

MINIMASI KONSUMSI ENERGI PADA PERMASALAHAN *HYBRID FLOW SHOP SCHEDULING* DENGAN ALGORITMA SALP SWARM

Bayu Nur Hidayat, Dana Marsetya Utama^{*}, Ikhlasul Amallynda

Jurusan Teknik Industri, Fakultas Teknik, Universitas Muhammadiyah Malang
e-mail: ^{*}dana@umm.ac.id

ABSTRAK

Kelangkaan dan kemungkinan berkurangnya sumber energi di masa depan disebabkan tingginya konsumsi energi pada industri manufaktur. Salah satu upaya untuk menangani permasalahan ini adalah dengan penjadwalan. Dalam artikel ini, kami meneliti tentang permasalahan Hybrid Flow shop Scheduling Problem (HFSSP) untuk meminimasi konsumsi energi. Prosedur algoritma metaheuristik diusulkan yang terinspirasi dari kawanan Salp di lautan, yaitu Salp Swarm Algorithm. Algoritma SSA diusulkan untuk menentukan sequencing yang optimal dengan mempertimbangkan konsumsi energi minimal. Temuan menarik dari penelitian ini adalah semakin tinggi populasi dan iterasi maka menghasilkan konsumsi energi yang lebih rendah. Selain itu, hasil optimasi menunjukkan bahwa total energi konsumsi disebabkan karena variasi konsumsi energi setup dan removal serta konsumsi energi mesin idle. Dari hasil numerik algoritma SSA dapat menghasilkan penjadwalan dengan konsumsi energi yang lebih minimal dibandingkan dengan prosedur heuristik.

Kata kunci: Hybrid Flow Shop Scheduling, Energy Consumption, Metaheuristic, Salp Swarm Algorithm

ABSTRACT

The scarcity and possible depletion of energy sources in the future is due to the high energy consumption in the manufacturing industry. One of the efforts to deal with this problem is scheduling. This article investigates the Hybrid Flow Shop Scheduling Problem (HFSSP) to minimize energy consumption. A metaheuristic algorithm procedure is proposed, inspired by the Salp swarm in the ocean, the Salp Swarm Algorithm. The SSA algorithm is proposed to determine the optimal sequencing considering minimal energy consumption. An interesting finding of this study is that the higher the population and iterations, the lower the energy consumption. In addition, the optimization results show that the total energy consumption is due to variations in setup and removal energy consumption and idle machine energy consumption. From the numerical results, the SSA algorithm can produce scheduling with minimal energy consumption compared to the heuristic procedure.

Keywords: Hybrid Flow Shop Scheduling, Energy Consumption, Metaheuristic, Salp Swarm Algorithm

PENDAHULUAN

Isu mengenai kelangkaan dan menurunnya cadangan energi di masa mendatang akibat pemanfaatan dalam industri manufaktur telah menjadi perhatian [1,2]. Di Tiongkok, sektor manufaktur menjadi konsumen terbesar dengan menyumbang lebih dari 50% dari total konsumsi energi listrik nasional [3]. Sementara di Jerman, lebih dari 47% dari penggunaan energi listrik nasional berasal dari industri manufaktur [4]. Kompleksitas situasi ini mendorong perlunya pengembangan strategi untuk mengurangi pemakaian sumber daya khususnya konsumsi energi [5]. Dalam beberapa tahun terakhir, studi mengenai produksi ramah lingkungan telah menarik perhatian para peneliti [6-8], mengingat peran industri manufaktur yang signifikan dalam menyumbang konsumsi energi global [9]. Efisiensi energi selama proses produksi menjadi sangat penting [10,11], dengan penekanan pada optimalisasi penggunaan energi melalui penjadwalan produksi yang efisien dalam konteks industri manufaktur [12-15]. Penjadwalan merupakan pengalokasian sumber daya untuk menyelesaikan rangkaian tugas pada mesin tertentu [16]. Salah satu jenis pendekatan yang relevan adalah penyelesaian masalah jadwal produksi adalah *Hybrid Flow Shop Scheduling Problem* (HFSSP) [17-19]. HFSSP merupakan permasalahan yang kompleks yang

diklasifikasikan sebagai NP-hard problem [20-23]. Pada HFSSP, serangkaian tugas dikerjakan pada sejumlah mesin sesuai urutan proses tertentu, dengan beberapa tahapan melibatkan lebih dari satu mesin [24,25]. Berbeda dari pendekatan *flow shop* murni yang hanya melibatkan satu mesin pada setiap tahapnya [16]. Pada permasalahan HFSSP, perlunya penentuan jadwal yang optimal untuk meminimalisir konsumsi energi dalam lingkup industri manufaktur.

