

BAB 3

METODOLOGI PENELITIAN

Bab ini membahas deskripsi masalah, tahapan penelitian, formulasi model masalah HFS, dan teknik penyelesaian menggunakan algoritma GWO.

3.1 Deskripsi Masalah

Permasalahan yang dikaji dalam penelitian ini adalah HFS, yaitu suatu sistem penjadwalan produksi yang terdiri dari beberapa tahapan pemrosesan berurutan, di mana setiap tahap memiliki lebih dari satu mesin yang bekerja secara paralel. Dalam sistem ini, sejumlah n *job* harus melewati m tahap produksi sesuai dengan urutan yang telah ditetapkan dan diproses pada salah satu mesin yang tersedia di setiap tahap.

Masalah utama yang dihadapi HFS adalah menentukan urutan *job* yang optimal agar setiap *job* dapat diproses secara berurutan di semua tahap dengan total waktu penyelesaian (*makespan*) seminimum mungkin. Pada setiap tahap, sebuah mesin hanya dapat mengerjakan satu *job* dalam satu waktu, sehingga *job* berikutnya harus menunggu hingga mesin yang dibutuhkan tersedia. Meskipun beberapa *job* dapat diproses secara bersamaan dalam satu tahap, setiap *job* harus dialokasikan ke mesin yang berbeda. Untuk menyelesaikan masalah ini, penelitian ini akan mengimplementasikan algoritma GWO guna menemukan solusi optimal dalam penjadwalan produksi.

3.2 Tahapan Penelitian

Tahapan-tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Studi Pustaka

Tahap studi pustaka dilakukan dengan mengkaji teori dan konsep yang berkaitan dengan HFS dan algoritma GWO dari berbagai sumber literatur, seperti buku, jurnal, maupun karya tulis ilmiah lainnya yang relevan.

2. Pengumpulan Data

Pada tahap ini, dilakukan pengumpulan data yang diperlukan untuk penelitian. Data yang digunakan mencakup data tahapan produksi, data

jumlah *job*, data jumlah mesin, serta data waktu pemrosesan pada setiap tahap produksi.

3. Formulasi Model Optimisasi

Pada tahap ini, akan diformulasikan model optimisasi untuk permasalahan HFS dengan terlebih dahulu mendefinisikan himpunan, indeks, dan parameter-parameter yang digunakan dalam model optimisasi.

4. Penyelesaian Model

Pada tahap ini, model optimisasi akan diselesaikan dengan menggunakan algoritma GWO.

5. Validasi

Pada tahap validasi, dilakukan pengujian terhadap model dan algoritma yang telah dibangun untuk memastikan keakuratannya dalam menyelesaikan masalah HFS. Pengujian dilakukan dengan memeriksa solusi yang diperoleh dengan perhitungan manual dan komputasi pada contoh kasus. Jika solusi yang diperoleh berbeda, maka proses akan kembali ke tahap pemodelan. Namun, jika solusi yang diperoleh sama, penelitian akan dilanjutkan ke tahapan implementasi.

6. Implementasi

Model optimisasi yang sudah valid akan diimplementasikan untuk menyelesaikan masalah HFS di suatu pabrik mebel dengan menggunakan algoritma GWO.

7. Penarikan Kesimpulan

Penarikan kesimpulan merupakan tahap yang dilakukan dengan cara menganalisis dan merangkum hasil penelitian yang telah dilakukan. Kesimpulan tersebut mencakup ringkasan mengenai kesesuaian antara hasil yang diperoleh dengan rumusan masalah, serta analisis hasil kinerja algoritma GWO dalam menyelesaikan HFS.

