



Jurnal Politeknik Caltex Riau

<https://jurnal.pcr.ac.id/index.php/elementer/>

| e- ISSN : 2460-5263 (Online) | p- ISSN : 2443-4167 (Print) |

---

## Implementasi *Hybrid Algorithm* Untuk Optimalisasi Konsumsi Energi Pada *Job Shop Scheduling*

Ferdinan Rinaldo Tampubolon<sup>1)</sup>, Rischa Devita<sup>2)</sup>

<sup>1)</sup>Teknik Elektro, Politeknik Negeri Medan, Jl.Almamater No 1, Medan, 20155

<sup>2)</sup>Teknik Elektro, Politeknik Negeri Medan, Jl.Almamater No 1, Medan, 20155

E-mail: ferdinantampubolon@polmed.ac.id (12 pt)

### [1] Abstrak

*Air pollution is one of the most concerned issues nowadays. One of the ways to reduce that issues is energy efficiency. This research will be focused in solving job shop scheduling which not only minimizing the makespan but also reducing energy consumption. This kind of scheduling will be solved by using Hybrid Algorithm that is the combination of Genetic Algorithm and Simulated Annealing. Simulated Annealing is a local search technique which can enhance the solution provided by Genetic Algorithm*

**Keywords:** *genetic, simulated annealing, makespan energy*

### [2] Abstract

*Pencemaran udara menjadi salah satu hal yang sedang banyak dibicarakan akhir-akhir ini. Salah satu cara untuk mengurangi pencemaran udara adalah dengan melakukan efisiensi terhadap konsumsi energi. Penelitian ini akan berfokus pada penyelesaian job shop yang tidak hanya meminimumkan waktu penyelesaian pekerjaan akan tetapi juga konsumsi energi. Permasalahan tersebut akan diselesaikan dengan menggunakan Algoritma Hybrid yaitu gabungan metode Hybrid Genetika dan Simulated Annealing. Algoritma Simulated Annealing digunakan untuk mengembangkan pencarian lokal pada solusi yang sudah diperoleh pada algoritma genetika.*

**Kata Kunci:** *genetika, simulated annealing, makespan, energy*

---

### 1. Pendahuluan

Pencemaran udara menjadi permasalahan yang sering dibicarakan akhir-akhir ini. Efeknya dapat mengakibatkan kerusakan lingkungan, perubahan iklim dan mengganggu kesehatan. Beberapa hal menjadi penyumbang polusi udara seperti kendaraan bermotor, industri dan lain-lain. Industri menghasilkan polusi udara melalui proses produksi yaitu dengan penggunaan energi pada mesin.

Berdasarkan penelitian (Cooper, Rossie, & Gutowski, 2017) banyak penggunaan energi pada industri yang tidak berkaitan langsung dengan proses produksi. Hal ini mengimplikasikan perlu adanya efisiensi penggunaan energi pada dunia industri, dalam hal ini adalah penjadwalan yang optimal untuk setiap mesin dan pekerjaan yang dilakukan pada interval waktu tertentu.

Penjadwalan *Job Shop* merupakan salah satu jenis penjadwalan yang digunakan pada dunia industry. Penjadwalan ini berkaitan dengan mengalokasikan sejumlah pekerjaan pada sebuah tiap mesin. Belakangan ini penelitian terkait *Job Shop* banyak difokuskan untuk tidak hanya mencari waktu penyelesaian akhir (*makespan*) akan tetapi juga keterlambatan (*tardiness*) dan konsumsi energi. Dengan demikian terdapat lebih dari satu fungsi tujuan dan menambah tingkat kesulitan penyelesaian permasalahan. Penyelesaian penjadwalan *job shop* dibagi menjadi dua metode yaitu eksak dan heuristik.

Penelitian menggunakan metode eksak dilakukan oleh (Ku & Beck, 2016) untuk mencari *makespan* dengan bantuan software CPLEX. Penggunaan metode heuristik dilakukan oleh (Kalshetty, Adamuthe, & Kumar, 2020) dengan menggunakan Algoritma Genetika. Penggunaan algoritma *simulated annealing* dilakukan oleh (Chakraborty & Bhowmik, 2015) untuk meminimumkan *makespan*. Penelitian yang melibatkan lebih dari satu fungsi tujuan yaitu *makespan* dan *tardiness* dilakukan oleh (Scaria, George, & Sebastian, 2016) dengan menggunakan algoritma *bee colony*.