Berbagai penelitian terdahulu sebagai upaya untuk mengatasi permasalahan HFSSP dengan fungsi tujuan mengurangi konsumsi energi telah dipublikasikan. Dalam penelitian yang dilakukan oleh Yan, Wen dan Li [26], disajikan pendekatan menggunakan Genetic Algorithm untuk meminimalkan penggunaan energi. Schulz [27] mengusulkan pendekatan menggunakan Mixed Integer Linear Programming. Ling, Feng & Zong [28], pada sisi lain, mengadopsi Particle Swarm Optimization untuk mereduksi penggunaan energi. Pada permasalahan HFSS dua tahap, Du et al. [29], telah mengenalkan Algoritma Ant Colony System sebagai alternatif penyelesaian. Sementara itu, Liu et al. [30] menawarkan model yang berfokus pada penyelesaian HFSS tiga tahap dengan pertimbangan tiga mesin di tahap 1, dua mesin di tahap 2, dan tiga mesin di tahap 3. Selanjutnya, mereka menerapkan pendekatan Mixed Integer Linear Programming (MILP) untuk meningkatkan efisiensi konsumsi energi. Pada penelitian yang sama, Liu et al. [31] juga mengajukan model serupa dengan modifikasi jumlah mesin di setiap tahap. Di lain pihak, Luo et al. [32], mengimplementasikan Algoritma Ant Colony Optimization untuk menyelesaikan permasalahan HFSSP. Discrete Imperialist Competitive Algorithm dikembangkan oleh Tao et al. [33] untuk penyelesaian HFSS dua tahap dengan fokus pada pengurangan konsumsi energi. Terdapat pula berbagai alternatif lainnya, seperti Algoritma Imperialist Competitive [18,34], Shuffled Frog-Leaping Algorithm [35], serta Improved Genetic Algorithm [36], yang juga diusulkan untuk tujuan serupa. Selain itu, beberapa pendekatan lain termasuk A- Novel Teaching Learning-Based Optimization Algorithm [37], Particle Swarm Optimization [28], Genetic Algorithm [38], dan Lagrangian Relaxation Algorithm [39], juga dikemukakan sebagai solusi potensial pada permasalahan minimasi konsumsi energi dalam HFSSP. Uraian tersebut menunjukkan bahwa prosedur metaheuristik merupakan prosedur yang populer untuk menyelesaikan masalah HFSSP [40,41].

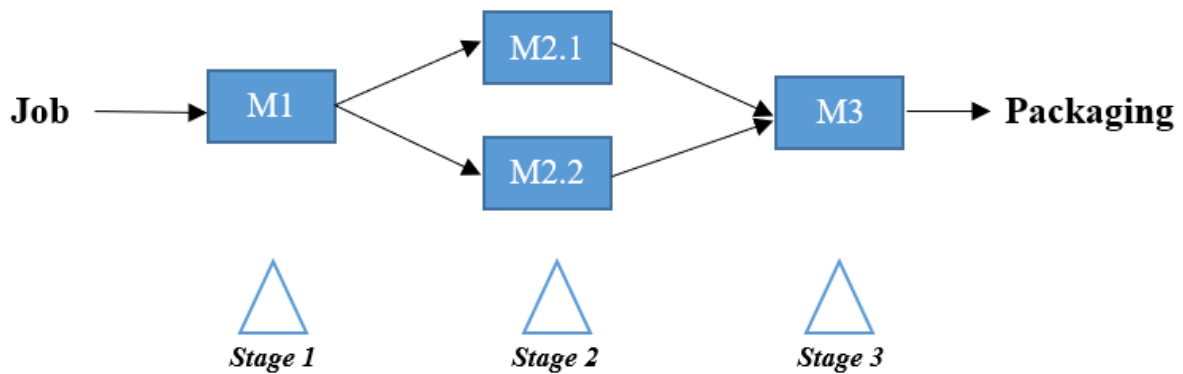
Berdasarkan uraian tersebut, berbagai prosedur metaheuristik telah ditawarkan untuk menyelesaikan masalah HFSSP. Namun, prosedur *Salp Swarm Algorithm* kurang mendapat perhatian untuk menyelesaikan masalah HFSSP. *Salp Swarm Algorithm* (SSA) dipilih untuk menentukan variabel keputusan urutan jadwal yang optimal dengan fungsi tujuan meminimasi konsumsi energi. *Salp Swarm Algorithm* (SSA) adalah algoritma metaheuristik terbaru yang di usulkan oleh Mirjalili et al. [42]. SSA adalah algoritma yang terinspirasi dari perilaku salp yang memiliki karakter mirip seperti ubur-ubur dalam mencari sumber makanan. Menurut Utama [43], dalam percobaan numerik SSA terbukti kompetitif dibandingkan *Genetic Algorithm* (GA) dan *Particle Swarm Optimization* (PSO). Selain itu, konsumsi energi pada proses produksi terjadi pada saat mesin memproses setiap *job*. Namun, sebagian besar energi juga di konsumsi pada saat mesin dalam kondisi *idle*. Di sisi lain, pada saat *setup* dan *removal job* juga mengkonsumsi energi karena mesin dalam kondisi menyala. Maka diperlukan penjadwalan produksi dengan mempertimbangkan *setup* dan *removal time* untuk meminimasi konsumsi energi listrik dengan algoritma SSA sebagai prosedur optimasi permasalahan HFSSP.

