The Development of Simulated Annealing Algorithm for Hybrid Flow Shop Scheduling Problem to Minimize Makespan and Total Tardiness

Budi Santosa and Ainur Rofiq
Department of Industrial Engineering
Institut Teknologi Sepuluh Nopember
Surabaya, Sukolilo 60111, Indonesia

Abstract

Hybrid Flowshop Scheduling (HFS) is one of the most comprehensively studied in scheduling problems. HFS has m-machines in each stage so that it is more complicated that regular flowshop scheduling. There are several production systems using HFS, such as steel manufacture, textile, and paper industry [8] [15]. Makespan and total tardiness are two important goals in scheduling to make the scheduling more efficient and to fulfill customer demands. HFS is also NP Hard problems, because the bigger problem will need more time for computation to solve the problem. This research offers simulated annealing algorithm to solve this problem to get minimum makespan and total tardiness. Simulated annealing had used for scheduling problems before and this algorithms gives the good performance. Simulated annealing has the strength to get out of local optima by receiving the worse solution. In this research, this algorithm will be modified by increasing the temperature once when temperature had reached very small value. From the experiment done, modified simulated annealing gives the better performance than regular simulated annealing, especially in the bigger problem.

Keywords
Hybrid Flowshop Scheduling, Makespan, Simulated Annealing, Total Tardiness

1. Introduction
Scheduling is one of the critical issues in a manufacturing system [1]. The problem in scheduling focuses on how to allocate the limited resources of production, such as machinery, material handling, operators, and other equipment to carry out the process in a series of operating activities (job) in a certain period of time to optimize certain objective function [17].

Hybrid flowshop (HFS) is the expansion of flowshop scheduling which has parallel machines in each stage. According to [4], the term hybrid flowshop firstly introduced by Gupta in 1998, in describing the flowshop scheduling with one machine on the first stage and two machines at the second stage. There are some objective functions of the flowshop scheduling problems, such as for minimizing makespan, minimizing the mean flow time, minimizing tardiness, and so on.

Flowshop scheduling problem, including a hybrid flowshop problem is non-polynomial hard (NP-Hard) problem because the bigger problem requires more time to get the optimal solution. Thus, the use of exact methods, such as branch and bound, linear programming and Lagrangian relaxation is not effective enough and needs other method which is able to give effective in terms of results and computation time.

More than an objective function is closer to the real existing conditions, so that scheduling problems become more complex with multi-objective. Therefore, it is required simultaneous consideration of multiple goals to generate the schedule so that it is able to optimize some objectives. Utility function approach is a method often used in multi-objective problem, in which each objective will be given the weight suits in order of priority. The purpose of this research is to optimize two objective functions, makespan and total tardiness in hybrid flowshop scheduling. Simulated annealing (SA) is a metaheuristic method that imitates the process of hot steel cooling slowly [20]. In some researches, this method is used as an optimization method for solving problems, such as traveling salesman problem, vehicle routing problem, scheduling jobs, and any other problems.

Several researches of hybrid flowshop scheduling also use the simulated annealing method. For example [13] applying simulated annealing method on HFS with the objective of minimizing total completion time and total tardiness with sequence dependent setup time. In [12] the simulated annealing method was applied in hybrid flowshop scheduling to minimize total completion time and total tardiness with sequence dependent setup time and transportation time, and any other researches. Simulated annealing method will also be used in this research in hybrid flowshop scheduling problem to minimize makespan and total tardiness with release date, where each job can only start to be processed after the release date. This research has not been conducted before. Simulated annealing will be modified by increasing the value of the temperature once when it reaches very little value.

