

**OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN
ALGORITMA GENETIKA DAN ALGORITMA *CAMPBELL*
*DUDEK SMITH***

Skripsi

Oleh

**VALENS ALVARO SIDABALOK
NPM. 2217031147**



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

2026

ABSTRACT

OPTIMIZATION OF JOB SHOP SCHEDULING USING GENETIC ALGORITHM AND CAMPBELL DUDEK SMITH ALGORITHM

By

Valens Alvaro Sidabalok

Optimization of production scheduling is essential to improve the efficiency of manufacturing systems, particularly in minimizing makespan. This study aims to minimize the makespan in a Job Shop Scheduling Problem (7 jobs, 7 machines) at PT Acosys Global Data, Bandar Lampung, using the Genetic Algorithm and the Campbell Dudek Smith (CDS) Algorithm. The Genetic Algorithm was applied through population initialization, fitness evaluation, selection, crossover, and mutation, while the CDS Algorithm utilized the formation of dummy machines and Johnson's rule. The results showed that the Genetic Algorithm produced the best job sequence [6,4,5,3,1,2,7] with a makespan of 546 minutes, outperforming the CDS Algorithm, which yielded a makespan of 560 minutes with the best job sequence [6,7,2,4,5,1,3]. In conclusion, the Genetic Algorithm is more optimal than the CDS Algorithm in minimizing makespan in a job shop production system

Keywords: Job Shop Scheduling Problem, makespan, Genetic Algorithm, Campbell Dudek Smith

ABSTRAK

OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN ALGORITMA GENETIKA DAN ALGORITMA *CAMPBELL* *DUDEK SMITH*

Oleh

Valens Alvaro Sidabalok

Optimalisasi penjadwalan produksi sangat diperlukan untuk meningkatkan efisiensi sistem manufaktur terutama dalam meminimalkan *makespan*. penelitian ini bertujuan untuk meminimalkan *makespan* pada *Job Shop Scheduling Problem* (7 job, 7 mesin) di PT Acosys Global Data Bandar Lampung menggunakan Algoritma Genetika dan Algoritma Campbell Dudek Smith (CDS). Algoritma Genetika diterapkan melalui inisialisasi populasi, evaluasi *fitness*, seleksi, *crossover*, dan mutasi, sedangkan Algoritma CDS menggunakan pembentukan mesin semu dan aturan Johnson. Hasil penelitian menunjukkan Algoritma Genetika menghasilkan urutan *job* terbaik [6,4,5,3,1,2,7] dengan *makespan* 546 menit, lebih baik dibandingkan Algoritma CDS yang menghasilkan *makespan* 560 menit dengan urutan *job* terbaik [6,7,2,4,5,1,3]. Kesimpulannya, Algoritma Genetika lebih optimal daripada Algoritma CDS dalam meminimalkan *makespan* pada sistem produksi *job shop*.

Kata-kata kunci: JSSP, *makespan*, Algoritma Genetika, CDS

**OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN
ALGORITMA GENETIKA DAN ALGORITMA *CAMPBELL*
*DUDEK SMITH***

VALENS ALVARO SIDABALOK

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**

2026

Judul Skripsi : **OPTIMALISASI PENJADWALAN JOB
SHOP MENGGUNAKAN ALGORITMA
GENETIKA DAN ALGORITMA
CAMPBELL DUDEK SMITH**

Nama Mahasiswa : **Valens Alvaro Sidabalok**

Nomor Pokok Mahasiswa : **2217031147**

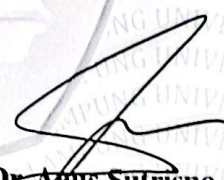
Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

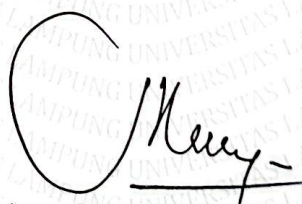
MENYETUJUI

1. Komisi Pembimbing


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


Dr. Agus Sutrisno, S.Si., M.Si.
NIP 197008311999031002

2. Ketua Jurusan Matematika
Matematika

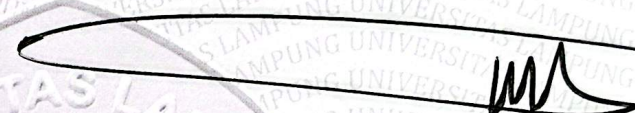

Dr. Aang Nuryaman, S.Si., M.Si.
NIP. 197403162005011001

MENGESAHKAN

1. Tim Penguji

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

Sekretaris : **Dr. Agus Sutrisno, S.Si., M.Si.** 

Penguji
Bukan Pembimbing : **Dr. Muslim Ansori, S.Si., M.Si.** 

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



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

Tanggal Lulus Ujian Skripsi: 19 Mei 2026

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

Nama : **Valens Alvaro Sidabalok**
Nomor Pokok Mahasiswa : **2217031147**
Jurusan : **Matematika**
Judul Skripsi : **OPTIMALISASI PENJADWALAN *JOB SHOP* MENGGUNAKAN ALGORITMA GENETIKA DAN ALGORITMA *CAMPBELL DUDEK SMITH***

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, 19 Mei 2026

Pennulis



Valens Alvaro Sidabalok

RIWAYAT HIDUP

Penulis memiliki nama lengkap Valens Alvaro Sidabalok yang lahir di Sidomulyo pada tanggal 11 Mei 2004. Penulis merupakan anak pertama dari tiga bersaudara, Putra dari pasangan Jondi Asi Sidabalok dan Eli Yusnita.

Penulis memulai pendidikan formal di TK Sejahtera pada tahun 2009 dan menyelesaikannya pada tahun 2010. Kemudian, penulis melanjutkan pendidikan di SDS Sejahtera 1 pada tahun 2010 sampai dengan tahun 2015. Pada tahun 2015, penulis pindah dan melanjutkan pendidikan di SDN 1 Sukamenanti hingga tahun 2016. Selanjutnya, penulis melanjutkan pendidikan di SMP Negeri 10 Bandar Lampung pada tahun 2016 sampai dengan tahun 2019. Setelah itu, penulis melanjutkan pendidikan di SMA YP Unila pada tahun 2019 dan menyelesaikannya pada tahun 2022.