METODE PENELITIAN

Definisi Masalah

Dalam permasalahan HFSSP ini terdapat n *job* yang akan diproses pada 3 *stage*. Keseluruhan *job* akan diproses melalui urutan yang sama mulai dari *stage* 1 ke *stage* 3.

Tujuan dari penelitian ini adalah untuk menghasilkan konsumsi energi listrik yang minimum. Skema *hybrid flowshop scheduling* dapat dilihat pada Gambar 1.



Gambar 1. Skema Permasalahan *Hybrid Flowshop Scheduling*

Asumsi

Beberapa batasan pada permasalahan hybrid flowshop scheduling adalah: (1) Setiap *job* memiliki *sequence dependent setup*, (2) Setiap mesin hanya dapat memproses 1 item, (3) Tidak adanya *iterupsi* pada *job*, (4) Setiap mesin akan memproses pada $t=0$, (5) Waktu *setup* terpisah dengan waktu proses, (6) Waktu *removal* hanya terdapat pada mesin di *stage* 1.

Formula untuk meminimasi konsumsi energi pada permasalahan HFSSP dapat dilihat dari persamaan berikut [30]:

$$EB = \sum_{j=1}^m \sum_{k=1}^{M_j} \sum_{i=1}^{N_{jk}} E_{ijk} \quad (1)$$

$$EOO = \sum_{j=1}^m \sum_{k=1}^{M_j} ER_{jk} \quad (2)$$

$$EI = \sum_{j=1}^m \sum_{k=1}^{M_j} \sum_{i=1}^{N_{jk}-1} EI_{ijk} (S_{(i+1)jk} - C_{ijk}) \quad (3)$$

$$TEC = EB + EOO + EI \quad (4)$$

Notasi dari permasalahan tersebut adalah sebagai berikut:

- n : jumlah *job*
- m : jumlah *stage*
- M_j : jumlah mesin paralel di *stage j*
- N_{jk} : jumlah *job* pada mesin k di *stage j*
- P_{ijk} : waktu proses saat *job i* diproses pada mesin k di *stage j*
- S_{ijk} : waktu mulai saat *job i* diproses pada mesin k di *stage j*
- C_{ijk} : waktu selesai saat *job i* selesai diproses pada mesin k di *stage j*
- I_{jk} : *job ke h* yang diproses pada mesin k di *stage j*
- E_{ijk} : konsumsi energi saat *job i* diproses pada mesin k di *stage j*
- EI_{jk} : konsumsi energi saat *idle* pada mesin k di *stage j*
- ER_{jk} : konsumsi energi saat *setup* dan *removal* mesin k di *stage j*
- EB : konsumsi energi saat proses
- EEO : konsumsi energi saat *setup* dan *removal*
- EI : konsumsi energi saat *idle*
- TEC : total konsumsi energi

Persamaan (1) digunakan untuk menghitung konsumsi energi pada saat mesin kondisi memproses *job*. Persamaan (2) digunakan untuk menghitung konsumsi energi pada saat mesin kondisi *setup* dan *removal job*. Persamaan (3) digunakan untuk menghitung konsumsi

energi pada saat mesin kondisi *idle*. Persamaan (4) menggambarkan model *hybrid flowshop scheduling* untuk meminimasi konsumsi energi (fungsi tujuan).

Salp Swarm Algorithm

Baru-baru ini, banyak algoritma metaheuristik yang berhasil diterapkan untuk memecahkan masalah yang sulit diselesaikan. Daya tarik menggunakan algoritma ini karena dapat memecahkan masalah yang kompleks dengan solusi yang optimal dan waktu yang cepat. Menurut Dokeroglu et al. [44], metaheuristik adalah gambaran dari heuristik dalam tingkat yang lebih tinggi untuk mengusulkan solusi dari masalah pengoptimalan. *Salp Swarm Algorithm* (SSA) merupakan algoritma terbaru yang di usulkan oleh [42]. *Salp* termasuk keluarga dari *salpidae* yang bersifat transparan dan berbentuk tong. Jaringan *salp* mirip dengan ubur-ubur. Mereka juga beroperasi mirip dengan ubur-ubur, dimana air dipompa melalui tubuh sebagai penggerak [45].

Menurut Mirjalili et al. [42], salah satu perilaku *salp* yang menarik adalah mereka mengerumuni kawanannya. Beberapa peneliti percaya bahwa perilaku ini dilakukan untuk mencapai pergerakan yang lebih baik dalam mencari makanan [46].

