

**OPTIMASI PENJADWALAN PRODUKSI PADA SISTEM *MULTI-
STAGE FLEXIBLE FLOW SHOP* DENGAN PENDEKATAN
*EVOLUTIONARY ALGORITHM***

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Strata-1
Program Studi Teknik Industri - Fakultas Teknologi Industri
Universitas Islam Indonesia**



Nama : Farah Khaerunnisa
No. Mahasiswa 22522086

**PROGRAM STUDI TEKNIK INDUSTRI PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2026**

PERNYATAAN KEASLIAN

Saya mengakui bahwa tugas akhir ini adalah hasil karya saya sendiri kecuali kutipan dan ringkasan yang seluruhnya sudah saya jelaskan sumbernya. Jika dikemudian hari ternyata terbukti pengakuan saya ini tidak benar dan melanggar peraturan yang sah maka saya bersedia ijazah yang telah saya terima ditarik kembali oleh Universitas Islam Indonesia.

Yogyakarta, 24 – 06 - 2026



(Farah Khaerunnisa)

NIM 22522086

SURAT BUKTI PENELITIAN



**FAKULTAS
TEKNOLOGI INDUSTRI**

Gedung KH. Was Mansur
Kampus Terpadu Universitas Islam Indonesia
Jl. Kaliurang km 14,5 Yogyakarta 55584
T : (0274) 898444 ext. 4110, 4100
F : (0274) 896007
E : ft@uii.ac.id
W : ft.uii.ac.id

SURAT KETERANGAN PENELITIAN

Nomor : 302/Ka.lab SIMANTI/20/Lab.SIMANTI/VI/2026

Assalamu'alaikum Warohmatullahi Wabarokaatuh

Dengan hormat,

Yang bertanda tangan dibawah ini, menerangkan bahwa:

Nama : Farah Khaerunnisa

Nim : 22522086

Jurusan : Teknik Industri

Dosen Pembimbing : Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM

Menyatakan bahwa mahasiswa tersebut diatas telah melaksanakan penelitian tugas akhir dengan judul **"OPTIMASI PENJADWALAN PRODUKSI PADA SISTEM MULTI-STAGE FLEXIBLE FLOW SHOP DENGAN PENDEKATAN EVOLUTIONARY ALGORITHM"** mulai pelaksanaan penelitian 12 Januari 2026 sampai 24 Juni 2026.

Demikian surat keterangan penelitian ini kami buat. Atas perhatiannya dan kerja samanya kami mengucapkan terima kasih.

Wassalamu'alaikum Warohmatullahi Wabarokaatuh

Yogyakarta, 25 Juni 2026
Kepala Laboratorium
Sistem Manufaktur Terintegrasi


Putri Dwi Annisa, S.T., M.Sc.

LEMBAR PENGESAHAN PEMBIMBING
**OPTIMASI PENJADWALAN PRODUKSI PADA SISTEM *MULTI-
STAGE FLEXIBLE FLOW SHOP* DENGAN PENDEKATAN**
EVOLUTIONARY ALGORITHM



Disusun Oleh :

Nama : Farah Khaerunnisa

No. Mahasiswa : 22522086

Yogyakarta, 30 - 06 - 2026

Dosen Pembimbing

(Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., PH.D., IPM)

LEMBAR PENGESAHAN DOSEN PENGUJI
OPTIMASI PENJADWALAN PRODUKSI PADA SISTEM *MULTI-STAGE FLEXIBLE FLOW SHOP* DENGAN PENDEKATAN
EVOLUTIONARY ALGORITHM

TUGAS AKHIR

Disusun Oleh :

Nama : Farah Khaerunnisa

No. Mahasiswa : 22 522 086

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Strata-I Teknik Industri Fakultas Teknologi Industri

Universitas Islam Indonesia

Yogyakarta, 09 - Juli - 2026

Tim Penguji

Ir. Muhammad Ridwan Andi Purnomo,
S.T., M.Sc., PH.D., IPM

Ketua

Wahyudhi Sutrisno, S.T., M.M., M.T

Anggota I

Putri Dwi Annisa, S.T., M.Sc.

Anggota II

Mengetahui,

Ketua Program Studi Teknik Industri Program Sarjana



Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., PH.D., IPM

NIK. 015220101

HALAMAN PERSEMBAHAN

Dengan rasa Syukur medalam, karya ini saya persembahkan kepada kedua orang tua tercinta, Ayah Mochamad Faris dan Ibu Rahmawati, yang doa dan kasih sayang nya senantiasa menemani setiap langkah saya hingga mencapai tahap ini. Saya juga mengucapkan terima kasih kepada dosen pembimbing saya, Bapak Ir. Muhammad Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM., atas segala, arahan, dukungan, dan bimbingannya selama proses penyusunan tugas akhir.

MOTTO

"Karena sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya bersama kesulitan ada kemudahan."

(QS. Al-Insyirah: 5–6)

"My heart is at ease knowing that what was meant for me will never miss me, and what misses me was never meant for me."

(Imam Syafi'i)

"Long story short, I survived."

(Taylor Swift)

KATA PENGANTAR

Alhamdulillahirabbil'alamin, segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, nikmat, pertolongan, kemudahan, serta kasih sayang-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **“Optimasi Penjadwalan Produksi Pada Sistem *Multi Stage Flexible Flow Shop* dengan Pendekatan *Evolutionary Algorithm* ”** dengan baik. Shalawat serta salam senantiasa tercurah kepada Nabi Muhammad SAW, keluarga, sahabat, dan seluruh umatnya hingga akhir zaman.

Penyusunan Tugas Akhir ini dilakukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia. Dalam proses penyusunannya, penulis menyadari bahwa perjalanan ini tidak selalu mudah. Berbagai tantangan, keterbatasan, dan kesulitan dihadapi, namun dengan doa, dukungan, bimbingan, serta kebaikan dari banyak pihak, seluruh proses tersebut dapat dilalui hingga akhirnya Tugas Akhir ini dapat terselesaikan.

Oleh karena itu, dengan segala kerendahan hati, penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Sri Kusumadewi, S. Si., M.T. selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
2. Bapak Dr. Drs. Imam Djati Widodo, M.Eng.Sc. selaku Ketua Jurusan Teknik Industri Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Bapak, Ir. M. Ridwan Andi Purnomo, ST., M.Sc., Ph.D., IPM. selaku Ketua Program Studi Teknik Industri Program Sarjana Universitas Islam Indonesia.
4. Bapak Ir. M. Ridwan Andi Purnomo, S.T., M.Sc., Ph.D., IPM. selaku dosen pembimbing Tugas Akhir yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan, ilmu, motivasi, serta dukungan kepada penulis dengan penuh kesabaran selama proses penyusunan Tugas Akhir ini. Segala masukan dan bimbingan yang diberikan menjadi pembelajaran yang sangat berharga bagi penulis, tidak hanya dalam menyelesaikan penelitian ini, tetapi juga dalam menghadapi tantangan di masa mendatang.
5. Kedua orang tua tercinta, Ayah dan Mama yang menjadi rumah pertama sekaligus tempat pulang paling nyaman bagi penulis. Terima kasih atas segala doa yang tidak pernah putus, kasih sayang yang tidak pernah berkurang, pengorbanan yang tidak pernah terhitung, serta kepercayaan yang selalu diberikan kepada penulis untuk terus melangkah dan bertumbuh. Setiap pencapaian dalam perjalanan ini tidak akan pernah terlepas dari cinta, dukungan, dan keikhlasan yang selalu menyertai di setiap langkah. Semoga Allah SWT senantiasa memberikan kesehatan, keberkahan umur, dan kebahagiaan yang berlimpah kepada Ayah dan Mama. Gelar ini mungkin tertulis atas nama penulis, tetapi sejatinya merupakan hasil dari doa, perjuangan, dan kasih sayang yang telah Ayah dan Mama berikan tanpa henti.
6. Adik tercinta, Muhammad Fakhri Syahreza, yang selalu menjadi sumber kebahagiaan dan semangat bagi penulis. Terima kasih karena telah tumbuh menjadi sosok yang membawa keceriaan dan warna dalam kehidupan penulis. Terima kasih atas dukungan, perhatian, dan hal-hal sederhana yang tanpa disadari mampu mengurangi rasa lelah selama proses ini berlangsung. Kehadiranmu selalu menjadi pengingat bagi penulis untuk terus berusaha menjadi kakak yang lebih baik dan seseorang yang dapat dibanggakan. Semoga Allah SWT senantiasa

menjaga, melindungi, memudahkan setiap langkahmu, serta mengabulkan seluruh cita-cita dan impian terbaik yang ingin kamu raih.

7. Kakak Keysha Sharira, Mba Ivana Utari Putri, Popy Zealinsky, dan kakak Viola Ramadhani, dan Zahra Fahriza yang telah menjadi tempat penulis berbagi cerita, keluh kesah, tangisan, kebingungan, dan berbagai rasa yang menyertai perjalanan penyusunan Tugas Akhir ini. Terima kasih karena selalu sabar mendengarkan setiap curhatan penulis, mendengarkan cerita yang mungkin berulang, memberikan perhatian, semangat, doa, serta dukungan yang begitu berarti di saat penulis merasa lelah dan hampir menyerah. Kehadiran kalian menjadi penguat yang sangat berharga, membuat perjalanan ini terasa lebih ringan dan tidak dijalani sendirian.
8. Sahabat Bismillah Sholat, Yovita, Khansa, dan Pingkan yang telah menemani penulis sejak awal perkuliahan hingga proses penyusunan Tugas Akhir ini. Terima kasih karena telah menjadi tempat berbagi cerita, saling menguatkan, mengingatkan dalam kebaikan, serta kebersamai perjalanan yang penuh dinamika ini. Kebersamaan yang terjalin bukan hanya tentang pertemanan selama di bangku kuliah, tetapi juga tentang tumbuh bersama, belajar bersama, dan saling mendoakan untuk masa depan yang lebih baik. Semoga pertemanan kita tetap berlanjut meskipun masa perkuliahan telah selesai, dan semoga kita tetap dipertemukan dalam banyak kebaikan lainnya. Daji, yang telah banyak membantu penulis selama penyusunan skripsi ini. Terima kasih atas kesabaran, bantuan, dan waktu yang telah diluangkan untuk mendengarkan, memberi masukan, dan menjadi tempat bertanya. Kehadiranmu sangat berarti dalam perjalanan menyelesaikan Tugas Akhir ini.
9. Sahabat sejak masa SMP, Alma Raisa dan Latifa Zahra, yang telah kebersamai penulis selama bertahun-tahun hingga saat ini. Terima kasih karena tetap bertahan dan berjalan bersama di tengah banyaknya perubahan, jarak, kesibukan, dan fase kehidupan yang terus berganti. Terima kasih karena selalu menjadi rumah yang hangat untuk pulang, tempat bercerita tanpa rasa takut untuk dihakimi, dan tempat penulis menemukan ketenangan di tengah berbagai hal yang tidak mudah untuk dilalui. Terima kasih atas doa, dukungan, cerita, tawa, dan kasih sayang yang selalu diberikan tanpa syarat. Persahabatan yang tumbuh sejak masa remaja hingga dewasa ini adalah anugerah yang sangat berharga bagi penulis. Semoga persahabatan ini senantiasa dipenuhi keberkahan, kebahagiaan, dan cerita-cerita indah di masa yang akan datang.
10. Seseorang yang pernah menjadi bagian penting dalam perjalanan perkuliahan penulis sejak semester pertama hingga tahap akhir penyusunan Tugas Akhir ini. Meskipun pada akhirnya jalan yang ditempuh tidak lagi berjalan beriringan, penulis tetap bersyukur atas setiap cerita, pelajaran, dukungan, dan kebersamaan yang pernah dihadirkan. Terima kasih karena telah menemani salah satu fase paling berharga dalam hidup penulis, menjadi tempat berbagi suka dan duka, serta memberikan banyak pembelajaran tentang proses bertumbuh dan memaknai kehidupan. Setiap pertemuan memiliki hikmahnya masing-masing, dan setiap kenangan akan selalu menjadi bagian dari perjalanan yang patut disyukuri. Semoga Allah SWT senantiasa memberikan kebahagiaan, keberkahan, dan kemudahan dalam setiap langkah yang akan ditempuh di masa mendatang.
11. Teman-teman penulis yang tidak dapat disebutkan satu per satu, tetapi telah menjadi bagian dari perjalanan selama masa perkuliahan maupun proses penyusunan Tugas Akhir ini. Terima kasih atas setiap dukungan, bantuan,

semangat, perhatian, kebersamaan, serta berbagai cerita yang telah dibagikan. Kehadiran kalian telah memberikan warna, pelajaran, dan kenangan yang berharga, sehingga perjalanan ini terasa lebih ringan dan menyenangkan untuk dijalani. Semoga setiap langkah yang akan ditempuh ke depannya senantiasa membawa kebahagiaan, kesuksesan, dan hal-hal baik yang pantas untuk dirayakan.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kata sempurna. Oleh karena itu, penulis sangat mengharapkan kritik dan saran yang membangun demi perbaikan di masa yang akan datang. Semoga Tugas Akhir ini dapat memberikan manfaat, baik bagi pengembangan ilmu pengetahuan maupun bagi pihak-pihak yang membutuhkan.

Yogyakarta, 24 Juni 2026

A handwritten signature in black ink, appearing to read 'Farah Khaerunnisa', written over a horizontal line.

Farah Khaerunnisa

ABSTRAK

Penjadwalan produksi merupakan salah satu aspek penting dalam meningkatkan efisiensi sistem manufaktur, khususnya pada perusahaan dengan sistem *Make to Stock* (MTS) yang memproduksi barang berdasarkan peramalan permintaan dan menyimpan produk sebagai persediaan untuk memenuhi kebutuhan pelanggan secara cepat. Pada sistem *Make to Stock*, kemampuan perusahaan dalam menyelesaikan produksi secara efisien menjadi penting karena berpengaruh terhadap tingkat ketersediaan produk, pemanfaatan sumber daya, dan pencapaian target produksi. Oleh karena itu, minimasi *makespan* diperlukan agar seluruh pekerjaan dapat diselesaikan lebih cepat sehingga kapasitas produksi dapat dimanfaatkan secara lebih efektif. PT XYZ merupakan perusahaan manufaktur elektronik yang memproduksi berbagai modul *transmitter* dan *receiver* melalui lima *work station* berurutan. Pada kondisi aktual, penjadwalan produksi masih dilakukan berdasarkan urutan produk P1 sampai P20, sehingga belum mempertimbangkan kombinasi urutan pekerjaan, pemilihan mesin, *release time*, dan *sequence-dependent setup time* secara optimal. Penelitian ini bertujuan untuk menentukan kombinasi *job assignment* dan *machine assignment* yang dapat meminimalkan *makespan* pada sistem *multi-stage flexible flow shop* serta menghitung nilai *makespan* setelah dilakukan optimasi. Metode yang digunakan adalah *Flexible Flow Shop Scheduling* dengan pendekatan *Evolutionary Algorithm* melalui *Microsoft Excel Solver*. Data yang digunakan meliputi 20 produk, lima *work station*, 12 mesin paralel, waktu proses, *release time*, urutan proses, dan matriks *setup time*. Hasil penelitian menunjukkan bahwa jadwal aktual perusahaan memiliki *makespan* sebesar 134 menit, sedangkan hasil optimasi *Solver* menghasilkan *makespan* sebesar 110 menit. Dengan demikian, terjadi penurunan *makespan* sebesar 24 menit atau 17,91%. Hasil optimasi juga menurunkan total *setup time* dari 228 menit menjadi 165 menit dan total *idle time* dari 188 menit menjadi 129 menit. Sebanyak 54 dari 100 keputusan alokasi mesin mengalami perubahan. Temuan ini menunjukkan bahwa perbaikan jadwal diperoleh melalui kombinasi perubahan urutan pekerjaan, pemilihan mesin, pengurangan *setup time*, dan pemanfaatan mesin paralel yang lebih efektif.