Pada tahun 2022, penulis diterima di Program Studi S1 Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), Universitas Lampung melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Pada tahun 2025, penulis melaksanakan Kerja Praktik (KP) di PT TASPEN KC Bandar Lampung selama 40 hari, terhitung mulai tanggal 1 Juli sampai dengan 9 Agustus 2025. Selain itu, sebagai bentuk pengabdian kepada masyarakat, penulis juga melaksanakan Kuliah Kerja Nyata (KKN) pada bulan Januari hingga Februari 2025 di Desa Kota Guring, Kecamatan Rajabasa, Kabupaten Lampung Selatan.

Selama masa studi, penulis menunjukkan ketekunan dan dedikasi dalam menyelesaikan berbagai tugas akademik. Penulis berharap hasil dari penelitian ini dapat memberikan manfaat bagi pengembangan ilmu pengetahuan, khususnya di bidang Penjadwalan

KATA INSPIRASI

"Fortis solus stabit"

"What we know is a drop, what we don't know is an ocean."

(Isaac Newton)

"Tak ada yang tahu kapan kau mencapai tuju. Dan percayalah, bukan urusanmu untuk menjawab itu. Katakan pada dirimu, besok mungkin kita sampai, besok mungkin tercapai."

(Besok Mungkin Kita Sampai–Hindia)

"Terus berenang, lanjutlah mendaki."

(33x–Perunggu)

PERSEMBAHAN

Alhamdulillahirobbil'alamin

Puji syukur kehadiran Allah Subhanahu Wata'ala atas segala rahmat, karunia, dan ridha-Nya sehingga skripsi ini dapat diselesaikan dengan baik.

Shalawat serta salam senantiasa tercurah kepada junjungan kita Nabi Muhammad Shallallahu Alaihi Wasallam.

Dengan rasa syukur dan Bahagia, saya persembahkan rasa terimakasih saya kepada:

Ayah

Terima kasih kepada Alm. Ayah tercinta atas kasih sayang, doa, semangat, dan pengorbanan yang telah diberikan. Meski tidak dapat menyaksikan pencapaian ini, segala nasihat dan perjuangan Ayah akan selalu menjadi inspirasi dalam hidupku.

Bunda

Terima kasih kepada Bundaku tercinta yang senantiasa memberikan cinta, doa, dukungan, dan kekuatan dalam setiap langkah hidupku. Terima kasih atas segala pengorbanan, kesabaran, dan kasih sayang yang telah mengantarkanku hingga menyelesaikan pendidikan ini.

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 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 "Optimalisasi Penjadwalan *Job Shop* Menggunakan Algoritma Genetika dan Algoritma *Campbell Dudek Smith*" 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 sepanjang proses penyusunan skripsi ini.
2. Bapak Dr. Agus Sutrisno, S.Si., M.Si. selaku Pembimbing 2 yang telah memberikan arahan, dukungan, serta doa sehingga penulis dapat menyelesaikan skripsi ini.
3. Bapak Dr. Muslim Ansori, S.Si., M.Si. selaku Penguji yang telah bersedia memberikan saran, kritik, serta evaluasi yang membangun sehingga penulis dapat menyelesaikan skripsi ini.
4. Ibu Widiarti, S.Si., M.Si. selaku dosen Pembimbing Akademik.
5. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Seluruh dosen, staff dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

7. Seluruh guru TK, SD, SMP, dan SMA yang telah mengajarkan dan memberikan ilmu hingga penulis bisa menduduki bangku perkuliahan.
8. Alm. Ayah tercinta yang telah membesarkan, merawat, dan mendidik penulis. Terima kasih atas kasih sayang, nasihat, semangat, dan pelajaran hidup yang telah diberikan.
9. Bunda tercinta yang selalu memberikan kasih sayang, doa, dukungan, dan semangat kepada penulis. Terima kasih atas segala perjuangan dan pengorbanan yang telah diberikan hingga penulis dapat menyelesaikan pendidikan ini.
10. Adik-adik penulis, Tata dan Jovan, yang selalu memberikan doa, dukungan, dan semangat selama proses penyusunan skripsi ini.
11. Opung Doli dan Opung Boru yang telah memberikan kasih sayang, perhatian, serta dukungan kepada penulis selama ini.
12. Amangboru dan Namboru yang senantiasa memberikan doa, dukungan, dan motivasi kepada penulis.
13. Om, Tante, dan Cicik yang selalu memberikan doa dan dukungan sehingga penulis dapat menyelesaikan skripsi ini.
14. Jiya yang senantiasa menjadi tempat penulis berbagi cerita, keluh kesah, dan semangat selama menjalani perkuliahan.
15. Teman-teman SMA penulis, Deo, Ridho, Jordy, Hafizd, Rafha, dan Valleta, yang telah kebersamai penulis sejak masa sekolah hingga akhir perkuliahan. Terima kasih atas kebersamaan, dukungan, dan motivasi yang diberikan.
16. Teman-teman perkuliahan penulis, Aprial, Najiya, Cyntia, April, dan Zahra, yang telah menjadi teman seperjuangan selama masa perkuliahan. Terima kasih atas bantuan, dukungan, dan kebersamaan yang diberikan kepada penulis.
17. Tante Armi serta seluruh tim Restu Bumi Adventure yang telah memberikan pengalaman, pengetahuan, dan kebersamaan yang berharga bagi penulis.
18. Kak Verna, Kak Toba, dan Kak Veli yang senantiasa memberikan semangat, motivasi, dan dukungan kepada penulis.

19. Diri sendiri, atas segala usaha, ketekunan, dan perjuangan yang telah dilakukan untuk menyelesaikan skripsi ini tepat pada waktunya. Terima kasih telah bertahan di tengah tantangan dan terus maju meski dihadapkan pada berbagai rintangan.