Dalam SSA ada dua jenis individu dalam segerombolan *salp*, yaitu *salp* pemimpin (*leader*) dan *salp* pengikut (*followers*). *Leader* adalah *salp* pertama yang memandu dari *salp* pengikut dalam gerakan mereka. Kawan X dari n *salp* dapat diwakili oleh matriks seperti yang ditunjukkan pada persamaan (5). Tujuan dari kawan ini adalah sumber makanan di ruang pencarian yang disebut F .

$$X_i = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & \vdots & \cdots & \vdots \\ x_1^n & x_2^n & \cdots & x_d^n \end{bmatrix} \quad (5)$$

Koefisien c_1 adalah parameter yang paling penting dalam *Salp Swarm Optimization*. Persamaan (6) menunjukkan bahwa pemimpin hanya memperbarui posisinya dengan menghormati sumber makanan. Menyeimbangkan eksplorasi (penjelajahan) dan eksploitasi (pengusahaan atau pendayagunaan) didefinisikan pada persamaan (6).

$$c_1 = 2e^{-\left(\frac{4t}{L}\right)^2} \quad (6)$$

Seperti yang disebutkan sebelumnya, populasi *salp* dibagi menjadi dua jenis yaitu *salp* pemimpin dan pengikut. Posisi *salp* pemimpin (*leader*) di *update* menggunakan persamaan (7)

$$x_j^i = \begin{cases} F_j + c_1 ((ub_j - lb_j) c_2 + lb_j) c_3 \geq 0,5 \\ F_j - c_1 ((ub_j - lb_j) c_2 + lb_j) c_3 < 0,5 \end{cases} \quad (7)$$

Dimana x_j^i menunjukkan posisi *salp* pemimpin (*leader*) pada dimensi j , F_j adalah posisi sumber makanan pada dimensi j th, ub_j menunjukkan batas atas pada dimensi j , dan lb_j menunjukkan batas bawah pada dimensi j . Parameter c_1 , c_2 dan c_3 adalah bilangan *random* yang dihasilkan secara seragam dalam interval $[0,1]$. Bahkan, mereka menentukan posisi selanjutnya dalam dimensi j harus menuju *infinity negative* serta ukuran langkah.

Untuk mengupdate posisi *salp* pengikut, persamaan (8) digunakan (hukum gerak newton)

$$x_j^i = \frac{1}{2}at^2 + v_0t \quad (8)$$

Dimana $i \geq 2$, x_j^i menunjukkan posisi pengikut di dimensi j , t adalah waktu, v_0 adalah kecepatan awal. Karena waktu dalam optimasi adalah iterasi, perbedaan antar iterasi sama dengan 1, dan mempertimbangkan konsideran $v_0 = 0$, kondisi demikian dapat diformulasikan sebagai persamaan (9)

$$x_j^i = \frac{1}{2}(x_j^i + x_j^{i-1}) \quad (9)$$

Dimana $i \geq 2$, x_j^i menunjukkan posisi *salp* pengikutnya di dimensi j th. Pseudocode dapat dilihat pada Gambar 2.

```

Initialize the salp population  $x_i$  ( $i = 1, 2, \dots, n$ ) considering ub and lb
while (end condition is not satisfied)
    Calculate the fitness based on energy consumption of each search agent (salp)
     $F$ =the best search agent
    Update  $c_1$  by Eq. (12)
    for each salp ( $x_i$ )
        if ( $i==1$ )
            Update the position of the leading salp by Eq. (7)
        else
            Update the position of the follower salp by Eq. (8)
        end
    end
    Amend the salps based on the upper and lower bound of variables
end
return  $F$ 
    
```

Gambar 2. Pseudocode Salp Swarm Algorithm

Data Penelitian dan Prosedur Percobaan

Data penelitian ini diambil dari studi kasus pada perusahaan percetakan di Indonesia. 18 *job* dijadwalkan melalui 3 tahap yang terdiri dari 1 mesin di tahap awal, 2 mesin parallel di tahap 2 dan 1 mesin di tahap ketiga. Dalam optimasi dengan SSA 3 variasi dari populasi dan iterasi disajikan. Pada parameter populasi menggunakan 10, 50 dan 100 populasi. Sementara itu, pada parameter iterasi menggunakan 50, 100, dan 200 iterasi. tiap percobaan dicatat TEC, EB, EOO, dan EI. Selanjutnya, prosedur SSA juga dibandingkan dengan prosedur heuristic First Come First Serve (FCFS) untuk menguji efektifitas algoritma. Algoritma FCFS digunakan sebagai prosedur penentuan urutan jadwal oleh perusahaan yang perlu dievaluasi keefektifannya.