Kata Kunci: penjadwalan produksi, *multi-stage flexible flow shop*, *Evolutionary Algorithm*, *makespan*, *Excel Solver*.

DAFTAR ISI

PERNYATAAN KEASLIAN.....	i
SURAT BUKTI PENELITIAN	ii
LEMBAR PENGESAHAN PEMBIMBING	iii
LEMBAR PENGESAHAN DOSEN PENGUJI	iv
HALAMAN PERSEMBAHAN.....	v
MOTTO	vi
KATA PENGANTAR	vii
ABSTRAK.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	7
1.3 Tujuan Penelitian	8
1.4 Manfaat Penelitian.....	8
1.5 Batasan Penelitian	8
1.6 Sistematika Penelitian	9
BAB II TINJAUAN PUSTAKA.....	11
2.1 Kajian Literatur	11
2.1.1 Perkembangan Metode Penjadwalan Produksi Berbasis Evolutionary Algorithm	11
2.1.2 Penjadwalan pada Sistem Multi-Stage dan Flexible Flow Shop	12
2.1.3 Perkembangan pada Flexible Job Shop dan Sistem yang Lebih Fleksibel	12
2.1.4 Integrasi Metode Hybrid dan Pendekatan Lanjutan	13
2.1.5 Perbandingan Berdasarkan Objective Function	14
2.2 <i>Research Gap</i>	16
2.3 Landasan Teori.....	18
2.3.1 Sistem Produksi Manufaktur	18
2.3.2 Production Planning and Inventory Control.....	19
2.3.3 Penjadwalan Produksi (Production Scheduling)	20
2.3.4 Flow Shop Scheduling.....	21
2.3.5 Flexible Flow Shop Scheduling (FFSP)	22
2.3.6 Permutation Flexible Flow Shop	23
2.3.7 Mesin Paralel dalam Sistem Produksi	24
2.3.8 Machine-Dependent Processing Time.....	25
2.3.9 Sequence-Dependent Setup Time (SDST)	25
2.3.10 <i>Makespan</i> sebagai Kriteria Optimasi.....	26
2.3.11 <i>Idle time</i> Mesin.....	27
2.3.12 Evolutionary Algorithm.....	27
2.3.13 Excel Solver Evolutionary sebagai Implementasi Metode Optimasi	28
BAB III METODE PENELITIAN.....	31
3.1 Subjek Penelitian.....	31
3.2 Objek Penelitian	31
3.3 Jenis Data	31

3.4	Komponen Model.....	32
3.4.1	Notasi Model	32
3.4.2	Variabel Keputusan dan Kriteria Performansi.....	33
3.4.3	Asumsi Model	33
3.5	Formulasi Model Penjadwalan.....	34
3.5.1	Fungsi Tujuan.....	34
3.5.2	Waktu Selesai Job.....	34
3.5.3	Waktu Mulai Proses dengan Setup Time	35
3.5.4	Rumus <i>Idle time</i>	35
3.5.5	Batasan Urutan Proses.....	35
3.5.6	Batasan Mesin	36
3.5.7	Batasan Setiap Job Dijadwalkan Satu Kali	36
3.5.8	Batasan Pemilihan Mesin	36
3.5.9	Persentase Perbaikan	37
3.6	Algoritma Pemecahan Masalah.....	37
3.7	Validasi Model	38
3.8	Teknik Pengumpulan Data.....	39
3.9	Alur Penelitian	39
BAB IV	PENGUMPULAN DAN PENGOLAHAN DATA	45
4.1	Pengumpulan Data	45
4.1.1	Data Produk.....	45
4.1.2	Data <i>Work station</i> dan Mesin	46
4.1.3	Data Waktu Proses	47
4.1.4	Data Release Time.....	48
4.1.5	Data Waktu Setup Time	49
4.1.6	Data Urutan Proses.....	50
4.2	Pengolahan Data.....	50
4.2.1	Pengolahan Jadwal Aktual Perusahaan	50
A.	Menentukan Job assignment	50
B.	Menentukan <i>Processor</i> pada Setiap <i>Work station</i>	50
C.	Menentukan <i>Previous Job</i>	51
D.	Menghitung <i>Setup Time</i>	52
E.	Menghitung <i>Start Time</i>	53
F.	Menghitung Waktu Proses (<i>Duration</i>)	55
G.	Menghitung <i>Finish Time</i>	55
H.	Menghitung <i>Makespan</i>	56
I.	Menghitung <i>Idle time</i>	57
J.	Menghitung <i>Total Idle time</i> pada Setiap <i>Work station</i>	57
K.	Hasil Perhitungan Jadwal Aktual Perusahaan	58
4.2.2	Pengolahan Jadwal Optimasi Solver	64
A.	Menentukan Variabel Keputusan <i>Solver</i>	64
B.	Menentukan Fungsi Tujuan <i>Solver</i>	64
C.	Menentukan Batasan <i>Solver</i>	65
D.	Menjalankan <i>Solver</i>	66
E.	Hasil Optimasi <i>Solver</i> Jadwal Aktual.....	67
BAB V	PEMBAHASAN DAN ANALISIS HASIL	73
5.1	Pengujian dan Validasi Model Penjadwalan	73
5.1.1	Pengujian Konsistensi Data dan Perhitungan.....	73
5.1.2	Pengujian Pemenuhan Batasan Penjadwalan	79

5.1.3	Pengujian Stabilitas Hasil Solver	81
5.2	Analisis Hasil Penjadwalan Aktual dan Hasil Optimasi <i>Solver</i>	82
5.2.1	Perubahan Job assignment dan Machine Assignment	83
A.	Perubahan <i>Job assignment</i> dan Pengaruhnya terhadap Aliran Produk	83
B.	Perubahan <i>Machine Assignment</i> dan Pemanfaatan Mesin Paralel	85
C.	Dampak perubahan Urutan dan Alokasi Mesin terhadap Waktu Produksi.	90
5.2.2	Perbandingan <i>Makespan</i>	91
A.	Perubahan Waktu Penyelesaian Setiap <i>Work station</i>	92
B.	Perubahan Waktu Selesai Produk	93
5.2.3	Analisis <i>Idle time</i>	94
5.2.4	Analisis Gantt Chart	95
A.	Analisis <i>Gantt Chart</i> Jadwal Aktual Perusahaan	96
B.	Analisis <i>Gantt Chart</i> Hasil Optimasi <i>Solver</i>	99
C.	Perbandingan Visual Kedua <i>Gantt Chart</i>	102
5.3	Temuan Utama dan Interpretasi Hasil	103
BAB VI	PENUTUP	112
6.1	Kesimpulan	112
6.2	Saran	113
DAFTAR PUSTAKA		115
LAMPIRAN		A-1

DAFTAR TABEL

Tabel 2. 1 <i>Research Gap</i>	17
Tabel 3. 1 Langkah Pengolahan Data	37
Tabel 4. 1 Nama Produk yang di Produksi PT XYZ.....	46
Tabel 4. 2 Pembagian Mesin Pada <i>Work station</i>	46
Tabel 4. 3 Waktu Proses Produk	47
Tabel 4. 4 <i>Release Time</i>	48
Tabel 4. 5 <i>Sequence-Dependent Setup Time</i>	49
Tabel 4. 6 Urutan Proses	50
Tabel 4. 7 <i>Processor</i> yang Digunakan Pada Setiap <i>Work station</i>	51
Tabel 4. 8 Perhitungan Jadwal Aktual <i>Work station</i> 1.....	58
Tabel 4. 9 Perhitungan Jadwal Aktual <i>Work station</i> 2.....	59
Tabel 4. 10 Perhitungan Jadwal Aktual <i>Work station</i> 3.....	59
Tabel 4. 11 Perhitungan Jadwal Aktual <i>Work station</i> 4.....	60
Tabel 4. 12 Perhitungan Jadwal Aktual <i>Work station</i> 5.....	61
Tabel 4. 13 <i>Idle time</i> Jadwal Aktual Perusahaan	61
Tabel 4. 14 Variabel Keputusan <i>Solver</i>	64
Tabel 4. 15 <i>Constraints</i> pada <i>Excel Solver</i>	66
Tabel 4. 16 Optimasi <i>Solver</i> Jadwal Aktual <i>Work station</i> 1	67
Tabel 4. 17 Optimasi <i>Solver</i> Jadwal Aktual <i>Work station</i> 2	68
Tabel 4. 18 Optimasi <i>Solver</i> Jadwal Aktual <i>Work station</i> 3	69
Tabel 4. 19 Optimasi <i>Solver</i> Jadwal Aktual <i>Work station</i> 4	69
Tabel 4. 20 Optimasi <i>Solver</i> Jadwal Aktual <i>Work station</i> 5	70
Tabel 4. 21 <i>Idle time</i> Optimasi <i>Solver</i>	70
Tabel 5. 1 Contoh Pengujian Konsistensi Perhitungan Waktu	74
Tabel 5. 2 Pengujian <i>Previous Job</i> pada Mesin MA di WS1 Hasil Optimasi <i>Solver</i>	75
Tabel 5. 3 Hasil <i>Makespan</i> Keseluruhan.....	78
Tabel 5. 4 Pengujian Non-Overlap pada Mesin MA di WS1 <i>Solver</i> 2.....	79
Tabel 5. 5 Hasil Lima Percobaan <i>Solver Evolutionary</i>	81
Tabel 5. 6 Perbandingan <i>Job assignment</i> dan <i>Machine Assignment</i> Jadwal Aktual.....	83
Tabel 5. 7 Perbandingan <i>Job assignment</i> dan <i>Machine Assignment</i> Hasil Optimasi <i>Solver</i>	84
Tabel 5. 8 Contoh Perubahan Alokasi Mesin Produk P14.....	86
Tabel 5. 9 Ringkasan Perubahan <i>Machine Assignment</i> pada Setiap <i>Work station</i>	87
Tabel 5. 10 Perbandingan Perubahan Urutan, Mesin, dan Waktu Penyelesaian Produk P14 dan P7.....	87
Tabel 5. 11 Contoh Pemanfaatan Mesin Paralel pada <i>Work station</i> 1 Hasil Optimasi <i>Solver</i>	89
Tabel 5. 12 Perbandingan Total Duration dan <i>Setup Time</i> antara Jadwal Aktual dan Hasil Optimasi <i>Solver</i>	90
Tabel 5. 13 Perbandingan <i>Makespan</i> Jadwal Aktual dan Hasil Optimasi <i>Solver</i>	91
Tabel 5. 14 Perbandingan Waktu Penyelesaian Terakhir pada Setiap <i>Work station</i>	92
Tabel 5. 15 Perbandingan Waktu Selesai Produk pada Jadwal Aktual dan Hasil Optimasi <i>Solver</i>	93
Tabel 5. 16 Perbandingan <i>Idle time</i> Jadwal Aktual dan Hasil Optimasi <i>Solver</i>	94
Tabel 5. 17 Ringkasan Temuan Utama Hasil Penjadwalan	103
Tabel 6. 1 Kombinasi Alokasi Mesin di Maing-Masing Produk	112

DAFTAR GAMBAR

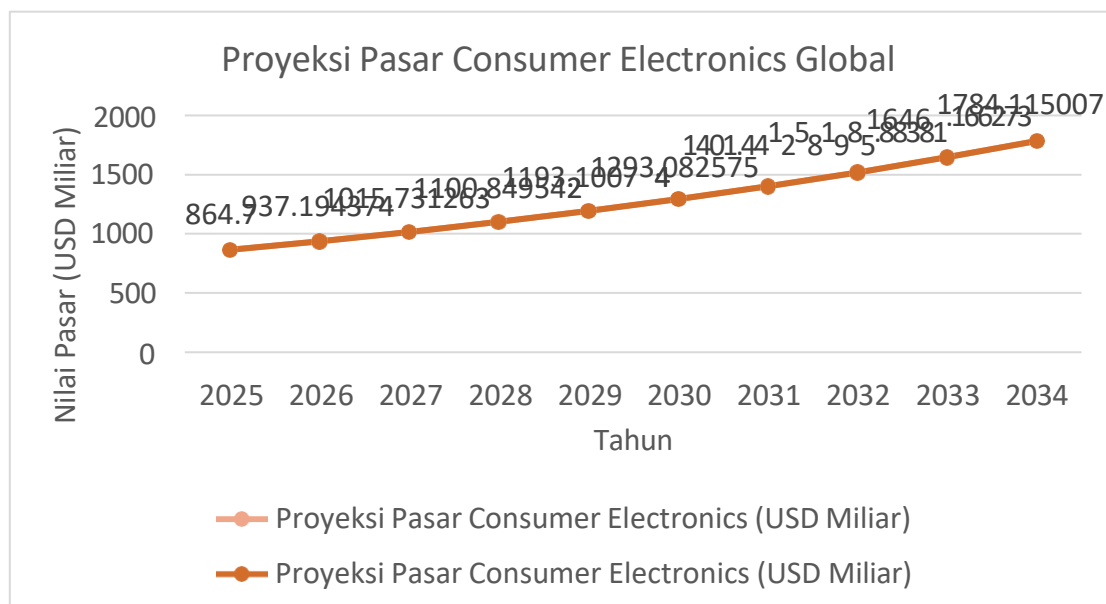
Gambar 1. 1 Proyeksi Pertumbuhan Nilai Pasar Elektronik Global (Sumber: https://www.fortunebusinessinsights.com/consumer-electronics-market-104693)	1
Gambar 1. 2 <i>Capacity Utilization</i> Sektor <i>Semiconductor & Electronic Components</i> dari FRED (Sumber: https://fred.stlouisfed.org/series/CAPUTLG3344S)	2
Gambar 3. 1 Alur Penelitian.....	40
Gambar 4. 1 Penentuan <i>Previous Job</i>	52
Gambar 4. 2 Perhitungan <i>Setup Time</i>	53
Gambar 4. 3 Perhitungan <i>Start Time</i> WS 1.....	54
Gambar 4. 4 Perhitungan <i>Start Time</i> WS 1 untuk P2 hingga P20	54
Gambar 4. 5 <i>Start Time</i> WS 2 hingga WS 5.....	55
Gambar 4. 6 Perhitungan <i>Duration</i>	55
Gambar 4. 7 Perhitungan <i>Finish Time</i>	56
Gambar 4. 8 Perhitungan <i>Makespan</i>	56
Gambar 4. 9 <i>Idle time</i> pada Setiap <i>Job</i>	57
Gambar 4. 10 <i>Total Idle time</i>	58
Gambar 4. 11 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 1</i>	63
Gambar 4. 12 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 2</i>	63
Gambar 4. 13 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 3</i>	63
Gambar 4. 14 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 4</i>	63
Gambar 4. 15 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 5</i>	63
Gambar 4. 16 Variabel Keputusan yang Akan Diubah <i>Solver</i>	64
Gambar 4. 17 Fungsi Tujuan <i>Solver</i>	65
Gambar 4. 18 <i>Constraints</i> untuk <i>Job assignment</i>	65
Gambar 4. 19 <i>Solver Parameters</i> Untuk Menjalankan <i>Solver</i>	67
Gambar 4. 20 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 1</i>	72
Gambar 4. 21 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 2</i>	72
Gambar 4. 22 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 3</i>	72
Gambar 4. 23 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 4</i>	72
Gambar 4. 24 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station</i>	72
Gambar 5. 1 Matriks <i>Setup Time</i>	76
Gambar 5. 2 Contoh Pengujian <i>Setup Time</i> pada Mesin MA <i>Work station 1</i> Hasil Optimasi <i>Solver</i>	77
Gambar 5. 3 Contoh Perhitungan <i>Start Time</i> pada <i>Work station 1</i>	77
Gambar 5. 4 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 1</i>	96
Gambar 5. 5 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 2</i>	96
Gambar 5. 6 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 3</i>	96
Gambar 5. 7 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 4</i>	96
Gambar 5. 8 <i>Gantt Chart</i> Aktual Perusahaan <i>Work station 5</i>	96
Gambar 5. 9 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 1</i>	99
Gambar 5. 10 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 2</i>	99
Gambar 5. 11 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 3</i>	99
Gambar 5. 12 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 4</i>	99
Gambar 5. 13 <i>Gantt Chart</i> Hasil Optimasi <i>Solver Work station 5</i>	99

