, the output of scheduling is visualized by the Gantt chart with different colors for each job. This Gantt chart is as shown in Figure 5.
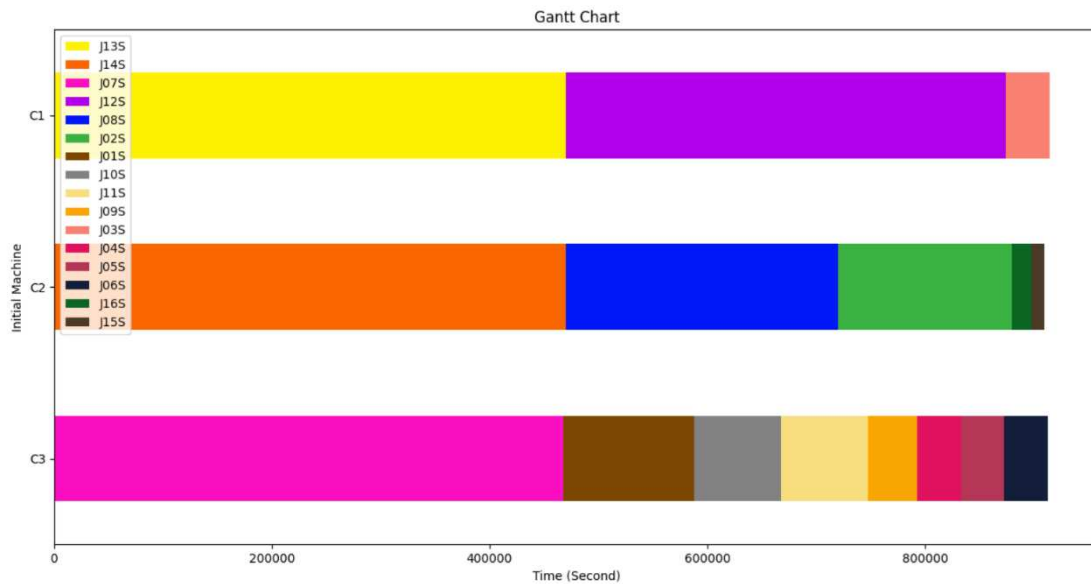
Figure 5 - Proposed Scheduling Gantt Chart

The results of data processing using PyCharm ended in the 15th iteration because the value of Ci is not between the boundary (LB ≤ C_i ≤ UB), and the value of the lower bound is more than the upper bound (LB ≥ UB). Because of that, the iteration is ended. The optimal scheduling is in the 14th iteration. The makespan of the proposed scheduling is 914412 seconds, with a total load of machine 1 (S3) is 914412 seconds, machine 2 (S9) is 910000 seconds, and load machine 3 (S10) is 913000 seconds.

Next, we will compare the actual scheduling with the proposed scheduling. First of all, the actual scheduling is does not have sequence priority jobs in the production schedule, while the proposed scheduling has sequence priority that must be finished first, then the other differences are in the completion time of each machine and job allocation, it is shown in Table 5 and Table 6.

Table 5 - Actual Scheduling

Machine	Actual Scheduling (Without order priority)	Machine Completion Time		
		Second	Hour	Day
C_1 (S3)	J01S, J02S, J03S, J04S, J05S, J06S	440000	122.222	8
C_2 (S9)	J09S, J10S, J12S, J13S, J14S	1469412	408.17	26
C_3 (S10)	J07S, J08S, J11S, J15S, J16S	828000	230	15

Table 6 - Proposed Scheduling

Machine	Actual Scheduling (With order priority)	Machine Completion Time		
		Second	Hour	Day
C_1 (S3)	J13S, J12S, J03S	914412	254.003	16
C_2 (S9)	J14S, J08S, J02S, J16S, J15S	910000	252.778	16
C_3 (S10)	J07S, J01S, J10S, J11S, J09S, J04S, J05S, J06S	913000	253.611	16

Based on Table 5 and Table 6, there are differences in completion time between the actual condition and the proposed condition. Table 7 is illustrated the percentage comparison of the completion times or load machines of each machine by calculating the completion time value of one of the machines by the total completion time of the three machines, then multiplied by 100%.

Table 7 - Comparison Completion Time or Work Load inter-Machine

Machine	Actual	Proposed
C_1 (S3)	16.07%	33.40%
C_2 (S9)	53.68%	33.24%
C_3 (S10)	30.25%	33.35%

In Table 7, the completion time of all machines in actual conditions has a large difference, the machine S9 has the largest percentage of the longest completion time from other machines, this shows that the job distribution in actual conditions is uneven for all machines. The completion time of all machines can be said balanced when the percentage of completion time or load machine does not have a large gap. In actual conditions, the difference between the S3 machine and the S9 machine is 37.61%, while in the proposed condition the difference between them is only 0.16%. The balance completion time or load machine of each machine can make the maximum completion time or makespan is short. It can be seen in Table 8 the comparison between the actual scheduling makespan with the proposed method.

Table 8 - Comparison Makespan Between Actual and Proposed Scheduling

Type of Scheduling	Cmax		
	Second	Hour	Day
Actual	1469412	408.17	26
Proposed	914412	254.003	16

The makespan value of the actual condition is 26 days, while the makespan with the proposed scheduling is 16 days, From the proposed scheduling with the suggested algorithm method, it can minimize the makespan of 38% from the actual scheduling.