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, 19 Mei 2026

Valens Alvaro Sidabalok

DAFTAR ISI

DAFTAR ISI	xiii
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Tujuan Penelitian	2
1.3 Manfaat Penelitian	2
II TINJAUAN PUSTAKA	4
2.1 Optimalisasi	4
2.2 Penjadwalan	4
2.3 Model Matematika Masalah Penjadwalan	5
2.4 Penjadwalan <i>Job Shop</i>	6
2.5 Algoritma Genetika	9
2.5.1 Inisialisasi Populasi Awal	10
2.5.2 Evaluasi <i>Fitness</i>	11
2.5.3 Seleksi	12
2.5.4 <i>Crossover</i>	12
2.5.5 Mutasi	13
2.5.6 Iterasi Baru	14
2.6 Algoritma Johnson	14
2.7 Algoritma <i>Campbell Dudek Smith</i>	14
III METODE PENELITIAN	16
3.1 Waktu dan Tempat Penelitian	16
3.2 Metode Penelitian	16
3.3 Tahapan Penelitian	17
IV HASIL DAN PEMBAHASAN	18
4.1 Data Penelitian JSSP	18
4.2 Model Matematika JSSP	19
4.2.1 Variabel Keputusan	19

4.2.2	Fungsi Objektif	20
4.2.3	Kendala-Kendala dengan Contoh Perhitungan	20
4.3	Perhitungan <i>Makespan</i>	21
4.4	Penyelesaian JSSP Menggunakan Algoritma Genetika	23
4.4.1	Perhitungan Manual Genetika hingga Generasi Ke-5	23
4.4.2	Tahapan Implementasi Algoritma Genetika Menggunakan <i>Python</i>	29
4.4.3	Perbandingan Hasil Perhitungan Manual dan Perhitungan Algoritma Genetika pada <i>Python</i>	35
4.5	Perhitungan Menggunakan Algoritma CDS	36
4.5.1	Perhitungan Manual Algoritma CDS	36
4.5.2	Tahapan Implementasi Algoritma <i>Campbell Dudek Smith</i> (CDS) Menggunakan <i>Python</i>	45
4.5.3	Perbandingan Hasil Perhitungan Manual dan Perhitungan Algoritma CDS pada <i>Python</i>	49
4.6	Pembahasan dan Perbandingan Hasil	49
V	Kesimpulan dan Saran	51
5.1	Kesimpulan	51
5.2	Saran	51
	DAFTAR PUSTAKA	53

DAFTAR GAMBAR

2.1	Ilustrasi dari Kromosom, Individu, Populasi	11
4.1	Import <i>library</i> yang digunakan dalam perhitungan <i>makespan</i>	21
4.2	Input data waktu proses <i>job</i> pada setiap mesin	21
4.3	Penentuan urutan <i>job</i> yang akan dihitung <i>makespan</i> -nya	22
4.4	Proses perhitungan <i>makespan</i> menggunakan program <i>Python</i>	22
4.5	Proses menjalankan program perhitungan <i>makespan</i>	22
4.6	Hasil output perhitungan <i>makespan</i>	23
4.7	Gantt Chart Perhitungan Manual GA Hingga 5 Generasi	28
4.8	Import <i>library</i> yang digunakan dalam algoritma genetika	29
4.9	Penentuan parameter algoritma genetika	30
4.10	Input data waktu proses <i>job</i> pada setiap mesin	30
4.11	Perhitungan nilai <i>makespan</i>	30
4.12	Inisialisasi populasi awal	31
4.13	Evaluasi nilai <i>fitness</i> (<i>makespan</i>)	31
4.14	Pengurutan individu berdasarkan nilai <i>makespan</i>	31
4.15	Output hasil setiap generasi	31
4.16	Penentuan individu terbaik berdasarkan <i>makespan</i>	32
4.17	Proses seleksi individu	32
4.18	Proses <i>crossover</i>	32
4.19	Proses mutasi	32
4.20	Pembentukan populasi baru	32
4.21	<i>Gantt Chart</i> Urutan Terbaik Dari 100 Generasi Menggunakan GA .	34
4.22	<i>Gantt Chart</i> Perhitungan Algoritma CDS	45
4.23	Import <i>library</i> yang digunakan pada algoritma CDS	45
4.24	Input data waktu proses <i>job</i> pada setiap mesin	45
4.25	Penentuan jumlah <i>job</i> dan mesin	46
4.26	Implementasi <i>Johnson Rule</i>	46
4.27	Perhitungan nilai <i>makespan</i>	47
4.28	Inisialisasi solusi terbaik	47
4.29	Pembentukan mesin semu	47

4.30	Penentuan urutan <i>job</i>	47
4.31	Perhitungan <i>makespan</i> tiap iterasi	48
4.32	Output hasil setiap iterasi CDS	48
4.33	Penentuan solusi terbaik	48
4.34	Hasil akhir algoritma CDS	48

DAFTAR TABEL

2.1	Contoh waktu proses penjadwalan job shop	8
2.2	Perbandingan nilai makespan	9
2.3	Kromosom A dan Hasil Mutasi	13
4.1	Durasi Waktu Proses Setiap Job pada Setiap Mesin (dalam menit) .	19
4.2	Populasi Generasi 0	23
4.3	Populasi Generasi 1	24
4.4	Populasi Generasi 2	25
4.5	Populasi Generasi 3	26
4.6	Populasi Generasi 4	26
4.7	Evaluasi <i>Fitness</i> Generasi 5	27
4.8	Waktu Mulai dan Selesai Setiap <i>Job</i> pada Masing-Masing Mesin (Generasi ke-5 Algoritma Genetika)	28
4.9	Perkembangan Nilai <i>Makespan</i> Tiap Generasi	33
4.10	Hasil Pemrograman Setiap Generasi	34
4.11	Waktu Mulai dan Selesai Setiap <i>Job</i> pada Masing-Masing Mesin (Urutan <i>Job</i> 6–4–5–3–1–2–7)	35
4.12	Hasil Akumulasi Waktu Mesin Semu Iterasi 1	36
4.13	Hasil Akumulasi Waktu Mesin Semu Iterasi 2	38
4.14	Hasil Akumulasi Waktu Mesin Semu Iterasi 3	39
4.15	Hasil Akumulasi Waktu Mesin Semu Iterasi 4	40
4.16	Hasil Akumulasi Waktu Mesin Semu Iterasi 5	41
4.17	Hasil Akumulasi Waktu Mesin Semu Iterasi 6	43
4.18	Perbandingan Urutan <i>Job</i> dan Nilai <i>Makespan</i>	44
4.19	Waktu Mulai dan Selesai Setiap <i>Job</i> pada Masing-Masing Mesin (Urutan <i>Job</i> 6–7–2–4–5–1–3)	44
4.20	Hasil Output Program	49
4.21	Perbandingan Hasil Terbaik Algoritma Genetika dan Algoritma Campbell Dudek Smith	49

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Penjadwalan merupakan salah satu fungsi kritis dalam sistem manufaktur yang secara langsung mempengaruhi efisiensi, biaya produksi, dan daya saing perusahaan. Dalam lingkungan produksi *job shop*, di mana setiap pekerjaan memiliki rute proses yang unik dan harus melalui berbagai mesin, tantangan penjadwalan menjadi semakin kompleks. Menurut Sempena, (2009) *Job Shop Scheduling Problem* (JSSP) termasuk dalam kategori masalah *NP-hard*, yang berarti tidak ada algoritma eksak yang dapat menyelesaikannya dalam waktu *polynomial* untuk skala besar. Keterbatasan ini mendorong para peneliti untuk mengembangkan berbagai pendekatan metaheuristik yang mampu memberikan solusi *near-optimal* dalam waktu komputasi yang *reasonable*.