BAB I PENDAHULUAN

1.1 Latar Belakang.

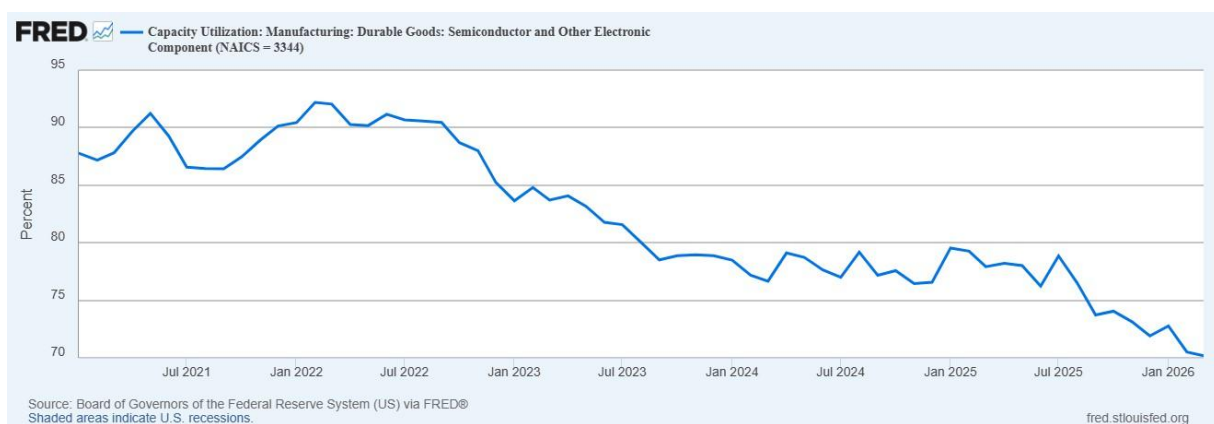
Perkembangan industri elektronik dalam beberapa dekade terakhir mengalami pertumbuhan yang sangat pesat seiring meningkatnya penggunaan perangkat elektronik pada berbagai sektor, seperti industri manufaktur, otomasi, telekomunikasi, sistem kendali, hingga perangkat rumah tangga. Produk elektronik modern tidak lagi hanya berfungsi sebagai alat bantu sederhana, tetapi berkembang menjadi perangkat yang mampu merespons kondisi lingkungan dan menjalankan fungsi otomatis secara lebih kompleks. Kondisi tersebut menyebabkan kebutuhan terhadap komponen elektronik terus meningkat, baik dari sisi jumlah maupun variasi produk yang diproduksi (Boylestad & Nashelsky, 2019).

Pertumbuhan industri elektronik global menunjukkan tren yang terus meningkat seiring dengan berkembangnya teknologi digital dan otomatisasi. Berdasarkan laporan dari (Fortune Business Insights, 2026), pasar consumer electronics global bernilai USD 864,73 miliar pada 2025 dan diproyeksikan meningkat dari USD 922,66 miliar pada 2026 menjadi USD 1.756,39 miliar pada 2034, dengan CAGR 8,38% selama periode 2026–2034. Peningkatan tersebut mencerminkan kebutuhan yang semakin besar terhadap perangkat elektronik, sistem otomasi, serta perangkat berbasis kontrol dan komunikasi sinyal, dan sistem digital modern.



Gambar 1. 1 Proyeksi Pertumbuhan Nilai Pasar Elektronik Global (Sumber: <https://www.fortunebusinessinsights.com/consumer-electronics-market-104693>)

Berdasarkan Gambar 1.1, nilai pasar consumer electronics global diproyeksikan terus meningkat dari tahun 2025 hingga 2034. Peningkatan tersebut menjadi gambaran bahwa industri elektronik perlu memiliki sistem produksi yang mampu merespons kebutuhan pasar secara lebih efektif. Di sisi lain, peningkatan permintaan tidak selalu dapat diimbangi secara langsung melalui penambahan kapasitas fisik, seperti pembelian mesin baru, perluasan fasilitas, penambahan tenaga kerja, lembur, atau penggunaan subcontracting. Alternatif tersebut membutuhkan biaya, waktu implementasi, dan pertimbangan operasional yang tidak sederhana. Oleh karena itu, pemanfaatan kapasitas produksi yang sudah tersedia menjadi salah satu pendekatan penting yang dapat dilakukan perusahaan.



Gambar 1. 2 Capacity Utilization Sektor Semiconductor & Electronic Components dari FRED (Sumber: <https://fred.stlouisfed.org/series/CAPUTLG3344S>)

Berdasarkan Gambar 1.2, tingkat capacity utilization pada sektor tersebut mengalami fluktuasi dan cenderung menurun pada periode 2021 hingga 2026. Data tersebut digunakan sebagai konteks industri, bukan untuk menjelaskan kondisi perusahaan secara spesifik. Namun, data tersebut dapat menjadi gambaran umum bahwa pemanfaatan kapasitas merupakan aspek penting dalam industri komponen elektronik. Oleh karena itu, dalam kondisi kapasitas fasilitas produksi tidak selalu dapat ditingkatkan secara cepat, perusahaan perlu mengoptimalkan pemanfaatan sumber daya produksi yang tersedia, salah satunya melalui penjadwalan produksi yang lebih efisien.

Penjadwalan produksi dipilih sebagai fokus penelitian karena penjadwalan berhubungan langsung dengan pemanfaatan kapasitas efektif di rantai produksi. Kapasitas efektif tidak hanya ditentukan oleh jumlah mesin yang tersedia, tetapi juga oleh bagaimana pekerjaan diurutkan, mesin dipilih, waktu setup diperhitungkan, serta waktu mulai proses ditentukan. Penjadwalan yang kurang optimal dapat menyebabkan waktu tunggu meningkat,

mesin menganggur, penumpukan pekerjaan, dan waktu penyelesaian produksi menjadi lebih panjang (Pinedo, 2016). Oleh karena itu, dalam kondisi kapasitas fisik sulit ditingkatkan secara cepat, penjadwalan produksi menjadi pendekatan yang relevan karena dapat membantu perusahaan memanfaatkan mesin yang tersedia secara lebih efisien tanpa harus menambah mesin atau fasilitas baru.

Dalam konteks *Production Planning and Inventory Control* (PPIC), penjadwalan produksi memiliki peran penting dalam mengatur aliran material, pemanfaatan kapasitas mesin, serta pencapaian target produksi. Salah satu pendekatan utama dalam penjadwalan produksi adalah dengan menentukan fungsi tujuan yang ingin dicapai. Salah satu fungsi tujuan yang umum digunakan adalah minimasi *makespan*, yaitu waktu total yang dibutuhkan untuk menyelesaikan seluruh pekerjaan dalam sistem produksi (Hasani et al., 2022a; Pinedo, 2022a). Dalam konteks penelitian ini, *makespan* dipilih sebagai fungsi tujuan utama karena menggambarkan waktu total penyelesaian seluruh pekerjaan dalam satu horizon penjadwalan. Minimasi *makespan* menjadi relevan terutama pada sistem produksi *Make to Stock* (MTS), karena semakin cepat seluruh batch produksi diselesaikan, maka semakin cepat pula kapasitas mesin dapat digunakan kembali untuk proses produksi berikutnya. Dengan demikian, minimasi *makespan* tidak hanya berfokus pada percepatan penyelesaian pekerjaan, tetapi juga mendukung pemanfaatan kapasitas mesin secara lebih efektif tanpa harus menambah kapasitas fisik secara langsung (Pinedo, 2016; Slack & Brandon-Jones, 2019).

Perusahaan yang menjadi subjek penelitian menerapkan sistem produksi *Make to Stock* (MTS), yaitu sistem produksi yang dilakukan berdasarkan hasil peramalan permintaan atau forecasting. Pada sistem ini, perusahaan menetapkan target produksi sebagai acuan jumlah produk yang harus diselesaikan dalam periode tertentu agar ketersediaan produk dapat terjaga dan kebutuhan pasar dapat dipenuhi dengan cepat (Slack & Brandon-Jones, 2019).

Namun, target produksi yang telah ditetapkan tidak selalu dapat tercapai sesuai kondisi aktual di rantai produksi. Perusahaan memiliki data target produksi yang disusun berdasarkan hasil forecasting dan juga memiliki catatan realisasi hasil produksi. Perbandingan antara target produksi dan realisasi produksi menunjukkan adanya kesenjangan antara kondisi ideal yang direncanakan dengan kondisi aktual yang terjadi. Berdasarkan hasil wawancara dengan manajer produksi, diketahui bahwa realisasi produksi saat ini masih berada di bawah target produksi yang telah ditetapkan. Kondisi tersebut menunjukkan bahwa sistem produksi belum sepenuhnya mampu mencapai *output* yang direncanakan.

Permasalahan tersebut tidak hanya berkaitan dengan besarnya permintaan, tetapi juga dengan bagaimana pekerjaan mengalir dan memanfaatkan sumber daya produksi yang tersedia di rantai produksi. Berdasarkan hasil wawancara dengan manajer produksi dan kondisi awal penjadwalan perusahaan, diketahui bahwa proses produksi masih dijalankan berdasarkan urutan produk secara berurutan dari P1 sampai P20, dan juga bukan disebabkan oleh kerusakan mesin (*breakdown*) ataupun kelangkaan bahan baku, melainkan teridentifikasi dari gejala operasional di rantai produksi yang mengarah pada ketidakefisienan penjadwalan. Pola tersebut menyebabkan pengaturan pekerjaan belum mempertimbangkan kesiapan produk untuk diproses, perbedaan kemampuan mesin, serta waktu persiapan antarproduk. Akibatnya, beberapa gejala operasional dapat muncul, seperti waktu tunggu (*idle time*) pada mesin karena produk belum tersedia sesuai *release time*, beban kerja mesin yang tidak seimbang karena pemilihan mesin belum mempertimbangkan perbedaan waktu proses, serta waktu setup yang lebih panjang karena urutan produk belum disusun berdasarkan hubungan setup antarproduk.

Pengabaian terhadap elemen-elemen penjadwalan tersebut dapat menimbulkan beberapa dampak operasional. Pertama, produk yang dijadwalkan lebih awal belum tentu telah tersedia untuk diproses karena setiap produk memiliki *release time* yang berbeda. Kondisi ini dapat menyebabkan mesin berada dalam keadaan siap beroperasi tetapi harus menunggu produk yang belum tersedia, sehingga terjadi waktu menganggur mesin, sehingga dapat menurunkan pemanfaatan sumber daya dan memperpanjang waktu penyelesaian pekerjaan (Pinedo, 2016). Waktu menganggur mesin menunjukkan adanya kapasitas yang tidak termanfaatkan secara produktif, sehingga *output* aktual yang dihasilkan dalam periode penjadwalan menjadi lebih rendah dibandingkan kapasitas yang seharusnya dapat digunakan.

Kedua, pemilihan mesin yang belum mempertimbangkan perbedaan waktu proses antar mesin dapat menyebabkan ketidakseimbangan beban kerja pada mesin paralel. Pada mesin paralel yang tidak identik, satu produk dapat membutuhkan waktu proses yang berbeda ketika dikerjakan pada mesin yang berbeda. Hasani et al. (2022b) menjelaskan bahwa pada *flexible flow shop* dengan *unrelated parallel machines*, perbedaan tingkat teknologi dan kecepatan proses menyebabkan waktu proses dan biaya produksi setiap mesin dapat berbeda. Oleh karena itu, apabila alokasi mesin tidak dilakukan secara tepat, sebagian mesin dapat memiliki beban kerja lebih tinggi dan menjadi titik penumpukan pekerjaan, sementara mesin lain kurang termanfaatkan. Ketidakseimbangan beban kerja tersebut dapat memperpanjang waktu tunggu antar *work station* dan memperbesar waktu penyelesaian keseluruhan pekerjaan.

Ketiga, urutan pekerjaan yang tidak mempertimbangkan *sequence-dependent setup time* dapat menyebabkan waktu persiapan mesin menjadi lebih panjang. Dalam kondisi *setup time* bergantung pada urutan produk sebelumnya, pemilihan urutan pekerjaan yang kurang tepat dapat meningkatkan frekuensi atau durasi setup antarproduk. Allahverdi (2015) menjelaskan bahwa setup time dan setup cost merupakan aspek penting dalam masalah penjadwalan karena berkaitan dengan penggunaan waktu dan kapasitas produksi. Akibatnya, waktu yang seharusnya dapat digunakan untuk memproses produk justru digunakan untuk aktivitas persiapan mesin. Secara operasional, kondisi tersebut menyebabkan hilangnya kapasitas efektif karena sebagian waktu kerja mesin tidak digunakan untuk menghasilkan *output*.

Akumulasi dari waktu menganggur mesin, ketidakseimbangan beban kerja, dan waktu setup yang panjang dapat berdampak pada meningkatnya *makespan*. *Makespan* yang lebih panjang menunjukkan bahwa seluruh pekerjaan membutuhkan waktu penyelesaian yang lebih lama, sehingga kapasitas produksi membutuhkan waktu lebih lama untuk dapat digunakan kembali pada periode produksi berikutnya. Dalam sistem *Make to Stock*, kondisi tersebut dapat menghambat kemampuan perusahaan dalam mencapai target produksi yang telah ditentukan berdasarkan *forecasting*. Apabila target produksi tidak tercapai, maka ketersediaan produk jadi berisiko tidak sesuai dengan kebutuhan yang direncanakan dan kemampuan perusahaan dalam merespons permintaan pasar menjadi kurang optimal. Dengan demikian, permasalahan penjadwalan tidak hanya berdampak pada aspek teknis di rantai produksi, tetapi juga berpengaruh terhadap pencapaian target produksi dan efektivitas kegiatan PPIC.

Perusahaan yang menjadi subjek penelitian merupakan perusahaan manufaktur elektronik yang memproduksi modul *transmitter* dan *receiver*. Dalam sistem elektronik, transmitter dan receiver bekerja sebagai pasangan komunikasi sinyal, di mana *transmitter* menghasilkan atau memancarkan sinyal, kemudian receiver menangkap, menguatkan, menyaring, mengondisikan, serta mengubah sinyal tersebut agar dapat diinterpretasikan oleh sistem pengukuran atau sistem kontrol (Pallas-Areny & Webster, 2012) (Helfrick & Cooper, 1990). Produk yang diproduksi terdiri dari sekitar 20 jenis produk dan diproduksi melalui lima *work station* yang harus dilalui secara berurutan. Pada setiap *work station*, tersedia dua hingga tiga mesin, sehingga terdapat beberapa alternatif mesin yang dapat digunakan untuk memproses setiap produk. Kondisi ini menyebabkan proses penjadwalan tidak hanya berkaitan dengan urutan pengerjaan produk, tetapi juga dengan pemilihan mesin pada setiap tahap proses.

Berdasarkan karakteristik tersebut, sistem produksi perusahaan dapat dikategorikan sebagai *multi-stage flexible flow shop*, karena seluruh produk melewati urutan proses yang sama, tetapi memiliki fleksibilitas dalam pemilihan mesin pada setiap *work station*. Pada sistem ini, keputusan penjadwalan tidak hanya berkaitan dengan penentuan urutan pekerjaan (*job assignment*), tetapi juga pemilihan mesin yang digunakan pada setiap tahap proses (*machine assignment*).