1. Mathematical Model of Hybrid Flowshop (HFS) Scheduling Problem

Before testing the algorithm, it is described the model of the problem that will be solved. Here is a mathematical model consists of the objective function (minimization of makespan and total tardiness) and its constraints quoted from Liao et al. [8]

j: job index
s: stage index
i: machine index
n: number of job; j = 1,2,3,… n.
k: number of stages; i = 1,2,3,… k.
m_s: number of machines at stage s; s=1,2, …k.
p_{js}: processing time of job j at stage s; s = 1,2,…k and j = 1,2,…n
S_{js}: starting time of job j at stage s
C_{js}: finishing time of job j at stage s
L: a very large constant number
X_{jis}: binary variable equal to 1 if job j assigned to machine I at stage s; 0 otherwise
Y_{hjs}: binary variable equal to 1 if job j precedes job I at stage 2; 0 otherwise

Minimize C_{max} (1)
Minimize T (2)
Subject to:
$C_{max} \geq C_{js}$ s = 1, …, k j=1, …, n (3)
$C_{js} = S_{js} + P_{js}$ s = 1, …, k j=1, …, n (4)
$\sum_{i=1}^{m_s} X_{jis} = 1$ s = 1, …, k j=1,…, n (5)
$C_{js} \leq S_{j(s+1)}$ s = 1, …, k-1 (6)
$S_{hs} \geq C_{js} - LY_{hjs}$ for all pairs of job (h,j), s = 1, …, k (7)
$S_{js} \geq C_{hs} - (1-Y_{hjs})L$ for all pairs of job (h,j), s = 1, …, k (8)
$S_j \geq R_j$ j=1, …, n (9)
$C_j^0 = r_j$ (10)
$T = \sum_{j=1}^{n} \max(C_{js} - d_j, 0)$ (11)
$X_{jis} \in (0,1)$, $Y_{hjs} \in (0,1)$all j = 1, …, n; i = 1, …, m_s; s = 1, …, k (12)

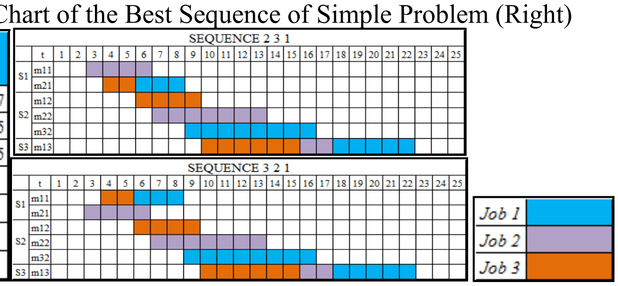
The objective is to minimze makespan (C_{max}) and total tardiness (T_{max}). Constraint (3) and (4) used to define makespan. Makespan obtained at least greater than or equal to the completion time of the last job. Constraint (4) is the completion time of job j at stage s. Constraint (5) is used to ensure each job is processed by one machine in each stage. Constraint (6) is used to ensure that each job can be processed in a stage starts only after the job is finished processing the previous stage. Constraint (6) and (7) are working together to ensure that each machine can process only one job at a time. When Y_{hjs}=1, and h job done before job j, constraint (7) is required. Constraint (8) shows that the starting time job j on stages must after completion time job h. When Y_{hjs}=0, then job j should be processed before job h. Constraint (9) is used to limit the time that the starting job should have after the ready time on the job. Constraint (10) is used specifically for when the job has not been processed, where the job can only be processed after the release date of the job. Constraint (11) is used to demonstrate the calculation of total tardiness time. Total tardiness is the sum of the maximum completion time difference with a predetermined due date. Constraint (12) is used to create variables on X_{jis} and Y_{hjs} binary value 0 or 1.

2. Development of Simulated Annealing Algorithm

After describing the model of the problem, then it will be doing validation and algorithm used before developing algorithm. Validation will be done is by enumeration of the simple problem and the along with the steps performed by using the proposed algorithm. Solving problem with enumeration is to try all the possibilities that exist with manual calculations. In the example with enumeration, it will use an example of a simple problem of hybrid flowshop scheduling with 3 jobs and 3 stages with 6 possible combinations of sequence job.

Table 1 Example of Simple Problem (Left) and Gantt Chart of the Best Sequence of Simple Problem (Right)

Job	*Release date*	Waktu Proses			*Due date*
		Stage 1	*Stage 2*	*Stage 3*	
j1	1	3	8	5	17
j2	2	4	7	2	15
j3	3	2	4	6	15
Jumlah Mesin					
Stage 1		2			
Stage 2		3			
Stage 3		1			



Here is the recapitulation of makespan and total tardiness results from all combinations sequence of 3 job and 3 stages of hybrid flowshop scheduling problem with release date.