HASIL DAN PEMBAHASAN

Hasil percobaan metode SSA ditunjukkan pada Tabel 1. Hasil ini menunjukkan bahwa semakin tinggi populasi dan iterasi maka menghasilkan konsumsi energi yang lebih rendah. Sementara itu, jika populasi dan iterasi semakin kecil konsumsi energi semakin tinggi. Terdapat sebuah temuan menarik dalam penelitian terkait HFSSP untuk mengoptimalkan konsumsi energi. Hasil penelitian ini menunjukkan bahwa tingkat populasi dan jumlah iterasi yang digunakan dalam algoritma SSA memiliki dampak signifikan terhadap tingkat konsumsi energi yang dihasilkan. Penelitian ini mengindikasikan bahwa semakin tinggi tingkat populasi dan jumlah iterasi yang diterapkan, maka hasilnya akan mengarah pada konsumsi energi yang lebih rendah. Dengan kata lain, penggunaan populasi dan iterasi yang besar dalam algoritma optimisasi mampu membawa dampak positif dalam upaya mengurangi pemakaian energi dalam permasalahan HFSSP. Sebaliknya, observasi ini juga mengungkapkan bahwa apabila tingkat populasi dan iterasi cenderung lebih kecil, maka akan berimplikasi pada peningkatan konsumsi energi. Oleh karena itu, temuan ini memberikan wawasan berharga bagi praktisi dan peneliti di bidang ini, bahwa pengaturan

yang tepat terhadap parameter populasi dan iterasi dalam algoritma metaheuristik dapat menjadi faktor penting dalam pencapaian tujuan mengurangi konsumsi energi dalam lingkungan HFSSP. Temuan ini sejalan dengan temuan penelitian Tang et al. [47] dan Hasani and Hosseini [48]. Hal ini dikarenakan semakin tinggi populasi dan iterasi, semakin banyak kemungkinan solusi yang dihasilkan, sehingga peluang untuk menemukan solusi terbaik semakin besar [49].

Selain itu, berdasarkan percobaan variasi iterasi dan populasi, temuan menarik dari penelitian ini adalah total energi konsumsi disebabkan karena variasi konsumsi energi *setup* dan *removal* serta konsumsi energi mesin *idle*. Konsumsi energi waktu proses selalu tetap dan tidak mempengaruhi total konsumsi energi. Pada permasalahan HFSSP, upaya untuk mengurangi konsumsi energi telah menjadi perhatian utama, dan penelitian-penelitian sebelumnya telah mengungkapkan beberapa aspek kunci yang berperan dalam total konsumsi energi. Temuan penelitian ini menyoroti bahwa variasi dalam konsumsi energi terutama terjadi pada tahap *setup* dan *removal*, serta ketika mesin berada dalam kondisi *idle*. Penting untuk dicatat bahwa konsumsi energi selama waktu proses berlangsung cenderung tetap dan tidak berpengaruh signifikan terhadap total konsumsi energi. Oleh karena itu, strategi efektif dalam upaya mengurangi konsumsi energi dalam penjadwalan hybrid flow shop akan fokus pada mengoptimalkan pengaturan tahap *setup* dan *removal*, serta mengelola periode ketika mesin tidak aktif (*idle*). Dengan mempertimbangkan temuan-temuan ini, langkah-langkah penjadwalan yang cermat dan penggunaan teknik-teknik optimisasi yang tepat dapat diimplementasikan untuk meminimalkan dampak variasi konsumsi energi *setup*, *removal*, dan energi *idle* pada konsumsi energi keseluruhan, yang pada akhirnya berpotensi mendukung pencapaian tujuan efisiensi energi dalam operasional manufaktur.

Tabel 1. Rekapitulasi Hasil Konsumsi Energi Metode SSA

Populasi	Iterasi	TEC	EB	EOO	EI
10	50	74.958,99	74.115	200,25	643,74
	100	74.948,43	74.115	190,65	642,78
	200	74.941,92	74.115	188,25	638,67
50	50	74.961,81	74.115	195,45	651,36
	100	74.953,92	74.115	197,85	641,07
	200	74.928,27	74.115	190,65	622,62
100	50	75.166,49	74.115	200,25	851,24
	100	75.123,15	74.115	190,65	817,50
	200	74.925,78	74.115	178,65	632,13

Perbandingan efisiensi konsumsi energi disajikan pada Tabel 2. Hasil ini menunjukkan bahwa algoritma SSA memberikan hasil kinerja yang lebih baik dibandingkan metode heuristic FCFS. Algoritma SSA menghasilkan solusi yang lebih baik daripada prosedur heuristic dalam optimasi karena beberapa alasan. SSA adalah teknik swarm intelligence yang meniru perilaku mencari makan dan navigasi salp biologi laut. Dengan meniru perilaku alami organisme ini, SSA memanfaatkan kecerdasan kolektif dari swarm untuk mengeksplorasi dan mengeksploitasi ruang pencarian secara efektif [42]. Lebih lanjut, SSA membagi populasi salp menjadi dua kelompok yaitu pemimpin dan pengikut. Pemimpin memandu proses pencarian dengan mengeksplorasi wilayah yang menjanjikan, sementara pengikut menyesuaikan posisi mereka berdasarkan informasi pemimpin. Mekanisme pemimpin-pengikut ini meningkatkan kemampuan eksplorasi dan eksploitasi algoritma, menghasilkan solusi yang lebih baik [50]. Sementara itu, SSA telah menunjukkan konvergensi yang tinggi dan kemampuan pencarian yang kuat dalam masalah optimasi [51]. SSA mampu mengoptimalkan fungsi benchmark tunggal, multi-modal, dan komposit, sehingga menjadi algoritma yang serbaguna dan dapat diterapkan pada berbagai skenario optimasi [42].