Permasalahan penjadwalan pada perusahaan menjadi semakin kompleks karena mesin-mesin paralel yang tersedia pada setiap *work station* tidak bersifat identik. Mesin paralel yang tidak identik memiliki perbedaan spesifikasi, kemampuan proses, dan waktu proses untuk produk yang sama. Selain itu, terdapat *release time* yang menunjukkan waktu paling awal suatu produk tersedia untuk diproses, serta *sequence-dependent setup time* yang menunjukkan waktu persiapan mesin yang bergantung pada urutan produk sebelumnya. Kondisi ini menunjukkan bahwa jadwal aktual yang masih disusun berdasarkan urutan P1 sampai P20 belum sepenuhnya merepresentasikan karakteristik sistem produksi perusahaan.

Berdasarkan permasalahan tersebut, penyelesaian yang sesuai adalah optimasi penjadwalan produksi yang mampu menentukan kombinasi urutan pekerjaan atau job sequencing dan pemilihan mesin atau *machine assignment* pada setiap *work station* dengan mempertimbangkan *release time*, waktu proses, dan *sequence-dependent setup time*. Penjadwalan ulang diperlukan agar pekerjaan tidak hanya diproses berdasarkan urutan aktual P1 sampai P20, tetapi dapat diatur berdasarkan kombinasi yang lebih efektif. Karena semakin banyak jumlah produk, *work station*, dan alternatif mesin, maka semakin besar pula kemungkinan solusi penjadwalan yang harus dipertimbangkan. Oleh karena itu, diperlukan metode optimasi yang mampu menentukan kombinasi urutan pekerjaan dan pemilihan mesin secara lebih efektif, sehingga waktu penyelesaian produksi dapat dipersingkat, waktu menganggur mesin dapat ditekan, dan pemanfaatan mesin yang tersedia menjadi lebih optimal.

Selain permasalahan praktis pada perusahaan, penelitian ini juga memiliki celah penelitian dari sisi pemodelan penjadwalan produksi. Penelitian terdahulu telah banyak membahas penjadwalan pada lingkungan *flow shop*, *permutation flow shop*, *flexible flow shop*, maupun *flexible job shop* menggunakan berbagai pendekatan metaheuristik. Namun, penelitian terapan yang secara spesifik memodelkan sistem produksi multi-stage flexible flow shop dengan mesin paralel tidak identik, *machine-dependent processing time*, *release time*, dan *sequence-dependent setup time*, kemudian membandingkan hasil optimasi dengan kondisi

aktual perusahaan, masih terbatas. Oleh karena itu, penelitian ini diarahkan untuk mengisi celah tersebut melalui optimasi job sequencing dan machine assignment menggunakan Evolutionary Algorithm pada Microsoft Excel Solver.

Permasalahan penjadwalan pada sistem *flexible flow shop* termasuk permasalahan yang kompleks dan bersifat *NP-hard*, sehingga sulit diselesaikan secara optimal menggunakan metode penjadwalan konvensional. Oleh karena itu, penelitian ini menggunakan pendekatan Flexible Flow Shop Scheduling (FFSS) untuk memodelkan sistem produksi perusahaan, karena pendekatan ini mampu merepresentasikan sistem produksi bertahap dengan mesin paralel pada setiap tahap proses. Selanjutnya, *Evolutionary Algorithm* (EA) digunakan sebagai metode optimasi karena mampu mengeksplorasi ruang solusi yang besar dan mencari solusi mendekati optimal melalui proses iteratif berbasis populasi (Eiben & Smith, 2015; Ruiz & Vázquez-Rodríguez, 2010).

Melalui penerapan *Flexible Flow Shop Scheduling* dan *Microsoft Excel Solver* metode Evolutionary, penelitian ini bertujuan memperoleh usulan jadwal produksi yang lebih baik dibandingkan kondisi aktual perusahaan. Hasil penjadwalan usulan akan dibandingkan dengan kondisi aktual perusahaan, yaitu urutan pengerjaan produk P1 sampai P20. Dengan nilai *makespan* yang lebih kecil, waktu penyelesaian produksi dapat dipersingkat, waktu menganggur mesin dapat ditekan, serta kapasitas mesin yang tersedia dapat dimanfaatkan secara lebih efektif dalam mendukung kegiatan *Production Planning and Inventory Control* (PPIC).

1.2 Rumusan Masalah

Berdasarkan latar belakang permasalahan yang telah diuraikan, maka rumusan masalah dalam penelitian Adalah sebagai berikut.

1. Apa kombinasi urutan pekerjaan (*job sequencing*) dan pemilihan mesin (*machine assignment*) yang dapat meminimalkan *makespan* pada sistem *multi-stage flexible flow shop*?
2. Berapa nilai *makespan* yang dihasilkan setelah dilakukan penjadwalan ulang pada proses produksi menggunakan pendekatan *Evolutionary Algorithm*?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, tujuan dari penelitian ini Adalah sebagai berikut.

- a) Menentukan kombinasi urutan pengerjaan (*Job assignment*) dan pemilihan mesin (*production routing*) yang optimal untuk meminimalkan *makespan*.
- b) Menghitung nilai *makespan* yang dihasilkan setelah dilakukan penjadwalan ulang pada proses produksi menggunakan pendekatan *Evolutionary Algorithm*.

1.4 Manfaat Penelitian

Manfaat yang diharapkan dari penelitian adalah sebagai berikut.

- a) Manfaat bagi Perusahaan
 - Memberikan rekomendasi penjadwalan produksi yang lebih optimal dalam menentukan alokasi produk ke mesin dan urutan pengerjaan.
 - Membantu perusahaan memperoleh usulan jadwal produksi yang lebih efisien melalui penentuan urutan pekerjaan dan pemilihan mesin yang menghasilkan waktu penyelesaian produksi lebih singkat.
 - Mendukung pengambilan keputusan dalam penyusunan jadwal produksi dan penentuan prioritas pengerjaan, khususnya pada kondisi kedatangan pesanan yang tidak pasti.
 - Meningkatkan efesiensi sistem produksi tanpa memerlukan penambahan mesin atau sumber daya baru.
- b) Manfaat bagi Dunia Akademik
 - Menambah referensi dan kajian ilmiah terkait penerapan *Flexible Flow Shop Scheduling* pada industry manufaktur.
 - Memberikan kontribusi dalam pengembangan metode optimasi penjadwalan.
 - Menjadi bahan acuan bagi penelitian selanjutnya di bidang *Production Planning and Inventory Control* (PPIC), khususnya pada permasalahan penjadwalan produksi *multi-stage* dengan mesin paralel.

1.5 Batasan Penelitian

Agar penelitian lebih terarah dan sesuai dengan tujuan yang telah ditetapkan, maka batasan penelitian dalam penelitian ini adalah sebagai berikut.

- a) Penelitian di fokuskan pada penjadwalan produksi pada sistem *multi-stage flexible flow shop* di perusahaan manufaktur sensor *receiver* elektronik.
- b) Objek penelitian dibatasi pada lima *work station* dengan jumlah mesin parallel dua hingga tiga mesin pada setiap *work station*.
- c) Produk yang diteliti terbatas pada sekitar 20 jenis produk dengan urutan proses yang sama dan variasi waktu proses.
- d) Fungsi tujuan utama adalah minimasi *makespan*. Batasan ini digunakan karena perusahaan menerapkan sistem *make to stock*, sehingga penyelesaian keseluruhan *batch* produksi secara lebih cepat diperlukan agar kapasitas mesin dapat dimanfaatkan kembali untuk proses produksi berikutnya.
- e) Penelitian ini tidak membahas aspek biaya produksi, kualitas produk, perawatan mesin, maupun faktor sumber daya manusia secara mendalam.
- f) Metode optimasi yang digunakan dibatasi pada *Flexible Flow Shop Scheduling* dan *Evolutionary Algorithm* sesuai dengan ruang lingkup penelitian.
- g) Proses pengerjaan pada setiap mesin bersifat *non-preemptive*, yaitu pekerjaan yang telah mulai diproses tidak dapat dihentikan atau disela hingga proses pada mesin tersebut selesai.

1.6 Sistematika Penelitian

Penulisan skripsi ini disusun secara sistematis ke dalam beberapa bab agar alur pembahasan penelitian dapat dipahami secara runtut dan terstruktur. Sistematika penulisan ini bertujuan untuk memberikan gambaran yang jelas mengenai isi dan hubungan antar bab dalam penelitian yang dilakukan

BAB I PENDAHULUAN

Bab ini memuat penjelasan awal yang menjadi landasan dilaksanakannya penelitian. Pembahasan pada bab ini meliputi latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, Batasan penelitian, serta sistematika penulisan yang digunakan sebagai panduan dalam penyusunan skripsi.

BAB II TINJAUAN PUSTAKA

Bab ini berisi kajian literatur dan penelitian terdahulu yang relevan dengan topik penelitian. Selain itu, bab ini juga membahas landasan teori yang

mendukung pemahaman terhadap konsep penjadwalan produksi, *flexible flow shop scheduling*, dan metode optimasi yang digunakan. Kajian literatur ini digunakan untuk mengidentifikasi *research gap* dan memperkuat posisi penelitian yang dilakukan.

BAB III METODE PENELITIAN

Bab ini menjelaskan metode penelitian yang digunakan, meliputi jenis dan pendekatan penelitian, objek dan subjek penelitian, data yang digunakan, asumsi model, serta tahapan penelitian. Selain itu, pada bab ini juga dipaparkan pemodelan sistem produksi dan metode penyelesaian yang diterapkan untuk mencapai tujuan penelitian.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini membahas proses pengumpulan data yang digunakan dalam penelitian serta tahapan pengolahan data berdasarkan metode yang telah ditetapkan. Hasil pengolahan data disajikan dalam bentuk tabel, grafik, dan ilustrasi yang mendukung analisis penjadwalan produksi.

BAB V PEMBAHASAN

Bab ini berisi analisis dan pembahasan terhadap hasil pengolahan data yang telah diperoleh pada bab sebelumnya. Pembahasan difokuskan pada interpretasi hasil penelitian serta keterkaitannya dengan tujuan dan rumusan masalah penelitian.

BAB VI PENUTUPAN

Bab ini memuat Kesimpulan yang diperoleh dari hasil penelitian serta saran yang dapat diberikan untuk pengembangan penelitian selanjutnya. Kesimpulan disusun berdasarkan hasil analisis dan pembahasan, sedangkan saran diberikan dengan mempertimbangkan keterbatasan penelitian yang telah dilakukan.

DAFTAR PUSTAKA Bagian ini memuat seluruh sumber referensi yang digunakan dalam penyusunan skripsi, baik berupa buku, jurnal ilmiah, maupun sumber lain yang relevan dan mendukung penelitian.

LAMPIRAN Bagian lampiran berisi dokumen pendukung penelitian, seperti data tambahan, perhitungan, tabel, dan gambar yang diperlukan untuk melengkapi pembahasan dalam skripsi.

BAB II

TINJAUAN PUSTAKA

2.1 Kajian Literatur

Kajian literatur dalam penelitian ini disusun menggunakan pendekatan induktif dan deduktif. Pendekatan induktif dilakukan dengan menelaah berbagai penelitian terdahulu secara umum untuk memperoleh gambaran, perkembangan topik, pola penelitian, serta metode yang banyak digunakan. Selanjutnya, pendekatan deduktif digunakan untuk mengerucutkan pembahasan dari konsep dan temuan yang bersifat umum menuju kajian yang lebih spesifik. Melalui kedua pendekatan tersebut, kajian literatur diharapkan mampu memberikan pemahaman yang sistematis dan terstruktur sebagai dasar dalam pembahasan selanjutnya. Selain itu, pada kajian literatur berikut ini disusun untuk mengidentifikasi perkembangan pendekatan penyelesaian masalah penjadwalan produksi, khususnya pada lingkungan *flow shop*, *flexible flow shop*, dan sistem sejenis yang memiliki tingkat kompleksitas tinggi.

2.1.1 Perkembangan Metode Penjadwalan Produksi Berbasis *Evolutionary Algorithm*

Pada tahap awal perkembangan, Khurshid et al. (2021) menunjukkan bahwa *evolutionary strategy* mampu memberikan penurunan *makespan* yang signifikan pada *permutation flow shop scheduling problem*. Hasil ini menegaskan bahwa pendekatan evolusioner efektif dalam menangani ruang solusi yang besar. Namun, penelitian tersebut masih terbatas pada struktur *flow shop* klasik yang tidak mempertimbangkan fleksibilitas mesin pada setiap *stage*. Dengan demikian, meskipun metode yang digunakan sudah kuat, model yang dikembangkan belum sepenuhnya merepresentasikan kondisi sistem produksi nyata.

Perkembangan metode kemudian bergerak ke arah integrasi algoritma untuk meningkatkan kualitas solusi. Saiful et al. (2022) mengusulkan pendekatan *hybrid genetic algorithm* dan *tabu search* yang mampu meningkatkan performa penjadwalan dengan mengombinasikan pencarian global dan lokal. Jika dibandingkan dengan penelitian menurut Khurshid et al. (2021), metode ini menunjukkan peningkatan kualitas solusi yang lebih stabil. Namun, kelemahan dari pendekatan ini adalah meningkatnya kompleksitas algoritma serta kebutuhan parameter tuning yang lebih rumit. Hal ini menunjukkan bahwa peningkatan

performa metode sering kali diikuti dengan peningkatan kompleksitas yang dapat mengurangi kemudahan implementasi.

2.1.2 Penjadwalan pada Sistem *Multi-Stage* dan *Flexible Flow Shop*

Perkembangan penelitian selanjutnya mengarah pada sistem penjadwalan yang lebih kompleks dan lebih mendekati kondisi nyata, yaitu sistem yang memiliki beberapa tahap proses dan mesin paralel.

(Huang et al., 2024) mengembangkan *co-evolutionary algorithm* untuk *two-stage hybrid flow shop* dengan mempertimbangkan ketidakpastian. Penelitian ini menunjukkan bahwa integrasi aspek ketidakpastian mampu meningkatkan kualitas solusi secara signifikan dibandingkan pendekatan deterministik. Namun, kelemahan utama dari pendekatan ini adalah kompleksitas model yang tinggi serta kebutuhan komputasi yang besar, sehingga kurang praktis untuk diterapkan pada penelitian skala terbatas.

(Song et al., 2025) juga menunjukkan bahwa integrasi *evolutionary algorithm* dengan *local search* mampu meningkatkan kualitas solusi pada sistem manufaktur yang kompleks. Berbeda dengan Huang et al., penelitian ini lebih menekankan efisiensi solusi daripada aspek ketidakpastian. Meskipun demikian, kedua penelitian tersebut memiliki kesamaan dalam hal meningkatnya kompleksitas metode, yang berpotensi mengurangi keterterapan praktisnya.

(Y. Wang et al., 2022) menunjukkan bahwa *hybrid metaheuristic* pada *flexible flow shop* mampu menurunkan *makespan* secara signifikan. Penelitian ini lebih dekat dengan struktur sistem penelitian ini, namun masih memiliki keterbatasan karena mempertimbangkan kondisi khusus seperti *blocking*, sehingga tidak sepenuhnya mewakili kondisi umum sistem produksi.

2.1.3 Perkembangan pada *Flexible Job Shop* dan Sistem yang Lebih Fleksibel

Selain *flow shop*, perkembangan penelitian juga banyak mengarah pada *flexible job shop scheduling* yang memiliki tingkat fleksibilitas lebih tinggi, khususnya dalam pemilihan rute proses dan alokasi mesin. Fleksibilitas ini memungkinkan sistem untuk memilih mesin yang berbeda pada setiap tahap, sehingga memberikan ruang optimasi yang lebih luas, namun sekaligus meningkatkan kompleksitas permasalahan secara signifikan.

