IOP Conf. Series: Materials Science and Engineering **114** (2016) 012060 doi:10.1088/1757-899X/114/1/012060

Flexible Job-Shop Scheduling with Dual-Resource Constraints to Minimize Tardiness Using Genetic Algorithm

A B N Paksi[1] **A Ma'ruf**[2]*
[1, 2]Department of Industrial Engineering, Faculty of Industrial Technology, Institut Teknologi Bandung, Bandung 40132, Indonesia

Abstract. In general, both machines and human resources are needed for processing a job on production floor. However, most classical scheduling problems have ignored the possible constraint caused by availability of workers and have considered only machines as a limited resource. In addition, along with production technology development, routing flexibility appears as a consequence of high product variety and medium demand for each product. Routing flexibility is caused by capability of machines that offers more than one machining process. This paper presents a method to address scheduling problem constrained by both machines and workers, considering routing flexibility. Scheduling in a Dual-Resource Constrained shop is categorized as NP-hard problem that needs long computational time. Meta-heuristic approach, based on Genetic Algorithm, is used due to its practical implementation in industry. Developed Genetic Algorithm uses indirect chromosome representative and procedure to transform chromosome into Gantt chart. Genetic operators, namely selection, elitism, crossover, and mutation are developed to search the best fitness value until steady state condition is achieved. A case study in a manufacturing SME is used to minimize tardiness as objective function. The algorithm has shown 25.6% reduction of tardiness, equal to 43.5 hours.

1. Introduction

Most of literatures on scheduling problem have considered only machines as a limited resource and have ignored the possible constraint caused by availability of workers with requisite skills to perform the required process. In fact, both machines and human resources are needed for processing a job on production floor. A Dual-Resource Constrained (DRC) system is one in which production capacity is constrained by both machines and workers [4]. Algorithm [1] has solved a DRC scheduling problem by using Genetic Algorithm (GA) to determine the best staffing level and dispatching rule with average flow time, average tardiness, and average waiting time as performance measures. Another study, Model [2] was developed to solve DRC scheduling problem in non-permutation flow-shop by using both analytical and meta-heuristic models with objective function to be achieved is to minimize work-in-progress inventory and to maximize service level.

In classical job-shop scheduling problem, there are a set of independent jobs and a set of machines. Each job has an ordered set of operations that must be processed on a predefined machines [3]. Along with

* Corresponding author. Tel.: +62 22 2506449; fax.: +62 22 2506449
E-mail address: anggoro.bintang@gmail.com (A.B.N. Paksi), maruf@ti.itb.ac.id (A. Maruf)

growing consumer demand variety, changing markets with global competition, and rapid development of new technologies, production floor should has flexibilities to accommodate that situations [6]. One of those is routing flexibility which means that a job can be processed on more than a possible route due to multiple number of similar machines. So that, a route of particular job depends on availability of machines. Flexible Job-shop Scheduling Problem (FJSP) model with routing and process plan flexibilities [3] was developed to minimize maximum completion time. Meanwhile, model [5] has developed to minimize tardiness by considering routing flexibility and interrupted job.

This research is a continuation research of scheduling problem that considers two constraints, namely machines and workers and also routing flexibility. Scheduling is categorized as NP-hard problem [8]. Computational time of NP-hard problem will increase exponentially along with increase of problem size. Therefore, meta-heuristic approach, namely Genetic Algorithm is used due to its practical implementation in industry.

2. Problem Description

A manufacturing SME with Dual-Resource Constrained system is chosen as case study object. In this case, objective of scheduling is to find feasible solution for a set of jobs that minimizes tardiness or positive difference between completion time and job due date. Flexible job-shop scheduling problem with Dual-Resource Constraints consists of a set of independent jobs. Each job has an ordered set of operations. Two considered constraints are machines and workers with less number of workers compared to machines. There are multiple number of similar machines. Each worker has ability to process operation in particular machines. Each operation will be scheduled into machine and worker with suitable skill needed.

