ISSN: 2456-3676

A Mathematical Model for Simultaneous Parallel Multiple Source Scheduling Problems

Felia Indah Kusuma[1], ParwadiMoengin[2]

[1]Department of Industrial Engineering, Trisakti University
Jln. Kyai Tapa No. 1 Jakarta 11440, Indonesia
felia@gmail.com

[2]Department of Industrial Engineering, Trisakti University
Jln. Kyai Tapa No. 1 Jakarta 11440, Indonesia
parwadi@trisakti.ac.id (corresponding author)

doi: 10.51505/ijaemr.2022.8104 URL: http://dx.doi.org/10.51505/ijaemr.2022.8104

Abstract

Manufacturing companies engaged in textile manufacturing engaged in the Job Order system with an extensive net work and market orientation require companies to pay attention to product quality and services that meet consumer requirements, one of which is by fulfilling orders at a pre-agreed time. The problem that is often found in companies like this is the frequent delays in completing orders. This is caused by the less optimal scheduling of job orders with different weights or penalties for each job which will ultimately result in a large total weighted delay originating from the accumulation of job delays with these different weights. This paper is intended to help companies create a new, more optimal schedule with the aim of minimizing the total weighted delay of order completion.

The model proposed in this paper is a mathematical model for simultaneous parallel multiple-source scheduling problems for multiple-jobs and multiple-operations. The initial step of using this model is the formulation of decision variables, the objective function is minimizing tardiness or total weighted delay, and limiting functions. After the jobs to be examined are formulated into a model, data processing is carried out using a computerized system. The software used in processing this data is Win QSB. At the end of the paper a numerical example is given to use the proposed model. From data processing using Win QSB, it can be seen how much total weighted delay is obtained from scheduling with an integer linear programming model for this scheduling.

Keywords: manufacturing, mathematical models, job orders

1. Introduction

Every manufacturing company engaged in a job order system with an extensive network realizes the importance of services that meet consumer requirements to win market competition, scheduling is an important element in fulfilling orders (orders) on time. Good scheduling can minimize delays and penalty risks due to delays, provide satisfaction and maintain customer loyalty, and can bring companies to competitive advantage. This delay will have a negative

Vol. 8, No. 01; 2023

ISSN: 2456-3676

impact in the form of a penalty that must be borne by the company. How to overcome the less than optimal scheduling of jobs with different weights for each job which can result in a large total weighted delay for the company. The purpose of this paper is to design a new, more optimal scheduling model using a mathematical model to solve the simultaneous multi-source multiple-source scheduling problem for multiple-operation multiple-jobs with the aim of minimizing the total weighted delay of order completion.

2. A Mathematical Model

2.1 Definition of Model

This model addresses the simultaneous parallel multi-resource scheduling problem as introduced by Kerzner (1995). Parallel means that the same type of source is used for different jobs at the same time, the execution of each job requires several sources simultaneously (simultaneously) using a mathematical model (Pinedo & Chao, 1999; Sipper & Bulfin, 1997). The term source is used to replace the term machine which is often found in scheduling theory. Each source can only perform one operation at a time and preemption is not allowed, which means an operation cannot be interrupted (Bodington, 1995).

2.2 Tardiness Weighted Objective Function

Weighted tardiness is a common objective function in scheduling problems. In a scheduling problem with a weighted tardiness objective function, each job has a due date and weight (importance). If the job's final completion time exceeds its due date, the job is subject to a penalty represented by its weight. Tardiness is defined as the time difference between the final settlement time and the due date if the final settlement time is greater than the due date. Total weighted tardiness is defined as the total of all delays from jobs with different weights (Suprayogi, et al, 2002).

2.3 Assumptions

Suppose there are N jobs and require H resource types. Each job i consists of N_i operations. Due date for each job is stated by D_i. Each operation j for job i requires processing time it. H_{ij} states the set of resources used to carry out operation j for job i. The time horizon is discretized in K time units. In k time slots, each source type h ($h \in H$) has M_{hk} available. The weight (importance) of each job i is expressed by w_i. The problem is to determine the work schedule for each operation for each job in order to obtain the minimum total weighted delay.

The dependency relationship of the operations for a job is assumed to form a directed acyclic graph, and each job is assumed to end with a single operation. Thus, the final completion time for each job C_i, is the same as the final completion time for the last operation in job i, namely $C_i = C_iN_i$.

Figure 1 shows an example of a process plan for a job. The execution of an operation is assumed to be non-preemptive (without interruption) so that a block of time with length t_{ij} is required to carry out operation j job i. The length of time for each operation on each source used is the same.

For each source type, each unit is assumed to be identical. All jobs are assumed to be available at time k = 1. The time horizon K is considered long enough to complete all jobs, $C_i \leq K$ for all i.