Tabel 2. Perbandingan Konsumsi Energi Metode FCFS dan Metode SSA

No	Metode	Urutan Job	Total Konsumsi Energi (KW)
1	Metode heuristic FCFS	1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18	75.245,98
2	Metode Usulan SSA	9-6-3-4-10-12-16-18-1-11-8-14-13-15-5-17-2-7	74.925,78
		Selisih	320,20
		Efisiensi	0,42%

KESIMPULAN

Dalam artikel ini, peneliti mengusulkan algoritma SSA untuk meminimasi konsumsi energi dari permasalahan HFSSP. Hasil optimasi dengan SSA menunjukkan bahwa semakin tinggi populasi dan iterasi maka menghasilkan konsumsi energi yang lebih rendah. Selain itu, hasil optimasi menunjukkan bahwa total energi konsumsi disebabkan karena variasi konsumsi energi *setup* dan removal serta konsumsi energi mesin idle. sementara itu, konsumsi energi waktu proses selalu tetap ketika percobaan variasi populasi dan iterasi pada prosedur SSA. Selanjutnya, algoritma SSA dibandingkan dengan metode heuristic FCFS. Dari hasil penelitian ini menunjukkan bahwa SSA menghasilkan konsumsi energi yang lebih minim dibandingkan dengan metode FCFS. Pada akhirnya, SSA dapat diterapkan untuk meminimasi konsumsi energi pada permasalahan HFSSP. Berdasarkan temuan penelitian tersebut, penelitian mendatang dapat mengeksplorasi lebih dalam terhadap mekanisme dan parameter-parameter dalam algoritma SSA untuk mengoptimalkan penyelesaian penjadwalan HFSSP dengan tujuan efisiensi energi yang lebih tinggi. Selain itu, eksplorasi kemungkinan kombinasi dengan teknik optimasi lain atau pendekatan yang berbeda juga dapat menjadi arah yang menarik untuk meningkatkan performa algoritma dalam menghadapi tantangan kompleksitas permasalahan penjadwalan HFSSP.

DAFTAR ISI

- [1] D. M. Utama, M. F. Ibrahim, D. S. Wijaya, D. S. Widodo, and M. D. Primayesti, "A Novel Hybrid Multi-Verse Optimizer Algorithm for Energy-Efficient Permutation Flow Shop Scheduling Problem," *Journal of Physics: Conference Series*, vol. 2394, no. 1, p. 012006, 2022.
- [2] D. M. Utama, "An effective hybrid crow search algorithm for energy-efficient flow shop scheduling," *AIP Conference Proceedings*, vol. 2453, no. 1, p. 020040, 2022.
- [3] Y. Liu, H. Dong, N. Lohse, S. Petrovic, and N. Gindy, "An investigation into minimising total energy consumption and total weighted tardiness in job shops," *Journal of Cleaner Production*, vol. 65, pp. 87-96, 2014.
- [4] M. Dai, D. Tang, A. Giret, M. A. Salido, and W. D. Li, "Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 5, pp. 418-429, 2013.
- [5] A. Sun, "The establishment of the green tax policy in China-To accelerate the construction of circular economy experimental zone in Qaidam Basin of Qinghai province as an example," *Asian Social Science*, vol. 9, no. 3, p. 148, 2013.
- [6] S. A. Mansouri, E. Aktas, and U. Besikci, "Green scheduling of a two-machine flowshop: Trade-off between makespan and energy consumption," *European Journal of Operational Research*, vol. 248, no. 3, pp. 772-788, 2016.
- [7] G.-S. Liu, Y. Zhou, and H.-D. Yang, "Minimizing energy consumption and tardiness penalty for fuzzy flow shop scheduling with state-dependent setup time," *Journal of cleaner production*, vol. 147, pp. 470-484, 2017.
- [8] D. M. Utama, "Pengembangan Algoritma NEH Dan CDS Untuk Meminimasi Consumption Energy Pada Penjadwalan Flow Shop," in *Prosiding SENTRA (Seminar Teknologi dan Rekayasa)*, 2019, no. 4, pp. 47-54.