(Xue et al., 2024) menunjukkan bahwa *multi-population genetic algorithm* mampu meningkatkan kualitas solusi secara signifikan pada sistem dengan mesin paralel. Pendekatan ini efektif dalam menjaga keberagaman solusi dan menghindari *premature convergence*. Namun, jika dibandingkan dengan sistem *flow shop*, fleksibilitas routing pada *job shop* membuat struktur permasalahan menjadi berbeda, sehingga hasil penelitian ini tidak dapat

langsung diterapkan pada sistem *multi-stage flexible flow shop* yang memiliki urutan proses tetap.

(Fan et al., 2022) mengembangkan *improved genetic algorithm* dengan integrasi *local search*, yang terbukti meningkatkan stabilitas dan kualitas solusi. Pendekatan ini menunjukkan bahwa kombinasi eksplorasi global dan eksploitasi lokal dapat memperbaiki performa algoritma. Akan tetapi, penelitian ini berfokus pada *total weighted tardiness*, sehingga orientasi optimasinya berbeda dengan penelitian ini yang berfokus pada *makespan*. Hal ini menunjukkan bahwa meskipun metode yang digunakan serupa, perbedaan *objective function* dapat menghasilkan arah optimasi yang berbeda.

(N. Chen et al., 2022) mengembangkan model *flexible job shop* dengan mempertimbangkan ketidakpastian waktu proses. Pendekatan ini memberikan kontribusi teoritis yang kuat dalam menangani sistem yang lebih realistis. Namun, kompleksitas model yang tinggi serta penggunaan pendekatan tambahan seperti *neighborhood search* menjadi kelemahan dalam konteks implementasi praktis, terutama pada penelitian dengan keterbatasan data dan komputasi.

Berdasarkan perbandingan tersebut, dapat disimpulkan bahwa pendekatan pada *flexible job shop* memberikan kontribusi penting dalam pengembangan metode optimasi, khususnya dalam meningkatkan kualitas solusi melalui berbagai variasi *genetic algorithm*. Namun, perbedaan struktur sistem, terutama pada fleksibilitas routing, membuat pendekatan tersebut tidak dapat langsung diterapkan pada *multi-stage flexible flow shop*. Oleh karena itu, penelitian ini mengambil pendekatan yang lebih terarah dengan tetap mempertahankan struktur *flow shop*, tetapi menambahkan fleksibilitas mesin untuk menjaga keseimbangan antara kompleksitas dan keterterapan metode.

2.1.4 Integrasi Metode Hybrid dan Pendekatan Lanjutan

Perkembangan terbaru dalam penelitian penjadwalan produksi menunjukkan bahwa *evolutionary algorithm* tidak lagi digunakan secara tunggal, tetapi mulai diintegrasikan dengan pendekatan yang lebih kompleks seperti pembelajaran mesin. Integrasi ini bertujuan untuk meningkatkan kemampuan algoritma dalam beradaptasi terhadap kondisi sistem yang dinamis serta memperbaiki kualitas solusi yang dihasilkan.

(Zhang et al., 2025) menggabungkan *genetic algorithm* dengan *deep reinforcement learning* untuk menangani sistem dinamis. Pendekatan ini menunjukkan performa yang tinggi karena mampu menyesuaikan keputusan penjadwalan secara lebih adaptif. Namun, kelemahan

utamanya terletak pada kebutuhan data yang besar, proses pelatihan model yang tidak sederhana, serta kompleksitas implementasi yang tinggi.

(Cheng et al., 2022) juga mengembangkan *Q-learning-based genetic algorithm* yang menunjukkan peningkatan kualitas Solusi pada sistem manufaktur yang kompleks. Jika dibandingkan dengan Zhang et al. (2025), pendekatan ini relatif lebih sederhana karena menggunakan mekanisme pembelajaran yang tidak sekompleks *deep reinforcement learning*. Meskipun demikian, metode ini tetap membutuhkan pemahaman tambahan terkait pembelajaran mesin, sehingga penerapannya masih lebih rumit dibanding pendekatan *genetic algorithm* konvensional.

Kedua penelitian tersebut menunjukkan bahwa arah perkembangan metode penjadwalan semakin mengarah pada pendekatan yang lebih adaptif dan cerdas. Namun, dalam konteks penelitian ini, pendekatan tersebut dinilai kurang sesuai karena kompleksitasnya terlalu tinggi untuk diterapkan secara praktis, terutama jika penelitian difokuskan pada sistem yang deterministik dan membutuhkan metode yang lebih sederhana namun tetap efektif.

2.1.5 Perbandingan Berdasarkan *Objective Function*

Penelitian terdahulu menunjukkan bahwa pemilihan metode sangat bergantung pada objective function yang digunakan. Sebagian besar penelitian dalam penjadwalan produksi menggunakan *makespan* sebagai indikator utama karena berkaitan langsung dengan total waktu penyelesaian seluruh pekerjaan dalam sistem. Namun demikian, terdapat pula penelitian yang menggunakan indikator lain sesuai dengan kebutuhan sistem produksi yang dikaji.

(Risma & Utama, 2023) berfokus pada minimasi konsumsi energi, yaitu pada *no-idle permutation flow shop scheduling problem*. Penelitian ini menunjukkan bahwa penggunaan algoritma optimasi tertentu dapat meningkatkan efisiensi energi sistem produksi. Namun, fokus pada konsumsi energi menyebabkan aspek waktu penyelesaian produksi tidak menjadi prioritas utama, sehingga hasil yang diperoleh tidak dapat langsung digunakan untuk mengevaluasi efisiensi waktu produksi dalam konteks *makespan*.

(Gurusinga et al., 2025) berfokus pada lateness dan keseimbangan beban mesin. Lalu, mengkaji permasalahan *parallel machine scheduling* dengan tujuan meminimalkan *lateness* dan meningkatkan keseimbangan beban mesin. Pendekatan ini lebih menekankan pada ketepatan waktu penyelesaian pekerjaan terhadap *due date* serta distribusi beban kerja yang merata antar mesin. Jika dibandingkan dengan penelitian yang berfokus pada *makespan*, pendekatan ini lebih relevan untuk sistem produksi yang memiliki tekanan tinggi terhadap

keterlambatan pengiriman. Namun, kelemahannya adalah tidak secara langsung mengoptimalkan total waktu penyelesaian produksi, sehingga peningkatan performa pada aspek *lateness* belum tentu diikuti oleh penurunan *makespan*.

(Larasati & Akbar, 2024) menggunakan pendekatan eksak untuk meminimalkan *tardiness* dan *workload imbalance* pada sistem dengan keterbatasan sumber daya ganda. Penelitian ini menunjukkan bahwa keseimbangan beban kerja dan ketepatan waktu dapat ditingkatkan melalui model matematis yang terstruktur. Akan tetapi, pendekatan ini cenderung memiliki keterbatasan dalam hal skalabilitas ketika diterapkan pada sistem dengan ukuran besar atau kompleksitas tinggi, serta tidak secara langsung menargetkan minimasi *makespan* sebagai indikator utama efisiensi waktu produksi

Perbedaan fokus *objective function* tersebut menunjukkan bahwa hasil dari masing-masing penelitian tidak dapat dibandingkan secara langsung, karena setiap penelitian mengoptimalkan aspek kinerja yang berbeda. Sebagai contoh, suatu metode yang optimal dalam meminimalkan konsumsi energi belum tentu menghasilkan *makespan* yang minimum, demikian pula metode yang optimal dalam mengurangi *lateness* belum tentu menghasilkan utilisasi waktu produksi yang paling efisien.

Berdasarkan analisis tersebut, penelitian ini memilih *makespan* sebagai *objective function* utama karena indikator ini memiliki keterkaitan langsung dengan efisiensi waktu produksi secara keseluruhan. Minimasi *makespan* tidak hanya mencerminkan percepatan penyelesaian produksi, tetapi juga berimplikasi pada pengurangan *idle time*, peningkatan utilisasi mesin, serta peningkatan kapasitas produksi. Dengan demikian, fokus pada *makespan* dinilai lebih sesuai dengan tujuan penelitian yang ingin meningkatkan efisiensi sistem produksi pada lingkungan *multi-stage flexible flow shop*. Selain itu, penggunaan *makespan* juga mempermudah proses perbandingan dengan metode pembandingan seperti *heuristic scheduling* dan *genetic algorithm*, yang umumnya menggunakan indikator yang sama dalam evaluasi kinerja.

Berdasarkan kajian literatur yang telah dilakukan, penelitian ini memposisikan kondisi aktual perusahaan sebagai dasar pembandingan terhadap hasil optimasi menggunakan Microsoft Excel Solver metode Evolutionary. Dengan demikian, evaluasi hasil penelitian difokuskan pada perbandingan antara jadwal aktual perusahaan dan jadwal usulan hasil optimasi, terutama berdasarkan nilai *makespan*, perubahan urutan pekerjaan, perubahan alokasi mesin, setup time, dan *idle time* sebagai indikator pendukung.

2.2 Research Gap

Berdasarkan kajian literatur yang telah dilakukan dengan pendekatan induktif dan deduktif, dapat diidentifikasi adanya beberapa *research gap* pada penelitian penjadwalan produksi. Secara umum, penelitian terdahulu telah banyak mengkaji permasalahan penjadwalan menggunakan berbagai pendekatan *evolutionary algorithm* dan *metaheuristic* pada lingkungan *flow shop*, *permutation flow shop*, *flexible job shop*, maupun sistem dinamis dan stokastik. Namun, masih terbatas penelitian terapan yang memodelkan sistem produksi dengan karakteristik *multi-stage flexible flow shop*, mesin paralel pada setiap *work station*, *machine-dependent processing time*, *sequence-dependent setup time*, dan *release time*, kemudian membandingkan hasil optimasi dengan kondisi aktual perusahaan.

Selain itu, meskipun beberapa penelitian telah mengusulkan metode *hybrid* yang mengombinasikan *evolutionary algorithm* dengan *local search*, *tabu search*, atau *reinforcement learning*, penerapan pendekatan tersebut umumnya dilakukan pada sistem satu tahap, dua tahap, atau *job shop* dinamis, sehingga masih terbatas penelitian yang secara khusus mengkaji sistem *multi-stage flexible flow shop* sebagai objek utama penelitian.

Penelitian ini mengisi gap tersebut dengan menyusun model penjadwalan produksi pada perusahaan manufaktur elektronik yang memiliki 20 produk dan lima *work station* berurutan. Pada setiap *work station* tersedia dua hingga tiga mesin paralel, sehingga diperlukan keputusan terkait urutan pekerjaan dan pemilihan mesin. Penelitian ini juga mempertimbangkan *release time* sebagai waktu paling awal produk tersedia untuk diproses serta *sequence-dependent setup time* sebagai waktu persiapan yang bergantung pada urutan produk sebelumnya pada mesin yang sama. Selanjutnya, *Microsoft Excel Solver* metode *Evolutionary* digunakan untuk memperoleh kombinasi urutan pekerjaan dan pemilihan mesin yang dapat meminimalkan *makespan*. Hasil optimasi kemudian dibandingkan dengan kondisi aktual perusahaan untuk mengetahui besarnya perbaikan jadwal produksi yang diperoleh.

Tabel 2. 1 *Research Gap*

No	Peneliti	<i>Multi-stage</i>	<i>Flexible Machine</i>	<i>Evolutionary Algorithm</i>	<i>Ketidakpastian Kedatangan</i>	<i>Makespan</i>
1.	(Khurshid et al., 2021)	√		√		√
2.	(Saiful et al., 2022)			√		√
3.	(Huang et al., 2024)	√	√	√		√
4.	(Xue et al., 2024)		√	√		√
5.	(Zhang et al., 2025)		√	√		√
6.	(Song et al., 2025)			√		√
7.	(Cheng et al., 2022)			√		√
8.	(Fan et al., 2022)		√	√		√
9.	(Y. Wang et al., 2022)			√		√
10.	(N. Chen et al., 2022)		√	√		√
11.	(Risma & Utama, 2023)			√		
12.	(Gurusinga et al., 2025)		√	√		
13.	(Larasati & Akbar, 2024)		√			
14.	(Hatim & Ahmad, 2022)	√	√	√		√
15.	(Yusriski et al., 2023)					
16.	Diusulkan	√	√	√	√	√

2.3 Landasan Teori

Berikut ini merupakan landasan teori yang

2.3.1 Sistem Produksi Manufaktur

Sistem produksi manufaktur merupakan suatu sistem yang terdiri dari serangkaian aktivitas yang digunakan untuk mengubah bahan baku menjadi produk jadi melalui proses transformasi menggunakan sumber daya mesin, tenaga kerja, dan material. Dalam sistem manufaktur modern, pengelolaan aliran produksi menjadi aspek penting karena mempengaruhi efisiensi operasional, utilisasi mesin, serta kemampuan perusahaan dalam memenuhi permintaan pasar. Sistem produksi yang baik harus mampu mengkoordinasikan berbagai aktivitas produksi secara terintegrasi sehingga proses produksi dapat berjalan secara efektif dan efisien (Groover, 2019).

Dalam praktiknya, sistem produksi dapat memiliki karakteristik yang berbeda tergantung pada jenis produk, jumlah variasi produk, serta konfigurasi proses produksi. Pada industri manufaktur elektronik, sistem produksi sering melibatkan beberapa tahapan proses yang harus dilalui secara berurutan oleh setiap produk dengan variasi waktu proses pada setiap tahap. Kompleksitas sistem produksi tersebut menuntut adanya pengelolaan yang baik terhadap perencanaan dan pengendalian produksi dapat berjalan optimal (Groover, 2019).

Perkembangan teknologi industri dalam beberapa dekade terakhir telah mendorong perubahan signifikan dalam sistem produksi manufaktur, perusahaan tidak lagi hanya berfokus pada kemampuan produksi semata, tetapi juga pada efisiensi operasional, fleksibilitas produksi, serta kemampuan untuk merespons perubahan permintaan pasar dengan cepat. Oleh karena itu, sistem produksi modern banyak mengadopsi pendekatan manajemen produksi yang terintegrasi dengan teknologi informasi, otomatisasi, serta sistem pengendalian yang lebih canggih (Slack & Brandon-Jones, 2019b).

Dalam praktiknya, sistem produksi manufaktur dapat diklasifikasikan berdasarkan pola aliran proses produksi seperti *job shop*, *batch production*, *flow shop*, dan *continuous production*. Setiap jenis sistem produksi memiliki karakteristik yang berbeda dalam hal urutan proses, variasi produk, serta tingkat fleksibilitas produksi. Sistem produksi dengan aliran proses yang terstruktur, seperti *flow shop*, umumnya digunakan pada industri yang memproduksi produk dalam jumlah besar dengan urutan proses yang relatif tetap (Pinedo, 2022a).

2.3.2 *Production Planning and Inventory Control*

Production Planning and Inventory Control (PPIC) merupakan kegiatan merencanakan serta mengendalikan jalannya material yang mengalir dari kegiatan produksi, guna mencapai efisiensi biaya dan ketepatan waktu pengiriman barang ke konsumen (Nasution & Prasetyawan, 2008). Lalu PPIC juga berfungsi manajemen produksi yang berperan dalam merencanakan, mengoordinasikan, serta mengendalikan aktivitas produksi seperti perencanaan kapasitas, pengendalian persediaan, pengaturan aliran material, serta penjadwalan produksi, sehingga dapat memenuhi permintaan pelanggan secara efisien. Melalui pengelolaan yang baik, PPIC dapat membantu Perusahaan dalam memastikan bahwa sumber daya produksi digunakan secara optimal serta produk dapat diproduksi sesuai dengan jadwal yang telah ditetapkan (Jacobs et al., 2024).