Table 2 The Result of Simple Problem

NO	SEQUENCE	*Makespan*	*Total Tardiness*	Z
1	1--2--3	23	13	18
2	1--3--2	22	10	16
3	2--1--3	23	13	18
4	2--3--1	22	7	14.5
5	3--1--2	22	10	16
6	3--2--1	22	7	14.5

Z value obtained from the sum of the individual objective functions that have been previously assigned weights. In this problem, makespan and total tardiness is considered equally important so that both the objective function is given equal weight, 0.5. From table 3, it can be concluded that the job sequence 2-3-1 and 3-2-1 have the smallest z value, so both job sequence is order the most optimal in this simple problem.

Then, the following are the steps of simulated annealing algorithm. Here is an illustration of the difference between regular simulated annealing and modified simulated annealing.

Figure 1 The Difference of Regular SA Cooling Schedule (Left) and Modified Simulated Annealing Cooling Schedule (Right)

These are steps in developing simulated annealing algorithm to solve the scheduling problem in hybrid flowshop simple problem above.

Step 1: Initializing Parameter

The parameters used in simulated annealing algorithm in this simple problem are as follows:
Initial temperature (To) = 50; Reduction factor = 0.4; Temperature reduction cycle = 2

Step 2: Initializing Initial Solution
Generating initial samples obtained initial objective function is being done by using a random sequence of job. The example results generated job sequence is 1-3-2. The objective function with the initial job sequence 1-3-2, are Makespan = 22 and Total tardiness = 10

Step 3: Determining Initial Iteration and Initial Cycle of Temperature Reduction
This step is the first step before going inside the iteration, is specifying the start of iteration. In this step, the iteration starts from a value of 0, which means that the iteration is not yet started. It is also for temperature reduction cycle.

Step 4: Generating Random Number to Determine Swap, Slide, and Flip to Obtain New Sequence
This step is used to obtain new solutions by swap, slide, or flip. If the random number generated between 0 – 0.33, flip method will be used. If the random number generated is between 0.34 - 0.67, then the swap method will be used. And then if the random number generated is between 0.68 - 1, then slide method will be used.
Random number generated was 0.23 which means flip method is used. The job will be flipped is job 2 and job 3. So, the previous sequence and the new sequence are:
Previous sequence → 1-3-2
New sequence →1-2-3

Step 5: Determining New Objective Function Value of the New Sequence
After getting the new sequence, then the new objective value of the new sequence is generated. New makespan and total tardiness of the new sequence (1-2-3) are Makespan = 23 and Total tardiness = 13

Step 6: Comparing Previous Solution and New Solution
Getting the single objective function from multiple objective functions is using utility approach function. In this research, makespan and total tardiness objective function are equally important and given equal weight, 0.5.
Previous Solution (Makespan = 22, total tardiness = 10)
$Z_1 = w_1*f(x_1) + w_2*f(x_2)$
$Z_1 = 0.5*22 + 0.5*10 = 16$
New Solution (Makespan = 23, total tardiness = 13)
$Z_2 = w_1*f(x_1) + w_2*f(x_2)$
$Z_2 = 0.5*23 + 0.5*13 = 18$

In simulated annealing algorithm, if the new solution is better than the old solution, the new solution will be accepted. However, if the new solution is not better than the old solution, it will be calculated by using metropolis criteria to determine whether new solution that is not better than the previous solution will be accepted or rejected and it will be compared with the results of random numbers generated. If the random number is smaller than the metropolis criterion, the "worse" new solution will be received, and otherwise. It is known that the new solution is no better than the old solution. Therefore, the calculation of Metropolis criteria is written as below.
$\Delta E = Z_2 - Z_1 = 18 - 16 = 2$; T = Recent temperature = 50
$P(E) = e^{-\Delta E/kT} = 0.9608$
ran = random number generated = 0.5639
Because ran<P (E), the new solution will be the previous solution for the next iteration.

Step 7: Stopping Criteria
Stopping criteria used is when the value of temperature is very small, which is 0.00000001.