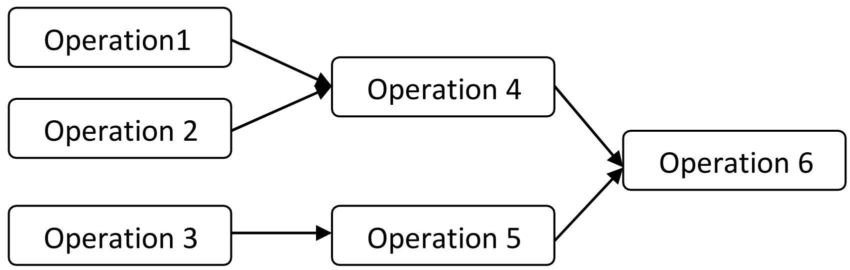

Figure 1: Example of a job process plan

2.4 Notations

The notations used in the formulation of this model are:

N	Job set
N_i	The set of operation son job i
P_{ij}	The set of all other operations that preceded the j operation on job i
H	Source type set
H_{ij}	The set of resource types used to perform operation j for job i ($H_{ij} \subseteq H, \quad H_{ij} \neq \emptyset$)
r_i	Release time job i
K	Length of planning time horizon
M_{hk}	A constant indicating the availability of h resources in k time slots (k = 1,…,K)
t_{ij}	The length of time for carrying out operation j for job I ($t_{ij} > 0$)
L	The tardiness of every job
E	Earliness of every job.
w_i	Job weight (importance) i ($w_i > 0$).
D_i	Due date job i ($D_i > 0$).
C_i	At the end of job i, $C_i = C_iN_i$, where $c_i\,N_i$ is the final completion time of the last operation on job i

b_{ij} At the beginning of the execution of operation j for job i

c_{ij} At the end of operation j for job i

b_{ijh} At the start of the execution of operation for job i on source h

c_{ijh} At the end of executing operation j for job i on source h

c_{ijhk} Binary variable 0-1; c_{ijhk} =1 if job i done at resource h is completed at time k, c_{ijhk} = 0, otherwise.

x_{ijhk} Binary variable 0-1; x_{ijhk} = 1 if operation j for job i uses resource in time slot k, x_{ijhk} = 0, otherwise

2.5 Model Formulation

The discrete time horizon integer programming model for the simultaneous parallel multiple-source scheduling problem with the objective function of minimizing the total weighted tardiness is as follows:

$$\text{Minimize} \quad Z = \sum_{i} w_i L_i \quad (1)$$

Subject to:

$$C_i - D_i - r_i - L_i + E_i = 0, \forall i \quad (2)$$

$$C_i = c_{iN_i}, \forall i \quad (3)$$

$$c_{ij} \geq c_{im} + t_{ij}, \forall i, j; j, m \in J_i, m \in P_{ij} \quad (4)$$

$$c_{ij} \geq t_{ij} + r_i, \forall i, j; j \in J_i \quad (5)$$

$$c_{ijh} = c_{ij}, \forall i, j, h; j \in J_i, h \in H_{ij} \quad (6)$$

$$c_{ijh} = \sum_{k} k c_{ijhk}, \forall i, j, h; j \in J_i, h \in H_{ij} \quad (7)$$

$$x_{ijhk} = \sum_{k \leq l \leq k + t_{ij} - 1} c_{ijhl}, \forall i, j, h; j \in J_i, h \in H_{ij} \quad (8)$$

$$\sum_{k} c_{ijhk} = 1, \forall i, j, h; j \in J_i, h \in H_{ij} \quad (9)$$

$$\sum_{i} \sum_{j \in J_i} x_{ijhk} \leq M_{hk}, \forall h, k; h \in H_{ij} \quad (10)$$

$$c_{ijhk} \in \{0,1\}, \forall h, k; j \in J_i, h \in H_{ij} \quad (11)$$

$$b_{ij} = c_{ij} - t_{ij} + 1, \forall i, j; j \in J_i \quad (12)$$

$$b_{ijh} = b_{ij}, \forall i, j, h, j \in J_i, h \in H_{ij} \quad (13)$$

Equation (1) is the objective function which minimizes total weighted tardiness. For the total weighted tardiness minimization objective function, the value of E is not taken into account

Vol. 8, No. 01; 2023

ISSN: 2456-3676

(assumed to be 0). Constraint (2) is a linear form of the tardiness objective function. Constraint (3) defines that the final completion time of each job is the same as the final completion time of the last operation on that job.

Constraint (4) is a limiting dependency relationship, namely the end time of an operation must be greater than or equal to the final time of completion of all preceding operations plus the length of time needed to carry out the operation. Constraint (5) guarantees that the final completion time for each job is always greater than or equal to the processing time.