3.3 Formulasi Model Optimisasi HFS

Dalam penyelesaian masalah HFS, diperlukan model optimisasi yang dapat menggambarkan sistem produksi dengan jelas dan mempertimbangkan faktor-

faktor yang mempengaruhi penjadwalan. Agar model dapat mencerminkan kondisi nyata dari sistem produksi, beberapa asumsi perlu ditetapkan sebagai dasar pemodelan. Asumsi-asumsi model masalah HFS yang digunakan adalah sebagai berikut:

1. Bahan baku selalu tersedia dalam jumlah yang cukup untuk proses produksi.
2. Seluruh mesin yang digunakan dalam produksi berfungsi dengan baik dan dalam kondisi optimal.
3. Waktu persiapan mesin sebelum dioperasikan sudah termasuk dalam perhitungan waktu proses.
4. Proses produksi berlangsung tanpa adanya kegagalan atau kebutuhan untuk memperbaiki produk cacat.

Setelah menetapkan asumsi-asumsi yang digunakan, tahap selanjutnya adalah menyusun formulasi model optimisasi HFS. Formulasi model optimisasi HFS dalam penelitian ini mengacu pada Hasanah (2021), dengan tahapan sebagai berikut::

- a. Pendefinisian himpunan dan parameter

Tahap pertama dalam formulasi model adalah mendefinisikan himpunan dan parameter yang akan digunakan sebagai dasar pemodelan. Himpunan dalam model ini terdiri dari himpunan tahap produksi, himpunan *job*, dan himpunan mesin. Sementara itu, parameter yang digunakan mencakup waktu pemrosesan *job*, waktu mulai dan selesai pemrosesan *job*, serta *makespan*.

- b. Pendefinisian variabel keputusan

Pada tahap ini, variabel keputusan didefinisikan untuk menentukan alokasi mesin yang mengerjakan setiap *job* di setiap tahap dan mengatur urutan pengerjaan *job* karena keterbatasan jumlah mesin.

- c. Pendefinisian fungsi tujuan

Fungsi tujuan dalam model ini didefinisikan untuk meminimalkan waktu penyelesaian *job* terakhir (*makespan*), yaitu waktu maksimum yang dibutuhkan untuk menyelesaikan seluruh *job*. Fungsi tujuan ini bertujuan mengoptimalkan jadwal produksi agar dapat diselesaikan dalam waktu sesingkat mungkin.

d. Menetapkan fungsi kendala

Kendala-kendala yang harus dipenuhi dalam penjadwalan adalah sebagai berikut:

- i. Setiap *job* harus diproses pada setiap tahap dengan hanya menggunakan satu mesin di tiap tahap.
- ii. Waktu mulai *job* pada tahap pertama harus lebih besar atau sama dengan 0.
- iii. *Job* hanya dapat melanjutkan ke tahap berikutnya setelah menyelesaikan proses pada tahap sebelumnya.
- iv. Urutan pengerjaan *job* harus jelas agar proses berjalan secara berurutan dan efisien.

e. Menentukan batasan model

Batasan model ditetapkan untuk memastikan bahwa setiap *job* hanya dapat diproses pada satu mesin di setiap tahap dan mengikuti urutan yang benar. Setiap *job* harus dialokasikan ke mesin yang sesuai, dan urutan pengerjaan *job* harus dipatuhi agar proses produksi berjalan secara efisien.

3.4 Teknik Penyelesaian Model dengan algoritma GWO

Penelitian ini menggunakan algoritma GWO untuk menyelesaikan masalah HFS. Algoritma GWO merupakan salah satu metode metaheuristik yang terinspirasi dari perilaku berburu kawanan serigala abu-abu merujuk pada penelitian Hashem, dkk. (2023). Dalam algoritma ini, solusi diwakili oleh posisi serigala yang terus diperbarui berdasarkan interaksi antar anggota kawanan, yaitu alpha, beta, dan delta untuk mendekati solusi optimal. Tujuan dari penggunaan algoritma GWO pada penelitian ini adalah untuk meminimumkan *makespan* dengan tetap memenuhi semua kendala penjadwalan yang telah ditentukan sebelumnya. Tahapan-tahapan algoritma GWO dalam menyelesaikan masalah HFS adalah sebagai berikut:

1. Inisialisasi populasi serigala abu-abu

Pada tahap ini, populasi serigala abu-abu dibangkitkan secara acak dalam ruang pencarian. Di mana n adalah jumlah serigala dalam kelompok, $X_i(t)$

merepresentasikan serigala ke- i , $i = 1, 2, \dots, n$ pada iterasi t dalam algoritma, dan T adalah jumlah iterasi maksimum. Setiap serigala dalam populasi merepresentasikan sebuah solusi kandidat, yaitu jadwal produksi untuk semua *job* pada setiap tahap dan mesin yang sesuai dengan kendala penjadwalan. Variasi dalam populasi awal bertujuan untuk meningkatkan peluang algoritma dalam menemukan solusi optimal.

2. Inisialisasi parameter a , A , dan C

Parameter a digunakan untuk mengatur sejauh mana serigala abu-abu menjelajahi ruang pencarian. Nilai a dimulai dari 2 dan secara perlahan menurun hingga 0 selama proses iterasi. Penurunan dilakukan dengan rumus:

$$a = 2 - \frac{2t}{T}$$

di mana t iterasi saat ini, dan T adalah jumlah iterasi maksimum. Parameter A mengontrol arah gerakan serigala, apakah mendekati atau menjauh dari solusi terbaik yang ditemukan. Ketika nilai $|A| < 1$, serigala cenderung bergerak mendekati solusi terbaik. Sebaliknya, jika nilai $|A| \geq 1$, serigala akan menjauh dari solusi terbaik untuk mengeksplorasi kemungkinan solusi lain. Nilai A dihitung menggunakan rumus berikut:

$$A = 2a \cdot r_1 - a$$

di mana r_1 adalah bilangan acak dalam interval $[0,1]$. Sementara itu, parameter C digunakan untuk menjaga agar algoritma tidak terjebak pada solusi lokal dan tetap dapat menjelajahi ruang pencarian dengan baik. Nilai C dapat dihitung dengan rumus:

$$C = 2 \cdot r_2$$

di mana r_2 adalah bilangan acak dalam interval $[0,1]$.

3. Evaluasi nilai *fitness* setiap agen pencarian

Setiap solusi dalam populasi awal dievaluasi menggunakan fungsi tujuan untuk menghitung nilai *fitness*. Fungsi tujuan dihitung berdasarkan *makespan* (C_{max}), yaitu waktu penyelesaian maksimum dari seluruh *job*. Semakin kecil nilai *makespan*, semakin baik kualitas solusi tersebut.

4. Penetapan tiga solusi terbaik (X_α , X_β , dan X_δ).

Setelah evaluasi *fitness*, tiga solusi terbaik dalam populasi ditetapkan sebagai alpha (X_α), beta (X_β), dan delta (X_δ). Solusi ini akan memandu pergerakan serigala lainnya dalam iterasi berikutnya, dengan alpha sebagai solusi terbaik utama.

5. Pembaruan posisi agen pencarian

Posisi setiap serigala diperbarui menggunakan persamaan (6) yang ada pada Subbab 2.4, dengan terlebih dahulu mencari nilai masing $\overrightarrow{D}_\alpha$, \overrightarrow{D}_β , dan $\overrightarrow{D}_\delta$ sesuai persamaan (10), (11), dan (12). Kemudian \overrightarrow{X}_1 , \overrightarrow{X}_2 , dan \overrightarrow{X}_3 dihitung dengan persamaan (7), (8), dan (9). Pembaruan posisi ini menghasilkan jadwal baru dengan urutan dan alokasi *job* yang berbeda.

6. Pembaruan parameter a , A , dan C

Setelah pembaruan posisi, nilai parameter a , A , dan C diperbarui untuk menyesuaikan mekanisme eksplorasi dan eksploitasi. Melalui penurunan bertahap pada a , algoritma secara dinamis mengurangi skala perubahan posisi serigala, sehingga konvergensi menuju solusi optimal menjadi lebih stabil.

