

**OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN
ALGORITMA GENETIKA DAN ALGORITMA NEH**

Skripsi

Oleh

**MUHAMAD HIDAYAT ARRIDHO SADRACH
NPM. 2117031118**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2025

ABSTRACT

OPTIMIZATION OF JOB SHOP SCHEDULING USING GENETIC ALGORITHM AND NEH ALGORITHM

By

Muhamad Hidayat Arridho Sadrach

This research discusses the optimization of Job Shop Scheduling Problem (JSSP) in the manufacturing industry using Genetic Algorithm (GA) and Nawaz-Enscore-Ham (NEH) Algorithm.. A case study was conducted on the tapioca flour production process at PT Teguh Wibawa Bhakti Persada. The objective of this research is to determine the minimum makespan and compare the effectiveness of both algorithms in solving scheduling problems. The research methodology includes data collection from the production process, algorithm implementation using Python programming language, and performance analysis based on iterations performed. The results indicate that the NEH Algorithm is more efficient in generating fast solutions with a smaller makespan compared to the Genetic Algorithm, which performs better in finding optimal solutions for complex cases. The simulation shows that NEH provides the best makespan of 149 min, while GA achieves a best makespan of 156:46 min after 100 generations. This research provides insights for the manufacturing industry in selecting the optimal scheduling method to improve production efficiency and resource utilization.

Keywords: Job Shop Scheduling, Genetic Algorithm, NEH Algorithm, scheduling optimization, makespan.

ABSTRAK

OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN ALGORITMA GENETIKA DAN ALGORITMA NEH

Oleh

Muhamad Hidayat Arridho Sadrach

Penelitian ini membahas tentang optimalisasi penjadwalan Job Shop Scheduling Problem (JSSP) dalam industri manufaktur menggunakan Algoritma Genetika (AG) dan Algoritma Nawaz-Enscore-Ham (NEH). Studi kasus dilakukan pada proses produksi tepung tapioka di PT Teguh Wibawa Bhakti Persada. Tujuan penelitian ini adalah untuk menentukan makespan minimum serta membandingkan efektivitas kedua algoritma dalam menyelesaikan permasalahan penjadwalan. Metode penelitian mencakup pengumpulan data produksi, implementasi algoritma dalam bahasa pemrograman Python, dan analisis kinerja berdasarkan iterasi yang dilakukan. Hasil penelitian menunjukkan bahwa Algoritma NEH lebih efisien dalam menghasilkan solusi cepat dengan makespan lebih kecil dibandingkan Algoritma Genetika, yang lebih unggul dalam menemukan solusi optimal untuk kasus kompleks. Simulasi menunjukkan bahwa NEH memberikan makespan terbaik sebesar 149 menit, sedangkan AG menghasilkan makespan terbaik sebesar 156:46 menit setelah 100 generasi. Penelitian ini memberikan wawasan bagi industri manufaktur dalam memilih metode penjadwalan yang optimal, guna meningkatkan efisiensi produksi dan pemanfaatan sumber daya.

Kata-kata kunci: Job Shop Scheduling, Algoritma Genetika, Algoritma NEH, optimasi penjadwalan, makespan.

**OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN
ALGORITMA GENETIKA DAN ALGORITMA NEH**

MUHAMAD HIDAYAT ARRIDHO SADRACH

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika

Fakultas Matematika dan Ilmu Pengetahuan Alam



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2025

Judul Skripsi : **OPTIMALISASI PENJADWALAN JOB SHOP MENGGUNAKAN ALGORITMA GENETIKA DAN ALGORITMA NEH**

Nama Mahasiswa : **Muhamad Hidayat Arridho Sadrach**

Nomor Pokok Mahasiswa : **2117031118**

Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

MENYETUJUI

1. Komisi Pembimbing

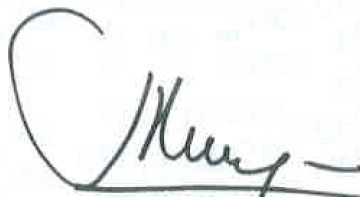


Dr. Notiragayu, S.Si.,M.Si.
NIP 197311092000122001



Dr. Muslim Ansori, S.Si.,M.Si.
NIP 197202271998021001

2. Ketua Jurusan Matematika



Dr.Aang Nuryaman, S.Si.,M.Si.
NIP. 197403162005011001

MENGESAHKAN

1. tim penguji

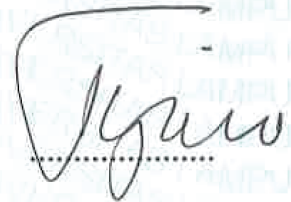
Ketua : **Dr. Notiragayu, S.Si.,M.Si.**



Sekretaris : **Dr. Muslim Ansori, S.Si.,M.Si.**



Penguji
Bukan Pembimbing : **Drs. Tiryono Ruby, M.sc.,Ph.D.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Ing. Heri Satria, S.Si., M.Si.
NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: **17 Maret 2025**

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

Nama : **Muhamad Hidayat Arridho Sadrach**
Nomor Pokok Mahasiswa : **2117031118**
Jurusan : **Matematika**
Judul Skripsi : **OPTIMALISASI PENJADWALAN JOB SHOP MENGGUNAKAN ALGORITMA GENETIKA DAN ALGORITMA NEH**

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri. Apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 17 Maret 2025

Penulis,



Muhamad Hidayat Arridho Sadrach

RIWAYAT HIDUP

Penulis bernama lengkap Muhamad Hidayat Arridho Sadrach yang lahir di Bekasi Kota pada tanggal 01 Juni 2001 sebagai anak pertama dari dua bersaudara dari pasangan Bapak Ferdi dan Ibu Lismiyati.

Penulis telah menempuh pendidikan formal di Taman Kanak-kanak (TK) An-Nursiah pada tahun 2006-2007, Sekolah Dasar (SD) Negeri Perwira 4 Bekasi pada tahun 2007-2013, Sekolah Menengah Pertama (SMP) Negeri 5 Bekasi pada tahun 2013-2016, dan Sekolah Menengah Atas (SMA) Negeri 10 Bekasi pada tahun 2016-2019.

Pada tahun 2021, penulis melanjutkan pendidikan di perguruan tinggi dan terdaftar sebagai mahasiswa S1 Jurusan Matematika Fakultas MIPA Unisversitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjadi mahasiswa, penulis aktif berorganisasi di Himpunan Mahasiswa Jurusan Matematika periode 2022-2023, Staf BEM FMIPA Unila periode 2022.

Sebagai bentuk pengabdian mahasiswa dan menjalankan Tri Dharma Perguruan Tinggi, pada tahun 2024 penulis melaksanakan kegiatan Kerja Praktik (KP) di Badan Perencanaan Pembangunan, Penelitian dan Pengembangan Daerah Bekasi (BAPPELITBANGDA), dan Kuliah Kerja Nyata (KKN) di Desa Tulung Balak, Kecamatan Batang Hari Nuban, Kabupaten Lampung Timur.

KATA INSPIRASI

“Sesungguhnya bersama kesulitan ada kemudahan. Maka, apabila engkau telah selesai (dari suatu urusan), tetaplah bekerja keras.”

(Q.S. Al-Insyirah, 94: 6-7)

“Keberhasilan bukanlah milik orang yang pintar, keberhasilan adalah kepunyaan mereka yang senantiasa berusaha.”

(B.J. Habibie)

”Tidak ada yang tau apa itu kunci sukses, tetapi satu hal yang pasti kunci gagal yaitu menyenangkan semua orang”

(Jerome Polin)

”Dipaksa-Terpaksa-Terbiasa”

(Hamba Allah)

”Terwujud Tak Terwujud Tetap Bersujud”

(Hamba Allah)

”Janganlah hendaknya kamu kuatir tentang apapun juga, tetapi nyatakanlah dalam segala hal keinginanmu kepada Allah dalam doa dan permohonan dengan ucapan syukur”

(Filipi 4:6)

”Walaupun dipersulit dalam hal mencintai, tapi setidaknya dipermudah dalam urusan skripsi”

(Kontrakan)

”Ketika kamu merasa ingin menyerah, ingat alasan mengapa kamu memulai”

(Desi Elena Sitompul)

PERSEMBAHAN

Dengan mengucapkan Alhamdulillah dan syukur kepada Allah SWT atas nikmat serta hidayah-Nya sehingga skripsi ini dapat terselesaikan dengan baik dan tepat pada waktunya. Dengan rasa syukur dan Bahagia, saya persembahkan rasa terimakasih saya kepada :

Ayah dan Ibuku Tercinta

Terimakasih kepada orang tuaku atas segala pengorbanan, motivasi, doa dan ridho serta dukungannya selama ini. Terimakasih telah memberikan pelajaran berharga kepada anakmu ini tentang makna perjalanan hidup yang sebenarnya sehingga kelak bisa menjadi orang yang bermanfaat bagi banyak orang.

