Job Scheduling Optimization with Optimal Work-in-Process in Flexible Flow Shops with Unrelated Parallel Machines

Benjaporn Chaiyanate
Master of Engineering Technology, Engineering Faculty
Thai-Nichi Institute of Technology
Bangkok, Thailand
2461150050@tni.ac.th

Anchalee Supithak
Industrial Engineering, Engineering Faculty
Thai-Nichi Institute of Technology
Bangkok, Thailand
anchalee_s@tni.ac.th

***Abstract*— This research proposes a heuristic method based on the Genetic Algorithm to determine the job schedule in the flexible flow shop with unrelated parallel machines. The algorithm objective is to solve for the job schedule minimizing the tardiness and the work-in-process. The interstage waiting time represents work-in-process of jobs in the flexible flow shop. The solutions from the developed heuristic are compared with the Brute Force search. The developed heuristic considers the integration of tardiness and interstage waiting time into the fitness function. The conventional method which is Earliest Due Date (EDD) and Earliest Completion Time (ECT) is applied to compare the performance of the heuristic. For the problem with five jobs, the developed algorithm with the tightness factor of 0.95, 1.15, and 1.40 gives the same solution as the brute force method for every cases. The deviation percentage represented the performance comparison between the developed algorithm and the brute force search. On the other hand, the average deviation percentage of the solutions from EDD and ECT is 144.51%, and the maximum value is 279.33%. The result indicates that the developed algorithm based on Genetic Algorithm with the fitness function of integration of tardiness and work-in-process achieves in finding the optimal solution.**

Keywords*— *Flexible Flow Shop, Genetic Algorithm, Interstage Waiting Time, Job Scheduling, Tardiness, Unrelated Parallel Machines, Work-in-Process

I. INTRODUCTION

The Flexible Flow Shop or FFS is the manufacturing process which composes of a series of processes or operation. Flexible Flow Shop is also known as Hybrid Flow Shop which each stage has more than one machine. The Flexible Flow Shop with unrelated parallel machines problem has been studied wildly, since it represents the real manufacturing shop floor. The order scheduling is complicated when each operation involves machines with difference performance. In the real situation, the nonidentical machine is interesting and complicated to find the scheduling solution. The jobs scheduling in FFS with unrelated parallel machines problem is considered as NP hard problem [1]. Many research focuseson solving for the schedule with minimization of average flow time or makespan and meet due date. However, jobs scheduling decision in FFS with unrelated parallel machines is complicated and becomes combinatorial explosion problem since the makespan depends on the job sequence and the selected machine in each operation which effect processing time. Thus, the Enumeration method or Branch-and-Bound may not be able to find the optimal solution within the practical period [2]. Moreover, the hybrid flow shop or flexible flow shop is studied wildly because the computational structure has become complex and interesting. The research of [3] presents the significant characteristic and the systematic solving approach of FFS problem. The Genetic Algorithm is wildly used in much research to solve the variance FFS with unrelated parallel machines problem especially with the complicated and large problem of job scheduling with machine selection [4]. In addition, not only the performance indicators of tardiness or makespan but also the work-in-process that flow in the manufacturing shop floor. The work-in-process effects throughput and cycle time of the FFS manufacturing. Little's law [5], [6] is introduced in queuing theory as shown in equation (1) to represent the relationship between the throughput and the cycle time.

$$WIP = TH \times CT \tag{1}$$

Work-in-process (WIP) increases as the cycle time increases. Thus, controlling the amount of WIP in the shop floor will affect the cycle time. Job sequencing effects completion time and the throughput rate as well. The determination of constant WIP or CONWIP is presented by [6] to determine the amount of constant WIP releasing into the shop floor. Prior studies have also investigated CONWIP setting in flow-shop environments, showing that WIP-Control policies can be incorporated into production systems to regulate the release of jobs into the shop floor [7]. More recent studies have extended CONWIP concepts in multi-stage production systems by considering WIP Capacity or WIP-Cap adjustment and system flexibility, indicating that WIP control remains an important issue in practical manufacturing environments [8]. Most research in the field of flexible flow shop unrelated parallel machine focuses on minimizing makespan or total tardiness. For FFS with unrelated parallel machines, the dual criteria considered in the objective function which is the sum of makespan and the number of tardiness is studied by [9]. Generally, the amount of WIP in the shop floor is decided separately from the job sequence determination. Specifically, the FFS unrelated parallel machine problem relates to the complex decision composing of job sequencing and machine assignment in each stage. According to literature review, the integration of completion time and due-date performance into fitness function is interesting. FFS with unrelated parallel machine involves both job sequencing and machine selection. This research aims to integrate interstage waiting time, which represents WIP, and total tardiness into the fitness function.