Data that are needed to solve this problem, namely sequence of operations, processing time, job due date, type and number of machines, and ability and number of workers. From the data, binary matrix of machine-operation is formed that shows ability of machines to perform processes. Value '1' in the matrix means that machine has ability to process operation, while value '0' means in the contrary. Another binary matrix of worker-machine is also formed with the same procedure. The following assumption are made for FJSP with DRC:

- The object has no problem in providing raw material and work-in-process, so jobs can be started right on the time scheduled.
- Pre-emption is not allowed.
- Workers are equally efficient at all machines, thus no loss in efficiency results.
- Machine breakdown is not considered.
- Workers may be transferred from one machine to another, subject to certain restriction (depending on skills).
- Each machine can process only one job at a time.
- Processing times include set-up. They are considered deterministic and known in advance.
- Transportation times are included in the processing times.
- Workers are occupied with both job and machine throughout loading, processing, and unloading.

3. Proposed Algorithm

In this research, Genetic Algorithm will be used as problem solving method for FJSP with Dual-Resource Constraints. GA search begins with codification process which covers chromosome and initial population forming. Then, decodification process is done to convert chromosome into solution that can be evaluated in terms of quality. Genetic operators are used to search the best solution. Search process will be done until termination condition is achieved. Solution that is achieved in the termination condition is likely to be the global optimum or at least close to the global optimum.

3.1. Genetic Algorithm Development

Overall, the development of Genetic Algorithm is shown in Figure 1.

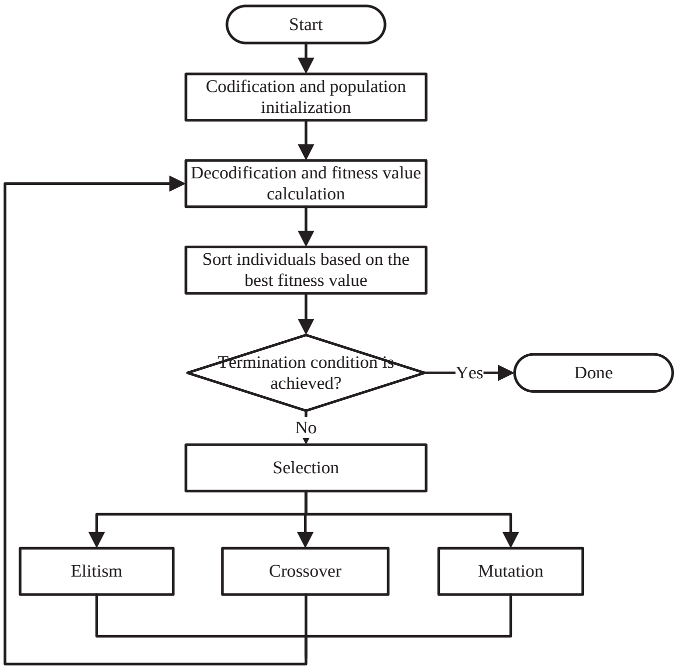

Figure 1. Flowchart of Genetic Algorithm Development

Chromosome used in this development is indirect chromosome representative that consists of two layers. This chromosome contains assignment data of machine and worker into each operation in each job. Meanwhile, information about starting time of each operation in each job will be accommodated in decodification process. Figure 2 shows illustration of chromosome representative used. The first layer determines in which machine an operation will be processed. While, worker who is responsible to process an operation is shown by second layer. For example, job 1 operation 1 will be processed in machine 1 and by worker 2.

	1-1	1-2	1-3	2-1	2-2	2-3	Job-Operation Number
Layer 1	M1	M3	M6	M2	M3	M1	Machine Number
Layer 2	W2	W4	W1	W2	W3	W4	Worker Number

Figure 2. Illustration of Chromosome Representative

Population initialization is done by randomizing gene value in each layer. Randomization of gene value has to ensure that machine and worker are capable to process corresponding operation.