Dosen Pembimbing dan Pembahas

Terimakasih kepada dosen pembimbing dan pembahas yang sudah sangat membantu, memberikan motivasi, memberikan arahan serta ilmu yang berharga.

Sahabat-sahabatku

Terimakasih kepada semua orang-orang baik yang telah memberikan pengalaman, semangat, motivasinya, serta doa-doanya dan senantiasa memberikan dukungan dalam hal apapun.

Almamater Tercinta

Universitas Lampung

SANWACANA

Alhamdulillah, puji dan syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini yang berjudul "Optimasi Penjadwalan Job Shop Menggunakan Algoritma Genetika Dan Algoritma Neh" dengan baik dan lancar serta tepat pada waktu yang telah ditentukan. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW.

Dalam proses penyusunan skripsi ini, banyak pihak yang telah membantu memberikan bimbingan, dukungan, arahan, motivasi serta saran sehingga skripsi ini dapat terselesaikan. Oleh karena itu, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Ibu Dr. Notiragayu, S.Si.,M.Si. selaku Pembimbing 1 yang telah banyak meluangkan waktunya untuk memberikan arahan, bimbingan, motivasi, saran serta dukungan kepada penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Dr. Muslim Ansori, S.Si.,M.Si. selaku Pembimbing II yang telah memberikan arahan, bimbingan dan dukungan kepada penulis sehingga dapat menyelesaikan skripsi ini.
3. Bapak Drs. Tiryono Ruby, M.Sc.,Ph.D. selaku Penguji yang telah bersedia memberikan kritik dan saran serta evaluasi kepada penulis sehingga dapat menjadi lebih baik lagi.
4. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Ibu Dr. Khoirin Nisa, M.Si. selaku dosen pembimbing akademik.
6. Seluruh dosen, staff dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

7. Pejuang Budi yaitu David, Zainal, Nanda, Jumi, Miranda, Rena, Reja, Jonathan dan Eri yang selalu mendukung dalam menyelesaikan skripsi ini.
8. Sobat Nyunyai yaitu Edo, Ael, Udin, Fajar, Luqman yang selalu membantu dalam menyelesaikan skripsi ini.
9. Genk Punk yaitu Eri, Rejak, Jenal, Najia, Manda yang selalu kebersamaian serta membantu dalam menyelesaikan skripsi ini.
10. Teman-teman Pimpinan HIMATIKA 2023 serta sobat MIKAT 2023 yang selalu memberikan semangat menyelesaikan Skripsi ini.
11. Circle eksternal terkeceku yang selalu mendukung dan memberikan masukan untuk skripsi ini.
12. Teman Skripsian saya yaitu Maria, Echi, Riska, Aqila, Desi, Prisko, Willy yang selalu mendukung serta memberi saran untuk menyelesaikan skripsi ini.
13. Terimakasih penuh saya ucapkan kepada Desi Elena Sitompul, dan Yunda Rani Agustina Siregar yang selalu menemani, membantu, mengajari, mendukung, serta memberi masukan dan saran untuk skripsi saya agar bisa dikerjakan dengan baik dan selesai tepat waktu.
14. Kepada sosok yang belum diketahui namanya namun sudah tertulis jelas di *lauhul mahfuz*. Terima kasih sudah menjadi salah satu sumber motivasi penulis untuk menyelesaikan skripsi ini sebagai upaya memantaskan diri. Semoga kita berjumpa di versi terbaik kita masing-masing.
15. *Last but not least, I wanna thank me. I wanna thank me for believing in me. I wanna thank me for all doing this hard work. I wanna thank me for having no days off. I wanna thank me for never quitting. I wanna thank me for always being a giver and trying give more than i receive. I wanna thank me for trying to do more right than wrong. I wanna thank me for just being me at all times.*
Semoga skripsi ini dapat bermanfaat bagi kita semua. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, sehingga penulis mengharapkan kritik dan saran yang membangun untuk menjadikan skripsi ini lebih baik lagi.

Bandar Lampung, 17 Maret 2025

Muhamad Hidayat Arridho Sadrach

DAFTAR ISI

DAFTAR ISI	xiv
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Tujuan Penelitian	3
1.3 Manfaat Penelitian	3
II TINJAUAN PUSTAKA	4
2.1 Optimalisasi	4
2.2 Penjadwalan	5
2.3 Model Matematika Masalah Penjadwalan	8
2.4 Penjadwalan Job Shop	9
2.5 Algoritma Genetika (AG)	11
2.5.1 Inisialisasi Populasi Awal	13
2.5.2 Representasi Individu dan Kromosom	13
2.5.3 Evaluasi Fitness	14
2.5.4 Seleksi	15
2.5.5 <i>Crossover</i>	15
2.5.6 Mutasi	16
2.5.7 Evaluasi Fitness Kembali dan Penempatan	16
2.5.8 Iterasi	17
2.6 Algoritma NEH	17
III METODE PENELITIAN	19
3.1 Waktu dan Tempat Penelitian	19
3.2 Data Penelitian	19
3.3 Tahapan Penelitian	20

IV HASIL DAN PEMBAHASAN	22
4.1 Data penelitian JSSP	22
4.2 Model Matematika JSSP	23
4.2.1 Variabel Keputusan	23
4.2.2 Fungsi Objektif	24
4.2.3 Kendala-Kendala dengan Contoh Perhitungan	24
4.3 Penyelesaian JSSP Menggunakan Algoritma Genetika	29
4.3.1 Perhitungan Manual hingga Generasi Ke-5	29
4.3.2 Penyelesaian JSSP Menggunakan Algoritma Genetika Dalam Bahasa Pemograman Python	34
4.3.3 Hasil Perhitungan	40
4.4 Penyelesaian JSSP Menggunakan Algoritma NEH	41
4.4.1 Perhitungan Manual	41
4.4.2 Penyelesaian JSSP Menggunakan Algoritma NEH Dalam Bahasa Pemograman Python	52
4.4.3 Hasil Perhitungan	54
V KESIMPULAN DAN SARAN	55
5.1 Kesimpulan	55
5.2 Saran	56
DAFTAR PUSTAKA	57

DAFTAR TABEL

2.1	Tabel Penjadwalan Job Shop	10
2.2	Kromosom A dan Hasil Mutasi	16
4.1	Durasi Waktu <i>Job</i> pada setiap mesin untuk 1 ton singkong (menit:detik)	23
4.3	Waktu Penyelesaian Operasi untuk Job 1 dalam Menit:Detik	25
4.4	Waktu Penyelesaian Operasi untuk Job 1 sampai Job 5 dalam Menit:Detik	26
4.5	Populasi Awal Generasi 0	29
4.6	Evaluasi Fitness Generasi 0	29
4.7	Evaluasi Fitness Generasi 1	30
4.8	Evaluasi Fitness Generasi 2	30
4.9	Evaluasi Fitness Generasi 3	31
4.10	Evaluasi Fitness Generasi 4	31
4.11	Evaluasi Fitness Generasi 5	32
4.12	Perhitungan dari solusi terbaik generasi ke-5 (menit:detik) .	33
4.13	Hasil 100 generasi menggunakan python	38
4.14	Hasil Pemograman Setiap Generasi	39
4.15	Perhitungan Makespan Terbaik	40
4.16	Total Waktu Pemrosesan Tiap Job	41
4.17	Perhitungan Makespan untuk Urutan [Job 2, Job 3]	42
4.18	Perhitungan Makespan untuk Urutan [Job 3, Job 2]	43
4.19	Perhitungan Makespan untuk Urutan [Job 5, Job 3, Job 2] . .	44
4.20	Perhitungan Makespan untuk Urutan [Job 3, Job 5, Job 2] . .	45
4.21	Perhitungan Makespan untuk Urutan [Job 3, Job 2, Job 5] . .	46
4.22	Perhitungan Makespan untuk Urutan [Job 4, Job 3, Job 5, Job 2]	47
4.23	Perhitungan Makespan untuk Urutan [Job 3, Job 4, Job 5, Job 2]	48

4.24	Perhitungan Makespan untuk Urutan [Job 3, Job 5, Job 2, Job 4]	49
4.25	Perhitungan Makespan untuk Urutan [Job 1, Job 4, Job 3, Job 5, Job 2]	50
4.26	Perhitungan Makespan untuk Urutan [Job 4, Job 1, Job 3, Job 5, Job 2]	51
4.27	Perhitungan Makespan untuk Urutan [Job 4, Job 3, Job 1, Job 5, Job 2]	52