Untuk penelitian yang berkaitan dengan konsumsi energi dilakukan oleh (Jiang, Zhang, Zhu, Gu, & Deng, 2018) dengan menggunakan *whale algorithm*, sementara (Yin, Li, Gao, Lu, & Zhang, 2017) dan (Piroozfard, Wong, & Wong, 2018) masing-masing dengan algoritma Genetika dan NSGA untuk kasus *flexible job shop*.

Penelitian ini akan menggunakan *hybrid algorithm* yaitu gabungan algoritma genetika dan *simulated annealing*, penggunaan gabungan dua algoritma sudah dilakukan oleh (Stastny, Skorpil, Balogh, & Klein, 2021) yaitu penyelesaian dengan fungsi tujuan *makespan* menggunakan gabungan Algoritma Genetika dan *Mean Graph Based*. Pada penelitian ini Algoritma Genetika akan digunakan untuk menemukan suatu solusi optimal pada penjadwalan. Kemudian pencarian lokal (*local search*) dilakukan pada solusi tersebut dengan mencari tetangga (*neighborhood*) pada solusi tersebut sehingga dihasilkan solusi yang lebih optimal. Fungsi tujuan yang digunakan yaitu *makespan* dan konsumsi energi. Hasil yang diperoleh pada penelitian ini akan dibandingkan dengan beberapa penelitian sebelumnya (*benchmark*) untuk menguji apakah metode tersebut cukup akurat untuk digunakan.

## 2. Metode Penelitian

Penelitian ini dilaksanakan dengan terlebih dahulu membentuk Algoritma genetika dan *simulated annealing* (GA+SA) untuk menyelesaikan penjadwalan *job shop* dengan fungsi tujuan *makespan*. Kemudian algoritma tersebut akan diuji dengan beberapa penelitian untuk menyelesaikan beberapa kasus (*instances*). Semakin kecil nilai *makespan* maka akan semakin baik performa algoritma tersebut

Contoh bentuk penjadwalan *job shop*.

Tabel 1. Contoh Penjadwalan *Job Shop*

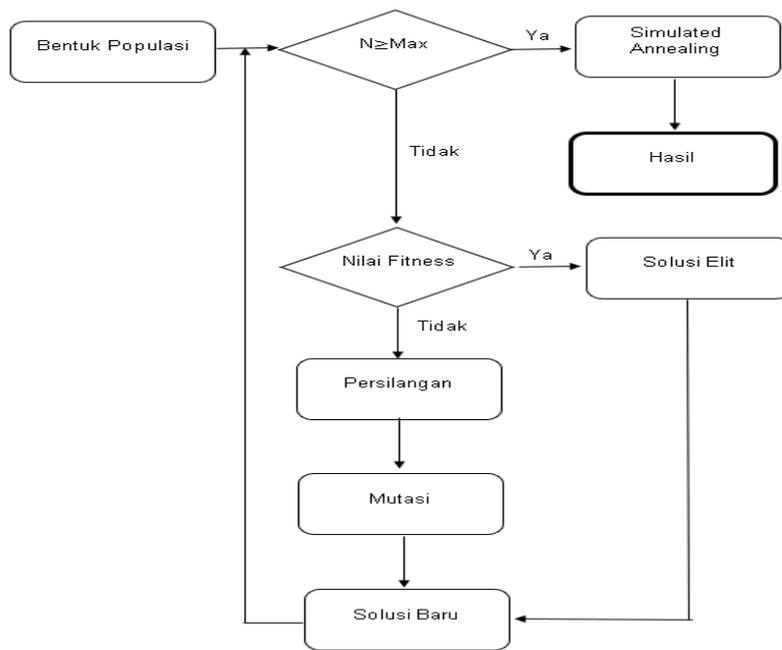
Pekerjaan	Operasi	Mesin-1	Mesin-2	Mesin-3
J <sub>1</sub>	O <sub>11</sub>	4	-	-
	O <sub>12</sub>	-	3	-
	O <sub>13</sub>	-	-	5
J <sub>2</sub>	O <sub>21</sub>	-	2	-
	O <sub>22</sub>	-	-	1
J <sub>3</sub>	O <sub>31</sub>	2	-	-
	O <sub>32</sub>	-	5	-
	O <sub>33</sub>	-	2	-

Algoritma GA+SA akan digunakan untuk menyelesaikan permasalahan *job shop* yang dimodifikasi tersebut. Populasi awal dibentuk dengan membangkitkan sejumlah kromosom dengan metode-metode acak, operasi terbanyak, dan waktu pengerjaan terbanyak. Gambar 1 menunjukkan Contoh kromosom *job shop* berdasarkan Tabel 1.