Di antara berbagai algoritma metaheuristik, Algoritma Genetika (GA) telah terbukti efektif dalam menangani JSSP. GA terinspirasi dari proses evolusi biologis yang menggunakan mekanisme seleksi, *crossover*, dan mutasi untuk mengeksplorasi ruang pencarian secara global (Abdolrazzagh-Nezhad dan Abdullah, 2024). Keunggulan utama algoritma genetika adalah kemampuannya menjaga keberagaman solusi sehingga tidak mudah terjebak pada solusi lokal.

Beberapa penelitian menunjukkan bahwa metode heuristik seperti *Campbell Dudek Smith* (CDS) dapat membantu menyusun urutan produksi yang lebih efisien. Ayu dkk., (2019) menemukan bahwa penerapan CDS pada sistem *job shop* mampu menurunkan makespan dan menyeimbangkan beban kerja mesin. Purwati dan Santika, (2021) juga menunjukkan bahwa CDS dapat mempercepat penyelesaian produksi pada perusahaan manufaktur dengan menghasilkan beberapa alternatif jadwal dan memilih yang paling optimal. Sementara itu, Maghfiroh dkk., (2013) menyatakan bahwa GA mampu mengeksplorasi ruang solusi yang luas dan

menghasilkan jadwal dengan makespan yang lebih kecil dibandingkan aturan prioritas sederhana.

Tinjauan penelitian terdahulu menunjukkan bahwa CDS efektif dalam menghasilkan urutan kerja yang efisien, sedangkan GA unggul dalam optimasi solusi. Namun, penggunaan kedua metode tersebut pada industri konveksi, khususnya konveksi baju di Bandar Lampung yang masih sangat terbatas. Dengan menggunakan CDS dan GA sebagai alat optimalisasi penjadwalan, diharapkan proses penjadwalan dapat dilakukan lebih sistematis dan mampu mengurangi waktu tunggu maupun tumpukan pekerjaan pada mesin tertentu.

Berdasarkan kondisi tersebut, penelitian ini bertujuan untuk mengoptimalkan penjadwalan produksi barang dari studi kasus Pabrik Konveksi PT Acosys Global Data Bandar Lampung dengan memanfaatkan metode *Campbell Dudek Smith* dan Algoritma Genetika. Penelitian ini diharapkan memberikan solusi penjadwalan yang lebih efisien, meminimalkan makespan, dan meningkatkan produktivitas konveksi. Selain memberikan manfaat praktis bagi pelaku usaha, penelitian ini juga memiliki signifikansi akademik karena berkontribusi pada pengembangan metode optimasi penjadwalan.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini yaitu:

1. Menentukan *makespan* minimum pada penjadwalan *job shop* menggunakan Algoritma Genetika dan Algoritma *Campbell Dudek Smith*.
2. Membandingkan hasil yang terbaik antara Algoritma Genetika dan Algoritma *Campbell Dudek Smith*.

1.3 Manfaat Penelitian

Manfaat yang ingin diperoleh dari penelitian ini yaitu:

1. Mengoptimalkan waktu dalam penjadwalan *job shop* dengan menggunakan Algoritma Genetika dan Algoritma *Campbell Dudek Smith* guna 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 dalam ilmu matematika dan komputasi merujuk pada proses pencarian solusi terbaik dari suatu permasalahan berdasarkan kriteria atau standar tertentu yang telah ditetapkan. Istilah “optimal” berarti yang terbaik atau tertinggi, sedangkan “mengoptimalkan” dimaknai sebagai upaya mencapai kualitas terbaik atau tertinggi (Luthfiyah dkk., 2025).. Konsep ini telah banyak diterapkan dalam berbagai bidang, antara lain teknik industri, ekonomi, manajemen operasi, dan ilmu komputer, karena perannya yang krusial dalam pengambilan keputusan dan pemanfaatan sumber daya secara efisien. Oleh karena itu, penelitian ini membahas konsep dasar optimalisasi beserta klasifikasinya, serta metode-metode optimalisasi yang relevan dengan permasalahan penjadwalan job shop.

Secara umum, optimalisasi dapat dipahami sebagai upaya sistematis untuk mencapai hasil yang paling diharapkan dengan cara yang efisien dan ekonomis. Dalam konteks komersial maupun operasional, optimalisasi bertujuan memaksimalkan tindakan agar menghasilkan konsekuensi yang paling menguntungkan. Proses optimalisasi melibatkan penentuan tujuan yang jelas, identifikasi dan penghindaran hambatan, pemilihan metode pemecahan masalah yang tepat dan andal, serta pengambilan keputusan yang efektif. Dengan demikian, optimalisasi dapat didefinisikan sebagai serangkaian prosedur atau metode yang digunakan untuk memanfaatkan sumber daya secara optimal guna memperoleh hasil yang ideal (Al Fikri dkk., 2024).

2.2 Penjadwalan

Penjadwalan merupakan proses pengaturan urutan dan waktu pelaksanaan suatu rangkaian kegiatan dengan tujuan memastikan pemanfaatan sumber daya

berlangsung secara efektif dan efisien. Dalam konteks sistem produksi, penjadwalan berfungsi untuk mengalokasikan pekerjaan ke mesin atau fasilitas dalam waktu tertentu sehingga alur proses menjadi optimal. Menurut Mashuri dkk., (2020), penjadwalan sangat penting untuk meminimalkan waktu penyelesaian (*makespan*), mengurangi waktu menganggur (*idle*), serta menghindari keterlambatan produksi, terutama pada sistem manufaktur yang terdiri atas banyak mesin dan job yang kompleks.

