



Journal Homepage: - www.journalijar.com
**INTERNATIONAL JOURNAL OF
 ADVANCED RESEARCH (IJAR)**

Article DOI: 10.21474/IJAR01/6308
 DOI URL: <http://dx.doi.org/10.21474/IJAR01/6308>



RESEARCH ARTICLE

MODEL MULTIOBJECTIVE FLEXIBLE JOB SHOP SCHEDULING WITH BATCH PROCESS

Ferdinan Rinaldo Tampubolon and Sawaluddin.
 Faculty of Math and Science Universitas Sumatera Utara

Manuscript Info

Manuscript History

Received: 14 November 2017
 Final Accepted: 16 December 2017
 Published: January 2018

Key words:-

Job Shop, Flexible, Batch, Tardiness,
 Makespan

Abstract

To compete in business competition, each manufacture must be able to fulfill the consumer demand precisely, While considering its lack of resource in terms of the amount of machine and the capacity of machine in processing each job. Scheduling is needed to assure the efficiency of the resource. Flexible job shop is a branch of scheduling which is the extension of Job Shop, it allows an operation to be processed by a set of machine instead of one machine. This research will develop the model of Flexible Job Shop Scheduling with multiple objective function, that is makespan and tardiness. This research is also considering the batch process, which allow some machine to have multiple capacity so it can process more than one job simultaneously.

Copy Right, IJAR, 2018,. All rights reserved.

Introduction:-

Business competition and consumer demand require each manufacture to complete each job as soon as possible. However, it is not easy because there is limitation in terms of the amount of machine or machine's ability which can only process some jobs in a interval of time. Considering this problem, scheduling plays a key role in using the resource economically, and gaining the profit abundantly.

Job Shop Scheduling Problem (JSSP) is a branch of scheduling, which allocate m machine to n_i operation of each job. While each operation must be processed sequentially. Flexible Job Shop Scheduling Problem (FJSSP) is the generalization of Job Shop which enable each job to be processed by a set of machine instead of one machine as in classical job shop. If μ_{ij} is a set of machine which can process the operation, in classical job shop μ_{ij} is a one member set while in flexible job shop μ_{ij} may have more than one member. It makes FJSSP more difficult than JSSP because it is not just about sequencing the operation but also assigning every machine to each operation.

Usually the objective function of FJSSP is minimizing the makespan, but in reality the properness of scheduling involves more than one criteria. Each of criteria may conflict one to each other. Thus, a Multiobjective Scheduling is more appropriate.

Some former research had discussed FJSSP. Hongbo Liu (2009) discussed FJSSP which the objective function is to minimize makespan and flowtime, but it does not involve tardiness in the job processing and the and the release time of job. Next Andy Ham (2016) discussed FJSSP wherein there are some machine which can deal with multiple job in a time interval, but it's only use one criteria, makespan. This research will develop Multi Objective Flexible Job Shop Scheduling Problem MOFJSSP model which the objective function is minimizing makespan and tardiness. It will also include a batch processing, that is doing multiple job in a time interval.

Method:-

Job Shop:-

Job Shop Scheduling Problem (JSSP) can be described as follows, suppose that there are amount of o machines $M_k = (k = 1, 2 \dots o)$, amount of n jobs $j_i = (i = 1, 2 \dots n)$ with n_i operation, $O_{i1}, O_{i2}, \dots, O_{in_i}$. Each machine will process each operation with processing time p_{ij} . μ_{ij} is set of machine which can process O_{ij} , the completion time for operation is c_{ij} , the completion can be provided by summing the processing time with the starting time $c_{ij} = p_{ij} + s_{ij}$. The goal of JSSP is to arrange every operation in such away that it's maximum completion time (c_{max}) will be minimized, there are some rules in JSSP, that is

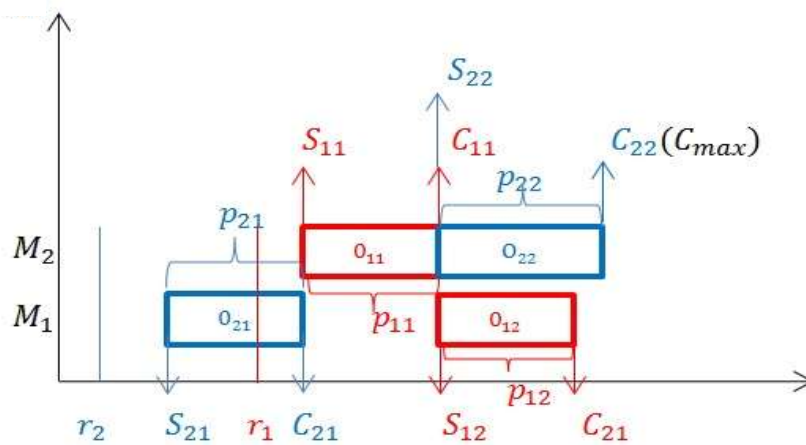
1. Each machine can only process one operation at a time.
2. Each operation must be finished in order to continue to another operation, in other word there is no delay.
3. Each operation can be processed once on exactly one machine.
4. There is only one capable machine which can processed an operation.
5. Each operation in a job can not be processed simultaneously.
6. Each first operation in a job can be processed after release time (r_i).
7. Each starting time, processing time, due time has been known certainly (deterministic).