DAFTAR GAMBAR

2.1	Ilustrasi Kromosom, Individu, Populasi	14
2.2	Siklus Algoritma Genetika	17
3.1	Tahapan Penelitian	21
4.1	Diagram Gantt Penjadwalan Job Shop (Generasi Ke-5)	33
4.2	Impor Script	34
4.3	Input Data	35
4.4	Parameter AG	35
4.5	Inisialisasi Populasi Awal	35
4.6	Evaluasi <i>Fitness</i>	35
4.7	Proses Seleksi	35
4.8	Proses rekombinasi (<i>Crossover</i>)	36
4.9	Proses Mutasi	36
4.10	Pengujian Data	36
4.11	Membuat Diagram Gantt	37
4.12	Menjalankan program	37
4.13	Diagram Gantt Urutan Pekerjaan Terbaik Dari 200 Generasi	39
4.14	<i>Script 1</i> Algoritma NEH Pada Python	53
4.15	<i>Script 2</i> Algoritma NEH Pada Python	53
4.16	<i>Script 3</i> Algoritma NEH Pada Python	53

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Penjadwalan merupakan bentuk pengambilan keputusan yang memegang peranan penting dalam banyak disiplin ilmu. Penjadwalan melibatkan pengalokasian sumber daya yang terbatas untuk berbagai aktivitas dengan tujuan mengoptimalkan satu atau lebih ukuran kinerja. Tergantung pada situasi, sumber daya dan aktivitas dapat bervariasi dalam bentuk penjadwalan. Sumber daya tersebut bisa berupa perawat di rumah sakit, pengemudi bus, mesin di pabrik perakitan CPU, mekanik di bengkel mobil, dan lain-lain. Aktivitas dapat berupa operasi dalam proses produksi, tugas perawat di rumah sakit, pelaksanaan program komputer, perbaikan mobil di bengkel mobil, dan sebagainya. Ada juga berbagai ukuran kinerja yang dapat dioptimalkan. Salah satu tujuannya adalah meminimalkan waktu rata-rata, sementara tujuannya lainnya adalah meminimalkan jumlah pekerjaan yang diselesaikan setelah tanggal jatuh tempo. (Blazewicz dkk., 2019)

Dalam dunia industri, efisiensi dan produktivitas merupakan faktor kunci yang menentukan keberhasilan sebuah perusahaan. Salah satu aspek penting dalam meningkatkan efisiensi adalah manajemen penjadwalan produksi. Penjadwalan yang baik tidak hanya berpengaruh pada waktu penyelesaian produksi, tetapi juga pada penggunaan sumber daya, biaya, dan kepuasan pelanggan. Salah satu jenis penjadwalan yang sering dihadapi dalam industri adalah penjadwalan job shop, di mana sejumlah pekerjaan harus dijadwalkan untuk diproses pada beberapa mesin dengan urutan tertentu. (Cheng Wu, 2006)

Penjadwalan job shop merupakan salah satu masalah yang kompleks dalam manajemen produksi, di mana berbagai jenis pekerjaan harus dijadwalkan

pada sejumlah mesin dengan urutan dan waktu pemrosesan yang berbeda (Pinedo, 2016). Tujuan utama dari penjadwalan ini adalah untuk meminimalkan waktu penyelesaian total (*makespan*), mengurangi waktu tunggu, dan meningkatkan efisiensi penggunaan sumber daya. Masalah ini memiliki sifat NP-hard, yang berarti bahwa tidak ada algoritma yang dapat menyelesaikannya secara efisien untuk semua kasus (Garey dan Johnson, 1979).

Dua metode yang sering digunakan dalam menyelesaikan masalah penjadwalan job shop adalah algoritma genetika (genetic algorithm) dan algoritma NEH (Nawaz-Enscore-Ham). Algoritma genetika adalah metode optimasi yang terinspirasi oleh proses evolusi di alam, menggunakan prinsip seleksi, crossover, dan mutasi untuk mengeksplorasi ruang solusi yang besar (Yang et al., 2020). Penelitian menunjukkan bahwa algoritma genetika dapat menghasilkan solusi yang baik untuk berbagai masalah penjadwalan, meskipun sering kali memerlukan waktu komputasi yang lebih lama dan pengaturan parameter yang cermat (Zhang dkk., 2019).

Di sisi lain, algoritma NEH merupakan metode heuristik yang dirancang khusus untuk masalah penjadwalan, terutama dalam konteks minimisasi makespan (Nawaz dkk., 1983). Metode ini bekerja dengan mengurutkan pekerjaan berdasarkan total waktu pemrosesan dan secara iteratif menempatkan pekerjaan ke dalam urutan yang optimal (Chen dkk., 2020). Kelebihan dari algoritma NEH adalah kemampuannya untuk memberikan solusi awal yang baik dengan waktu komputasi yang relatif rendah, sehingga sering digunakan sebagai metode dasar dalam penelitian penjadwalan (Kumar dan Gupta, 2020).

Perbandingan antara kedua metode ini penting untuk memahami keunggulan dan kelemahan masing-masing dalam konteks penjadwalan job shop (Lee dkk., 2018). Beberapa studi menunjukkan bahwa algoritma genetika sering kali lebih unggul dalam menemukan solusi yang lebih baik dalam ruang solusi yang lebih besar, sedangkan algoritma NEH lebih efisien dalam memberikan solusi yang kompetitif dalam waktu yang lebih singkat (Kumar dan Singh, 2021). Hasil Penelitian yang dilakukan oleh Nguyen dapat menentukan metode mana yang lebih efektif dalam menghasilkan solusi untuk berbagai skenario penjadwalan, serta untuk mengeksplorasi kemungkinan penggabungan kedua metode tersebut dalam pendekatan hybrid yang dapat meningkatkan hasil (Nguyen dkk., 2021).

Melalui penelitian ini, Penulis juga ingin membandingkan metode Algoritma Genetika dengan metode Algoritma NEH pada data produksi tepung tapioka PT Teguh Wibawa Bhakti Persada. Diharapkan dapat memberikan wawasan yang lebih dalam mengenai efektivitas algoritma genetika dan algoritma NEH dalam penjadwalan job shop, serta memberikan rekomendasi praktis bagi industri dalam memilih metode yang paling sesuai untuk meningkatkan efisiensi sistem produksi. Hasil penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan teori dan praktik penjadwalan yang lebih baik.

1.2 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

- 1) Menentukan *makespan* minimum pada penjadwalan job shop menggunakan Algoritma Genetika dan Algoritma NEH.
- 2) Membandingkan hasil yang diperoleh menggunakan Algoritma Genetika dan Algoritma NEH mana yang memberikan hasil terbaik.

1.3 Manfaat Penelitian

Manfaat dari penelitian ini adalah:

- 1) Mengoptimalkan waktu dalam penjadwalan job shop dengan menggunakan Algoritma Genetika dan Algoritma NEH, sehingga meningkatkan efisiensi produksi.
- 2) Membantu Meningkatkan efisiensi operasional dan produktivitas dalam industri manufaktur.
- 3) Manufaktur dapat memperoleh strategi penjadwalan yang lebih efisien guna meningkatkan produktivitas dan mengoptimalkan penggunaan sumber daya yang tersedia.

BAB II

TINJAUAN PUSTAKA

2.1 Optimalisasi

Optimalisasi adalah proses mencari nilai terbaik dari suatu fungsi atau sistem di bawah kendala tertentu. Dalam berbagai disiplin ilmu, konsep ini memiliki aplikasi yang luas, mulai dari matematika, ekonomi, teknik, hingga ilmu komputer. Dalam tinjauan pustaka ini, akan dibahas beberapa aspek penting dari optimalisasi, termasuk definisi, metode, dan aplikasinya.

Definisi 2.1.1 Optimalisasi dapat didefinisikan sebagai proses menemukan solusi terbaik dari suatu masalah yang melibatkan variabel-variabel tertentu. Menurut Winston (2004), optimalisasi adalah "proses memilih nilai terbaik dari fungsi objektif yang dipengaruhi oleh sejumlah kendala". Fungsi objektif adalah fungsi yang ingin dimaksimalkan atau diminimalkan, sedangkan kendala adalah batasan yang harus dipenuhi oleh solusi.