Arifudin, (2009) menyebutkan bahwa penjadwalan sering mengalami kesulitan jika diselesaikan secara matematis secara langsung, khususnya ketika faktor sumber daya seperti biaya harus diseimbangkan sepanjang durasi kegiatan. Oleh karena itu, pendekatan berbasis optimasi termasuk heuristik maupun metaheuristik seperti Algoritma Genetika dan Algoritma *Campbell Dudek Smith* digunakan untuk menghasilkan jadwal yang lebih stabil, merata, dan efisien. Penjadwalan yang dioptimalkan tidak hanya memastikan urutan aktivitas berjalan dengan benar, tetapi juga membantu penggunaan sumber daya menjadi lebih merata serta mengurangi pemborosan biaya dan waktu, sehingga hasilnya lebih efektif dan lebih mudah diterapkan dalam kondisi nyata.

2.3 Model Matematika Masalah Penjadwalan

Variabel Keputusan

- S_{ij} : Menyatakan waktu mulai (*start time*) job ke- i pada mesin ke- j .
- C_{ij} : Menyatakan waktu selesai (*completion time*) job ke- i pada mesin ke- j .
- P_{ij} : Waktu pemrosesan operasi - ij pada Mesin ke - j .

Fungsi Objektif

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

$$\min C_{\max} \quad (2.3.1)$$

Dengan

$$C_{\max} = \max(C_{ij}), \quad \forall (i, j) \quad (2.3.2)$$

Kendala-Kendala

1. Urutan Operasi dalam *Job*:

Operasi pada *job* harus dikerjakan secara berurutan:

$$S_{i,j+1} \geq C_{ij}, \quad \forall i = 1, 2, \dots, n, \quad \forall j = 1, 2, \dots, n-1 \quad (2.3.3)$$

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

2. Satu mesin tidak bisa memproses dua *job* bersamaan:

Setiap mesin hanya dapat menjalankan satu operasi dalam satu waktu:

$$S_{ij} \geq C_{\ell j} \text{ atau } S_{\ell j} \geq C_{ij}, \quad \forall (i, j) \neq (\ell, j) \quad (2.3.4)$$

jika operasi dijalankan pada mesin yang sama.

3. Waktu penyelesaian operasi:

$$C_{ij} = S_{ij} + P_{ij}, \quad \forall (i, j) \quad (2.3.5)$$

di mana P_{ij} adalah durasi operasi ij

4. Makespan:

Makespan harus lebih besar atau sama dengan seluruh waktu selesai pekerjaan.

$$C_{\max} \geq C_{ij} \quad (2.3.6)$$

2.4 Penjadwalan *Job Shop*

Penjadwalan *job shop* didefinisikan sebagai permasalahan di mana sekumpulan n *job* (pekerjaan) harus diproses pada sekumpulan m mesin yang berbeda setiap *job* terdiri dari serangkaian operasi, dan setiap operasi ditetapkan untuk dilakukan pada mesin tertentu selama waktu pemrosesan yang telah ditentukan. Dalam model JSSP, setiap mesin hanya dapat menangani satu operasi pada satu waktu, dan operasi-operasi dari pekerjaan yang sama tidak boleh diproses secara bersamaan di mesin berbeda.

Masalah utama adalah menemukan urutan (*sequence*) operasi untuk tiap mesin sehingga memenuhi urutan operasi tiap *job* dan tujuan optimisasi (misalnya meminimalkan total penyelesaian *makespan*). Dengan demikian, penjadwalan job shop mencerminkan problem optimasi kombinatorial kompleks, di mana solusi harus menjaga *constraint* mesin tunggal sekaligus meminimalkan waktu total.

Menurut Bedworth pada jurnal Ramdani dan Erfina, (2023) tujuan dari penjadwalan yaitu:

- Meningkatkan penggunaan sumber daya atau mengurangi waktu tunggu, sehingga total waktu proses dapat berkurang dan produktivitas dapat meningkat.
- Mengurangi sejumlah pekerjaan menunggu dalam antrian ketika sumber daya yang ada masih mengerjakan tugas yang lain.
- Mengurangi beberapa kelambatan pada pekerjaan yang mempunyai batas waktu penyelesaian sehingga akan meminimalisasi *penalty cost* (biaya kelambatan).

Berdasarkan ketiga tujuan tersebut, kinerja penjadwalan dinilai menggunakan beberapa kriteria penjadwalan, antara lain:

- *Minimum makespan* (keseluruhan waktu yang digunakan dalam proses produksi).
- *Minimum mean flow time* (rata-rata waktu proses)
- *Minimum tardiness* (keterlambatan).

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.

Sebagai ilustrasi sederhana, misalkan tiga produk A, B, C diproses pada dua mesin M_1 dan M_2 dengan waktu proses sebagai berikut.

Tabel 2.1 Contoh waktu proses penjadwalan job shop

Job	M_1	M_2
A	4	6
B	5	2
C	3	4

Dalam permasalahan penjadwalan job shop, urutan pengerjaan job merupakan faktor penting yang memengaruhi nilai makespan. Oleh karena itu, pada contoh berikut ditunjukkan dua urutan job yang berbeda, yaitu $A \rightarrow B \rightarrow C$ dan $B \rightarrow C \rightarrow A$, untuk memperlihatkan perbedaan hasil penjadwalan.

Berdasarkan urutan $A \rightarrow B \rightarrow C$, penjadwalan pada masing-masing mesin adalah sebagai berikut.

- Mesin M1:
 - Job A: 0 – 4
 - Job B: 4 – 9
 - Job C: 9 – 12
- Mesin M2:
 - Job A: 4 – 10
 - Job B: 10 – 12
 - Job C: 12 – 16

Job terakhir selesai pada waktu 16, sehingga nilai makespan untuk urutan ini adalah

$$C_{\max}^{ABC} = 16.$$

Untuk urutan $B \rightarrow C \rightarrow A$, penjadwalan diperoleh sebagai berikut.

- Mesin M1:
 - Job B: 0 – 5
 - Job C: 5 – 8
 - Job A: 8 – 12
- Mesin M2:
 - Job B: 5 – 7
 - Job C: 8 – 12
 - Job A: 12 – 18

Job terakhir selesai pada waktu 18, sehingga nilai makespan untuk urutan ini adalah

$$C_{\max}^{BCA} = 18.$$

Perbandingan nilai makespan dari kedua urutan tersebut ditunjukkan pada Tabel 2.2.