Work-in-Process in this research is considered as job accumulated in the shop floor and is represented by time-based proxy according to Little' Law theory. This problem is considered as NP- hard problem and the decisions is complicated. Genetic Algorithm is applied to develop the heuristic method. The solution is compared to the brute force method for the small problem size with the number of job is up to 5 jobs. General schedule decision is based on Earliest Due Date or EDD and Earliest Completion Time or ECT for machine selection. The machine selection decision based on ECT is choosing the fastest machine available for every operation. The decision related to job sequencing and machine selection of each stage affect directly to job waiting in shop floor and the completion time. Thus, the sequence aiming to minimize tardiness may result in high work-in-process. On the other hand, balancing work of each station may affect the job completion time. Therefore, the scheduling decisions impact on interstage waiting time (used as a time-based proxy of WIP) and tardiness. Both measures are subsequently integrated into a unified objective function to reflect overall system performance in FFS-UPM environments.

The research methodology is to develop heuristic based on genetic algorithm. The objective function is to minimize the fitness function composing of tardiness and interstage waiting time. The developed heuristic is validated by comparing the solutions to the solutions from brute force method with the small problem of five jobs.

II. Research Methodology

A. Problem Description and Assumptions

This research aims to solve for the job schedule of Flexible Flow Shop with Unrelated Parallel Machines (FFS-UPM) problem with n jobs and a operations. Each operation composes of m machines in parallel and non-identical. The related notation is described in table 1.

TABLE I. NOTATION USED IN THE MATHEMATICAL MODEL

Symbol	Description
i	Job Index, i = 1, 2, …, n
j	Operation Index, j = 1, 2, …, a
k	Machine Index, k = 1, 2, …, m
n	Number of jobs
a	Number of operations
m	Number of machines of each operation
J_i	Job i
M_j	Set of machines of operation j
$p_{i,j,k}$	Processing time of job i at stage j on machine k
d_i	Due date of job i
$S_{i,j}$	Completion time of job i at stage s
C_i	Completion time of job i at the final stage
T_i	Tardiness of job i, $T_i = \max(0,\ C_i - d_i)$
$W_{i,j}$	Interstage waiting time of job i between stage s and s+1
WIP_wait_{total}	Total interstage witing time,
θ	Due-date tightness parameter
k	Scaling parameter in the objective function
$\bar{p}$	Average processing time ($p_{i,j,k}$) computed by averaging of the total processing time of all jobs, which is computed from the total processing time for all operations while processing time of each operation computed from the average processing time of all machines.
α	Scaling coefficient, $\alpha = k\bar{p}$
p	Job permutation (job sequence vector defining the processing order across all stages)
A_m	Current available time of machine m
$ST_{i,j}$	Start time of job i at stage s (decision-dependent)

B. Instance Generation and Data Structure

To perform the replicated experimental design, all instances are prepared by performing deterministic randomness which generated from a fix starting point called a seed. Data of processing time for every job (i), every operation (j), and every machine (k) including the due date are recorded in Excel file. The processing time of job i, operation j, and machine k is generated according to discrete distribution of $p_{i,j,k} \sim U\{1{,}99\}$ for all $i = 1,2,\dots,n$, $j = 1,2,\dots,a$, and $k \in M_j$. To represent the best performance of the production, the due date of each job is determined from the minimum processing path of that job as the following equation.

$$\mathrm{P_{i,min}} = \sum_{j=1}^{a} min_k\{P_{i,j,k}\} \tag{2}$$

Then, due date is calculated as equation (3) depending on the tightness of due date value (θ).

$$d_i = \left\lceil \theta\, P_i^{\min} \right\rceil, \theta \in \{0.95, 1.15, 1.40\} \tag{3}$$