Optimalisasi dapat dibedakan menjadi beberapa jenis, antara lain:

- 1) Optimalisasi Linier: Melibatkan fungsi objektif dan kendala yang bersifat linier. Metode Simplex adalah salah satu teknik yang umum digunakan dalam optimalisasi linier (Dantzig, 1963).
- 2) Optimalisasi Non-linier: Terjadi ketika fungsi objektif atau kendala tidak linier. Metode Lagrange dan metode gradient descent adalah beberapa teknik yang digunakan dalam konteks ini (Boyd Vandenberghe, 2004).
- 3) Optimalisasi Diskrit: Berfokus pada masalah di mana variabel keputusan hanya dapat mengambil nilai tertentu, seperti dalam pemrograman integer (Papadimitriou Steiglitz, 1998).
- 4) Optimalisasi Stokastik: Menghadapi ketidakpastian dalam data atau model. Pendekatan ini sering digunakan dalam ekonomi dan keuangan (Birge Louveaux, 2011).

Berbagai metode telah dikembangkan untuk menyelesaikan masalah optimisasi. Beberapa metode yang umum digunakan antara lain:

- 1) Metode Gradient: Digunakan untuk menemukan titik maksimum atau minimum dari fungsi dengan menghitung gradiennya (Nocedal Wright, 2006).
- 2) Metode Simplex: Khusus untuk masalah optimalisasi linier, metode ini mencari solusi optimal dengan berpindah dari satu titik sudut ke titik sudut lainnya dalam ruang solusi (Dantzig, 1963).
- 3) Pemrograman Dinamis: Metode ini digunakan untuk menyelesaikan masalah yang dapat dipecah menjadi sub-masalah yang lebih kecil dan lebih sederhana (Bellman, 1957).
- 4) Algoritma Genetika: Merupakan metode heuristik yang terinspirasi oleh proses evolusi alam. Algoritma ini digunakan untuk menyelesaikan masalah yang kompleks dan tidak dapat diselesaikan dengan metode tradisional (Holland, 1975).
- 5) Algoritma NEH (Nawaz, Enscore, dan Ham) : salah satu metode heuristik yang digunakan dalam bidang penjadwalan, khususnya untuk masalah penjadwalan mesin (machine scheduling) dengan tujuan meminimalkan makespan, yaitu waktu total yang dibutuhkan untuk menyelesaikan semua pekerjaan.

Optimalisasi adalah bidang yang penting dan luas dengan banyak aplikasi praktis. Dengan pemahaman yang mendalam tentang metode dan teknik yang tersedia, individu dan organisasi dapat membuat keputusan yang lebih baik dan lebih efisien. Penelitian lebih lanjut dalam bidang ini dapat menghasilkan metode baru dan inovatif yang dapat diterapkan dalam berbagai konteks.

2.2 Penjadwalan

Penjadwalan adalah proses pengaturan dan pengorganisasian sumber daya untuk melaksanakan serangkaian tugas atau pekerjaan dalam waktu tertentu. Dalam berbagai bidang, seperti manufaktur, transportasi, dan manajemen proyek, penjadwalan memainkan peran penting dalam meningkatkan efisiensi dan produktivitas. Tinjauan pustaka ini akan membahas konsep dasar penjadwalan, jenis-jenis penjadwalan, metode yang digunakan, serta tantangan dan aplikasi dalam berbagai konteks.

Definisi 2.2.1 Penjadwalan dapat didefinisikan sebagai proses menentukan urutan dan waktu pelaksanaan tugas atau pekerjaan untuk mencapai tujuan tertentu, seperti meminimalkan waktu penyelesaian, mengurangi biaya, atau meningkatkan penggunaan sumber daya (Pinedo, 2016). Penjadwalan melibatkan pengalokasian sumber daya, seperti mesin, tenaga kerja, dan bahan, untuk menyelesaikan pekerjaan dalam batas waktu yang ditentukan.

Penjadwalan dapat dibedakan menjadi beberapa jenis, antara lain:

- 1) Penjadwalan Proyek: Mengatur urutan dan waktu pelaksanaan tugas dalam proyek untuk mencapai tujuan proyek secara efisien. Metode seperti PERT (Program Evaluation and Review Technique) dan CPM (Critical Path Method) sering digunakan dalam konteks ini (Kerzner, 2017).
- 2) Penjadwalan Produksi: Mengatur urutan dan waktu produksi barang di pabrik. Ini termasuk penjadwalan mesin, penjadwalan tenaga kerja, dan penjadwalan bahan baku (Graham, 2000).
- 3) Penjadwalan Transportasi: Mengatur rute dan waktu pengiriman barang atau penumpang. Ini melibatkan pengoptimalan rute untuk mengurangi biaya dan waktu perjalanan (Toth Vigo, 2001).
- 4) Penjadwalan Komputasi: Mengatur tugas dalam sistem komputer untuk memaksimalkan penggunaan CPU dan meminimalkan waktu tunggu. Ini termasuk penjadwalan proses dan penjadwalan tugas dalam sistem operasi (Silberschatz et al., 2018).

Berbagai metode dikembangkan untuk menyelesaikan masalah penjadwalan. Beberapa metode yang umum digunakan antara lain:

- 1) Metode Heuristik: Pendekatan berbasis aturan yang digunakan untuk menemukan solusi yang baik dalam waktu yang lebih singkat. Contoh metode heuristik termasuk Algoritma Genetik, Algoritma NEH, Algoritma Jonson dan Algoritma Greedy (Holland, 1975; Nawaz et al., 1983).
- 2) Pemrograman Linier: Metode matematis yang digunakan untuk memodelkan dan menyelesaikan masalah penjadwalan dengan kendala linier. Metode Simplex adalah salah satu teknik yang umum digunakan dalam pemrograman linier (Dantzig, 1963).
- 3) Pemrograman Dinamis: Metode yang digunakan untuk menyelesaikan masalah penjadwalan yang dapat dipecah menjadi sub-masalah yang lebih kecil dan lebih sederhana (Bellman, 1957).

- 4) Simulated Annealing dan Tabu Search: Metode yang digunakan untuk menemukan solusi optimal dengan mengeksplorasi ruang solusi secara acak dan menghindari solusi yang sudah pernah dieksplorasi sebelumnya (Kirkpatrick et al., 1983; Glover, 1989).

Penjadwalan sering kali menghadapi berbagai tantangan, antara lain:

- 1) Ketidakpastian: Ketidakpastian dalam waktu penyelesaian tugas, ketersediaan sumber daya, dan permintaan dapat mempengaruhi efektivitas penjadwalan (Baker Trietsch, 2009).
- 2) Kendala Sumber Daya: Keterbatasan dalam jumlah mesin, tenaga kerja, dan bahan dapat membatasi fleksibilitas dalam penjadwalan (Pinedo, 2016).
- 3) Kompleksitas Masalah: Masalah penjadwalan sering kali NP-Hard, yang berarti tidak ada algoritma yang dapat menyelesaikannya dalam waktu polinomial untuk semua kasus (Garey Johnson, 1979).

Penjadwalan memiliki aplikasi yang luas di berbagai bidang, antara lain:

- 1) Industri Manufaktur: Dalam pengaturan produksi, penjadwalan digunakan untuk mengoptimalkan penggunaan mesin dan tenaga kerja, serta meminimalkan waktu tunggu dan biaya produksi (Graham, 2000).
- 2) Manajemen Proyek: Dalam manajemen proyek, penjadwalan digunakan untuk merencanakan dan mengelola urutan tugas, memastikan bahwa proyek selesai tepat waktu dan sesuai anggaran (Kerzner, 2017).
- 3) Transportasi dan Logistik: Penjadwalan rute pengiriman dan pengaturan jadwal transportasi sangat penting untuk efisiensi operasional dan pengurangan biaya (Toth Vigo, 2001).
- 4) Sistem Komputer: Dalam sistem operasi, penjadwalan proses digunakan untuk mengelola eksekusi tugas dan memaksimalkan penggunaan sumber daya komputer (Silberschatz et al., 2018).

Penjadwalan adalah aspek krusial dalam manajemen sumber daya yang mempengaruhi efisiensi dan produktivitas di berbagai bidang. Dengan memahami berbagai jenis penjadwalan, metode yang digunakan, serta tantangan yang dihadapi, para peneliti dan praktisi dapat mengembangkan strategi yang lebih baik untuk mengatasi masalah penjadwalan yang kompleks. Penelitian lebih lanjut diperlukan untuk mengembangkan metode baru yang dapat mengatasi

tantangan yang ada dan meningkatkan efektivitas penjadwalan dalam konteks yang berbeda.

2.3 Model Matematika Masalah Penjadwalan

Variabel Keputusan

- x_{ijk} : Variabel biner, bernilai 1 jika operasi- ij (operasi ke- j pada job ke- i) dijadwalkan pada mesin k , bernilai 0 jika tidak.
- C_{ij} : Waktu penyelesaian operasi- ij .
- p_{ij} : Waktu pemrosesan operasi- ij pada Mesin ke- j
- S_{ij} : Waktu mulai operasi- ij .