Tabel 2.2 Perbandingan nilai makespan

Urutan Job	Makespan
$A \rightarrow B \rightarrow C$	16
$B \rightarrow C \rightarrow A$	18

2.5 Algoritma Genetika

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).

Salah satu pendekatan yang cukup efektif untuk menyelesaikan permasalahan

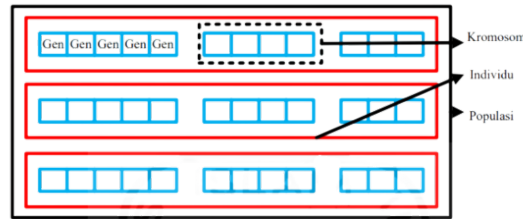
NP-Hard adalah menggunakan algoritma metaheuristik. Meskipun algoritma ini tidak dapat menjamin solusi yang paling optimal, namun ia mampu menangani permasalahan yang kompleks dan berskala besar secara efisien, dengan mencari solusi yang cukup baik dan memenuhi berbagai kriteria dalam waktu komputasi yang relatif singkat. Salah satu jenis algoritma heuristik yang cukup dikenal adalah *Genetic neighborhood search methods*, yang mencakup tiga teknik utama, yaitu: *Genetic Algorithm*, *Simulated Annealing*, dan *Tabu Search*.

Algoritma Genetika (*Genetic Algorithm* (GA)) adalah metode optimasi berbasis heuristik 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). 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. Dalam penerapannya Algoritma Genetika melibatkan 7 langkah utama (Radhamani dan Baburaj, 2013):

1. Inisialisasi populasi awal.
2. Evaluasi *fitness*.
3. Seleksi.
4. *Crossover*.
5. Mutasi.
6. Iterasi baru.
7. Berhenti.

2.5.1 Inisialisasi Populasi Awal

Menurut Alfandianto dkk., (2017), inisialisasi populasi awal ialah suatu pendekatan untuk memperoleh kromosom-kromosom awal. Jumlah individu pada populasi awal merupakan masukan dari pengguna. Setelah total individu pada populasi awal didapatkan, dibangkitkan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Pada penelitian ini inisialisasi populasi dibangkitkan secara random.



Gambar 2.1 Ilustrasi dari Kromosom, Individu, Populasi

Pada tahap inisialisasi populasi awal, setiap individu terdiri atas sekumpulan kromosom, sedangkan setiap kromosom merupakan rangkaian gen. Populasi merupakan kumpulan beberapa individu, dan setiap individu merepresentasikan kombinasi permutasi penjadwalan pekerjaan yang dijalankan oleh mesin, mulai 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).

2.5.2 Evaluasi *Fitness*

Evaluasi *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, yakni yang diperlukan untuk menyelesaikan semua pekerjaan yang ada di *shop* (manufaktur) 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 :

$$Makespan = \max\{C_1, C_2, \dots, C_n\} \quad (2.5.7)$$

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.

2.5.3 Seleksi

Setiap kromosom yang terdapat dalam populasi akan melalui proses seleksi untuk dipilih menjadi orang tua. Sesuai dengan teori Evolusi Darwin maka kromosom yang baik akan bertahan dan menghasilkan keturunan yang baru untuk generasi selanjutnya. Pilih 2 buah individu dengan nilai *fitness* terkecil untuk diproses pada tahap selanjutnya yaitu tahap *crossover* dan mutasi. Kedua buah individu yang telah dipilih untuk selanjutnya disebut orang tua.

2.5.4 Crossover

Dalam algoritma genetika, "*crossover*" adalah menyilangkan dua kromosom sehingga membentuk kromosom baru yang harapannya lebih baik dari pada induknya (Lukas dkk., 2005). 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 orang tua, tetapi tidak berarti generasi yang baru akan sama persis dengan generasi yang lama. *crossover* memiliki beberapa metode yang umum digunakan seperti *one point crossover*, *two point crossover*, *order crossover* (OX), *partially mapped crossover* (PMX).

Teknik *crossover* yang digunakan pada penelitian ini adalah *order crossover*. *Order crossover* (OX) diperkenalkan oleh Davis. Teknik ini diawali dengan membangkitkan dua bilangan acak. Gen yang berada di antara kedua bilangan acak tersebut kemudian disalin ke *offspring* dengan posisi yang sama. Langkah berikutnya untuk mendapatkan *offspring* pertama adalah mengurutkan gen dari *parent* kedua dengan urutan: gen setelah bilangan acak kedua, diikuti gen sebelum bilangan acak pertama, dan diakhiri gen di antara kedua bilangan acak. Urutan gen yang telah disusun tersebut kemudian dibandingkan dengan *offspring* pertama. Jika gen tersebut sudah terdapat dalam *offspring* pertama, maka gen diabaikan dari urutan tadi. Selanjutnya, urutan gen yang baru dimasukkan ke dalam *offspring* dengan menempatkan gen setelah bilangan acak kedua terlebih dahulu, dan sisa gen dimasukkan ke posisi sebelum bilangan acak pertama. Proses serupa juga diterapkan untuk menghasilkan *offspring* kedua (Lukas dkk., 2005).

Berikut contoh implementasi order crossover:

- *Parent1* : (1, 2, |3, 4, 5, 6|, 7, 8)
- *Parent2* : (2, 3, |1, 6, 5, 4|, 8, 7)

Salin bagian yang dipilih

- *Offspring1* : ($x, x, |3, 4, 5, 6|, x, x$)
- *Offspring2* : ($x, x, |1, 6, 5, 4|, x, x$)

Pada urutan *parent2* : (2, 3, 1, 6, 5, 4, 8, 7) hapus (3, 4, 5, 6) pada urutan dan letakan pada *offspring1*

- *Offspring1* : ($x, x, |3, 4, 5, 6|, x, x$) = (2, 1, |3, 4, 5, 6|, 8, 7) lakukan hal yang sama pada *offspring 2*
- *Offspring2* : (2, 3, |1, 6, 5, 4|, 7, 8)

2.5.5 Mutasi

Pada tahap ini menurut Radhamani dan Baburaj, (2013) Operasi mutasi dilakukan untuk menghasilkan keturunan baru yang bertujuan meningkatkan keragaman dalam populasi solusi pada masalah *Job Shop Scheduling*. Mutasi diterapkan menggunakan operasi pergantian. Secara umum, menghindari hilangnya informasi penting dari individu selama proses evolusi. Proses mutasi dilakukan dengan memilih dua pekerjaan secara acak dalam satu individu, kemudian menukarkan posisinya pada mesin atau prosesor yang dipilih secara acak. Apabila mutasi tidak terjadi, maka individu hasil *crossover* akan tetap sama dengan induknya. Sebaliknya, jika mutasi terjadi, maka susunan gen pada individu akan berubah sehingga dihasilkan solusi penjadwalan baru yang berbeda.