3. Experimental Results
Data testing used comes from research journal conducted by Carlier and Neron [3] in 5 job 5 stages, 5 stages and 10 jobs and 15 jobs 10 stage. Then, data testing of Mousavi et al [11] used for the case of 20 jobs 10 stages and 30 jobs 10 is also taken. Only several data testing of both journals used the measure the performance of the algorithm. Besides, there is also generating additional data such as the data release date and due date.

Table 4 Configuration of Data Testing

Faktor	**Data Uji**				
Jumlah tahap operasi (*stage*)	5		10		
Jumlah *job*	5	10	15	20	30
Jumlah mesin paralel tiap tahap operasi	3-1-3-3-1	2-1-4-2-3	5-4-6-5-3-4-3-1-2-2		
waktu proses *job* tiap tahap operasi	U(3,20)		U(20,100)		
release date job	U(1,10)				
due date job	U(*μ*ns,μ*(nj/(nm/ns))+ns-1)*				

The first experiment conducted is the experimental of parameter testing. Parameters are important elements in the algorithm so that it needs to be tested to obtain good parameter values in order to obtain better results. The first parameter testing is temperature reducing factor testing (c) that ranges between 0 and 1. In this research, reduction factor parameter testing is a value of 0.2, 0.5, and 0.9. Parameter testing is conducted using data testing of 10 job 10 stage.

From the experimental results of the parameter testing in table 5, temperature reduction factor will be used is 0.5 because it gives the best performance than the other.

Then, parameter testing conducted for the initial temperature using the similar data testing. Initial temperature will be tested is 50, 200, and 1000 for 10 jobs 10 stages problem. The result of parameter testing of initial temperatures listed in table 6.

Table 5 Reduction Factor Parameter Testing

To = 200, c = 0.2													
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job										Waktu Komputasi
1	144	184	6	9	2	5	10	8	3	1	7	4	1.3728
2	144	195	2	9	6	10	5	8	3	1	7	4	1.3572
3	157	210	9	6	5	1	3	10	8	4	7	2	0.8736
To = 200, c = 0.5													
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job										Waktu Komputasi
1	144	177	6	9	2	5	10	1	3	8	7	4	1.7472
2	144	177	6	9	2	5	10	1	3	8	7	4	1.8408
3	144	177	6	9	2	5	10	1	3	8	7	4	2.2308
To = 200, c = 0.9													
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job										Waktu Komputasi
1	144	177	9	6	2	5	10	1	8	7	3	4	7.5816
2	144	177	9	2	5	6	10	1	3	8	7	4	9.4849
3	144	184	9	6	2	5	10	1	8	7	3	4	9.8125

Table 6 Initial Temperature Parameter Testing

To = 50, c = 0.5				
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job	Waktu Komputasi
1	144	203	5 9 6 10 8 1 2 3 7 4	1.6848
2	144	193	6 9 2 10 5 3 8 7 1 4	1.8876
3	146	211	8 6 2 5 10 1 9 3 7 4	1.5912
To = 200, c = 0.5				
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job	Waktu Komputasi
1	144	177	6 9 2 5 10 1 3 8 7 4	1.7472
2	144	177	6 9 2 5 10 1 3 8 7 4	1.8408
3	144	177	6 9 2 5 10 1 3 8 7 4	2.2308
To = 1000, c = 0.5				
No Percobaan	*Makespan*	*Total Tardiness*	Urutan Job	Waktu Komputasi
1	154	172	9 6 2 5 10 3 1 8 4 7	1.8096
2	144	193	9 6 2 5 8 1 10 3 7 4	1.6848
3	154	193	5 9 2 6 1 10 4 8 3 7	2.0904

From the result of initial temperature parameter testing, the best initial temperature is using the value that approach with the result of the objective value. So, the bigger problem will use the bigger initial temperature, too.

After parameters testing, then the experiment using regular simulated annealing and simulated annealing modified will be conducted on five problems of the existing data from the simple problem to the complex problems. Each problem will be experimentally tested 30 times of each algorithm.