Dalam sistem manufaktur modern, PPIC memiliki peran yang sangat penting karena berhubungan langsung dengan kinerja operasional perusahaan. Fungsi utama PPIC meliputi perencanaan produksi jangka panjang, perencanaan kebutuhan material, pengendalian persediaan, serta pengaturan penjadwalan produksi. Aktivitas tersebut bertujuan untuk memastikan bahwa proses produksi dapat berjalan secara lancar tanpa mengalami kekurangan material, kelebihan persediaan, maupun ketidakseimbangan kapasitas produksi (Jacobs et al., 2024).

Penjadwalan produksi merupakan salah satu aktivitas utama dalam PPIC yang memiliki peran penting dalam mengatur aliran pekerjaan dalam sistem produksi. Penjadwalan produksi bertujuan untuk menentukan urutan pengerjaan pekerjaan pada mesin serta menentukan waktu pelaksanaan setiap aktivitas produksi. Keputusan penjadwalan yang tepat dapat meningkatkan efisiensi sistem produksi, meminimalkan waktu tunggu pekerjaan, serta meningkatkan utilisasi mesin (Pinedo, 2022a).

Sebaliknya, penjadwalan produksi yang tidak direncanakan dengan baik dapat menimbulkan berbagai permasalahan operasional seperti penumpukan pekerjaan pada mesin tertentu, meningkatnya waktu tunggu pekerjaan, serta rendahnya tingkat pemanfaatan mesin. Kondisi tersebut dapat menyebabkan menurunnya kinerja sistem produksi serta meningkatkan biaya operasional perusahaan. Oleh karena itu, penerapan metode penjadwalan produksi yang sistematis menjadi sangat penting dalam mendukung fungsi PPIC dalam suatu perusahaan manufaktur.

Selain itu, PPIC juga berperan dalam memastikan bahwa kegiatan produksi dapat berjalan secara terkoordinasi dengan aktivitas lain dalam rantai pasok, seperti pengadaan material

dan distribusi produk. Integrasi antara perencanaan produksi, pengendalian persediaan, serta penjadwalan produksi memungkinkan Perusahaan untuk meningkatkan efisiensi operasional serta meningkatkan kemampuan perusahaan dalam memenuhi permintaan pelanggan secara tepat waktu.

2.3.3 Penjadwalan Produksi (*Production Scheduling*)

Penjadwalan produksi merupakan proses pengambilan keputusan untuk menentukan urutan pengerjaan pekerjaan (*jobs*) pada sejumlah sumber daya atau mesin dalam suatu horizon waktu tertentu dengan tujuan mengoptimalkan ukuran kinerja tertentu, seperti *makespan*, keterlambatan atau utilisasi mesin. Menurut (Pinedo, 2022a), mendefinisikan penjadwalan sebagai pengalokasian sumber daya terhadap aktivitas selama periode waktu tertentu dengan tujuan mengoptimalkan ukuran kinerja tertentu, seperti waktu penyelesaian, keterlambatan, atau utilisasi. Sejalan dengan itu, (Brucker, 2007) menjelaskan bahwa penjadwalan dapat dipandang sebagai masalah optimasi kombinatorial karena melibatkan pemilihan urutan/penugasan yang terbaik dari jumlah kemungkinan yang sangat besar.

Dalam kajian manajemen operasi, penjadwalan dibedakan dari perencanaan tingkat tinggi karena penjadwalan berada pada level keputusan operasional yang lebih detail, yakni “kapan” dan “di mana” pekerjaan di proses. Keputusan ini berdampak langsung pada performa sistem seperti waktu tunggu, tingkat keterlambatan, serta tingkat pemanfaatan mesin. Literatur penjadwalan juga menekankan bahwa lingkungan produksi memengaruhi struktur masalah, misalnya apakah hanya ada satu mesin, mesin paralel, aliran proses bertahap, atau lingkungan *job shop* yang rutennya berbeda-beda (Pinedo, 2022a).

Selain itu, sebagian besar masalah penjadwalan yang realistis bersifat sulit secara komputasi. (Brucker, 2007) menekankan bahwa banyak variasi masalah penjadwalan termasuk kategori *NP-hard*, sehingga untuk ukuran masalah menengah-besar, penyelesaian eksak sering tidak praktis. Karena itu, penelitian penjadwalan modern banyak mengembangkan pendekatan heuristik dan metaheuristik untuk memperoleh solusi yang baik dalam waktu komputasi yang masuk akal.

Berdasarkan definisi dan perspektif tersebut, penjadwalan produksi dapat disimpulkan sebagai proses pengambilan keputusan operasional untuk mengatur urutan serta alokasi sumber daya yang terbatas guna mengoptimalkan ukuran kinerja tertentu. Penjadwalan tidak hanya berkaitan dengan urutan pekerjaan, namun juga menyangkut

efisiensi sistem secara keseluruhan, dan secara komputasi sering menuntut pendekatan optimasi yang tepat agar layak diterapkan pada masalah berskala nyata.

2.3.4 Flow Shop Scheduling

Flow shop scheduling merupakan suatu metode penjadwalan produksi di mana setiap pekerjaan (*job*) harus melalui beberapa tahap proses atau mesin dengan urutan yang sama. Dalam sistem ini, semua pekerjaan mengikuti aliran proses yang identic dari tahap pertama hingga tahap terakhir, sehingga sistem produksi bersifat linier dan terstruktur (Pan, 2013). Permasalahan *flow shop scheduling* (FSP) pada dasarnya merupakan model teoretis yang digunakan untuk merepresentasikan kondisi nyata dalam sistem produksi. Dalam model ini, digunakan beberapa asumsi penyederhanaan agar permasalahan dapat dianalisis secara sistematis. Beberapa asumsi tersebut antara lain seluruh pekerjaan (*job*) dianggap saling independen dan sudah tersedia sejak awal proses (waktu 0), mesin selalu dalam kondisi siap digunakan, serta setiap pekerjaan pada suatu waktu hanya berada dalam kondisi menunggu atau sedang diproses. Selain itu, setiap mesin hanya dapat mengerjakan satu pekerjaan pada satu waktu..

Berdasarkan informasi yang didapatkan dari jurnal (Pan, 2013), Solusi dari permasalahan FSP berupa penentuan urutan pengerjaan seluruh pekerjaan dalam bentuk jadwal produksi yang bertujuan untuk mengoptimalkan kriteria tertentu. Dalam penjadwalan produksi, kriteria optimasi umumnya berkaitan dengan waktu penyelesaian pekerjaan (*completion time*), yang dilambangkan sebagai C_j . Waktu penyelesaian suatu pekerjaan pada mesin ke- i dinotasikan sebagai $C_{i,j}$, sedangkan waktu penyelesaian pekerjaan pada tahap akhir atau mesin terakhir dinyatakan sebagai C_j . Parameter ini menjadi dasar dalam mengevaluasi performansi suatu jadwal produksi.

Selain itu, salah satu kriteria optimasi yang paling umum digunakan dalam flow shop scheduling adalah minimisasi *makespan* atau C_{max} , yaitu waktu total yang dibutuhkan untuk menyelesaikan seluruh pekerjaan dalam sistem produksi. Minimasi *makespan* menjadi sangat penting terutama pada kondisi di mana sejumlah pekerjaan harus diselesaikan dalam waktu sesingkat mungkin, seperti pada pesanan yang terdiri dari beberapa produk dari satu pelanggan yang harus dikirim secara bersamaan. Selain itu, penurunan nilai *makespan* juga berdampak pada peningkatan efisiensi sistem produksi karena dapat meningkatkan tingkat pemanfaatan mesin serta mengurangi waktu menganggur (*idle time*).

Dengan demikian, *flow shop scheduling* menjadi dasar penting dalam penelitian penjadwalan produksi karena dapat digunakan untuk merepresentasikan sistem produksi yang memiliki alur proses berurutan. Pemahaman terhadap model ini juga menjadi landasan dalam pengembangan model yang lebih kompleks, seperti flexible flow shop scheduling yang digunakan dalam penelitian ini.

2.3.5 Flexible Flow Shop Scheduling (FFSP)

Flexible Flow Shop Scheduling (FFSP) merupakan model penjadwalan produksi yang menggabungkan karakteristik *flow shop* dan mesin parallel, di mana setiap *job* di proses melalui beberapa tahap secara berurutan dengan fleksibilitas pemilihan mesin pada setiap tahap (Xu et al., 2026). Di mana model ini mampu merepresentasikan sistem produksimodern yang kompleks, terutama dalam menangani variasi produk, alur proses yang beragam, serta keterbatasan sumber daya..

Perbedaan utama antara flow shop dan flexible flow shop terletak pada adanya keputusan tambahan dalam pemilihan mesin. Jika pada flow shop hanya ditentukan urutan *job*, maka pada FFSS harus ditentukan dua hal sekaligus, yaitu urutan *job* dan alokasi *job* ke mesin. Hal ini menyebabkan ruang solusi menjadi jauh lebih besar dan kompleks dibandingkan flow shop konvensional (Brucker, 2007).

Dalam pengembangannya, model *Flexible Flow Shop Scheduling* tidak hanya mempertimbangkan waktu proses, tetapi juga aspek lain yang memengaruhi kinerja sistem produksi, salah satunya Adalah waktu tunggu antar tahap (*inter-stage waiting time*). Berdasarkan ada atau tidaknya waktu tunggu antar tahap (*inter-stage waiting time*), sistem produksi dapat dibagi menjadi dua jenis (Xu et al., 2026) (B. Wang et al., 2018), yaitu.

1. **Nonactive-wait (Kontinu)** : Pekerjaan harus langsung berpindah ke tahap berikutnya setelah menyelesaikan tahap sebelumnya tanpa adanya penundaan
2. **Active-wait (Diskrit)** : Pekerjaan harus menunggu dalam rentang waktu tertentu sebelum masuk ke tahap berikutnya.

Waktu tunggu aktif ini memiliki batas minimum dan maksimum (*waiting time window*). Berbeda dengan waktu tunggu akibat hambatan produksi (*blocking*), waktu tunggu aktif sengaja dirancang untuk memenuhi kebutuhan proses. Waktu tunggu ini sangat penting karena dapat memengaruhi kualitas produk. Misalnya pada proses fermentasi makanan

atau pengolahan farmasi, waktu tunggu yang terlalu singkat atau terlalu lama dapat menurunkan kualitas produk bahkan menyebabkan cacat.

2.3.6 *Multi-Stage Flesible Flow Shop*

Sistem manufaktur *Multistage Flexible Flow Shop*, yang dalam berbagai literatur juga sering disebut *Hybrid Flow Shop*, adalah tata letak produksi yang mengintegrasikan struktur *flow shop* konvensional dengan lingkungan mesin paralel (Risma & Utama, 2023). Dalam sistem ini, proses produksi terdiri dari sejumlah tahap pengerjaan (k) secara berurutan, di mana syarat mutlak adalah $k \geq 2$ (Hidri, 2023). Sebuah himpunan pekerjaan atau *jobs* berukuran n wajib diproses melalui setiap tahap tersebut secara sekuensial, mulai dari tahap pertama hingga tahap terakhir (Gong et al., 2023).

Karakteristik utama yang membuat sistem ini disebut "fleksibel" adalah adanya minimal satu tahap pengerjaan yang memiliki lebih dari satu mesin paralel (Han et al., 2019). Secara lebih rinci, karakteristik utama *multi stage flexible flow shop* meliputi:

- *Routing* Terbatas namun Fleksibel
Setiap *job* wajib melewati urutan tahapan yang baku, namun pada tahap yang memiliki mesin paralel, sistem harus mengalokasikan *job* tersebut ke salah satu mesin yang tersedia (Han et al., 2019).
- Kategorisasi Mesin Paralel
Mesin-mesin yang berada dalam satu tahap yang sama dapat dikategorikan menjadi mesin identik, mesin dengan kecepatan seragam (*uniform*), atau mesin yang sama sekali tidak terkait (*unrelated*) satu sama lain (Risma & Utama, 2023).
- Kompleksitas *NP-Hard*
Secara matematis, masalah penjadwalan pada *multi stage flexible flow shop* diklasifikasikan sebagai optimasi kombinatorial Tingkat *NP-Hard* karena menuntut dua Keputusan krusial secara simultan, penentuan alokasi mesin pada setiap tahap dan penentuan urutan pengerjaan *job* di setiap mesin (Han et al., 2019; Hidri, 2023).

2.3.7 *Permutation Flexible Flow Shop*

Permutation Flexible Flow Shop (PFFS) merupakan sebuah varian spesifik dari masalah penjadwalan produksi *Hybrid Flow Shop*. Menurut (Ruiz & Vázquez-Rodríguez, 2010), karakteristik utama dari PFFS adalah adanya kombinasi antara stasiun kerja paralel (*flexible*) dengan batasan urutan yang mengikat (*permutation*). Dalam sistem ini, proses produksi terdiri dari beberapa stasiun kerja berturutan di mana tiap stasiun memiliki satu

atau lebih mesin paralel. Meskipun setiap pekerjaan (*job*) dapat dialokasikan pada mesin paralel yang berbeda di setiap stasiunnya, urutan relatif pengerjaan antar *job* harus tetap sama atau konsisten dari stasiun pertama hingga stasiun terakhir.

Penerapan batasan permutasi ini sangat umum ditemukan pada manufaktur riil karena mampu menyederhanakan manajemen logistik di rantai produksi serta meminimalkan penumpukan barang dalam proses (*Work-in-Process*) akibat keterbatasan area penyimpanan sementara antar stasiun kerja (Pinedo, 2016). Dalam pemodelan optimasi, sifat permutasi ini membatasi ruang pencarian solusi (*search space*), sehingga algoritma metaheuristik dapat difokuskan untuk mencari satu urutan permutasi terbaik yang dapat meminimalkan total waktu penyelesaian (*makespan*).

2.3.8 Mesin Paralel dalam Sistem Produksi

Mesin paralel merupakan kondisi dalam sistem produksi di mana terdapat lebih dari satu mesin yang memiliki fungsi serupa dan dapat digunakan secara bersamaan untuk memproses pekerjaan. Keberadaan mesin paralel memberikan fleksibilitas dalam sistem produksi karena memungkinkan pekerjaan dialokasikan ke beberapa mesin secara simultan, sehingga kapasitas produksi dapat ditingkatkan. Dalam lingkungan ini, suatu pekerjaan dapat diproses oleh lebih dari satu mesin, sehingga diperlukan keputusan penugasan yang tepat untuk mencapai tujuan penjadwalan yang diinginkan (Kim & Glock, 2018).

Dalam sistem produksi *multi-stage*, penggunaan mesin paralel menjadi salah satu strategi untuk meningkatkan efisiensi dan respons terhadap permintaan. Sistem dengan beberapa mesin memungkinkan perusahaan untuk menyesuaikan kapasitas produksi terhadap kebutuhan, serta memberikan fleksibilitas dalam pengoperasian mesin. Namun demikian, kondisi ini juga menimbulkan tantangan tambahan, karena perusahaan harus menentukan mesin mana yang digunakan serta bagaimana mesin tersebut dioperasikan untuk memenuhi permintaan secara optimal.

Salah satu permasalahan utama dalam sistem mesin paralel adalah penentuan alokasi pekerjaan ke mesin yang tersedia. Ketidaktepatan dalam distribusi pekerjaan dapat menyebabkan ketidakseimbangan beban kerja, di mana sebagian mesin mengalami kelebihan beban, sementara mesin lainnya tidak dimanfaatkan secara optimal. Kondisi ini dapat menurunkan efisiensi sistem produksi secara keseluruhan. Oleh karena itu, diperlukan pendekatan penjadwalan yang mampu mendistribusikan pekerjaan secara merata untuk meningkatkan utilisasi mesin.