Proses mutasi ditampilkan pada tabel

Tabel 2.3 Kromosom A dan Hasil Mutasi

A	Job7	Job6	Job4	Job5	Job3	Job2	Job1
Hasil	Job7	Job6	Job1	Job5	Job3	Job2	Job4

2.5.6 Iterasi Baru

Proses pemilihan, *crossover*, mutasi, *evaluasi fitness*, dan penempatan kembali 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 batasan akhir proses seleksi, persilangan, dan mutasi. Jumlah generasi yang besar berarti semakin banyak iterasi yang dilakukan dan semakin luas domain solusi yang dieksplorasi, namun Algoritma Genetika akan berjalan lebih lambat.

2.6 Algoritma Johnson

Algoritma Johnson pertama kali diperkenalkan oleh S. M. Johnson pada tahun 1954 melalui makalah berjudul “*Optimal Two- and Three-Stage Production Schedules with Setup Times Included*”, di mana Johnson memformulasikan solusi optimal untuk masalah penjadwalan pada sistem *flow-shop* dua mesin (setiap *job* harus diproses terlebih dahulu di mesin pertama lalu mesin kedua) (Johnson, 1954). Prinsip dasar algoritma ini adalah memilih *job* dengan waktu proses terkecil pada mesin pertama atau mesin kedua, kemudian menempatkannya di posisi awal atau akhir urutan sesuai mesin mana yang memiliki waktu minimum aturan sederhana namun efektif untuk meminimalkan total waktu penyelesaian (*makespan*) (Brucker, 2007).

2.7 Algoritma Campbell Dudek Smith

Metode *Campbell Dudek and Smith* atau biasa dikenal dengan algoritma CDS pertama kali ditemukan pada tahun 1965. Algoritma CDS merupakan *continuous improvement* dari algoritma Johnson yang digunakan untuk melakukan perbaikan penjadwalan *job shop multi stage*. Pendekatan heuristik CDS bertujuan untuk memecahkan permasalahan jumlah pekerjaan (*n-job*) dan jenis mesin tiap pengerjaan yang disusun seri dibagi (*m-mesin*) ke dalam beberapa grup dengan menentukan urutan waktu prioritas terpendek dalam mengkombinasikan work center (Makarim dkk., 2023). Langkah menentukan metode CDS Pada model produksi *job shop* untuk meminimasi makespan waktu terkecil sebagai urutan *job* penjadwalan yang optimal adalah sebagai berikut:

1. Mulai urutan pekerjaan dengan menentukan iterasi $k = m - 1$ dan jumlah *work center* m (jumlah mesin).
2. Untuk seluruh job, gunakan Algoritma Johnson untuk menentukan nilai $P_{i,1}^k$ $P_{i,2}^k$ yang terkecil (waktu proses terpendek pada mesin pertama dan kedua semu), di mana:

$$P_{i,1}^k = \sum_{j=1}^k P_{i,j} \quad (2.7.8)$$

$$P_{i,2}^k = \sum_{j=m-k+1}^m P_{i,j} \quad (2.7.9)$$

3. Jika waktu terpendek berada pada mesin semu pertama ($P_{i,1}^k$), jadwalkan job tersebut pada urutan awal; jika pada mesin semu kedua ($P_{i,2}^k$), jadwalkan pada urutan akhir.
4. Kembalikan job yang telah dijadwalkan ke daftar job. Jika masih ada job yang belum dijadwalkan, ulangi langkah 2; jika semua job telah dijadwalkan, urutan untuk iterasi k selesai.
5. Ulangi langkah 1–4 untuk $k = 2, 3, \dots, m - 1$ guna menghasilkan beberapa alternatif urutan.
6. Pilih urutan dengan makespan terkecil sebagai solusi optimal.

Keterangan:

$P_{i,1}^k$	= Waktu proses tiap <i>job</i> ke- i pada mesin pertama
$P_{i,2}^k$	= Waktu proses tiap <i>job</i> ke- i pada mesin kedua
k	= $1, 2, 3, \dots, (m - 1)$
m	= jenis mesin yang digunakan pada <i>work center</i>
i	= Jenis urutan proses
j	= Pekerjaan atau <i>job</i>

BAB III

METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilaksanakan pada semester genap tahun ajaran 2025/2026 di Jurusan 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 Metode Penelitian

Penelitian ini menggunakan metode pengumpulan data melalui wawancara dengan narasumber serta didukung oleh studi literatur. Wawancara yang dilakukan bersama narasumber bertujuan untuk mendapatkan data terkait Pabrik Konveksi PT Acosys Global Data Bandar Lampung. Penyelesaian masalah dilakukan menggunakan

Algoritma Genetika dan Algoritma Campbell Dudek Smith (CDS). Pada Algoritma Genetika, proses optimasi dilakukan melalui tahapan inisialisasi populasi, evaluasi fitness berdasarkan nilai makespan, seleksi, crossover, mutasi, dan pembentukan generasi baru hingga diperoleh solusi terbaik. Sementara itu, algoritma CDS digunakan dengan membentuk mesin semu pada setiap iterasi dan menentukan urutan job menggunakan aturan Johnson untuk memperoleh nilai makespan minimum. Hasil dari kedua algoritma kemudian dibandingkan untuk mengetahui metode yang menghasilkan solusi penjadwalan paling optimal berdasarkan nilai makespan terkecil.

3.3 Tahapan Penelitian

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

1. Menentukan dengan jelas masalah penjadwalan *Job Shop* yang akan diteliti.
2. Mengumpulkan data penjadwalan produksi barang dari studi kasus Pabrik Konveksi PT Acosys Global Data Bandar Lampung.
3. Menerapkan Algoritma Genetika dan Algoritma *Campbell Dudek Smith* untuk memecahkan masalah penjadwalan *Job Shop*.
4. Menentukan solusi yang dihasilkan oleh masing-masing algoritma.
5. Menganalisis dan membandingkan solusi dari Algoritma Genetika dan Algoritma *Campbell Dudek Smith*.
6. Menyimpulkan hasil analisis dan perbandingan.