Table 7 Experimental Results

No	Problem	Average of Makespan Modified SA	Average of Makespan Regular SA	Makespan GAP Average	Average of Total Tardiness Modified SA	Average of Total Tardiness Regular SA	Total Tardiness GAP Average
1	5 Job 5 Stage	109	109	0.00%	97	97	0.00%
2	10 Job 5 Stage	145	145	0.00%	177.1	182	2.53%
3	15 Job 10 Stage	1285.73	1299.37	1.05%	2458.33	2717.23	9.53%
4	20 Job 10 Stage	1714	1737.37	1.34%	5971.87	6285.8	4.99%
5	30 Job 10 Stage	2011.6	2036.23	1.21%	9342.67	10621.6	12.04%

4. Analysis

In the experimental of parameter testing, the value of reducing factors tested was 0.2, 0.5, and 0.9. From the results of experiments that have been conducted on these three values of reduction factor the parameter test with 3 trials for each value; the best results are obtained on reducing the value of factor 0.5 compared with a value of 0.2 and a reducing factor of 0.9. The value of 0.2 doesn’t give the good performance. Then, when reduction factor of 0.9 compared with the reduction factor of 0.5, the solution obtained does not differ much, but the computation time required by using a reduction factor of 0.5 is shorter. It can be concluded that by using a reducing factor of 0.5 is enough to get a good solution with a reliable computation time.

In the experiment conducted for 5 job 5 stage using regular simulated annealing algorithm and modified simulated annealing, the obtained solution is same in 10 running for each algorithm. This is because the data used to test both algorithms is still relatively small problem so that both algorithms are still able to give optimal results. Then, in the results of experiments conducted on both regular SA and modified SA algorithm using data testing 10 jobs and 5 stages, there is the difference solution obtained even though the difference is not too significant. However, from the experiment results of 10 with 10 jobs 5 stages was already appeared that the performance of modified SA is little much better than regular SA, where the modification of modified SA is by increasing the temperature once again after the temperature reached very small value. Modified SA is effective enough to avoid the solution from a local optima trap.

From the experimental results of the two algorithms in the problem of 15 jobs 10 stages and 20 jobs 10 stages, both algorithms showed quite different results where the modified SA gives better performance than regular SA. On this test, the replication is conducted in 30 times for each algorithm and the results obtained often show different solutions. It also shows the diversity of the results of a solution for the bigger problem. The biggest problem in the experiment is using 30 job 15 stages and modified simulated annealing gives the better results than regular simulated annealing. The average gap for makespan is 1.21% and total tardiness is 12.04%.

The performance of modified SA on big problems showed that the development of the algorithm carried out effectively enough to make the solution get out of the local optima trap, which is a weakness of ordinary SA algorithm. Regular SA will be difficult to get out of local optima when the temperature reached little value, even though there is no guarantee the result in the little value of temperature obtained is approaching optimal solution or still trapped on local optima.

5. Conclusion

Regular simulated annealing algorithm and modified simulated annealing can be used in solving the hybrid flowshop scheduling problems with release date to minimize makespan and total tardiness. Then, modified simulated annealing by increasing the temperature when the temperature reached very little value proven to give better solutions than regular simulated annealing. Performance of modified SA algorithm starts to look better when the data used include big problem.

Acknowledgement

The authors thank to the reviewers for the comments and suggestions that improved the research of the paper.

References

Allahverdi, A., Gupta, J. N. D., & Aldowaisan, T., A review of scheduling research involving setup considerations, 27, 1999.

Behnamian, J., Ghomi, S. M. T. F., Jolai, F., & Amirtaheri, O., Minimizing makespan on a three-machine flowshop batch scheduling problem with transportation using genetic algorithm, Applied Soft Computing Journal, 12(2), 768-777, 2012.

Carlier, J., & Neron, E., An exact method for solving the multiprocessor flowshop, R.A.I.R.O- R.O, 34, 1-25, 2000.

Gómez-gasquet, P., Andrés, C., & Lario, F.-c., An agent-based genetic algorithm for hybrid flowshops with sequence dependent setup times to minimise makespan, 39, 8095-8107, 2012.

Jungwattanakit, J., Reodecha, M., Chaovalitwongse, P., & Werner, F, Constructive and Simulated annealing Algorithms for Hybrid flowshop Problems with Unrelated Parallel Machines, Thammasat Int. J. Sc. Tech.,Vol. 12, No. 1, 2007.