Constraint (6) determines that the completion time for each operation is the same as the completion time for that operation at each source used. Equation (7) determines the final completion time of each operation on each source. In equation (7), because c_{ijhk} is equal to zero except for the final settlement time, the sum on the right-hand side is equal to k* times 1, where k* is the final completion time.

Equation (8) guarantees that if the operation is completed at time k, then between the time slots k- t_i +1 and k the execution of operation j on job i uses the source ($h \in H_{ij}$). Constraint (9) guarantees that each operation j for job i on each source h ($h \in H_i$) is completed only once throughout the time horizon. Constraint (10) is a limiting resource availability. Constraint (11) is a binary constraint for the c_{ijhk} decision variable. Here, although the decision variable x_{ijhk} is not restricted to being a binary 0 - 1 variable, the value of the decision variable is always zero or one based on the relationship between equations (8) and (11). The initial time for each operation is determined using equation (12). The start time for each operation for each type of source is determined using equation (13). This initial time can be determined by adding the two equations above directly to the model formulation above, or it can be determined after the optimal solution is obtained. The second way is more advantageous because it reduces the number of decision and limiting variables involved.

3. Numerical Example

3.1 The Data

The following is a numerical example for implementing the proposed mathematical model. General Production Data required are:

1. Type of manufacturing process.
2. Production Process and Operation Process Map.
3. Time of each operation.
4. Data on the type and number of machines.
5. Job order data along with the date of order and delivery.
6. The weight of each job.

In this study, data were collected from 2 jobs in the first week (in a 7-daytime slot).

Table1 Formulation of data collection results

No	Job	Quantity (pieces)	Operation	Source 1	Source 2	Source 3	Source 4	Source 5	Source 6	Time t_{ij} (days)	Order date	Delivery date	Due date (slots to)	Weight
1	Job 1	2750	1 *Weaving*	O						2	1-Sep	5-Sep	5	1
			2 *Dyeing*		O	O				1				
			3 *Hemming*				O			1				
			4 *Sewing*					O		2				
			5 *Packing*						O	1				
2	Job 2	1430	1 *Weaving*	O						2	2-Sep	6-Sep	6	2
			2 *Dyeing*		O	O				1				
			3 *Hemming*				O			1				
			4 *Sewing*					O		1				
			5 *Packing*						O	1				

Table description:

1. Resources are the machines used for operations
   Source 1: Weaving Machines
   Source 2: Dyeing Machine
   Source 3: The Tumbler Machine
   Source 4: Long Hemming Machines
   Source 5: Cross Sewing Machines
   Source 6: Packing Machines
2. The symbol O in the table above indicates that job i performed operation j using source h.
3. Order quantity for Job 1 is 2500 pcs. The company's policy is to produce 10% more than the order quantity to anticipate defects and other damage that may occur in the production process, so the quantity scheduled to be produced is (2500+10%*2500) = 2750 pcs. Likewise for Job 2 the quantity scheduled to be produced is (1300+10%*1300) = 1430pcs.

3.2 Data Processing

The first step of data processing is defining the problem based on the integer linear programming model for scheduling problems starting with:

1. Formulation of notations and decision variables

L The tardiness of each job

ISSN: 2456-3676

Z Total weighted tardiness

C_i At the end of job i, $C_i = c_iN_i$, where c_iN_i is the final completion time of the last operation on job i

b_{ij} At the beginning of the execution of operation for job i

c_{ij} At the end of operation j for job i

b_{ijh} At the start of the execution of operation j for job I on source h

c_{ijh} At the end of executing operation j for job i on source h

c_{ijhk} Binary variable 0-1; c_{ijhk} = 1if job i done at resource h is completed at time k, c_{ijhk}= 0 otherwise

X_{ijhk} Binary variable 0-1; x_{ijhk} = 1 if operation j for job i uses resource h in time slot k, x_{ijhk} = 0 otherwise

2. Formulation of the objective function and constraints
The next step of data processing is carried out with the WinQSB Software by inputting data into the WinQSB software in accordance with the formulation of the model that has been made. WinQSB will run the computing process, process the data and generate an answer report. The result is translated into the scheduling results table.