BAB V

Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan dapat diambil kesimpulan sebagai berikut:

1. Penerapan Algoritma *Campbell Dudek Smith* CDS pada permasalahan penjadwalan *flow shop* menghasilkan beberapa alternatif urutan *job* melalui pembentukan mesin semu dan penerapan aturan *Johnson*. Berdasarkan perhitungan manual yang dilakukan, diperoleh urutan *job* [6,7,2,4,5,1,3] menghasilkan nilai *makespan* terbaik sebesar 560 menit.
2. Algoritma Genetika yang diterapkan melalui tahapan inialisasi populasi, evaluasi *fitness*, seleksi, *crossover*, dan *mutasi* mampu menghasilkan solusi penjadwalan yang lebih optimal. Hasil perhitungan menggunakan pemrograman *Python* hingga generasi ke-100 dengan urutan *job* [6,4,5,3,1,2,7] menghasilkan nilai *makespan* terbaik sebesar 546 menit.
3. Implementasi Algoritma CDS dan Algoritma Genetika menggunakan bahasa pemrograman *Python* menghasilkan nilai *makespan* yang konsisten dengan hasil perhitungan manual, sehingga dapat disimpulkan bahwa model dan algoritma yang digunakan telah diimplementasikan dengan benar.
4. Berdasarkan hasil perbandingan, Algoritma Genetika menghasilkan nilai *makespan* yang lebih kecil dibandingkan Algoritma CDS, sehingga lebih efektif dalam meminimalkan waktu *job shop* pada kasus yang diteliti.

5.2 Saran

Berdasarkan hasil penelitian ini, disarankan agar metode penjadwalan yang dikembangkan dapat diimplementasikan sebagai bahan evaluasi dalam penyusunan

jadwal produksi di perusahaan. Penelitian selanjutnya diharapkan dapat mengembangkan model penjadwalan dengan mempertimbangkan kriteria kinerja lain selain makespan, serta menguji metode optimasi pada skala permasalahan yang lebih besar dan kondisi produksi yang dinamis.

DAFTAR PUSTAKA

- Abdolrazzagh-Nezhad, M., & Abdullah, S. (2024). A Review on Metaheuristic Approaches for Job-Shop Scheduling Problems. *DATA SCIENCE: Journal of Computing and Applied Informatics (JoCAI)*, 8(1), 45–63.
- Alfandianto, A., Nugroho, Y. A., & Setiafindari, W. (2017). Penjadwalan Produksi Menggunakan Pendekatan Algoritma Genetika di PT Pertani (Persero) Cabang D.I. Yogyakarta. *Jurnal DISPROTEK*, 8(2), 1–7.
- Al Fikri, M. F., Irawan, J. D., & Zahro', H. Z. (2024). Penerapan Algoritma Genetika Pada Penjadwalan Produksi Di PT. Cahaya Metal Perkasa Untuk Meminimalkan makespan. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(2), 1361–1367.
- Apriana, I. W. R., Tastrawati, N. K. T., & Sari, K. (2016). Implementasi Algoritma Cat Swarm Optimization dalam Menyelesaikan Job Shop Scheduling Problem (JSSP). *E-Jurnal Matematika*, 5(3), 90–97.
- Arifudin, R. (2009). Optimasi Penjadwalan Proyek Dengan Penyeimbangan Biaya Menggunakan Kombinasi CPM Dan Algoritma Genetika. *Jurnal Matematika*, 12(1), 1–9.
- Ayu, T. Z., Yugowati, A., & Choirul, A. (2019). Optimalisasi Penjadwalan Mesin Produksi Flowshop Menggunakan Metode CDS. *CDMA: Jurnal Pengabdian Masyarakat*, 2(2), 26–33.
- Brucker, P. (2007). *Scheduling Algorithms* (5th ed.). Springer.
- Hartono, M. (2001.). Penjadualan Produksi n job m mesin Untuk Sistem Produksi Job Shop di Perusahaan & Industri Mesin Aneka Mesin Malang. 10-18.
- Johnson, S. M. (1954). Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1(1), 61–68.
- 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 (SNATI 2005)*, 1–6.
- Luthfiyah, A., Azzahra, N., Alghifari, A., & Kusumaningrum, H. (2025). Optimalisasi Sarana dan Prasarana untuk Mendukung Proses Pembelajaran. *Harmoni Pendidikan: Jurnal Ilmu Pendidikan*, 2(1), 90–103.
- Maghfiroh, M., Darmawan, D., & Yu, V. F. (2013). Genetic Algorithm for Job Shop Scheduling Problem: A Case Study. *International Journal of Industrial and Manufacturing Engineering*, 7(4), 258–265.
- Makarim, R. A., Maksum, A. H., & Rachmat, M. T. (2023). Optimasi Penyusunan Penjadwalan Menggunakan Pendekatan Heuristik Algoritma Campbell Dudek Smith (CDS) di PT OSIN. *JURNAL REKAYASA SISTEM DAN INDUSTRI*, 10(1), 25–31.
- Mashuri, C., Mujiyanto, A. H., Sucipto, H., & Arsamba, R. Y. (2020). Penerapan Algoritma Campbell Dudek Smith (CDS) untuk Optimasi Waktu Produksi Pada Penjadwalan Produksi. *Jurnal Sistem Informasi Bisnis (JSINBIS)*, 10(2), 131–136.
- 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.
- Purwati, S., & Santika, A. (2021). Analisis Penjadwalan Produksi dengan Metode CDS. *OPSI: Jurnal Optimasi Sistem Industri*, 14(1), 18–25.
- Radhamani, A. S., & Baburaj, E. (2013). Performance Efficient Heterogeneous Multi Core Scheduling Strategy Based on Genetic Algorithm. *ARPJN Journal of Engineering and Applied Sciences*, 8(1), 27–34.
- Ramdani, M., & Erfina, A. (2023). Aplikasi Penjadwalan Operasi di Instalasi Bedah Sentral RSUD R.Syamsudin, SH. *SISMATIK (Seminar Nasional Sistem Informasi dan Manajemen Informatika)*, 1–10.
- Sempena, S. (2009). Algoritma Genetik Hibrida dalam Penyelesaian Job-shop Scheduling Problem. *Makalah*, 1–10.