Decodification process is done by plotting operations into time event. After assigning resources to each operation in codification process, determination of starting time will be done. Scheduling approach used is forward scheduling with ready time of all jobs is at $t = 0$. Earliest Due Date (EDD) dispatching rule is used to determine which job that will be processed first. Figure 3 shows flowchart of decodification process.

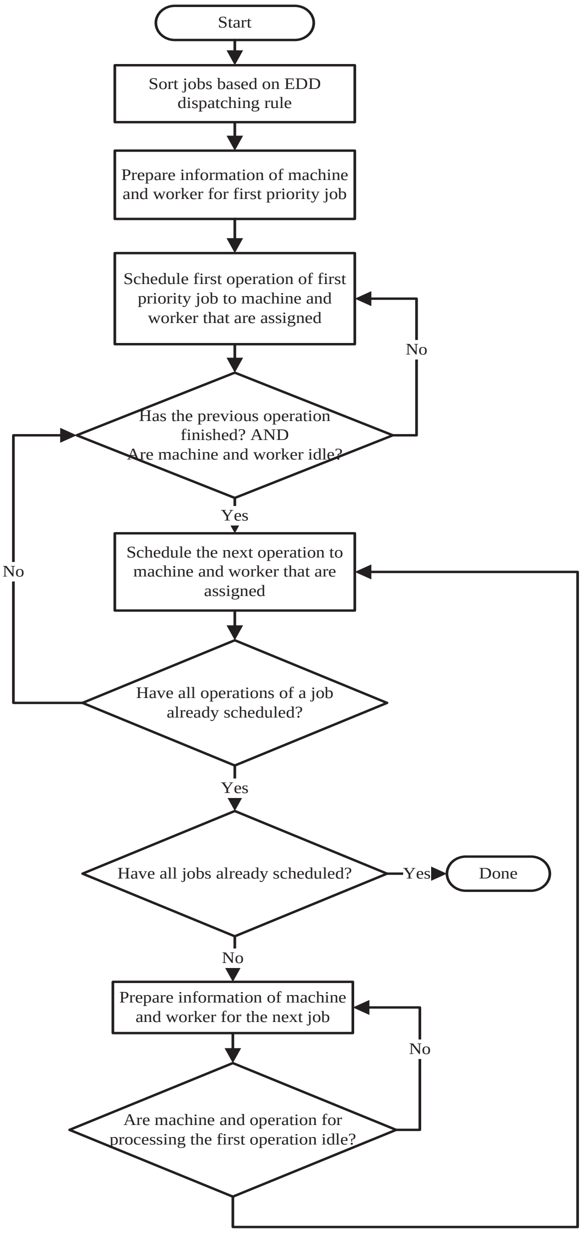

Figure 3. Flowchart of Decodification Process

Fitness value is objective function of Genetic Algorithm that indicates how good solution is and how close an individual with optimum solution [7]. Objective function of this research is to minimize tardiness or positive difference between completion time (*CT*) and due date (*DD*) of each job ($m = 1,..,M$)

$$Minimize \sum_{m=1}^{M} CT_m - DD_m \quad (1)$$

Selection process is used to choose parent candidate that will go through crossover and mutation processes. If population size is P, number of chromosome that will selected is population number minus the best (elite) chromosome or (P-1). Selection process will produce List of Parent Candidate (LPC) that consists of (P-1) parent candidate of crossover and mutation processes. Sampling mechanism that is used is roulette wheel procedure. Principle of this method is linear search by using roulette wheel. Slot for each individual is proportional with its fitness value. Elitism process is determination of an individual that has the best fitness value in a population. Individuals that have the best fitness value are collected in List of Elite Individual (LEI). Then, an individual in the LEI is chosen randomly to be included to List of New Individual (LNI). Crossover is process that combines two parents to produce two offspring that have some shared characteristics of their parents. Crossover operator that is used in this research is two points crossover. This crossover operator exchanges gene values that exist between two points that are randomly selected. Mutation has a role to make small changes in solution randomly. This changes will add some characteristics in solution. Mutation process that is developed is by changing gene value of a locus that is selected randomly. After selecting a locus, gene values will be changed with another feasible values randomly.