Fungsi Objektif

Fungsi objektif untuk meminimalkan *makespan* (total waktu penyelesaian):

$$\min C_{\max}$$

dengan

$$C_{\max} = \max(C_{ij}), \quad \forall i, j$$

Kendala-Kendala

(a) Urutan Operasi dalam Job:

Operasi pada job harus dikerjakan secara berurutan:

$$S_{i,j+1} \geq C_{ij}, \quad \forall i = 1, 2, \dots, n \quad \forall j = 1, 2, \dots, n - 1$$

(Waktu mulai operasi ke- $j + 1$ dari job i harus lebih besar atau sama dengan waktu selesai operasi ke- j dari job yang sama).

(b) Satu Mesin Tidak Bisa Memproses Dua Job Bersamaan:

Setiap mesin hanya dapat menjalankan satu operasi dalam satu waktu:

$$S_{ij} \geq C_{lj} \text{ atau } S_{lj} \geq C_{ij}, \quad \forall (i, j) \neq (l, j),$$

jika operasi dijalankan pada mesin yang sama.

(c) **Waktu Penyelesaian Operasi:**

Waktu penyelesaian operasi dihitung berdasarkan waktu mulai dan durasi operasi:

$$C_{ij} = S_{ij} + p_{ij}, \quad \forall i, j$$

di mana p_{ij} adalah durasi operasi- ij .

(d) **Alokasi Mesin:**

Operasi hanya dapat dijalankan pada mesin yang sesuai:

$$\sum_k x_{ijk} = 1, \quad \forall i, j$$

2.4 Penjadwalan Job Shop

Penjadwalan job shop adalah metode pengaturan proses produksi yang digunakan untuk mengelola berbagai jenis pekerjaan yang berbeda dalam satu fasilitas produksi. Sistem ini ditandai dengan fleksibilitas tinggi, di mana setiap pekerjaan dapat memiliki urutan dan waktu proses yang berbeda (Pinedo, 2016). Konsep utama dalam penjadwalan job shop adalah untuk meminimalkan waktu penyelesaian total (*makespan*), keterlambatan (*tardiness*), mengurangi waktu tunggu, dan memaksimalkan penggunaan sumber daya.

Kriteria yang umum digunakan dalam penjadwalan job shop meliputi *makespan* (total waktu penyelesaian), *tardiness* (keterlambatan), dan total waktu tunggu (Cheng et al., 2005). Penjadwalan yang efektif tidak hanya berfokus pada pengurangan waktu penyelesaian, tetapi juga pada pengoptimalan penggunaan sumber daya dan peningkatan kepuasan pelanggan.

Makespan adalah waktu total yang diperlukan untuk menyelesaikan semua pekerjaan dalam sistem penjadwalan. Ini diukur dari waktu mulai pekerjaan pertama hingga waktu selesai pekerjaan terakhir. *Makespan* yang lebih pendek menunjukkan efisiensi yang lebih baik dalam penjadwalan, karena berarti semua pekerjaan dapat diselesaikan lebih cepat. Dalam konteks industri, mengurangi *makespan* dapat meningkatkan produktivitas dan mengurangi biaya operasional. Sedangkan *Tardiness*

mengacu pada seberapa lama suatu pekerjaan terlambat dalam penyelesaiannya dibandingkan dengan waktu yang dijadwalkan. Jika pekerjaan selesai setelah tenggat waktu yang ditentukan, maka pekerjaan tersebut dianggap terlambat. Keterlambatan dapat berdampak negatif pada kepuasan pelanggan dan dapat menyebabkan penalti atau biaya tambahan. Oleh karena itu, mengurangi tingkat keterlambatan adalah salah satu tujuan penting dalam penjadwalan. Begitu juga dengan Total waktu tunggu mengacu pada jumlah waktu yang dihabiskan oleh pekerjaan dalam antrean sebelum mulai diproses. Waktu tunggu yang tinggi dapat menunjukkan adanya kemacetan dalam sistem penjadwalan dan dapat mengindikasikan bahwa sumber daya tidak digunakan secara efisien. Mengurangi total waktu tunggu dapat meningkatkan throughput (jumlah pekerjaan yang diselesaikan dalam periode tertentu) dan mengurangi waktu siklus keseluruhan.

Tantangan utama dalam penjadwalan job shop termasuk ketidakpastian dalam waktu proses, perubahan permintaan, dan keterbatasan sumber daya (Baker dan Kanet, 2005). Penjadwalan yang adaptif dan responsif menjadi semakin penting dalam menghadapi dinamika pasar yang cepat berubah.

Sebagai contoh penerapan penjadwalan job shop, pertimbangkan sebuah pabrik yang memproduksi tiga jenis produk (A, B, dan C) dengan dua mesin (M1 dan M2). Setiap produk memiliki waktu proses yang berbeda pada masing-masing mesin:

Tabel 2.1 Tabel Penjadwalan Job Shop

<i>Produk</i>	<i>M1</i>	<i>M2</i>
<i>A</i>	3	2
<i>B</i>	2	1
<i>C</i>	4	3

Dalam penjadwalan ini, manajer produksi harus menentukan urutan pekerjaan untuk meminimalkan makespan. Misalnya, jika urutan pekerjaan adalah $A \rightarrow B \rightarrow C$, maka total waktu penyelesaian dapat dihitung

berdasarkan waktu proses pada masing-masing mesin.

2.5 Algoritma Genetika (AG)

Algoritma Genetika (AG) dikembangkan pada tahun 1975 di Universitas Michigan oleh Jhon Holland. AG diartikan sebagai algoritma yang menggunakan proses seleksi alam atau bisa juga disebut proses evolusi sesuai dengan yang diutarakan oleh Charles Darwin. Individu yang ada akan melakukan perubahan gen secara terus-menerus hingga sesuai dengan lingkungannya (Mubarok dan Chotijah, 2021). Menurut (Lukas, dkk., 2005), proses AG terdapat beberapa langkah, yaitu pengkodean (encoding), seleksi (selection), persilangan (crossover), dan mutasi (mutation).

Salah satu metode penyelesaian permasalahan np-hard yang cukup efektif adalah metode algoritma heuristik (Heuristic Algorithm), yaitu suatu jenis algoritma yang termasuk ke dalam jenis algoritma sub-optimal. Meskipun algoritma heuristik termasuk ke dalam jenis algoritma sub-optimal yang tidak dapat menjamin tercapainya suatu solusi yang optimal (best solution), namun algoritma heuristik akan dapat secara efektif mengatasi permasalahan yang cukup sulit dan berskala besar dengan cara mencari solusi terbaik yang dapat memuaskan semua kriteria dengan waktu komputasi yang relatif kecil. Salah satu metode algoritma heuristik yang cukup populer adalah neighborhood search methods, dimana didalamnya terdapat tiga macam metode, yaitu : Genetic Algorithm, Simulated Annealing dan Tabu Search.

Algoritma Genetika (Genetic Algorithm, GA) adalah metode optimasi berbasis *heuristic* yang terinspirasi oleh prinsip evolusi biologis, seperti seleksi alam, *crossover*, dan mutasi. Algoritma ini digunakan untuk menyelesaikan permasalahan optimasi kompleks, termasuk penjadwalan **Job Shop Scheduling Problem (JSSP)**.

Job Shop Scheduling Problem (JSSP) adalah masalah optimasi yang bertujuan untuk menjadwalkan serangkaian pekerjaan (*job*) pada sejumlah mesin agar memenuhi fungsi objektif tertentu, seperti:

- **Makespan:** Total waktu penyelesaian semua pekerjaan.
- **Flow time:** Jumlah total waktu penyelesaian semua pekerjaan.
- **Tardiness:** Penundaan pekerjaan yang selesai setelah tenggat waktu.

Karena JSSP termasuk dalam kelas masalah **NP-hard**, metode optimasi seperti algoritma genetika cocok digunakan untuk mencari solusi yang mendekati optimal dalam waktu komputasi yang terbatas.

Penggunaan Algoritma Genetika

1. Representasi Solusi

Solusi dalam GA direpresentasikan sebagai kromosom. Dalam konteks JSSP, kromosom dapat berupa urutan operasi untuk setiap mesin.

Contoh kromosom:

$$[O_{11}, O_{12}, O_{21}, O_{22}, \dots]$$

2. Evaluasi Fitness

Setiap kromosom dievaluasi berdasarkan fungsi objektif (C_{\max}).

3. Operator Evolusi

- **Seleksi:** Memilih individu terbaik berdasarkan nilai *fitness*.
- **Crossover:** Menggabungkan dua solusi untuk menghasilkan solusi baru.
- **Mutasi:** Memodifikasi solusi untuk menjaga keberagaman populasi.

4. Iterasi

Algoritma dijalankan hingga konvergensi atau mencapai batas iterasi yang ditentukan.