The gantt chart of feasible solution of JSSP can be seen in **Fig 1**.

Flexible Job Shop:-

Flexible Job Shop Scheduling Problem is a generalized form of Job Shop which enable every operation to be processed by more than one machine. An operation may have a various completion time if it is processed by a diffrent machine, in FJSSP, assumption (3) can be ignored, because some operation can be processed by several capable machine. Flexible Job Shop mainly have two sub-problem, routing, which is assign each machine to an operation, sequencing, which is arrange each operation properly. Yunus Demir (2013) divided the model of flexible job shop in 3 type based on definition binary variable, that is,

1. Sequence-position variable, it involves variable $x_{ijkl} = 1$ if O_{ij} is assigned to k^{th} machine and l^{th} position, $x_{ijkl} = 0$, otherwise.
2. Precedence variable, it depends on variable $z_{ijhghk} = 1$ if O_{ij} precedes O_{gh} on k^{th} machine and zero for otherwise
3. Time-indexed variable, it involves $x_{ijk_u} = 1$, if O_{ij} is processed on k^{th} machine in period u .



Batch Process:-

Considering a machine which can process multiple job simultaneously. The processing time for each operation in a batch may not be the same and the completion time of the batch is determined by the operation which have the longest completion time. Assumption (5) will be ignored. In this problem there are some machine which can processed more than one job simultaneously, which is depend on the capacity of the machine. Suppose that b denote the maximum amount of operation which can be inserted into the batch (machine capacity). If $b = 1$, then the

problem will not be taken into account, while if $b = \text{inf}$ then the minimization of the makespan will be trivial. Thus, $b = n$ then $2 \leq b \leq n - 1$.

Result:-

Parameter:-

- i : index of job, $i = 1, 2, \dots, n$
- j : index of operation, $j = 1, 2, \dots, m$
- k : index of machine, $k = 1, 2, \dots, o$
- b : index of batch, $b = 1, 2, \dots, p$
- d_i : due date of job i
- L_i : lateness of job i
- T_i : tardiness of job i
- K_k : capacity of machine k
- p_{kb} : processing time of the k^{th} machine in batch b
- O_{ij} : the j^{th} operation in job i
- p_{ijk} : processing time of the j^{th} operation of job i which is processed by machine k
- s_{kb} : starting time of the k^{th} machine in batch b
- c_{kb} : completion time of the k^{th} machine in batch b
- r_i : release time of job i
- c_i : completion time of job i
- M : big positive number

Binary Variable:-

$$x_{ijkb} = \begin{cases} 1, & \text{if } O_{ij} \text{ is processed by machine } k \text{ on } b^{\text{th}} \text{ batch} \\ 0, & \text{otherwise} \end{cases}$$

$$a_{ijk} = \begin{cases} 1, & \text{if } O_{ij} \text{ can be processed by machine } k \\ 0, & \text{otherwise} \end{cases}$$

$$v_{ijk} = \begin{cases} 1, & \text{if } O_{ij} \text{ is processed by machine } k \\ 0, & \text{otherwise} \end{cases}$$

Constarints:-

Assignment Constraint:-

Equation (1) ensures each job must be processed by exactly one machine, equation (2) ensure that each job must be assigned to one batch, equation (3) shows relation between the capable machine for each job and the chosen machine, equation (4) ensures that the amount operation in a batch will not exceed the capacity of the machine.

$$\sum_{k=1}^o v_{ijk} = 1, \quad \forall i, j \tag{1}$$

$$\sum_{b=1}^p x_{ijkb} = v_{ijk}, \quad \forall i, j, k \tag{2}$$

$$v_{ijk} \leq a_{ijk}, \quad \forall i, j, k \tag{3}$$

$$\sum_{k=1}^o \sum_{b=1}^p x_{ijkb} \leq K_k, \quad \forall i, j \tag{4}$$

Sequencing Constraints:-

Equation (5) shows the processing of a job can not be started before release time, Equation (6) ensures the precedence relation between each operation in a job.

$$s_{ij} \leq r_i, \quad \forall i, j, k \quad (5)$$

$$s_{i(j+1)} \geq s_{ij} + \sum_{k=1}^o \sum_{b=1}^p x_{ijkb} \cdot p_{ijk}, \quad \forall i, j \quad (6)$$

Batch Constraints:-

Equation (22) ensures the precedence relation of the batch processing in the same machine

$$s_{k(b+1)} \geq s_{kb} + \sum_{k=1}^o \sum_{b=1}^p x_{ijkb} \cdot p_{kb}, \quad \forall i, j \quad (7)$$

Starting Time and Completion Time Constraints:-

Equation (8) ensures that the processing of batch can be started when each operation in a batch are ready to be processed, equation (9) defines the processing time of the batch, that is the longest processing time of each operation in the batch.