Genetic Algorithm search is done until a termination condition is achieved. In this research, termination condition is achieved if steady state happens. Steady state means that repetition of the best value happens multiple times. In this research, number of repetition that is needed to reach steady state is set 30 times.

3.2. Scheduling Application

Scheduling application, based on Visual Basic for Application Microsoft Excel is designed for GA implementation. Application that is designed consists of main algorithm and some small algorithms. Steps that is done in main algorithm are as follows:

1. Input data that is needed, namely number and sequence of operations of each job, processing time of each operation, due date of each job, matrix of machine-operation, and matrix of machine-worker.
2. Input parameters that are needed, namely population size (P), percentage of crossover (P_c), and number of repetition to achieve steady state (SS).
3. Calculate number of individual that will go through crossover ($P_{crossover}$) and number of individual that will go through mutation ($P_{mutation}$).
4. Set $g = 0$; $i = 0$; and $f_{now} = 999999$.
5. Codification and population initialization (Algorithm A).
6. Set $p = 1$.
7. Decodification and calculation of fitness value (Algorithm B).
8. Is $p = P$?
   8.1. If all individuals have passed decodification process, then next to step 9.
   8.2. If there is still individual that has not passed decodification process yet, then $p = p+1$.
9. Sort individuals based on the best fitness value.
10. Set the best fitness value = f_{best}.
11. Is $f_{best} < f_{now}$?
    11.1. If $f_{best} < f_{now,}$ then next to step 12.
    11.2. If $f_{best} = f_{now}$, then $i = i+1$.
        11.2.1. If i = SS, then algorithm is done.
        11.2.2. If $i \neq$ SS, then next to step 13.
12. Set $f_{now} = f_{best}$; $i = 0$.
13. Selection process (Algorithm C).
14. Population development process
    14.1. Elitism process (Algorithm D).
    14.2. Crossover process (Algorithm E).
    14.3. Mutation process (Algorithm F).
15. Repeat step 6.

4. Numerical Example

Data that is used for this research can be seen in Table 1-6.

Table 1. Number, Sequence, and Processing Time of Each Operation

Job	Operation	Machine	Processing Time (Hour)
1	1	BU	0,75
	2	FR	2,25
	3	FR CNC	2,75
2	1	BU	2,25
	2	FR	2,25
	3	FR CNC	9,25
	4	BU	4,25
3	1	FR	0,75
	2	BU	0,75
4	1	FR CNC	8,25
5	1	FR CNC	8,25
6	1	BU	1,25
7	1	FR	2,25
	2	GD	2,25
	3	FR CNC	3,25
8	1	FR	2,25
	2	GD	2,25
	3	FR CNC	3,25
9	1	BU	8,25
	2	BOK	0,75
	3	EDM	6,25
10	1	FR CNC	7,25
	2	BU	3,25
	3	BOK	2,25
11	1	BU	6,25
	2	BOK	3,25
12	1	BU	2,25
	2	BOK	1,25
13	1	FR	1,25
	2	BOK	0,75
	3	FR CNC	2,25

Job	Operation	Machine	Processing Time (Hour)
14	1	BU	2,25
	2	FR	1,25
15	1	BU	1,25
	2	FR	0,75
16	1	BU	2,75
17	1	BU	1,25
	2	FR	0,75
18	1	FR	0,75
	2	BOK	0,75
19	1	FR	2,25
	2	GD	1,75
	3	BOK	1,25
20	1	FR	2,25
	2	FR CNC	4,25
	3	GD	1,25
	4	BOK	1,25
21	1	FR	3,25
	2	GD	1,75
	3	BOK	1,25
22	1	BU	0,75
	2	GS	6,25
23	1	FR CNC	4,75
	2	FR	0,75
24	1	FR CNC	3,75
	2	FR	0,75
25	1	EDM	8,25
26	1	EDM	8,25
27	1	BU	0,75