Beberapa generasi baru yang terbentuk dan memiliki ketahanan hidup yang kuat akan bermunculan didalam populasi tersebut, yang kemudian dari proses seleksi, kawin silang, dan mutasi yang berkelanjutan meng-

hasilkan generasi dengan kemampuan bertahan hidup yang bagus. Algoritma ini berbeda dari algoritma tradisional pada beberapa hal (Goldberg, 1989) yaitu:

- 1) AG berkeja dalam bentuk kode sekumpulan parameter, bukan parameter itu sendiri.
- 2) AG mencari solusi dari sekumpulan populasi, bukan dari satu titik solusi.
- 3) AG langsung menggunakan informasi berupa fungsi tujuan dan bukan turunannya.
- 4) AG menggunakan aturan probabilitas, bukan aturan deterministik. Penerapan metode Algoritma Genetika melibatkan delapan langkah utama (Chairul Saleh, 2002) :
 - a) Inisialisasi populasi
 - b) Representasi individu dan kromosom
 - c) Evaluasi fitness
 - d) Seleksi
 - e) Crossover
 - f) Mutasi
 - g) Penempatan
 - h) Iterasi

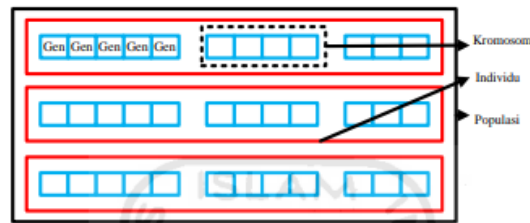
2.5.1 Inisialisasi Populasi Awal

Ada dua cara untuk menentukan inisialisasi populasi awal, yaitu secara random dan secara heuristic (Gen dan Cheng, 2000). Dalam penelitian ini inisialisasi populasi dibangkitkan secara random.

2.5.2 Representasi Individu dan Kromosom

Representasi memberikan pengaruh yang kuat terhadap performansi Algoritma Genetika. Representasi individu bergantung pada masalah yang akan diselesaikan. Individu menyatakan salah satu solusi yang mungkin. Setiap individu merupakan kumpulan kromosom sedangkan kromosom merupakan kumpulan dari gen. Sedangkan populasi merupakan kumpulan dari beberapa individu. Pada gambar 2.1 diilustrasikan perbedaan

istilah-istilah tersebut.



Gambar 2.1 Ilustrasi Kromosom, Individu, Populasi

Individu merupakan kombinasi permutasi dari penjadwalan pekerjaan yang dilakukan oleh mesin, dari mesin-1 hingga mesin-n. Setiap individu terdiri dari n kromosom. Kromosom menunjukkan urutan job yang dibebankan pada mesin tersebut. Sebagai contoh, misalkan ada 2 mesin dan jumlah job yang melalui mesin satu yaitu job 1, 2, 3 dan job yang melalui mesin 2 yaitu job 1, 2, 3 maka representasinya yaitu (1-2-3)-(1-2-3).

2.5.3 Evaluasi Fitness

Fungsi fitness digunakan untuk mengevaluasi individu dalam populasi. Fitness adalah sebuah perhitungan nilai yang harus dimaksimalkan, diminimalkan atau disesuaikan dengan nilai terdekat paling mungkin yang diberikan. Pada penjadwalan, waktu yang diperlukan untuk menyelesaikan semua pekerjaan yang ada di shop (makespan) harus diminimalkan. Pada penelitian ini, fitness (obyektif) yang dicari yaitu makespan. Obyektif makespan adalah total waktu yang dibutuhkan untuk menyelesaikan semua job yang ada atau dapat juga dikatakan sebagai waktu saat selesainya job terakhir. Obyektif makespan ini dapat dinyatakan dengan persamaan matematis :

$$Makespan = Max[C1, C2, \dots, Cn] \quad (2.5.1)$$

Individu yang mempunyai nilai fungsi fitness terkecil adalah individu terbaik yang akan mendekati optimal. Tujuan penjadwalan pada penelitian adalah minimasi fungsi fitness F supaya didapatkan nilai makespan yang minimal. Dengan nilai makespan yang kecil maka diharapkan akan mengurangi keterlambatan dalam penyelesaian pembuatan produk.

2.5.4 Seleksi

Prosedur seleksi pada intinya mengarahkan pencarian pada Algoritma Genetik ke daerah yang lebih menjanjikan pada ruang pencarian. Terdapat tiga hal yang perlu diperhatikan dalam fase seleksi, meliputi ruang sampling, mekanisme sampling dan peluang sampling (Gen and Cheng, 2000). Ada dua metode seleksi, yaitu proporsional dan ranking. Dalam penelitian ini akan digunakan metode ranking yaitu dengan cara mengurutkan hasil evaluasi fitness setiap individu lalu akan dipilih individu dengan hasil terbaik.

2.5.5 Crossover

Dalam algoritma genetika, "crossover" adalah proses menggabungkan dua individu untuk membuat individu baru karena hal tersebut *crossover* umumnya diistilahkan dengan X. Dua individu baru tersebut dibentuk oleh dua individu induk dalam populasi melalui proses seleksi sebelumnya. Dalam proses persilangan, perlu ditentukan probabilitas persilangan (*Crossover Probability*) yang menunjukkan kemungkinan crossover terjadi antara 2 individu. Jika tidak terjadi *crossover*, maka keturunannya akan sama persis dengan individu orangtua, tetapi tidak berarti generasi yang baru akan sama persis dengan generasi yang lama. *Crossover* dilakukan dengan harapan bahwa individu yang baru akan lebih baik.

Crossover memiliki beberapa metode yang umum digunakan yaitu:

- 1) *One-Point Crossover* (Persilangan Satu Titik) : Dalam metode ini, memilih satu titik pada kromosom induk, lalu menukar bagian kromosom di sebelah kanan dari titik tersebut antara kedua induk.
- 2) *Two-Point Crossover* (Persilangan Dua Titik) : Dalam metode ini, memilih dua titik pada kromosom induk, lalu menukar bagian di antara kedua titik tersebut antara kedua induk.
- 3) *Order Crossover (OX)* : Metode *Order Crossover (OX)* adalah salah satu metode yang cocok untuk masalah penjadwalan, di mana harus memastikan bahwa setiap pekerjaan hanya muncul sekali dalam kromosom anak. Metode ini menjaga urutan gen dari satu induk, sambil menyisipkan elemen yang hilang dari induk lain.

- 4) *Partially Mapped Crossover* (PMX) : Metode PMX adalah metode *crossover* lain yang cocok untuk masalah penjadwalan. Dalam metode ini, menukar segmen antara dua titik potong antara dua kromosom induk, kemudian memperbaiki duplikasi yang mungkin terjadi.

2.5.6 Mutasi

Mutasi adalah sebuah proses dari perubahan gen dalam kondisi random untuk membentuk gen baru dalam kelompok gen. Dalam Algoritma Genetika, mutasi memberikan aturan untuk menempatkan kembali gen yang hilang dari populasi selama proses seleksi sehingga dapat memberikan gen yang tidak ada pada populasi sebelumnya. Mutasi digunakan untuk mencegah generasi premature dalam operasi *crossover*. Pada proses mutasi, diperlukan adanya probabilitas mutasi. Probabilitas Mutasi (Mutation Probability) menunjukkan kemungkinan mutasi terjadi pada gen-gen yang menyusun sebuah individu. Jika tidak terjadi mutasi maka keturunan yang dihasilkan setelah *crossover* tidak berubah. Jika terjadi mutasi bagian individu akan berubah. Tabel 2.2 menunjukkan proses mutasi pada gen ketiga dan ketujuh.

Tabel 2.2 Kromosom A dan Hasil Mutasi

<i>A</i>	<i>Job7</i>	<i>Job6</i>	<i>Job4</i>	<i>Job5</i>	<i>Job3</i>	<i>Job2</i>	<i>Job1</i>
<i>Hasil</i>	<i>Job7</i>	<i>Job6</i>	<i>Job1</i>	<i>Job5</i>	<i>Job3</i>	<i>Job2</i>	<i>Job4</i>

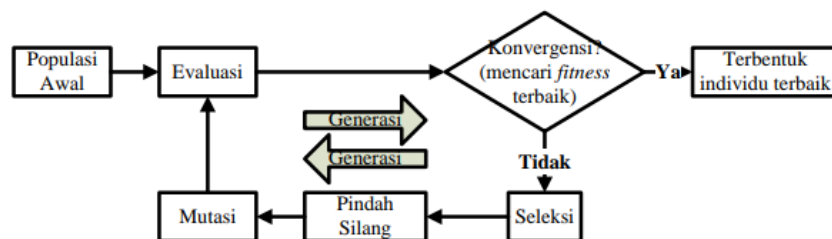
2.5.7 Evaluasi Fitness Kembali dan Penempatan

Setelah mutasi, perlu mengevaluasi fitness individu baru ini dengan cara yang sama seperti sebelumnya. Hitung waktu penyelesaian total untuk individu yang telah dimutasi. Berdasarkan pada pendekatan pembentukan generasi, semua populasi ditempatkan kembali dalam masing-masing generasi (Goldberg, 1989). Dalam penelitian ini menggunakan generator Algoritma Genetika. Dalam generator ini, jumlah dari n populasi terbaik dalam generasi terakhir akan digunakan untuk menempatkan kembali populasi yang terburuk dari generasi sebelumnya. Metode

ini membuat sejumlah individu terbaik tiap generasi akan otomatis diturunkan ke generasi berikutnya (Koza, 2001).