The performance measurement composes of the total tardiness and the interstage waiting time. The total tardiness is calculated as follow:

$$T_i = \max\,(0, C_i - d_i) \tag{4}$$

$$Total\ \ Tardiness = \sum_{i=1}^{n} T_i \tag{5}$$

The work-in-process is represented by interstage waiting time as follow:

$$W_{i,j} = \max\,(0, ST_{i,j+1} - C_{i,j}) \tag{6}$$

The relationship between interstage waiting time and work-in-process is

$$\mathrm{WIP\_wait_{total}} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \sum_{\mathrm{j=1}}^{\mathrm{a-1}} \mathrm{W_{i,j}} \tag{7}$$

Thus, the objective function is

$$Fitness = WIP\_wait_{total} + \alpha \sum_{i=1}^{n} T_i \tag{8}$$

When $\alpha = \kappa\bar{p}$, and $\bar{p}$ represents the scale normalization parameter of tardiness component. This research applies κ as constant value and equal to 10 ($\kappa = 10$). The $\bar{p}$ is calculated as follow:

$$\bar{p} = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{a} \frac{1}{|M_j|} \sum_{k=1}^{|M_j|} P_{i,j,k} \right) \tag{9}$$

The decision of starting time of job i at operation j, ($ST_{i,s}$) is

$$ST_{i,j} = \max\left(C_{i,a-1}, A_m\right) \tag{10}$$

and the completion time of job *i* at operation *j* is

$$C_{i,j} = ST_{i,j} + p_{i,j,k} \tag{11}$$

While A_m is available time of machine *m* at the current stage.

To evaluate the performance of the developed heuristic, the solution is compared to Brute force method for small size problem, since brute force enumeration can find the optimal solution for the small size problem. Therefore, the research plan is comparing the solutions from the developed heuristic to the Brute force method and the EDD+ECT method.

C. Genetic Algorithm Configuration

This research applied genetic algorithm (GA) to solve FFS-UPM problem under two criterions of fitness function. The genetic algorithm works well with multiple decisions which are job sequencing and machine assignment with the same chromosome by evaluating with the schedule simulation and fitness value calculation. The developed heuristic follows figure 2 and are described as follow.

Step 1 Chromosome Representation

The chromosome representation is divided into two parts. The first part is permutation gene which is the vector of *n* sequence job. The second part is machine-assignment gene which defines the selected machine of job *i* at operation *j* represented by $m(i,j)$. For example, the problem of four jobs ($n = 4$) with three operations ($a = 3$) and two identical machines ($m = 2$) for each operation, the number of gene in chromosome is 16 (from $4 + (4 \times 3)$).

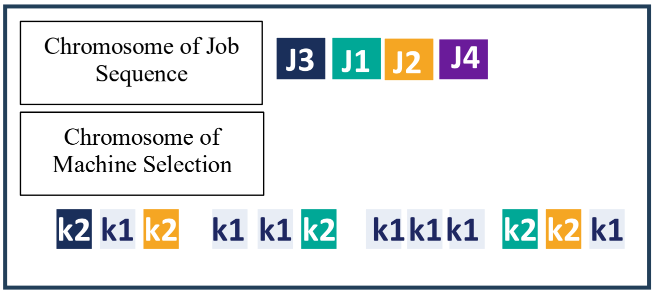

Fig. 1 The Chromosome Representation.

After decoding, the solution is job schedule which is used to simulate the starting and finishing time of the job in each stage. Then, $WIP_{wait,total}$ and tardiness are calculated to get the fitness value. The developed heuristic follows these steps.

Step 2 Initiation

Randomly generate an initial population of potential solutions (π). For every (i, j), randomly select machine from the set of machine of operation *j*.

Step 3 Fitness Evaluation

For each chromosome, decoding $(\pi, m(i,j))$ and generate job schedule to calculate $ST_{i,j}$ and $C_{i,j}$. Then, the total tardiness and the fitness value is calculated as equation (12) and equation (13).

$$\sum_{i=1}^{n} T_i = \sum_{i=1}^{n} \max\left(0, C_i - d_i\right) \tag{12}$$

$$Fitness = WIP\_wait_{total} + \alpha \sum_{i=1}^{n} T_i \tag{13}$$

This research applies $\alpha = \kappa\bar{p}$, where $\kappa = 10$ and constant to keep the stability and un-bias results. The κ value is scale normalization parameter of tardiness in fitness function to avoid the local optimal solution.