Selain itu, variasi waktu proses antar pekerjaan juga menjadi faktor penting dalam pengambilan keputusan penjadwalan. Pekerjaan dengan waktu proses yang lebih lama berpotensi menyebabkan penumpukan pada mesin tertentu apabila tidak dialokasikan dengan baik. Dengan demikian, proses penjadwalan harus mempertimbangkan kombinasi antara waktu proses, kapasitas mesin, serta strategi distribusi pekerjaan agar sistem produksi dapat berjalan secara optimal.

Dengan demikian, keberadaan mesin paralel memberikan keuntungan dalam hal fleksibilitas dan peningkatan kapasitas produksi, namun di sisi lain juga meningkatkan kompleksitas pengambilan keputusan dalam penjadwalan. Oleh karena itu, diperlukan metode optimasi yang tepat untuk mengelola sistem produksi dengan mesin paralel secara efektif.

2.3.9 Machine-Dependent Processing Time

Dalam pemodelan sistem manufaktur yang fleksibel, estimasi waktu proses tidak selalu bersifat seragam pada setiap fasilitas. Menurut (Pinedo, 2016), *Machine-dependent processing time* merupakan karakteristik utama yang ditemukan pada lingkungan stasiun kerja paralel yang tidak seragam (*unrelated parallel machines*). Pada kondisi ini, waktu pemrosesan suatu pekerjaan (*job*) sepenuhnya bergantung pada mesin spesifik mana pekerjaan tersebut dialokasikan.

Secara matematis, jika pekerjaan j diproses pada mesin i , maka waktu prosesnya dinyatakan sebagai p_{ij} . Variasi nilai p_{ij} ini disebabkan oleh adanya perbedaan teknologi, kapasitas, usia teknis, maupun kecepatan operasional antar mesin paralel di stasiun kerja tersebut. Dalam perhitungan total waktu penyelesaian (*makespan*), algoritma optimasi harus mampu memetakan penugasan mesin secara tepat agar mesin dengan kemampuan lebih tinggi dapat dimanfaatkan secara maksimal guna meminimalkan akumulasi waktu tunggu.

2.3.10 Sequence-Dependent Setup Time (SDST)

Keberhasilan penjadwalan di rantai produksi tidak hanya ditentukan oleh durasi proses produksinya saja, melainkan juga oleh waktu persiapan (*setup*) mesin sebelum pekerjaan dimulai. Menurut (Pinedo, 2016), kondisi yang dinamakan *Sequence-Dependent Setup Time* (SDST) terjadi ketika waktu penyetelan awal sebuah pekerjaan sangat dipengaruhi oleh jenis pekerjaan yang diselesaikan tepat sebelumnya pada mesin yang sama.

Secara matematis, jika pekerjaan j langsung dikerjakan setelah pekerjaan k selesai, maka waktu persiapannya ditulis sebagai nilai matriks s_{kj} . SDST ini sangat sering

dijumpai pada industri nyata seperti manufaktur elektronik atau kimia. Sebagai contoh, jika urutan pengerjaan tidak diatur dengan baik, mesin akan terlalu lama menganggur (*idle*) hanya untuk bongkar-pasang alat atau membersihkan sisa komponen dari pekerjaan sebelumnya. Oleh karena itu, mengabaikan faktor SDST dalam pemodelan jadwal akan menghasilkan *output* yang tidak realistis saat diterapkan di lapangan. Agar hasil perhitungan *makespan* C_{max} benar-benar optimal dan akurat, algoritma metaheuristik yang digunakan wajib memperhitungkan urutan penugasan mesin sekaligus runtunan waktu pengesetan ini.

2.3.11 *Makespan* sebagai Kriteria Optimasi

Makespan merupakan salah satu ukuran performansi yang paling umum digunakan dalam penjadwalan produksi. *Makespan* dapat didefinisikan sebagai waktu antara dimulai tugas pertama hingga selesainya tugas akhir atau total waktu yang dibutuhkan sistem untuk menyelesaikan seluruh tugas berdasarkan jadwal tertentu (Madraki et al., n.d.). Lalu, *makespan* juga merupakan ukuran efisiensi produksi dan didefinisikan sebagai waktu yang berlalu sebelum penyelesaian serangkaian pekerjaan pada sejumlah mesin tertentu (Y.-Y. Chen et al., 2021).

Secara matematis *makespan* umumnya dinotasikan sebagai C_{max} , yaitu *completion time maximum* atau waktu penyelesaian terbesar dari seluruh pekerjaan yang dijadwalkan. Apabila C_j menyatakan waktu selesai dari pekerjaan ke- j , maka nilai *makespan* diperoleh dari waktu penyelesaian paling akhir di antara seluruh pekerjaan. Dengan kata lain, *makespan* menunjukkan kapan seluruh *job* dalam sistem produksi selesai diproses. Oleh karena itu, dalam permasalahan penjadwalan produksi, tujuan utama yang sering digunakan adalah meminimalkan C_{max} agar total waktu penyelesaian produksi menjadi lebih singkat dan efisien (Tremblet et al., 2022). Rumusnya dapat dituliskan sebagai berikut:

$$C_{max} = \max(C_1, C_2, \dots, C_n) \quad (2.1)$$

Artinya, apabila terdapat beberapa pekerjaan dengan waktu penyelesaian yang berbeda-beda, maka *makespan* diambil dari pekerjaan yang selesai paling terakhir.

Minimasi *makespan* menjadi tujuan utama dalam banyak penelitian penjadwalan karena nilai *makespan* lebih kecil menunjukkan bahwa sistem produksi mampu menyelesaikan seluruh pekerjaan dalam waktu yang lebih singkat. Hal ini berpengaruh langsung terhadap efisiensi produksi serta kemampuan perusahaan dalam memenuhi permintaan pelanggan tepat waktu.

2.3.12 *Idle time* Mesin

Idle time (waktu menganggur) adalah interval waktu di mana sumber daya (seperti mesin atau tenaga kerja) dalam kondisi siap dan tersedia untuk bekerja, namun tidak melakukan kegiatan produktif atau tidak dapat menyelesaikan pekerjaan mereka karena alasan diluar kendali mereka. Hal ini biasanya terjadi karena adanya hambatan eksternal seperti menunggu bahan baku, kesalahan teknis, atau ketidakefisienan dalam penjadwalan (Zeschke & Zacher, 2023).

Secara matematis, apabila $S_{i,j}$ adalah waktu mulai pekerjaan ke- j pada mesin ke- i , dan $C_{i,j-1}$ adalah waktu selesai pekerjaan sebelumnya ($j - 1$) pada mesin yang sama, maka *idle time* dapat dirumuskan sebagai berikut:

$$IT_{i,j} = S_{i,j} - C_{i,j-1} \quad (2.2)$$

Persamaan tersebut menunjukkan bahwa *idle time* terjadi ketika terdapat selisih waktu antara selesainya pekerjaan sebelumnya dan dimulainya pekerjaan berikutnya. Semakin besar nilai *idle time*, maka semakin besar pula waktu mesin tidak produktif. Oleh karena itu, dalam penjadwalan produksi, *idle time* perlu diminimalkan agar pemanfaatan mesin menjadi lebih optimal dan aliran proses produksi dapat berjalan lebih efisien.

Berdasarkan literatur yang dikutip dari (Kanet & Sridharan, 2000), klasifikasi *idle time* digunakan untuk membedakan penyebab terjadinya waktu mengangsur dalam sistem produksi. Secara umum *idle time* dapat terbagi menjadi dua, yaitu:

1. Normal *Idle time*: Waktu menganggur yang sudah direncanakan (misalnya: waktu istirahat, perawatan rutin, *set-up* mesin, atau pergantian *shift*). Jenis *idle time* ini masih dapat diterima karena menjadi bagian dari kebutuhan operasional perusahaan.
2. Abnormal *Idle time*: Waktu menganggur yang tidak terencana (misalnya: kerusakan mesin mendadak, keterlambatan operator, atau belum tersedianya pekerjaan yang siap diproses. atau keterlambatan pasokan material). Abnormal *idle time* perlu menjadi perhatian karena dapat menurunkan produktivitas, memperpanjang waktu penyelesaian produksi, dan menyebabkan pemanfaatan mesin menjadi tidak optimal.

2.3.13 *Evolutionary Algorithm*

Algoritma Evolusioner (EA) merupakan kelas algoritma optimasi berbasis populasi yang bekerja berdasarkan prinsip-prinsip evolusi biologis seperti seleksi alam, rekombinasi, dan mutasi. Menurut (Eiben & Smith, 2015) dalam buku *Introduction to Evolutionary Computing*, EA beroperasi dengan memelihara sekumpulan solusi kandidat (populasi) yang secara iteratif diperbaiki melalui proses seleksi berdasarkan nilai kesesuaian (*fitness*

function). Individu yang memiliki nilai *fitness* lebih baik memiliki peluang lebih besar untuk bertahan hidup dan mewariskan informasi genetiknya ke generasi berikutnya.

Secara umum, keluarga besar EA biasanya dibagi menjadi empat paradigma utama yang berkembang secara historis:

1. *Genetic Algorithms* (GA): merupakan salah satu paradigma EA yang banyak digunakan untuk permasalahan optimasi. GA umumnya merepresentasikan solusi dalam bentuk kromosom dan menggunakan operator seperti seleksi, *crossover*, dan mutasi untuk menghasilkan solusi baru.
2. *Evolution Strategies* (ES): Fokus pada optimasi parameter numerik atau kontinu, dengan penekanan utama pada proses mutasi dan adaptasi parameter.
3. *Genetic Programming* (GP): Evolusi untuk menghasilkan program komputer atau ekspresi matematika.
4. *Evolutionary Programming* (EP): Mirip dengan ES, namun awalnya difokuskan pada evolusi *Finite State Machines*.

Secara umum, sebuah algoritma evolusioner diawali dengan pembentukan populasi awal secara acak, yang kemudian diperbaiki melalui penggunaan operator pusat, yaitu: seleksi, *crossover* (rekombinasi), dan mutasi (Eiben & Smith, 2015). Proses ini mengikuti beberapa tahapan kunci sebagai berikut (Dasgupta & Michalewicz, 1997) :

1. Seleksi dan *Fitness*: Individu dalam populasi yang memiliki nilai *fitness* terbaik *diprioritaskan* untuk dipertahankan dan disebar ke generasi berikutnya. Hal ini memastikan bahwa fitur-fitur solusi yang menjanjikan tetap terjaga dan diperkuat dalam populasi.
2. *Crossover*: Operator ini memungkinkan pertukaran atribut antara induk (*parents*) yang terpilih. Perannya sangat krusial dalam menyeimbangkan antara eksplorasi ruang solusi (*exploration*) serta pemanfaatan area solusi yang berkualitas baik (*exploitation*).
3. Mutasi: Berfungsi untuk menyuntikkan perubahan acak ke dalam populasi. Hal ini bertujuan untuk mencegah konvergensi berlebihan dan menghindari terjebak algoritma pada solusi optimum lokal atau konvergensi prematur.

2.3.14 Excel Solver Evolutionary sebagai Implementasi Metode Optimasi

Optimasi merupakan proses komputasi matematis yang esensial untuk menemukan solusi terbaik berdasarkan sekumpulan batasan operasional yang telah ditetapkan (Goldberg, 2013). Dalam lingkungan *spreadsheet*, Microsoft mengintegrasikan fitur *Add-in Solver*

yang secara efektif mengonversi relasi fungsi-fungsi sel Excel menjadi model matematika formal untuk memfasilitasi pengambilan keputusan manajerial (Fylstra et al., 1998). Meskipun mesin standar pada perangkat *Solver* tersebut seperti *Generalized Reduced Gradient* (GRG) sangat mampu untuk menyelesaikan model non-linear yang berkesinambungan (kontinu), metode kalkulus ini dipastikan akan mengalami kegagalan kalkulasi saat menghadapi fungsi diskontinu di Excel seperti penggunaan rumus logika IF, CHOOSE, dan penelusuran VLOOKUP. Untuk memecahkan kebuntuan masalah fungsi yang rumit dan tidak mulus (*non-smooth*) tersebut, Excel mengimplementasikan mesin algoritma *Evolutionary* (Fylstra et al., 1998).

Sebagai implementasi metode optimasi, pendekatan *Evolutionary* di dalam Excel *Solver* bekerja berdasarkan prinsip Algoritma Evolusioner (EA) dan Algoritma Genetika (GA) yang terinspirasi dari teori evolusi biologis dan seleksi alam (Goldberg, 2013). Berbeda dengan algoritma aljabar tradisional yang mencari solusi statis dari satu titik ke titik lain melalui perhitungan turunan gradien, metode *Evolutionary* mencari solusi secara probabilistik melalui sebuah "populasi" atau sekumpulan kandidat solusi (Frontline Systems, 2024). Dalam implementasinya pada perangkat lunak, proses ini berjalan melalui siklus operasional berikut (Yu & Gen, 2021):

1. Inisialisasi

Solver membangkitkan secara acak kumpulan nilai awal pada *decision variables* (sel variabel yang dapat diubah) yang masih memenuhi *constraints* (batasan) model. Setiap kombinasi nilai ini direpresentasikan sebagai sebuah "kromosom"

2. Evaluasi *Fitness* (Target Objektif)

Algoritma menguji setiap kromosom ke dalam model spreadsheet untuk melihat hasil akhirnya pada sel target (objektif). Nilai sel target ini menjadi ukuran "kebugaran" (*fitness*); semakin mendekati tujuan (Maksimum/Minimum/Nilai tertentu), semakin tinggi nilai *fitness*-nya

3. Seleksi dan Pindah Silang (*Crossover*)

Solver memilih kombinasi variabel terbaik untuk dipertahankan, lalu menyilangkan nilai-nilai dari dua solusi terbaik ("induk") untuk menghasilkan solusi baru ("keturunan") pada iterasi berikutnya guna mengeksplorasi kombinasi yang unggul.

4. Mutasi (*Mutation*)

Untuk mencegah algoritma berhenti pada *local optima* (titik optimal lokal), *Solver* secara acak memutasi atau mengubah sedikit nilai variabel. Hal ini memungkinkan algoritma "melompat" keluar dari area sub-optimal dan mengeksplorasi area solusi baru secara global.

5. Terminasi (*Termination*)

Iterasi generasi akan terus berlanjut hingga solusi tidak lagi mengalami perbaikan yang signifikan (*convergence*), atau batas waktu komputasi maksimal yang diatur pengguna telah habis.

BAB III

METODE PENELITIAN

3.1 Subjek Penelitian

Subjek penelitian ini adalah PT XYZ, sebuah perusahaan yang bergerak di sektor industri elektronik, khususnya dalam produksi modul *transmitter* dan *receiver* elektronik. PT XYZ bertanggung jawab untuk memproduksi berbagai jenis modul yang digunakan, Perusahaan ini memproduksi sekitar 20 jenis produk yang digunakan dalam berbagai sistem elektronik dan otomasi.

Penelitian ini difokuskan pada aktivitas penjadwalan produksi di PT XYZ, yang melibatkan penentuan urutan pengerjaan produk (*job sequencing*) dan pemilihan mesin (*production routing*) pada setiap *work station* untuk meningkatkan efisiensi sistem produksi dan untuk meminimalkan waktu penyelesaian produksi (*makespan*). Penelitian ini akan mengoptimalkan *multi-stage flexible flow shop scheduling*, dengan tujuan meningkatkan efisiensi produksi dengan meminimasi waktu siklus yang dapat dilakukan dengan cara mengurangi *idle time* pada mesin di setiap *work station*.

3.2 Objek Penelitian

Objek penelitian ini adalah sistem penjadwalan produksi pada PT XYZ yang memiliki karakteristik *multi-stage flexible flow shop scheduling problem*. Dalam sistem ini, setiap produk (*job*) harus melewati beberapa *work station*, yang masing-masing memiliki 2-3 mesin paralel. Terdapat sekitar 20 jenis produk dengan urutan proses yang sama namun waktu proses yang berbeda pada setiap mesin.