2.5.8 Iterasi

Proses pemilihan, crossover, mutasi, evaluasi fitness, dan penempatan diulang selama beberapa generasi. Setiap generasi diharapkan menghasilkan individu dengan waktu penyelesaian yang lebih baik. Proses ini berlanjut hingga mencapai solusi yang memuaskan atau hingga jumlah generasi tertentu tercapai. Jumlah generasi (jumlah iterasi digunakan sebagai batas akhir proses seleksi, persilangan dan mutasi. Jumlah generasi besar berarti semakin banyak iterasi yang dilakukan, dan semakin besar domain solusi yang akan dieksplorasi tetapi Algoritma Genetik akan berjalan lambat. Jika diilustrasikan, maka proses Algoritma Genetika yang dilakukan dapat terlihat seperti di bawah ini :



Gambar 2.2 Siklus Algoritma Genetika

2.6 Algoritma NEH

Metode ini dikembangkan oleh Nawaz, Enscore dan Ham pada tahun 1983. Metode ini juga disebut metode Incremental Construction Algorithms. Algoritma NEH dapat ditunjukkan melalui langkah-langkah berikut ini :

Langkah 1:

- a) Jumlahkan waktu proses setiap job.
- b) Urutkan job-job menurut waktu prosesnya dimulai dari yang terbesar hingga terkecil.
- c) Hasil urutan ini disebut sebagai daftar pengurutan job.

Langkah 2:

- a) Ambil job yang menempati urutan pertama dan kedua pada daftar pengurutan job-job.
- b) Buat dua alternatif calon urutan parsial baru.
- c) Hitung makespan parsial dan mean flow time parsial dari calon urutan parsial baru.
- d) Pilih calon urutan parsial baru yang memiliki makespan parsial yang terkecil. Jika ada urutan calon parsial baru yang memiliki makespan parsial terkecil yang sama, pilihlah calon urutan parsial yang baru tadi yang memiliki mean flow time parsial yang lebih kecil. Jika masih sama, maka pilihlah calon urutan parsial baru tadi secara acak.
- e) Calon urutan parsial yang baru yang terpilih menjadi urutan parsial baru.
- f) Coret job-job yang diambil dari daftar pengurutan job-job.
- g) Periksa apakah $k = n$ (n adalah jumlah job yang ada). Jika ya, lanjutkan ke langkah 4. Jika tidak lanjutkan ke langkah 3.

Langkah 3:

- a) Ambil job yang menempati urutan pertama dari daftar pengurutan job-job.
- b) Hasilkan sebanyak k calon urutan parsial baru dengan memasukkan job yang diambil ke dalam slot urutan parsial sebelumnya.
- c) Hitung makespan parsial dan mean flow time parsial dari calon urutan parsial baru.
- d) Pilih calon urutan parsial baru yang memiliki makespan parsial yang terkecil. Jika ada urutan calon parsial baru yang memiliki makespan parsial terkecil yang sama, pilihlah calon urutan parsial yang baru tadi yang memiliki mean flow time parsial yang lebih kecil. Jika masih sama, maka pilihlah calon urutan parsial baru tadi secara acak.
- e) Calon urutan parsial yang baru yang terpilih menjadi urutan parsial baru.
- f) Coret job-job yang diambil dari daftar pengurutan job-job.
- g) Periksa apakah $k = n$ (n adalah jumlah job yang ada). Jika ya, lanjutkan ke langkah 4. Jika tidak ulangi lagi langkah 3.

Langkah 4:

- a) Urutan parsial baru menjadi urutan final.
- b) Proses perhitungan berhenti.

BAB III

METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada Semester Genap Tahun Ajaran 2023/2024 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang beralamatkan di Jalan Prof. Dr. Ir. Soemantri Brojonegoro, Gedong Meneng, Kecamatan Rajabasa, Kota Bandar Lampung, Lampung.

3.2 Data Penelitian

Penelitian ini menggunakan metode pengumpulan data dengan melakukan wawancara Bersama narasumber serta didukung oleh studi literatur yang diperoleh. Wawancara yang dilakukan Bersama narasumber bertujuan untuk mendapatkan data terkait PRODUKSI TEPUNG TAPIOKA PT TEGUH WIBAWA BHAKTI PERSADA.

Tahapan dalam penelitian ini adalah identifikasi masalah, perumusan masalah, penyelesaian masalah, dan penarikan kesimpulan. Identifikasi masalah dilakukan untuk mendapat gambaran mengenai permasalahan yang akan diselesaikan. Data pada penelitian ini diperoleh dengan survey langsung ke lapangan. Penyelesaian masalah dilakukan dengan menggunakan Algoritma Genetika dan Algoritma NEH.

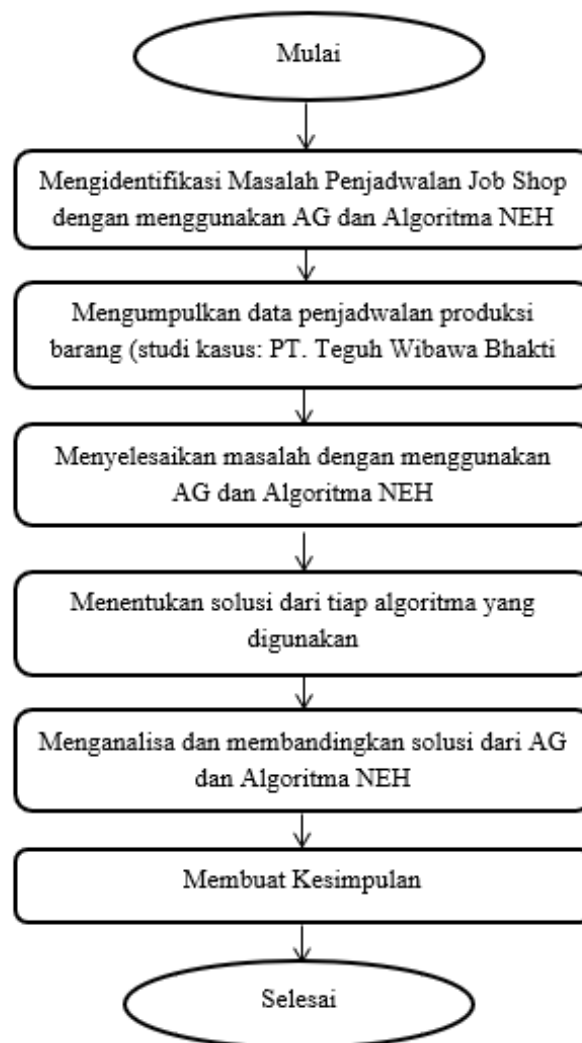
3.3 Tahapan Penelitian

Metode penelitian ini meliputi jenis penelitian, populasi dan sampel, teknik pengumpulan data, serta teknik analisis data. Penjelasan ini diharapkan dapat memberikan gambaran yang jelas tentang cara penulis melakukan penelitian.

Langkah-langkah yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Menentukan dengan jelas masalah penjadwalan Job Shop yang akan diteliti.
2. Memilih metode yang tepat, seperti Algoritma Genetika dan NEH, untuk menyelesaikan masalah.
3. Mengumpulkan data penjadwalan produksi barang dari studi kasus PT. Teguh Wibawa Bhakti Persada, Lampung.
4. Menerapkan Algoritma Genetika dan Algoritma NEH untuk memecahkan masalah penjadwalan Job Shop.
5. Menentukan solusi yang dihasilkan oleh masing-masing algoritma.
6. Menganalisis solusi yang dihasilkan oleh masing-masing algoritma.
7. Membandingkan solusi dari Algoritma Genetika dan Algoritma NEH.
8. Menyimpulkan hasil analisis dan perbandingan, serta memberikan rekomendasi untuk solusi terbaik.