4. Results
The results obtained are translated in to a scheduling table for 2 jobs, where each job consists of 5 multiple operations with 6 sources. Scheduling is done in a predetermined time slot, which is 7 days. The scheduling table is:

ISSN: 2456-3676

Table 2 New Scheduling Results

Source	Job	Operation	Time slots (days)						
			1	2	3	4	5	6	7
1	1	1	1	1	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
	2	1	0	1	1	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
Source used			1	2	1	0	0	0	0
Sources available			118	118	118	118	118	118	118
2	1	1	0	0	0	0	0	0	0
		2	0	0	1	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
	2	1	0	0	0	0	0	0	0
		2	0	0	0	1	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
Source used			0	0	1	1	0	0	0
Sources available			14	14	14	14	14	14	14
3	1	1	0	0	0	0	0	0	0
		2	0	0	1	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
	2	1	0	0	0	0	0	0	0
		2	0	0	0	1	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0

ISSN: 2456-3676

Source used			0	0	1	1	0	0	0
Sources available			4	4	4	4	4	4	4
4	1	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	1	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
	2	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	1	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	0
Source used			0	0	0	1	1	0	0
Sources available			12	12	12	12	12	12	12
5	1	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	1	1	0
		5	0	0	0	0	0	0	0
	2	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	1	0
		5	0	0	0	0	0	0	0
Source used			0	0	0	0	1	2	0
Sources available			120	120	120	120	120	120	120
6	1	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	1
	2	1	0	0	0	0	0	0	0
		2	0	0	0	0	0	0	0
		3	0	0	0	0	0	0	0
		4	0	0	0	0	0	0	0
		5	0	0	0	0	0	0	1
Source used			0	0	0	0	0	0	2
Sources available			6	6	6	6	6	6	6

ISSN: 2456-3676

Table description: X_{ijhk} is a binary variable 0-1; x_{ijhk} = 1 if j operation for job uses source h in k time slot, x_{ijhk} = 0 otherwise.

4. Discussion

4.1 Comparison of the old schedule with the new schedule

Table 3 Old schedule (obtained through data collection)

	Time slots								
	1	2	3	4	5	6	7	8	9
Job1									
Job2									

From these two jobs, the total weighted delay is calculated as Z = 1.L1 + 2.L2. L1 is the delay for job 1which is obtained from the completion time minus the due date which is 4 days while L2 is the delay for job 2 which is obtained from the completion time minus the due date which is 2 days.

Based on the previously agreed weight based on the penalty that will be imposed on the company (the weight for job 1 is 1 and the weight for job 2 is 2), the total weighted delay for the two jobs is: 1*4 days+ 2*2 days= 8 days.

Table 4 New scheduling (is a simplified form of Table 2)

	Timeslot								
	1	2	3	4	5	6	7	8	9
Job1									
Job2									

From these two jobs, the total weighted delay is calculated as Z = 1.L1 + 2.L2. L1 is the delay for job 1which is obtained from the completion time minus the due date which is 2 days while L2 is the delay for job 2 which is obtained from the completion time minus the due date which is 1 day. With a weight for job 1 of 1 and a weight of 2 for job 2, the total weighted delay for both jobsis:1*2 days+ 2*1 days= 4 days.

4.2 Analysis of the advantages and disadvantages of the model

The integer linear programming model for this scheduling problem is quite accurate and complete enough to take into account the variables involved in scheduling and provide a more optimal scheduling solution according to its objective to minimize the total weighted delay.

From the results of scheduling using this model, it can be seen directly the variable data such as:

- At the beginning and at the end of each job and each operation on each existing source.
- Beginning time and completion time for each job as a whole.
- Availability of resource sat each time slot.

For many scheduling models, the known exact algorithms are those based on enumeration (singing one at a time). In fact, the natural combination of scheduling problems makes the modeling and computational processes difficult. This is where the shortcoming of the integer linear programming model for this scheduling problem lies. This model is difficult to apply to large scheduling cases with long time horizons because adding variables to jobs, operations, sources, or time slots will cause the number of variables and constraints to increase exponentially. This is closely related to the complexity of the problem formulation process based on the model and the limited capacity of the software and the relatively long computational time to process large amounts of data.

5. Conclusion

The new scheduling design using the integer linear programming model for the simultaneous multi-source multiple-source scheduling problem for the multiple-operation multiple-job for the case study can minimize the total weighted delay of order completion compared to the oldschedulingfrom8 days to 4 days

References

Bodington, C. Edward. (1995). Planning, Scheduling, and Control Integration In The Process Industries. McGraw-Hill, Inc USA.

Kerzner,Harold.(1995).ProjectManagement:ASystemApproachtoPlanning,Scheduling,andControlling.InternationalThompsonPublishingInc.NewYork.USA.

Pinedo, Michael., Xiuli Chao.(1999).Operations Scheduling: With Aplications in Manufacturing and Services. McGraw-HillInc. Singapore.

Sipper, Daniel., Robert L. BulfinJr.(1997). Production Planning, Control, and Integration.

McGraw-HillInc.USA.

Suprayogi. Isa Setiasyah Toha. (2002). Jurnal TMI22: Model Pemrograman LinierBilangan Bulat Untuk Masalah Penjadwalan Sumber-Majemuk Paralel Simultan.ITB. Bandung..