4. CONCLUSION

Based on the research that has been done, there are problems in the length of time for completion or makespan on the spot-welding machine at XYZ company, this job scheduling is still used a random method without any work order. From these problems, a suggestion was made regarding the scheduling of spot-welding machines at XYZ company to minimize makespan using the suggested algorithm method. The proposed scheduling gives more optimal results where the operator knows the job to be done first, and by using the suggested algorithm method can balance the completion time of each machine. The iteration is carried out using the suggested algorithm method, namely stopping at the 15th iteration with the results of the makespan value of 16 days, which can minimize the makespan from the actual condition by 38%. With this research, there were many advantages for companies, the production schedule has a priority sequence, and by using this scheduling, the makespan will be minimized, and also the cost.

Disclaimer
The authors whose names are written certify that they have no conflict of interest.

REFERENCES

[1] H. Suwa and H. Sandoh, Online Scheduling in Manufacturing A Cumulative Delay Approach. Osaka: Springer, 2013.

[2] N. I. Lesmana, "Penjadwalan Produksi Untuk Meminimalkan Waktu Produksi Dengan Menggunakan Metode Branch and Bound," J. Tek. Ind., vol. 17, no. 1, p. 42, 2017

[3] D. M. Utama, "Algoritma LPT-Branch and Bound Pada Penjadwalan Flexible Flowshop untuk Meminimasi Makespan," PROZIMA (Productivity, Optim. Manuf. Syst. Eng., vol. 2, no. 1, p. 20, 2019

[4] H. Wibowo, "Penjadwalan Mesin Screw Press Stasiun Kempa Pada Produksi Cpo (Crude Palm Oil) Dan Kernel Dengan Menggunakan Metode Indikator," Spektrum Ind., vol. 14, no. 1, p. 45, 2016

[5] M. D. Prasetio, "An Approaching Machine Learning Model: Tile Inspection Case Study," Int. J. Innov. Enterp. Syst., vol. 4, no. 01, pp. 12–22, 2020

[6] V. Gaspersz, Production Planning and Inventory Control Berdasarkan Pendekatan Sistem Terintegrasi MRP II dan JIT menuju Manufakturing 21. Jakarta: Gramedia Pustaka Utama, 2005.

[7] R. Ginting, Penjadwalan Mesin. Yogyakarta: Graha Ilmu, 2009.

[8] D. Setiya Widodo, P. Budi Santoso, and E. Siswanto, "Pendekatan Algoritma Cross Entropy-Genetic Algorithm Untuk Menurunkan Makespan Pada Penjadwalan Flow Shop," J. Eng. Manag. Industial Syst., vol. 2, no. 1, pp. 41–49, 2014

[9] Y. Muharni, E. Febianti, and N. N. Sofa, "Minimasi Makespan Pada Penjadwalan Flow Shop Mesin Paralel Produk Steel Bridge B-60 Menggunakan Metode Longest Processing Time Dan Particle Swarm Optimization," J. Ind. Serv., vol. 4, no. 2, 2019, doi: 10.36055/jiss.v4i2.5154.

[10] R. I. Safitri, "Analisis Sistem Penjadwalan Produksi Berdasarkan Pesanan Pelanggan dengan Metode FCFS, LPT, SPT dan EDD Pada PD. X," J. Optimasi Tek. Ind., vol. 1, no. 2, p. 26, 2019

[11] Y. Mauergauz, Advanced Planning and Scheduling in Manufacturing and Supply Chains. Switzerland: Springer, 2016.

[12] Y. German, I. Badi, A. Bakir, and A. Shetwan, "Scheduling to Minimize Makespan on Identical Parallel Machines," Int. J. Sci. Eng. Res., vol. 7, no. 3, 2016.

[13] M. H. Riaja, P. P. Suryadhini, and A. Oktafiani, "PENJADWALAN IDENTICAL PARALELL MACHINE MENGGUNAKAN METODE SUGGESTED ALGORITHM DAN BRANCH AND BOUND UNTUK MEMINIMASI MAKESPAN PADA PROSES INJECTION MOLDING DI CV. GRADIENT," e-Proceeding Eng., vol. 6, pp. 6565–6572, 2019.

[14] H. Habiba, A. Hassam, Z. Sari, C. M. Amine, and T. Souad, "Minimizing Makespan on Identical Parallel Machines," Proc. - 2019 3rd Int. Conf. Appl. Autom. Ind. Diagnostics, ICAAID 2019, no. September, pp. 25–27, 2019

[15] M. Widyastuti, E. Irawan, and A. P. Windarto, "Penerapan Metode Gantt Chart dalam Menentukan Penjadwalan Kinerja Karyawan," Pros. Semin. Nas. Ris. Inf. Sci., vol. 1, no. September, p. 557, 2019

[16] T. D. SANTOSO, D. FAUJI, and R. KURNIAWAN, "Metode Cpm & Gantt Chart Untuk Penjadwalan Dan Analisis Aktivitas Kritis Pada Proyek Pembangunan Perumahan Griya Utama …," no. 2017, 2020.