Secara singkat proses optimalisasi penjadwalan Job Shop menggunakan Algoritma Genetika dan Algoritma NEH sebagai berikut :



Gambar 3.1 Tahapan Penelitian

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan mengenai optimalisasi penjadwalan Job Shop menggunakan Algoritma Genetika dan Algoritma NEH, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Algoritma Genetika (AG) dan Algoritma NEH Dapat digunakan untuk menyelesaikan masalah penjadwalan Job Shop Scheduling Problem (JSSP) yang kompleks. Kedua algoritma ini menghasilkan solusi yang dapat meningkatkan efisiensi waktu produksi.
2. Hasil penelitian menunjukkan bahwa Algoritma Genetika mampu menghasilkan solusi dengan makespan yang lebih optimal setelah beberapa iterasi. Dengan melakukan seleksi, crossover, dan mutasi, AG secara bertahap menemukan solusi yang lebih baik.
3. Algoritma NEH, sebagai algoritma heuristik, memberikan hasil yang kompetitif dengan perhitungan manual yang lebih cepat yaitu 151:35 dibandingkan AG sebagai algoritma heuristik dengan hasil 154:48 Menit.
4. Berdasarkan hasil simulasi menggunakan bahasa pemrograman Python, AG menghasilkan makespan terbaik sebesar 156:46 dari 100 generasi. Algoritma NEH memberikan hasil yang paling optimal yaitu 149 Menit tetapi dengan iterasi yang lebih sedikit.
5. Perbandingan antara kedua metode menunjukkan bahwa Algoritma NEH lebih efisien dalam menghasilkan solusi cepat dengan makespan lebih kecil dibandingkan Algoritma Genetika, yang lebih unggul dalam menemukan solusi optimal untuk kasus kompleks.

5.2 Saran

Sebagai tindak lanjut dari penelitian ini, beberapa saran yang dapat diberikan untuk pengembangan lebih lanjut adalah sebagai berikut:

1. **Eksperimen dengan Parameter Berbeda:** Dalam implementasi Algoritma Genetika, perlu dilakukan pengujian lebih lanjut terhadap parameter seperti jumlah populasi, tingkat mutasi, dan probabilitas crossover untuk melihat dampaknya terhadap hasil akhir.
2. **Pengujian dengan Dataset Lebih Besar:** Untuk menguji kehandalan metode yang digunakan, penelitian lebih lanjut dapat dilakukan dengan dataset yang lebih variatif.
3. **Implementasi pada Industri Lain:** Penelitian ini dapat diperluas dengan mengaplikasikan metode yang sama pada industri lain yang memiliki permasalahan penjadwalan produksi serupa, seperti industri otomotif atau manufaktur lainnya.
4. **Optimasi Komputasi:** Untuk meningkatkan efisiensi pemrosesan, dapat dilakukan eksplorasi terhadap metode komputasi paralel atau penggunaan perangkat keras yang lebih canggih seperti GPU dalam menjalankan Algoritma Genetika.

DAFTAR PUSTAKA

- Baker, K. R., dan Kanet, J. J. (2005). *Introduction to Scheduling. In Production and Operations Management.* (pp. 123-145). New York: Wiley
- Baker, K. R., dan Trietsch, D. (2009). *Principles of Scheduling.* Wiley.
- Bierwirth, C., D.C. Mattfield, (1999). "Production Scheduling and Rescheduling with Genetic Algorithm", *Evolutionary Computation*,. 7 (1), 1999, 1-17.
- Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., dan Weglarz, J. (2019). *Handbook on scheduling: Algorithms and systems.* Springer.
- Bellman, R. (1957). *Dynamic Programming.* Princeton University Press.
- Birge, J. R., dan Louveaux, F. (2011). *Introduction to Stochastic Programming* Springer.
- Boyd, S., dan Vandenberghe, L. (2004). *Group Theory and Physics* Cambridge University Press.
- Chairul S, Azmi H, Md. Yusoff J, (2002), Optimasi Ukuran Lot Menggunakan Pengulangan Ganda Algoritma Genetika. Proceeding Seminar Nasional Teknik Industri dan Manajemen Produksi TIMP, 207-212, Surabaya.
- Cheng, T. C. E., Gupta, J. N. D., dan Kuo, R. J. (2005). *Scheduling in a Job Shop Environment: A Review.* *European Journal of Operational Research*, 164(3), 689-709.
- Cheng, T. C. E., dan Wu, Y. (2006). "A Genetic Algorithm for Job Shop Scheduling". *International Journal of Advanced Manufacturing Technology*, 29(3-4), 341-349.

- Chen, L., Wang, J., da Zhang, Y. (2020). "An Improved NEH Heuristic Algorithm for Job Shop Scheduling Problem." *Journal of Intelligent Manufacturing*, 31(3), 715-726.
- Dantzig, G. B. (1963). *Linear Programming and Extensions*. Princeton University Press.
- Garey, M. R., dan Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company.
- Gen, M. dan Cheng, R. (2000), *Genetic Algorithms and Engineering Design*, John Wiley and Sons.
- Glover, F. (1989). *Tabu Search—Part I*. *ORSA Journal on Computing*, 1(3), 190-206.
- Goldberg, David E., (1989). *Genetic Algorithms, in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Co. Inc.
- Graham, R. L. (2000). *An Efficient Algorithm for Scheduling Independent Tasks*. *Operations Research*, 28(1), 156-165.
- Garey, M. R., dan Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press.
- Kerzner, H. (2017). *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. Wiley.
- Kirkpatrick, S., Gelatt, C. D., dan Vecchi, M. P. (1983). *Optimization by Simulated Annealing*. *Science*, 220(4598), 671-680.
- Koza, J., (2001). *Genetic algorithms*. In *Genetic Programming: On the Programming of Computers by Means of Natural Selection* 1-20, MIT Press.
- Kumar, A., dan Gupta, R. (2020). *Performance Evaluation of NEH Algorithm in Job Shop Scheduling*." *International Journal of Advanced Manufacturing Technology*, 118(1-2), 345-356.

- Kumar, A., dan Singh, P. (2021). "A Comparative Study of Genetic Algorithm and NEH for Job Shop Scheduling." *Journal of Manufacturing Systems*, 58, 123-134.
- Lee, S., Kim, H., dan Park, J. (2018). "Comparative Analysis of Genetic Algorithm and NEH Heuristic for Job Shop Scheduling." *Computers and Industrial Engineering*, 115, 1-10.
- Lukas, S., Anwar, T., Yuliani, W. (2005). Penerapan Algoritma Genetika untuk *Traveling Salesman Problem* dengan Menggunakan Metode *Order Crossover* dan *Insertion Mutation*. Seminar Nasional Aplikasi Teknologi Informasi 2005. ISBN: 979-756-061-6.
- Mubarok, A. Y., Chotijah, U. (2021). Penerapan Algoritma Genetika untuk Mencari Optimasi Kombinasi Jalur Terpendek dalam Kasus *Travelling Salesman Problem*. *Jurnal Teknologi Terpadu*. 7(2): 77-82
- Nawaz, M., Enscore, E. E., dan Ham, I. (1983). "A Heuristic Algorithm for the Job Shop Scheduling Problem." *International Journal of Production Research*, 21(1), 91-98.
- Nguyen, T. T., Le, T. H., dan Tran, Q. (2021). "Hybrid Approaches for Job Shop Scheduling: A Review." *Journal of Scheduling*, 24(5), 543-558.
- Nocedal, J., dan Wright, S. J. (2006). *Numerical Optimization*. Springer.
- Papadimitriou, C. H., dan Steiglitz, K. (1998). *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall.
- Pinedo, M. (2016). *Scheduling: Theory, Algorithms, and Systems*. Springer.
- Silberschatz, A., Galvin, P. B., dan Gagne, G. (2018). *Operating System Concepts*. Wiley.
- Toth, P., dan Vigo, D. (2001). *The Vehicle Routing Problem*. *SIAM Monographs on Discrete Mathematics and Applications*.
- Wang, Y., Xu, Z., dan Wang, H. (2014). A Survey of Job Shop Scheduling Problems. *Journal of Scheduling*, 17(3), 297-308.

Winston, W. L. (2004). *Operations Research: Applications and Algorithms (4th ed.)*. Duxbury Press.

Yang, S., Zhang, Y., dan Liu, H. (2020). "A Novel Genetic Algorithm for Job Shop Scheduling Problem." *Soft Computing*, <http://cs.felk.cvut.cz/xobitko/ga/intro.html>

Zhang, Y., Wang, J., dan Chen, L. (2019). "Performance Analysis of Genetic Algorithms for Job Shop Scheduling." *Journal of Intelligent Manufacturing*, 30(5), 1987-2000.