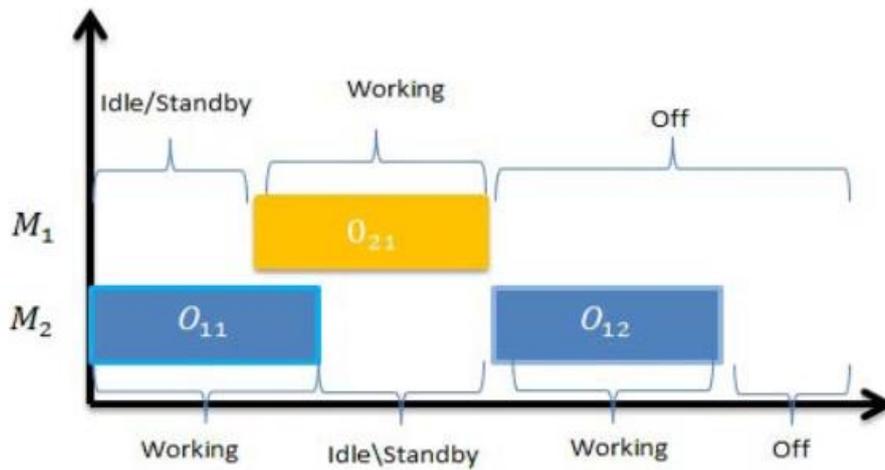
Gambar 1. Contoh Kromosom *Job Shop*

Nilai *fitness* dihitung berdasarkan fungsi tujuan  $F(x)$ . Pada penelitian ini terdapat dua fungsi tujuan sehingga diperlukan pembobotan pada masing-masing fungsi tujuan (Yin et al., 2017). Sejumlah kromosom yang memiliki nilai terbaik akan disimpan sehingga tidak mengalami proses persilangan maupun mutasi. Proses persilangan dilakukan dengan mempertukarkan beberapa gen antar dua kromosom sehingga menghasilkan dua kromosom baru. Mutasi dilakukan dengan mengubah susunan gen suatu kromosom. Proses ini dilakukan berulang sampai sejumlah iterasi yang ditentukan. Setelah suatu solusi ditemukan maka akan dilakukan proses *simulated annealing*, dimana gen-gen pada solusi tersebut akan dipertukarkan sampai suhu akhir tercapai. *Flowchart* Algoritma Genetika dan *Simulated Annealing* pada Gambar 2.



Gambar 2. Flowchart GA+SA

Kemudian kasus (*instances*) tadi akan dimodifikasi dengan menambahkan konsumsi energi pada saat mesin bekerja (*working*) dan pada saat mesin dalam keadaan *standby* (*idle*) untuk kemudian diselesaikan dengan Algoritma Genetika dan *Simulated Annealing*



Gambar 3. Diagram Chant Konsumsi Energi

### 3. Hasil Dan Pembahasan

Algoritma ini dijalankan dengan menggunakan software Matlab 2021a. Beberapa *instances* dari (Lawrence, 1984) yaitu la04, la08, la12, la16, la21, la29, la35 dan la38 digunakan untuk menguji performa Algoritma GA+SA. Parameter yang digunakan untuk algoritma genetika: jumlah populasi 100; jumlah iterasi 200; probabilitas *crossover* 0,5; probabilitas mutasi 0,6; probabilitas elit 0.1; Parameter yang digunakan untuk *simulated annealing*: Suhu awal 100; Suhu akhir 0,0001; penurunan suhu: 0.9; jumlah iterasi: 300. Algoritma dijalankan sebanyak 10 kali. Hasil yang diperoleh dari algoritma GA+SA akan dibandingkan dengan hasil *makespan* terbaik (*best known solution*), penelitian dari (Kalshetty et al., 2020) dan (Stastny et al., 2021). Hasil yang diperoleh ditampilkan pada Tabel 2