Khabbazi, M. R., A simulated annealing algorithm approach to hybrid flowshop scheduling with sequence-dependent setup times, 965-978, 2011.

Lee, G.-c., Estimating order lead times in hybrid flowshops with different scheduling rules, Computers & Industrial Engineering, 56(4), 1668-1674, 2009.

Liao, C.-j., Tjandradjaja, E., & Chung, T.-p. (2012). An approach using particle swarm optimization and bottleneck heuristic to solve hybrid flowshop scheduling problem. Applied Soft Computing Journal, 12(6), 1755-1764.

Liu, H., Gao, L., & Pan, Q. (2011). A hybrid particle swarm optimization with estimation of distribution algorithm for solving permutation flowshop scheduling problem. Expert Systems with Applications, 38(4), 4348-4360.

Moursli, O., & Pochet, Y., A branch-and-bound algorithm for the hybrid flowshop, 64, 113-125, 2000.

Mousavi, S. M., Zandieh, M., & Yazdani, M., A simulated annealing / lokal search to minimize the makespan and total tardiness on a hybrid flowshop. 369-388, 2013.

Naderi, B., Tavakkoli-moghaddam, R., & Khalili, M., Electromagnetism-like mechanism and simulated annealing algorithms for flowshop scheduling problems minimizing the total weighted tardiness and makespan, Knowledge-Based Systems, 23(2), 77-85, 2010.

Naderi, B., Zandieh, M., Ghoshe, A. K., & Roshanaei, V., An improved simulated annealing for hybrid flowshops with sequence-dependent setup and transportation times to minimize total completion time and total tardiness, Expert Systems with Applications, 36(6), 9625-9633, 2009.

Naderi, B., Zandieh, M., & Roshanaei, V., Scheduling hybrid flowshops with sequence dependent setup times to minimize makespan and maximum tardiness, 1186-1198, 2009.

Niu, Q., A Quantum-Inspired Immune Algorithm for Hybrid flowshop with Makespan Criterion, 15(4), 765-785, 2009.

Pan, Q.-k., Fatih, M., & Liang, Y.-c.,A discrete differential evolution algorithm for the permutation flowshop scheduling problem, Computers & Industrial Engineering, 55(4), 61-82, 2008.
Pinedo, M. L., Scheduling:Theory, Algorithms, and Systems (2 ed.). USA: Springer, 2002.
Riyanto, O. A. W. , Algoritma Differential Evolution-Variable Neighborhood Search untuk Minimasi Makespan dan Maximum Lateness pada Penjadwalan Job Hybrid flowshop with Job-Sequence Dependent Setup-Time, Institut Teknologi Sepuluh Nopember, Surabaya, 2011.
Ruiz, R., & Vázquez-rodríguez, J. A., The hybrid flowshop scheduling problem, European Journal of Operational Research, 205(1), 1-18, 2010.
Santosa, B., & Willy, P., Metoda Metaheuristik Konsep dan Implementasi. Surabaya: Guna Widya, 2011.
Wang, X., & Tang, L., A tabu search heuristic for the hybrid flowshop scheduling with finite intermediate buffers, 36, 907-918, 2009.

Biography

Budi Santosa is a professor and Head of Department at Industrial Engineering Department, Institut Teknologi Sepuluh Nopember (ITS), Surabaya Indonesia. He earned B.S. in Industrial Engineering from Institut Teknologi Bandung (ITB), Indonesia, Masters and PhD in Industrial Engineering from University of Oklahoma, USA. He has published journal and conference papers. Dr Budi Santosa has done research projects in application of methaeuristics techniques in logistics, transportation and scheduling. His research interests include optimization, metaheuristics, scheduling and data mining.

Ainur Rofiq is Bachelor of Industrial Engineering, Institut Teknologi Sepuluh Nopember (ITS) Surabaya. His study is majoring in computation and optimization. He is also laboratory assistant and lecturers' assistant for several disciplines, such as operation research, engineering optimization, simulation, and statistic. Becoming a laboratory assistant, he got many experiences and knowledge, especially in industrial and computation discipline. As a student, he is also active in academic and non academic field so that he got many experiences coordinating some events held in ITS Surabaya.