Table 2. Due Date of Each Job

Job	Due Date (Hour)
1	8
2	24
3	8
4	16
5	16
6	8
7	8
8	8
9	16
10	16
11	16
12	8
13	8
14	8

Job	Due Date (Hour)
15	8
16	8
17	8
18	8
19	8
20	16
21	8
22	8
23	8
24	8
25	16
26	16
27	8

Table 3. Machine Numbering

Machine Number	Machine
1	GD
2	GD
3	GS
4	FR
5	FR
6	BOK
7	BU
8	BU
9	EDM
10	EDM
11	FR CNC
12	FR CNC
13	BU CNC

Table 4. Matrix of Machine – Operation (1)

Operation	1-1	1-2	1-3	2-1	2-2	2-3	2-4	3-1	3-2	4-1	5-1	6-1	7-1	7-2	7-3	8-1	8-2	8-3	9-1	9-2	9-3	10-1	10-2	10-3	11-1	11-2	12-1	12-2	13-1	13-2	13-3	14-1	14-2
Machine 1														1			1																
Machine 2														1			1																
Machine 3																																	
Machine 4		1			1			1					1			1													1				1
Machine 5		1			1			1					1			1													1				1
Machine 6																				1				1		1		1		1			
Machine 7	1			1			1		1			1							1				1		1		1					1	
Machine 8	1			1			1		1			1							1				1		1		1					1	
Machine 9																					1												
Machine 10																					1												
Machine 11			1			1				1	1				1			1				1									1		
Machine 12			1			1				1	1				1			1				1									1		
Machine 13																																	

Table 5. Matrix of Machine – Operation (2)

Operation	15-1	15-2	16-1	17-1	17-2	18-1	18-2	19-1	19-2	19-3	20-1	20-2	20-3	20-4	21-1	21-2	21-3	22-1	22-2	23-1	23-2	24-1	24-2	25-1	26-1	27-1
Machine 1									1				1			1										
Machine 2									1				1			1										
Machine 3																			1							
Machine 4		1			1	1		1			1				1						1		1			
Machine 5		1			1	1		1			1				1						1		1			
Machine 6							1			1				1			1									
Machine 7	1		1	1														1								1
Machine 8	1		1	1														1								1
Machine 9																								1	1	
Machine 10																								1	1	
Machine 11												1								1		1				
Machine 12												1								1		1				
Machine 13																										

Table 6. Matrix of Worker – Machine

Machine	1	2	3	4	5	6	7	8	9	10	11	12	13
Worker 1	1	1	1										
Worker 2				1	1								
Worker 3						1							
Worker 4							1	1					
Worker 5									1	1			1
Worker 6											1	1	
Worker 7											1	1	
Worker 8	1	1	1	1	1	1	1	1	1	1			
Worker 9	1	1	1	1	1	1	1	1	1	1			
Worker 10	1	1	1	1	1	1	1	1	1	1			

Algorithm parameters that are used in this research are population size, percentage of crossover, and number of repetition to achieve steady state. Value of population size that are used are 30, 40, 50, 60, and 70. While, percentage of crossover that are used are 0.6; 0.7; 0.8; 0.9. Fix number of repetition as a steady state requirement is 30. After running scheduling application with 20 combinational sets of parameters and two replications, average fitness values and computational times are shown in Table 7.

According to Table 7, there is no a general pattern of fitness value. Increase of population size or percentage of crossover result in fluctuated change of fitness value. On the other hand, there is a general pattern of computational time that is influenced by population size. Computational time will increase along with increase of population size. It is caused by wider solution space that is explored. However, similar condition is not resulted by changing percentage of crossover. Up and down varieties of computational time happen along with increase of percentage of crossover.