$$s_{kb} \geq s_{ij} + M(x_{ijkb} - 1), \quad \forall i, j, k, b \quad (8)$$

$$p_{kb} \geq p_{ijk} \cdot x_{ijkb}, \quad \forall i, j, k, b \quad (9)$$

Completion time and Lateness Constraints:-

equation (10) defines the completion time for the batch, equation (11) defines the lateness in processing each job, equation (12) specifies the completion time of job, that is the completion time of the last operation in a job.

$$c_{kb} \geq s_{kb} + p_{kb}, \quad \forall k, b \quad (10)$$

$$L_i \geq C_i - d_i, \quad \forall i \quad (11)$$

$$c_i \geq r_i + \sum_{k=1}^o \sum_{b=1}^p x_{ijkb} \cdot c_{kb}, \quad \forall k, b \quad (12)$$

Non-negative Constraint:-

$$c_{kb} \geq 0, s_{kb} \geq 0, s_{ij} \geq 0 \quad \forall k, b \quad (13)$$

Objective Function:-

$$\text{Minimize } C_{max} = \max_{1 \leq k \leq o} \left\{ \max_{1 \leq b \leq p} c_{kb} \right\}, \quad \forall k, b \quad (14)$$

$$\text{Minimize } T = \sum_{i=1}^n \max\{0, L_i\}, \quad \forall i \quad (15)$$

Equation (13) defines the makespan, that is the completion time of the last operation. Equation (14) defines the total tardiness, which is lateness with positive value.

Conclusion:-

The model of Flexible Job Shop with batch process use the same binary variable with flexible job shop, that is sequence position variable. The differences is in batch process there can be more than one operation which can be

assigned to one machine and one position simultaneously. While in flexible job shop, there is only operation for one machine and position. The definition of processing time and starting time also different.

References:-

1. Ahmadi, Ehsan., Mostafa, Zandieh., Motjaba, Farrokh., Seyed, Mohammad Emami (2009). A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms. *Computers and Operations Research*, Volume 431, Issue 10, 56-66.
2. Brucker, Peter. (2007). *Scheduling Algorithms*, fifth edition, Springer, German.
3. Peter Brucker } Brucker, Peter & Sigrid Knust (2005). *Complex Scheduling*, Springer, German.
4. Damodaran, Purushothaman., Miguel Rojas-Santiago (2011). *Mathematical Formulation to Minimize Makespan in a Job Shop with a Batch Processing Machine*.
5. Ninth LACCEI Latin American and Caribbean Conference (LACCEI'2011), WE1-1 - WE1-10
6. French, Simon (1981). *Sequencing and Scheduling: An Introduction to the Mathematics of Job Shop*. John Wiley & Sons Edition, Manchester.
7. Ham, Andy. (2016). Flexible Job Shop Scheduling Problem with Parallel Batch Processing. *Proceedings of the 2016 Winter Simulation Conference*, 2740--2749.
8. Ham, Andy., Eray Cakici. (2016). Flexible job shop scheduling problem with parallel batch processing machines: MIP and CP approaches. *Computers & Industrial Engineering* (2016). 160-165.
9. Kethley, R. Bryan., Bahram Alidaee., Haibo Wang. (2014). Single machine scheduling to minimize a modified total late work function with multiple due dates *Production & Manufacturing Research: An Open Access Journal*. Volume 2 No 1, 624-640.
10. Liu, Hongbo. (2009). A Multi-swarm Approach to Multi-objective Flexible Job-Shop Scheduling Problems. *Fundamenta Informaticae* 95 (2009),1-25.
11. Hsiang, Cheng Liu & Ding-Hsiang Huang. (2014). Reduction of power consumption and carbon footprints by applying multi-objective optimisation via genetic algorithms. *International Journal of Production Research*, Volume 52 No 2, 337-352.
12. Hwa Rong Huang., Chang-Lin Yang., Wei-Che Cheng. (2013). Flexible job shop scheduling with due window - a two-pheromone ant colony approach. *Int. J. Production Economics*, 685-697.
13. Pinedo, Michael.L. (2008). *Scheduling: Theory, Algorithms, and System*, Prentice Hall, New Jersey.
14. Piroozfard, Hamed., Kuan Yew Wong., Wai Peng Wong (2016). Minimizing total carbon footprint and total late work criterion in flexible job shop scheduling by using an improved multi-objective genetic algorithm. *Resource Conservation and Recycling* (2016).
15. Sakawa, Masatoshi. (2013). *Linear and Multiobjective Programming with Fuzzy Stochastic Extensions*. Springer, Volume 203.
16. Scaria, Abyson., Kiran, George., Jobin, Sebastian. (2016). An artificial bee colony approach for multi-objective job shop scheduling. *Procedia Technology* 25 (2016), 1030--1037.
17. Yin, Lvjiang., Xinyu, Li., Liang, Gao., Chao, Lu., Zhao, Zhang. (2016). A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem. *Sustainable Computing: Informatics and Systems*, 15-30.
18. Demir Yunus., & Isleyen, S, Kursat. (2016). Evaluation of Mathematical models for flexible job-shop scheduling problems. *Applied Mathematical Modelling*, 977-988.