Tabel 2. Perbandingan Hasil dengan *benchmark*

Kasus	ukuran	Optimal	GA	GA+Graph Based		GA+SA		
				Makespan	Mean	Makespan	Mean	%dev
la04	10x10	590	604	595	641,77	590	604,9	0
la08	15x5	863	863	863	869,58	863	863,5	0
la12	20x5	1039	1039	1039	1039,11	1039	1042,4	0
la16	10x10	945	973	982	1057,65	969	1026,2	2,5
la21	15x10	1053	1159	1122	1230,36	1096	1338,3	4,08
la29	20x10	1152	1343	1316	1402,03	1270	1524,3	10,2
la35	30x10	1888	1962	1980	2115,15	1889	2366,9	0,05
la38	15x15	1196	1406	1381	1446,89	1298	1592,1	9

Dari tabel 2 terlihat algoritma sudah berjalan baik yaitu dengan persentase deviasi (perbedaan hasil dari (*best known solution*) dibawah 10 persen untuk tiap kasus kecuali pada kasus la29. Hasil makespan yang diperoleh juga cukup baik dibandingkan Algoritma Genetika, meskipun pada beberapa kasus memiliki nilai hasil rata-rata (mean) makespan lebih rendah dari algoritma GA dan Graph Based.

Untuk meminimumkan konsumsi energi maka ditambahkan variabel konsumsi energi mesin pada saat bekerja ( $P_{work}^k$ ) maupun pada saat idle ( $P_{idle}^k$ ). Bilangan acak untuk tiap-tiap mesin untuk posisi idle yaitu sembarang nilai dengan rentang (1,4) sementara untuk posisi bekerja yaitu (6,18).

Untuk menentukan nilai fitness maka terlebih dahulu dilakukan normalisasi baik pada makespan maupun konsumsi energy.

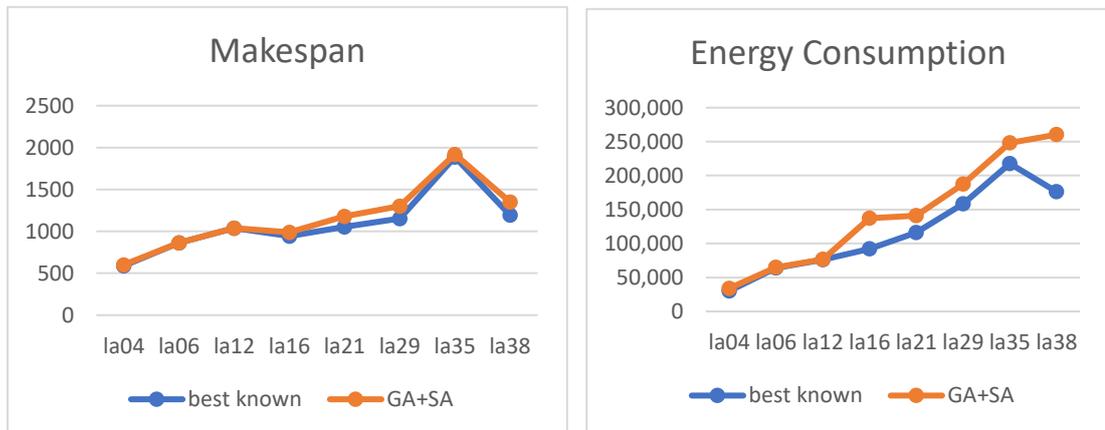
$$F(x) = w_1 \frac{C(x) - C_{min}}{C_{max} - C_{min}} + w_2 \frac{E(x) - E_{min}}{E_{max} - E_{min}}$$

$w_1$  merupakan bobot untuk fungsi tujuan waktu penyelesaian,  $w_2$  merupakan bobot untuk fungsi tujuan konsumsi energi.  $C(x)$  merupakan waktu penyelesaian untuk kromosom,  $E(x)$  adalah konsumsi energi pada kromosom,  $C_{max}$  dan  $C_{min}$  masing-masing menunjukkan waktu penyelesaian terbesar dan terkecil yang dapat diperoleh.  $E_{max}$  dan  $E_{min}$  masing-masing menunjukkan konsumsi energi terbesar dan terkecil yang dapat diperoleh.