Step 4 Selection

The tournament is applied with the sample size of 3 ($k = 3$) [10]. By sample three samples, the minimal one is selected as the parent for the next step.

Step 5 Crossover

The crossover composes of two parts according to gene. For permutation gene, the Order Crossover is applied to keep legality and relative order preservation [10]. For machine gene, the one-point crossover to swap machine selection between parents. The crossover probability is 0.9 ($P_c = 0.9$).

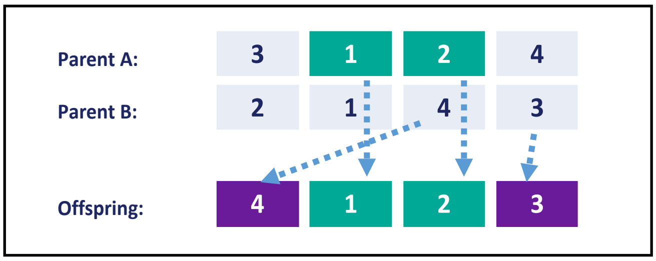

Fig. 2 The Example of Crossover of the Job Sequence

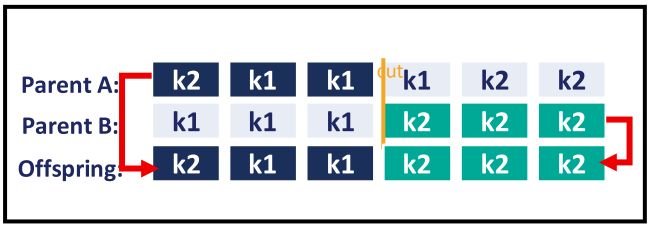

Fig. 3 The Example of One-Point Crossover of the Machine Selection of Job i and Operation k

Step 6 Mutation

The combination operations of mutation is applied to retain diversity of population, increase searching area, and avoid local optimal search. The permutation gene applied swap mutation and the machine gene applied the probability of changing machine of 0.25 ($P_m = 0.25$) to switch to another machine in the same stage. The mutation rate is constant for every instances.

Step 7 Elitism and Replacement

This research keeps the two best individuals (Elitism = 2) and replace the left with the offspring generation from selection–crossover–mutation to keep the solution quality.

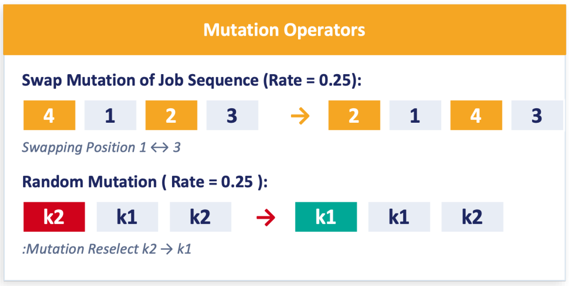

Fig. 4 The Example of Mutation Operation

Step 8 Termination and Parameter Settings

For every instance, the following parameters are set to terminate the developed algorithm.

Population size = 80

Number of generations = 250

$P_c = 0.9$, $P_m = 0.25$

Elitism = 2

$\kappa = 10$ and $\alpha = \kappa\bar{p}$

This research considers the tie breaking rule and comparison protocol as well. If there is more than one fitness value being equal, the following deterministic comparison decision is applied.

- Select the one with lower value of makespan, otherwise

- Select the solution with lower mean flow time

These criterions are applied to the brute force method, the developed heuristic with GA based, and the EDD+ECT.

The pseudo code description is as follow:

Input data: n, a, $P_{i,j,k}$, D, GA Parameters

Output: The Grant Chart of the job sequence.

1. Calculate the average processing time, for example, $\bar{P} = 27.875$, then $\alpha = k\bar{P} = 278.75$
2. Generate an initial population of 80 random chromosomes.
3. Evaluate the Fitness of every chromosome, then determine the current best solution.
4. Repeat for 250 generations:

4.1 Elitism: Copy the 2 best chromosomes to the next generation.

4.2 Tournament Selection: Randomly select 3 candidate chromosomes, then select the chromosome with the lowest Fitness value as a parent.

4.3 Genetic Crossover: Use Crossover OX [32] (job sequence gene) + One-point crossover (machine gene) with a crossover rate = 0.90.