7. Evaluasi ulang nilai *fitness* semua agen pencarian

Setelah posisi serigala diperbarui, nilai *fitness* kembali dihitung untuk mengevaluasi jadwal baru. Jadwal yang lebih baik memiliki nilai C_{max} yang lebih kecil.

8. Pembaruan solusi terbaik (X_α , X_β , dan X_δ)

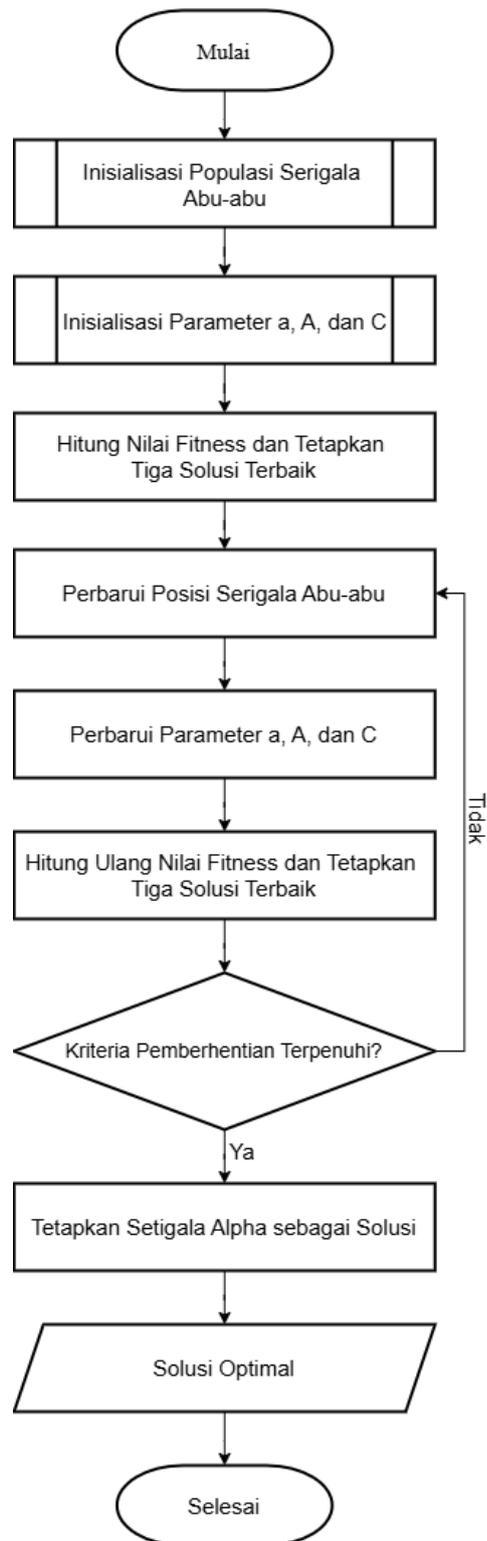
Jika posisi serigala yang diperbarui menghasilkan nilai *fitness* lebih baik daripada solusi alpha, beta, atau delta sebelumnya, solusi ini akan menggantikan posisi alpha, beta, atau delta sesuai urutannya.

9. Pengecekan kriteria pemberhentian

Proses iterasi dilanjutkan hingga salah satu kriteria pemberhentian tercapai, yaitu apabila jumlah iterasi maksimum terpenuhi atau tidak ada perbaikan signifikan pada nilai *fitness* selama sejumlah iterasi berturut-turut. Setelah proses iterasi selesai, posisi alpha (X_α) dipilih sebagai solusi optimal. Solusi ini memberikan jadwal produksi dengan *makespan* minimum, yang

sesuai dengan tujuan optimasi dan memenuhi semua kendala penjadwalan yang telah ditentukan.

Untuk memperjelas tahapan algoritma GWO, berikut disajikan *flowchart* pada Gambar 3.1 yang menggambarkan alur kerja algoritma secara sistematis.



Gambar 3.1 *Flowchart* Algoritma GWO