Table 7. Average Result of Algorithm Testing Using 20 Combinational Sets of Parameters

% Crossover		0.6		0.7		0.8		0.9	
Criterion		Tardiness (Hour)	Time (Second)	Tardiness (Hour)	Time (Second)	Tardiness (Hour)	Time (Second)	Tardiness (Hour)	Time (Second)
Population Size	30	143.00	742	136.25	883	143.75	567	140.25	745
	40	139.75	1079	145.00	1136	129.50	653	140	871
	50	140.50	1363	147.75	1651	137.25	893	133.5	926
	60	147.50	1895	127.50	1788	154.50	1069	143.5	1120
	70	140.25	2075	**124.75**	1943	134.25	1251	131.75	2430

The largest computational time is produced by combination of population size at 70 and percentage of crossover at 0.9, which is 2430 seconds. Upper limit of computational time needed is still possible to be implemented in industry.

Based on testing of 20 combinational sets of parameters, the best fitness value is produced by combination of population size at 70 and percentage of crossover at 0.7, which is 124.75 hours. Compared with as-is condition, there is 25.58% reduction of tardiness value, equal to 43.5 hours.

5. Conclusion

A proposed algorithm of scheduling problem that covers Dual-Resource Constraints and routing flexibility is developed. Meta-heuristic approach, based on GA, is used to solve this NP-hard problem.

Developed Genetic Algorithm uses indirect chromosome representative that consists of two layers. Fitness value, namely tardiness value is evaluated by using an additional procedure in decodification process. Genetic operators, namely roulette wheel selection, one point elitism, two points crossover, and one point mutation are used until termination condition is achieved that requires 30 times repetition of fitness value. Implementation of GA is done by designing an application using Visual Basic Application Microsoft Excel.

A case study is done in manufacturing Small Medium Enterprise that has characteristics of DRC and routing flexibility. According to the testing, algorithm produces the best solution with population size at 70 and percentage of crossover at 0.7. However, this combinational parameters only give the best feasible solution in data that are used in this research. Compared with existing condition, proposed algorithm gives 43.5 hours reduction of tardiness value. There is 25.58% improvement that is contributed by proposed algorithm.

References

[1] ElMaraghy H, Patel V and Abdallah I 2000 Scheduling of Manufacturing System Under Dual-Resource Constraints Using Genetic Algorithms *J. of Manuf. Sys.* **20** 186-201
[2] Mehravaran Y and Logendran R 2013 Non-permutation Flowshop Scheduling with Dual Resources *Exp. Sys. with Appl.* **40** 5061-76
[3] Ozguven C, Ozbakir L and Yavuz Y 2010 Mathematical models for job-shop scheduling problems with routing and process plan flexibility *Appl. Math. Model.* **34** 1539-48
[4] Patel V 1997 *Scheduling in A Dual Resource Constrined System Using Genetic Algoritms* (Windsor: University of Windsor)
[5] Prasetio I 2013 Usulan Model Penjadwalan untuk Meminimasi Keterlambatan Pengiriman Produk di Politeknik Manufaktur Negeri Bandung *Proc. of 7th National Industrial Engineering Conference (NIEC) (Surabaya, Indonesia, 10 October 2013)* pp 77-83

[6] Roshanaei V 2012 *Mathematical Modelling and Optimization of Flexible Job Shops Scheduling Problem* (Windsor: Electronic Theses and Dissertations) p 157
[7] Sivanandam S N and Deepa S N 2008 *Introduction to Genetic Algorithms* (Berlin: Springer)
[8] Xu J, Xu X and Xie S Q 2011 Recent Developments in Dual Resource Constrained (DRC) System Research *Eur. J. of Op. Res.* **215** 309-18