- [9] Y. Li *et al.*, "A discrete artificial bee colony algorithm for distributed hybrid flowshop scheduling problem with sequence-dependent setup times," *International Journal of Production Research*, vol. 59, no. 13, pp. 3880-3899, 2021.
- [10] S. Wang, X. Wang, F. Chu, and J. Yu, "An energy-efficient two-stage hybrid flow shop scheduling problem in a glass production," *International Journal of Production Research*, vol. 58, no. 8, pp. 2283-2314, 2020.
- [11] D. M. Utama, T. Baroto, and D. S. Widodo, "Energy-Efficient Flow Shop Scheduling Using Hybrid Grasshopper Algorithm Optimization," *Jurnal Ilmiah Teknik Industri*, vol. 19, no. 1, pp. 30-38, 2020.
- [12] M. Akbar and T. Irohara, "Scheduling for sustainable manufacturing: A review," *Journal of cleaner production*, vol. 205, pp. 866-883, 2018.
- [13] D. Utama, "Minimizing Number of Tardy Jobs in Flow Shop Scheduling Using A Hybrid Whale Optimization Algorithm," in *Journal of Physics: Conference Series*, 2021, vol. 1845, no. 1, p. 012017: IOP Publishing.
- [14] D. M. Utama, D. S. Widodo, M. F. Ibrahim, and S. K. Dewi, "An effective hybrid ant lion algorithm to minimize mean tardiness on permutation flow shop scheduling problem," *International Journal of Advances in Intelligent Informatics*, vol. 6, no. 1, pp. 23-35, 2020.
- [15] D. S. Widodo and D. M. Utama, "The Hybrid Ant Lion Optimization Flow Shop Scheduling Problem for Minimizing Completion Time," in *Journal of Physics: Conference Series*, 2020, vol. 1569, no. 2, p. 022097: IOP Publishing.
- [16] D. M. Utama, D. S. Widodo, M. F. Ibrahim, K. Hidayat, T. Baroto, and A. Yurifah, "The hybrid whale optimization algorithm: A new metaheuristic algorithm for energy-efficient on flow shop with dependent sequence setup," in *Journal of Physics: Conference Series*, 2020, vol. 1569, no. 2, p. 022094: IOP Publishing.
- [17] 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.
- [18] M. Li, D. Lei, and J. Cai, "Two-level imperialist competitive algorithm for energy-efficient hybrid flow shop scheduling problem with relative importance of objectives," *Swarm and Evolutionary Computation*, vol. 49, pp. 34-43, 2019.
- [19] D. M. Utama, A. K. Garside, and W. Wicaksono, "Pengembangan Algoritma Hybrid Flowshop Three-Stage Dengan Mempertimbangkan Waktu Setup," *Jurnal Ilmiah Teknik Industri*, vol. 18, no. 1, pp. 72-78, 2019.
- [20] D. A. Rossit, F. Tohmé, and M. Frutos, "The Non-Permutation Flow-Shop scheduling problem: A literature review," *Omega*, vol. 77, pp. 143-153, 2018.
- [21] C. Yu, Q. Semeraro, and A. Matta, "A genetic algorithm for the hybrid flow shop scheduling with unrelated machines and machine eligibility," *Computers & Operations Research*, vol. 100, pp. 211-229, 2018.
- [22] L. De FELICE, "A simulation model for solving the flow shop scheduling problem under uncertainty," 2018.
- [23] J.-q. Li, H.-y. Sang, Y.-y. Han, C.-g. Wang, and K.-z. Gao, "Efficient multi-objective optimization algorithm for hybrid flow shop scheduling problems with setup energy consumptions," *Journal of Cleaner Production*, vol. 181, pp. 584-598, 2018.
- [24] V. Fernandez-Viagas, J. M. Molina-Pariante, and J. M. Framinan, "New efficient constructive heuristics for the hybrid flowshop to minimise makespan: A computational evaluation of heuristics," *Expert Systems with Applications*, vol. 114, pp. 345-356, 2018.
- [25] V. Fernandez-Viagas, P. Perez-Gonzalez, and J. M. Framinan, "Efficiency of the solution representations for the hybrid flow shop scheduling problem with makespan objective," *Computers & Operations Research*, vol. 109, pp. 77-88, 2019.