4.4 Mutation: Use Mutation Swap (job sequence gene) + Random Reset (machine gene) with a mutation rate = 0.25 per gene.

4.5 Evaluate Fitness, then update the current best solution if it is better.

5. Output the chromosome with the lowest Fitness

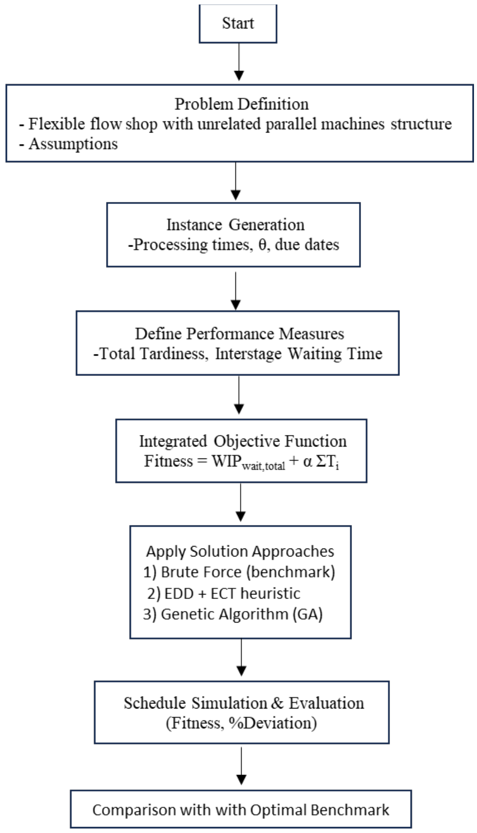

Fig 5. Overall Research Framework

III. Research Results

The research experiment is applying the developed heuristic (GA) to the instances of small problem size which the brute force method can solved. The performance of the developed heuristic is compared to the brute force method and the EDD+ECT method by calculating the average percentage deviation as equation (14) and (15), respective.

$$\%Deviation = \frac{Fitness_{GA} - Fitness_{Brute\ Force}}{Fitness_{Brute\ Force}} \times 100 \qquad (14)$$

$$\%Deviation = \frac{Fitness_{EDD+ECT} - Fitness_{GA}}{Fitness_{GA}} \times 100 \qquad (15)$$

A. Experimental Setup

The developed heuristic is applied to five jobs problem, since this is the maximum number of jobs that the brute force can solved in the acceptable running time. The experiment is performed on 5 instances or replications, and each instance is the five jobs scheduling FFS-UPM problem. The tightness factor of due date are set as three values which are 0.95, 1.15, and 1.40. The total experiment run is 15 for three levels of tightness value and five replicates.

$$\theta \in \{0.95, 1.15, 1.40\}$$

The fitness value from developed heuristics is compared to the solution from Brute Force by the term average percentage deviation.

B. Experimental Results

For the problem of five jobs (n = 5) of the FFS-UPM, the tightness (θ) is varied with three values which are 0.95, 1.15, and 1.40. The results in table II and table III shows that the develop heuristic (GA) gives the same solution as the brute force method for every tightness values (%Deviation = 0%).

TABLE II. AVERAGE ABSOLUTE FITNESS VALUES (N = 5)

θ	Fitness Value		
	Brute Force	GA	EDD+ECT
0.95	231,241	231,241	317,405
1.15	107,998	107,998	186,692
1.4	21,983	21,983	54,099

TABLE III. PERFORMANCE RELATIVE TO OPTIMAL SOLUTION (N = 5)

θ	Method	Average %Deviation	Maximum %Deviation
0.95	GA	0	0
0.95	EDD+ECT	37.88	62.59
1.15	GA	0	0
1.15	EDD+ECT	86.33	187.42
1.40	GA	0	0
1.40	EDD+ECT	144.51	279.33

From table II, when the value of θ increases which mean that the tightness due date is more flexible, the fitness value which composes of tardiness and interstage waiting time decreases, since the tardiness tremendously decreases with relaxing of job due date. From table III, the develop heuristic (GA) gives the same solution as the brute force method which is the optimal solution for every instances, since the value of %Deviation is zero for the small problem (n=5). When comparing to the method of earliest due date with earliest completion time of machine selection (EDD+ECT), the GA gives better solution. The value of %Deviation is 37.88%, 86.33%, 144.51% for the tightness value of 0.95, 1.15, and 1.40, respectively. From the results, when the tightness of due date increase, the EDD scheduling approach gives the inferior solution because the interstage waiting time and tardiness increase and can not be controlled. Therefore, the work-in-process increases when the due date tightness increases. On the other hand, the developed heuristic based on GA with the two objectives fitness function can solve for the job schedule in flexible flow shop with unrelated parallel machines. For the small problem with the number of job no more than five jobs, the developed heuristic gives the optimal solution which is proved by the brute force method.