Tabel 3. Hasil *Makespan* dan *Energy Consumption*

Kasus	Makespan		Energy		Fungsi Objektif		
	$C_{min}$	$C_{max}$	$E_{min}$	$E_{max}$	Makespan	Energy	F. Obj ( $F(x)$ )
1a04	590	2471	30.069	52.600	602	34.016	0,0908
1a08	863	3698	64.048	100.840	863	64.977	0,0126
1a12	1039	4453	76.048	119.070	1039	76.731	0,0079
1a16	945	4104	91.993	171.190	990	137080	0,2921
1a21	1053	6614	116.310	236.940	1180	140.910	0,1134
1a29	1152	8055	158.400	300.630	1300	187.520	0,1129
1a35	1888	5343	217.690	407.290	1920	248.000	0,0853
1a38	1196	7952	176.470	420.710	1350	260.380	0,1834

Tabel 3 menunjukkan hasil makespan maupun konsumsi energi yang dihasilkan, untuk bobot yang sama untuk makespan dan konsumsi energi  $w_1 = 0,5$  dan  $w_2 = 0,5$ . Terlihat untuk makespan hasil yang diperoleh cukup mendekati dengan *best known solution*. Sementara itu untuk konsumsi energi hasil yang diperoleh terlihat cukup baik kecuali pada kasus terakhir 1a38 seperti terlihat pada Gambar 3.



Gambar 3. Konsumsi Energi

#### 4. Simpulan

Algoritma Genetika dan Simulated Annealing mempunyai performa yang cukup baik dalam menyelesaikan penjadwalan *job shop*, hal ini dikarenakan adanya penambahan pencarian lokal pada solusi yang sudah ditemukan oleh Algoritma Genetika. Meski demikian Algoritma ini sedikit mempunyai sedikit penurunan performa pada kasus yang memiliki jumlah pekerjaan dan mesin yang besar.

**Daftar Pustaka**

- [1] Chakraborty, S., & Bhowmik, S. (2015). An efficient approach to job shop scheduling problem using simulated annealing. *International Journal of Hybrid Information Technology*, 8(11), 273-284.
- [2] Cooper, D. R., Rossie, K. E., & Gutowski, T. G. (2017). The energy requirements and environmental impacts of sheet metal forming: An analysis of five forming processes. *Journal of Materials Processing Technology*, 244, 116-135.
- [3] Jiang, T., Zhang, C., Zhu, H., Gu, J., & Deng, G. (2018). Energy-efficient scheduling for a job shop using an improved whale optimization algorithm. *Mathematics*, 6(11), 220.
- [4] Kalshetty, Y. R., Adamuthe, A. C., & Kumar, S. P. (2020). Genetic algorithms with feasible operators for solving job shop scheduling problem. *Journal of Scientific Research*, 64(1).
- [5] Ku, W.-Y., & Beck, J. C. (2016). Mixed integer programming models for job shop scheduling: A computational analysis. *Computers & Operations Research*, 73, 165-173.
- [6] Lawrence, S. (1984). Resource constrained project scheduling: An experimental investigation of heuristic scheduling techniques (Supplement). *Graduate School of Industrial Administration, Carnegie-Mellon University*.
- [7] Piroozfard, H., Wong, K. Y., & Wong, W. P. (2018). Minimizing total carbon footprint and total late work criterion in flexible job shop scheduling by using an improved multi-objective genetic algorithm. *Resources, Conservation and Recycling*, 128, 267-283.
- [8] Scaria, A., George, K., & Sebastian, J. (2016). An artificial bee colony approach for multi-objective job shop scheduling. *Procedia Technology*, 25, 1030-1037.
- [9] Stastny, J., Skorpil, V., Balogh, Z., & Klein, R. (2021). Job Shop Scheduling Problem Optimization by Means of Graph-Based Algorithm. *Applied Sciences*, 11(4), 1921.
- [10] Yin, L., Li, X., Gao, L., Lu, C., & Zhang, Z. (2017). A novel mathematical model and multi-objective method for the low-carbon flexible job shop scheduling problem. *Sustainable Computing: Informatics and Systems*, 13, 15-30.