- [26] J. Yan, J. Wen, and L. Li, "Genetic Algorithm Based Optimization for Energy-aware Hybrid Flow Shop Scheduling," in *Proceedings on the international conference on artificial intelligence (ICAI)*, 2014, p. 1: The Steering Committee of The World Congress in Computer Science, Computer
- [27] S. Schulz, "A multi-criteria MILP formulation for energy aware hybrid flow shop scheduling," in *Operations Research Proceedings 2016*: Springer, 2018, pp. 543-549.
- [28] Z. Ling-Li, Z. Feng-Xing, X. Xiao-hong, and G. Zheng, "Dynamic scheduling of multi-task for hybrid flow-shop based on energy consumption," in *2009 International Conference on Information and Automation*, 2009, pp. 478-482: IEEE.
- [29] B. Du, H. Chen, G. Q. Huang, and H. Yang, "Preference vector ant colony system for minimising make-span and energy consumption in a hybrid flow shop," in *Multi-objective evolutionary optimisation for product design and manufacturing*: Springer, 2011, pp. 279-304.
- [30] X. Liu, F. Zou, and X. Zhang, "Mathematical model and genetic optimization for hybrid flow shop scheduling problem based on energy consumption," in *2008 Chinese Control and Decision Conference*, 2008, pp. 1002-1007: IEEE.
- [31] Z. Liu, J. Yan, Q. Cheng, C. Yang, S. Sun, and D. Xue, "The mixed production mode considering continuous and intermittent processing for an energy-efficient hybrid flow shop scheduling," *Journal of Cleaner Production*, vol. 246, p. 119071, 2020.
- [32] H. Luo, B. Du, G. Q. Huang, H. Chen, and X. Li, "Hybrid flow shop scheduling considering machine electricity consumption cost," *International Journal of Production Economics*, vol. 146, no. 2, pp. 423-439, 2013.
- [33] X.-r. Tao, J.-q. Li, T.-h. Huang, and P. Duan, "Discrete imperialist competitive algorithm for the resource-constrained hybrid flowshop problem with energy consumption," *Complex & Intelligent Systems*, pp. 1-16, 2020.
- [34] R. Zhou, D. Lei, and X. Zhou, "Multi-objective energy-efficient interval scheduling in hybrid flow shop using imperialist competitive algorithm," *Ieee Access*, vol. 7, pp. 85029-85041, 2019.
- [35] D. Lei and T. Wang, "Solving distributed two-stage hybrid flowshop scheduling using a shuffled frog-leaping algorithm with memplex grouping," *Engineering Optimization*, vol. 52, no. 9, pp. 1461-1474, 2020.
- [36] L. Meng, C. Zhang, X. Shao, Y. Ren, and C. Ren, "Mathematical modelling and optimisation of energy-conscious hybrid flow shop scheduling problem with unrelated parallel machines," *International Journal of Production Research*, vol. 57, no. 4, pp. 1119-1145, 2019.
- [37] D. Lei, L. Gao, and Y. Zheng, "A novel teaching-learning-based optimization algorithm for energy-efficient scheduling in hybrid flow shop," *IEEE Transactions on Engineering Management*, vol. 65, no. 2, pp. 330-340, 2017.
- [38] S. Schulz, "A genetic algorithm to solve the hybrid flow shop scheduling problem with subcontracting options and energy cost consideration," in *International Conference on Information Systems Architecture and Technology*, 2018, pp. 263-273: Springer.
- [39] X. L. Ding, J. Zhu, and C. Liu, "Lagrangian Relaxation Algorithms For Hybrid Flow-Shop Scheduling Problems with Energy Saving," in *Advanced Materials Research*, 2014, vol. 997, pp. 821-826: Trans Tech Publ.
- [40] D. M. Utama and M. D. Primayesti, "A novel hybrid Aquila optimizer for energy-efficient hybrid flow shop scheduling," *Results in Control and Optimization*, vol. 9, p. 100177, 2022.
- [41] D. M. Utama, M. D. Primayesti, S. Z. Umamy, B. M. N. Kholifa, and A. D. Yasa, "A systematic literature review on energy-efficient hybrid flow shop scheduling," *Cogent Engineering*, vol. 10, no. 1, p. 2206074, 2023.

- [42] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163-191, 2017.
- [43] D. M. Utama, "Salp Swarm Algorithm Untuk Meminimasi Konsumsi Energi Pada Penjadwalan Flow Shop Dengan Set Up Dan Removal Time," in *Prosiding SENTRA (Seminar Teknologi dan Rekayasa)*, 2019, no. 5, pp. 79-85.
- [44] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Computers & Industrial Engineering*, vol. 137, p. 106040, 2019.
- [45] L. P. Madin, "Aspects of jet propulsion in salps," *Canadian Journal of Zoology*, vol. 68, no. 4, pp. 765-777, 1990.
- [46] P. Anderson and Q. Bone, "Communication between individuals in salp chains. II. Physiology," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 210, no. 1181, pp. 559-574, 1980.
- [47] H. Tang, J. Zhou, Y. Shao, and Z. Yang, "Hybrid Flow-Shop Scheduling Problems with Missing and Re-Entrant Operations Considering Process Scheduling and Production of Energy Consumption," *Journal*, Type of Article vol. 15, no. 10, 2023.
- [48] A. Hasani and S. M. H. Hosseini, "A bi-objective flexible flow shop scheduling problem with machine-dependent processing stages: Trade-off between production costs and energy consumption," *Applied Mathematics and Computation*, vol. 386, p. 125533, 2020.
- [49] K. Geng, L. Liu, and Z. Wu, "Energy-efficient distributed heterogeneous re-entrant hybrid flow shop scheduling problem with sequence dependent setup times considering factory eligibility constraints," *Scientific Reports*, vol. 12, no. 1, p. 18741, 2022.
- [50] Z. Wang *et al.*, "Orthogonal pinhole-imaging-based learning salp swarm algorithm with self-adaptive structure for global optimization," *Frontiers in Bioengineering and Biotechnology*, vol. 10, p. 1018895, 2022.
- [51] Z. Yi, Z. Yangkun, Y. Hongda, and W. Hong, "Application of an improved Discrete Salp Swarm Algorithm to the wireless rechargeable sensor network problem," *Frontiers in Bioengineering and Biotechnology*, vol. 10, p. 923798, 2022.