The result supports the research concept of dual criteria fitness function of tardiness and interstage waiting time which give better stability in finding solution than one criteria fitness function. When θ increases, the fitness values for every cases decrease corresponding to the more flexible due date, the lower tardiness value.

C. The Results for Large Problem

The developed heuristic is applied to simulate and solve problems with 20, 50 jobs and increasing the number of machines of each operation from 2 to 3 machines. The results are illustrated in table IV. The developed heuristic can solve the large problems. When the number of machine increases, the total tardiness decreases. From the sensitivity analysis results in table IV, with the setting fitness function, increasing the number of machines for 2 to 3 with the large problem (50 jobs) does not improve the Total WIP_wait to represent interstage waiting time, while the total tardiness is improved significantly.

TABLE IV. THE SIMULATION RESULTS OF IMPLEMENTING DEVELOPED HEURISTICS TO LARGE PROBLEMS

No. of machines	No. Job	Total WIP_wait	Total Tardiness	Makespan	Mean Flow Time
2–2–2	20	1,168	4,127	662	338.5
3–3–3	20	807	3,127	505	256.7
2–2–2	50	6,380	30,297	1,430	724.0
3–3–3	50	6,433	23,897	1,090	569.7

IV. CONCLUSION

This research proposes the heuristic based on genetic algorithm with dual criteria fitness function which are tardiness and interstage waiting time to solve for the job schedule in flexible flow shop with unrelated-parallel machine problem. For the small size problem (n = 5), the heuristic gives the same solution as the brute force method. In addition, the heuristic gives the better solution than the Earliest Due Date with Earliest Completion Time, since the deviation percentage is 37.88%, 86.33%, 144.51% for the due date tightness value of 0.95, 1.15, and 1.40, respectively. In conclusion, the due date tightness increases, the fitness value decreases. When the due date increases, the EDD approach focuses on only decreasing tardiness not the interstage waiting time. Therefore, the dual criteria fitness function is more efficient to solve for a job schedule that minimize tardiness and work-in-process than applying EDD approach.

REFERENCES

[1] J. Blazewicz, K. H. Ecker, E. Pesch, G. Schmidt, and J. Weglarz, Scheduling Computer and Manufacturing Processes, 2nd ed. Berlin, Germany: Springer, 2001.

[2] E. Taillard, “Benchmarks for basic scheduling problems,” European Journal of Operational Research, vol. 64, no. 2, pp. 278–285, 1993.

[3] R. Ruiz and J. A. Vázquez-Rodríguez, “The hybrid flow shop scheduling problem,” *European Journal of Operational Research*, vol. 205, no. 1, pp. 1–18, 2010.

[4] C. Yu, Y. Chen, and X. Li, “A genetic algorithm for the hybrid flow shop scheduling with unrelated machines and machine eligibility,” *Computers & Operations Research*, vol. 100, pp. 211–230, 2018.

[5] J. D. C. Little, “A proof for the queuing formula: L = λW,” Operations Research, vol. 9, no. 3, pp. 383–387, 1961.

[6] W. J. Hopp and M. L. Spearman, Factory Physics, 3rd ed. New York, NY, USA: McGraw-Hill, 2011.

[7] M. Braglia, M. Frosolini, R. Gabbrielli, and F. Zammoria, “CONWIP card setting in a flow-shop system with a batch production machine,” International Journal of Industrial Engineering Computations, vol. 2, no. 1, pp. 1–18, 2011.

[8] B. Bokor and K. Altendorfer, “Extending ConWIP by flexible capacity and WIP-Cap adjustment for a make-to-order multi-item multi-stage production system,” Flexible Services and Manufacturing Journal, vol. 37, pp. 443–472, 2024.

[9] 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,” *Computers & Operations Research*, vol. 36, no. 2, pp. 358–378, 2009.

[10] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, MA, USA: Addison-Wesley, 1989