Permasalahan utama dalam penelitian ini adalah apa kombinasi urutan pengerjaan (*Job assignment*) dan alokasi mesin (*machine assignment*) yang menghasilkan nilai *makespan* minimum dengan mempertimbangkan *release time* dan *setup time*

3.3 Jenis Data

Jenis data yang digunakan dalam penelitian ini adalah data kuantitatif yang mencakup parameter-parameter terkait sistem produksi di PT XYZ. Data yang digunakan meliputi:

1. Data Jumlah Produk (*Job*)

Penelitian ini menggunakan 20 jenis produk yang diproduksi oleh PT XYZ. Setiap produk direpresentasikan sebagai satu *job* dalam sistem penjadwalan. Berikut ini merupakan daftar produk yang digunakan dalam penelitian ini.

2. Data Waktu Proses

Waktu yang dibutuhkan setiap produk pada setiap *work station*. Waktu ini dapat bervariasi antar produk dan antar mesin pada setiap *stage*.

3. Data Waktu *Setup*

Data waktu setup adalah data waktu persiapan yang dibutuhkan mesin sebelum memproses produk, terutama saat terjadi pergantian produk pada mesin yang sama.

4. Data Jumlah Mesin

Jumlah mesin yang tersedia pada setiap *work station* dalam proses produksi.

5. Data Urutan Proses

Urutan proses produksi yang harus dilalui oleh setiap produk dalam sistem *flow shop*.

6. Data *Release Time*

Data *release time* adalah data waktu paling awal suatu pekerjaan atau produk tersedia dan dapat mulai diproses dalam sistem produksi.

3.4 Komponen Model

3.4.1 Notasi Model

Berikut merupakan notasi model yang digunakan dalam penelitian ini.

a. Indeks

i : indeks pekerjaan/*job*, $i = 1, 2, 3, \dots, n$

j : indeks pekerjaan sebelumnya pada mesin yang sama

k : indeks *work station*, $k = 1, 2, 3, \dots, K$

m : indeks mesin pada *work station*

r : indeks posisi urutan pengerjaan, $r = 1, 2, 3, \dots, n$

b. Parameter

n : jumlah pekerjaan/*job*

K : jumlah *work station*

J : Himpunan seluruh *job*

- M_k : himpunan mesin yang tersedia pada *work station* ke- k
 $Z_{i,r}$: Bernilai 1 jika *job* i berada pada urutan ke- r , dan bernilai 0 jika tidak
 $p_{i,k,m}$: waktu proses *job* i pada mesin m di *work station* k
 $S_{j,i}$: waktu setup dari *job* sebelumnya j menuju *job* i
 r_i : *release time* *job* i
 $C_{j,k}$: waktu selesai *job* sebelumnya j pada mesin yang sama di *work station* k

c. Variabel Keputusan

- π_r : *job* yang ditempatkan pada urutan pengerjaan ke- r
 $m_{i,k}$: mesin yang dipilih untuk memproses *job* i pada *work station* k

d. Variabel Perhitungan

- $S_{i,k}$: waktu mulai proses *job* i pada *work station* k
 $C_{i,k}$: waktu selesai *job* i pada *work station* k
 $C_{j,k}$: waktu selesai *previous job* pada *work station* k
 $IT_{i,k}$: *idle time* sebelum *job* i diproses pada *work station* k
 TIT_k : total *idle time* pada *work station* k
 TIT : total *idle time* seluruh *work station*
 C_{max} : *makespan* atau waktu penyelesaian seluruh pekerjaan

3.4.2 Variabel Keputusan dan Kriteria Performansi

Variabel keputusan dalam penelitian ini terdiri dari dua hal utama, yaitu urutan pengerjaan *job* (*job sequencing*) dan pemilihan mesin pada setiap *work station* (*production routing*). Urutan pengerjaan *job* digunakan untuk menentukan produk mana yang diproses terlebih dahulu, sedangkan *production routing* digunakan untuk menentukan mesin yang digunakan oleh setiap produk pada masing-masing *work station*.

Kriteria performansi yang digunakan dalam penelitian ini adalah *makespan*. *Makespan* merupakan waktu total yang dibutuhkan untuk menyelesaikan seluruh pekerjaan pada sistem produksi. Nilai *makespan* digunakan sebagai fungsi tujuan yang akan diminimalkan dalam proses optimasi.

3.4.3 Asumsi Model

Asumsi model yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Seluruh produk diproses melalui lima *work station* secara berurutan.
2. Setiap produk hanya dapat diproses oleh satu mesin pada setiap *work station*.

3. Setiap mesin hanya dapat memproses satu produk dalam satu waktu.
4. Proses produksi bersifat *non-preemptive*, sehingga pekerjaan yang sedang diproses tidak dapat dihentikan atau disela sebelum selesai.
5. Waktu proses setiap produk pada setiap mesin telah diketahui.
6. Waktu setup diperhitungkan apabila terjadi pergantian produk pada mesin yang sama.
7. Waktu setup bernilai nol apabila tidak terdapat pekerjaan sebelumnya pada mesin yang sama.
8. Setiap *job* memiliki *release time* masing-masing, sehingga *job* hanya dapat mulai diproses pada *work station* pertama setelah waktu rilisnya terpenuhi
9. Fungsi tujuan utama dalam penelitian ini adalah meminimalkan *makespan*.
10. *Idle time* digunakan sebagai indikator pendukung untuk menjelaskan waktu tunggu mesin antar pekerjaan, tetapi tidak menjadi fungsi tujuan utama.

3.5 Formulasi Model Penjadwalan

3.5.1 Fungsi Tujuan

Fungsi tujuan dalam penelitian ini adalah meminimalkan *makespan*. *Makespan* merupakan waktu penyelesaian seluruh pekerjaan pada *work station* terakhir. Semakin kecil nilai *makespan*, semakin cepat seluruh pekerjaan dapat diselesaikan. Untuk rumus *makespan* dapat dilihat pada rumus (2.1), dengan penyesuaian rumus berdasarkan permasalahan yang ada:

$$C_{max} = \max_{i \in J} C_{i,K}$$

Keterangan:

$C_{i,k}$: waktu selesai *job* i pada *work station* k

C_{max} : *makespan* atau waktu penyelesaian seluruh pekerjaan

J : Himpunan seluruh *job*

K : Jumlah *work station*

3.5.2 Waktu Selesai *Job*

Waktu selesai *job* dihitung berdasarkan waktu mulai proses dan waktu proses pada mesin yang dipilih.

$$C_{i,k} = S_{i,k} + p_{i,k,m} \quad (2.3)$$

Keterangan:

- $C_{i,k}$: waktu selesai *job i* pada *work station k*
 $S_{i,k}$: waktu mulai proses *job i* pada *work station k*
 $p_{i,k,m}$: waktu proses *job i* pada mesin *mdi work station k*

3.5.3 Waktu Mulai Proses dengan *Setup Time*

Waktu mulai proses ditentukan berdasarkan waktu ketersediaan *job*, waktu selesai *job* pada *work station* sebelumnya, waktu selesai *job* sebelumnya pada mesin yang sama, dan waktu setup.

Untuk *work station* pertama:

$$S_{i,1} = \max(r_i, C_{j,1}) + s_{j,i} \quad (2.4)$$

Untuk *work station* kedua hingga terakhir:

$$S_{i,k} = \max(C_{i,k-1}, C_{j,k}) + s_{j,i} \quad (2.5)$$

Keterangan:

- $S_{i,k}$: waktu mulai proses *job i* pada *work station k*
 r_i : *release time job i*
 $C_{i,k-1}$: waktu selesai *job i* pada *work station* sebelumnya
 $s_{j,i}$: waktu setup dari *job* sebelumnya *j* menuju *job i*

Apabila tidak terdapat *job* sebelumnya pada mesin yang sama, maka:

$$s_{j,i} = 0$$

dan:

$$C_{j,k} = 0$$

3.5.4 Rumus *Idle time*

Idle time terjadi apabila terdapat jeda antara waktu mesin selesai melakukan *setup* dan waktu mulai pekerjaan berikutnya. Dengan demikian, *setup time* tidak dihitung sebagai *idle time* karena *setup* merupakan aktivitas persiapan mesin sebelum proses produksi.

$$IT_{i,k} = S_{i,k} - (C_{j,k} + s_{j,i}) \quad (2.6)$$

Jika hasilnya negatif, gunakan 0:

$$T_{i,k} = \max(0, S_{i,k} - (C_{j,k} + s_{j,i})) \quad (2.7)$$

3.5.5 Batasan Urutan Proses

Setiap *job* hanya dapat diproses pada *work station* berikutnya setelah proses pada *work station* sebelumnya selesai.

$$S_{i,k} \geq C_{i,k-1} \quad (2.8)$$

Keterangan:

$S_{i,k}$: waktu mulai proses *job i* pada *work station k*

$C_{i,k-1}$: waktu selesai *job i* pada *work station* sebelumnya

3.5.6 Batasan Mesin

Setiap mesin hanya dapat memproses satu *job* dalam satu waktu. *Job* berikutnya pada mesin yang sama hanya dapat dimulai setelah *job* sebelumnya selesai dan waktu setup telah terpenuhi.

$$S_{i,k} \geq C_{j,k} + s_{ji} \quad (2.9)$$

Keterangan:

s_{ji} : waktu setup dari *job* sebelumnya *j* menuju *job i*

$C_{j,k}$: waktu selesai *job* sebelumnya *j* pada mesin yang sama di *work station k*

$S_{i,k}$: waktu mulai proses *job i* pada *work station k*

3.5.7 Batasan Setiap *Job* Dijadwalkan Satu Kali

Setiap *job* hanya boleh muncul satu kali dalam urutan penjadwalan. Batasan ini digunakan agar tidak ada *job* yang dijadwalkan lebih dari satu kali atau tidak dijadwalkan sama sekali.

$$\sum_{r=1}^n Z_{i,r} = 1 \quad (2.10)$$

Keterangan:

n : jumlah pekerjaan/*job*

$Z_{i,r}$: bernilai 1 apabila *job i* ditempatkan pada urutan ke-*r*, dan bernilai 0 apabila tidak

Pada pengolahan data menggunakan *Microsoft Excel*, batasan ini dikontrol melalui pengecekan jumlah kemunculan *job* menggunakan fungsi *COUNTIF*.

3.5.8 Batasan Pemilihan Mesin

Setiap *job* hanya dapat diproses oleh satu mesin pada setiap *work station*. Mesin yang dipilih harus termasuk dalam daftar mesin yang tersedia pada *work station* tersebut.

$$m_{i,k} \in M_k \quad (2.11)$$

Keterangan:

$m_{i,k}$: mesin yang dipilih untuk *job i* pada *work station k*

M_k : himpunan mesin yang tersedia pada *work station k*

Sebagai contoh, apabila *work station* 1 memiliki tiga mesin, maka:

$$M_1 = \{1,2,3\}$$

3.5.9 Persentase Perbaikan

$$\frac{C_{maxAktual} - C_{maxOptimasi Solver}}{C_{maxAktual}} \times 100\% \quad (2.12)$$

3.6 Algoritma Pemecahan Masalah

Algoritma pemecahan masalah merupakan tahapan sistematis yang digunakan untuk menyelesaikan permasalahan penjadwalan produksi pada penelitian ini. Pengolahan data dilakukan dengan bantuan Microsoft Excel dan fitur *Solver* menggunakan metode *Evolutionary Algorithm*. Tahapan pengolahan data dalam penelitian ini adalah sebagai berikut.

Tabel 3. 1 Langkah Pengolahan Data

No	Langkah Pengolahan Data
1.	Mengumpulkan data yang dibutuhkan dalam penelitian, meliputi data produk, data <i>workstation</i> , jumlah mesin, waktu proses, <i>setup time</i> , urutan proses, dan <i>release time</i> .
2.	Menentukan karakteristik sistem produksi sebagai <i>multi-stage flexible flow shop</i> .
3.	Menyusun model penjadwalan produksi berdasarkan data produk, urutan proses, waktu proses, <i>setup time</i> , jumlah mesin, dan <i>release time</i> .
4.	Menyusun jadwal kondisi aktual perusahaan berdasarkan urutan pengerjaan produk secara berurutan dari P1 sampai P20.
5.	Menentukan variabel keputusan, yaitu urutan pengerjaan <i>job</i> dan pemilihan mesin pada setiap <i>workstation</i> .
6.	Menentukan <i>job</i> sebelumnya pada mesin yang sama untuk menghitung <i>setup time</i> antarproduk.
7.	Menghitung <i>setup time</i> berdasarkan <i>job</i> sebelumnya pada mesin yang sama. Apabila tidak terdapat <i>job</i> sebelumnya, maka <i>setup time</i> bernilai nol.
8.	Menghitung waktu mulai proses pada setiap <i>workstation</i> dengan mempertimbangkan <i>release time</i> , waktu selesai <i>job</i> pada <i>workstation</i> sebelumnya, dan waktu kesiapan mesin setelah <i>setup</i> .
9.	Menghitung waktu proses berdasarkan <i>job</i> dan mesin yang dipilih.

No	Langkah Pengolahan Data
10.	Menghitung waktu selesai <i>job</i> pada setiap <i>berdasarkan</i> waktu mulai proses dan waktu proses.
11.	Menghitung <i>idle time</i> mesin sebagai indikator pendukung, yaitu selisih antara waktu mulai <i>job</i> berikutnya dengan waktu kesiapan mesin setelah setup.
12.	Menghitung nilai <i>makespan</i> kondisi aktual sebagai nilai maksimum waktu selesai <i>job</i> pada <i>workstation</i> terakhir.
13.	Menjalankan <i>Solver</i> dengan metode <i>Evolutionary Algorithm</i> untuk meminimalkan <i>makespan</i> .
14.	Memperoleh hasil penjadwalan setelah optimasi (<i>after Solver</i>).
15.	Membandingkan nilai <i>makespan</i> sebelum dan sesudah optimasi.
16.	Menghitung selisih penurunan <i>makespan</i> dan persentase perbaikan <i>makespan</i> .
17.	Menganalisis hasil perubahan urutan <i>job</i> , pemilihan mesin, <i>idle time</i> , serta perbaikan nilai <i>makespan</i> sebagai dasar usulan penjadwalan produksi.

3.7 Validasi Model

Validasi model dilakukan untuk memastikan bahwa model penjadwalan yang digunakan telah sesuai dengan karakteristik sistem produksi aktual. Validasi dilakukan dengan memeriksa kesesuaian logika model, rumus perhitungan, serta batasan yang digunakan dalam pengolahan data.

1. Validasi Urutan Proses

Validasi ini dilakukan untuk memastikan bahwa setiap *job* melewati *work station* secara berurutan dari tahap pertama hingga tahap akhir.

2. Validasi Pemilihan Mesin

Validasi ini dilakukan untuk memastikan bahwa setiap *job* hanya diproses oleh satu mesin pada setiap *work station*, dan mesin yang dipilih sesuai dengan jumlah mesin yang tersedia.

3. Validasi *Setup Time*

Validasi ini dilakukan untuk memastikan bahwa waktu *setup* dihitung berdasarkan *job* sebelumnya pada mesin yang sama. Apabila tidak terdapat *job* sebelumnya, maka *setup time* bernilai nol.

4. Validasi *Non-Preemptive*

Validasi ini dilakukan untuk memastikan bahwa setiap pekerjaan yang sudah mulai diproses tidak mengalami interupsi hingga *work station* terakhir.

5. Validasi *Makespan*

Validasi ini dilakukan untuk memastikan bahwa nilai *makespan* diperoleh dari waktu terbesar pada *work station* terakhir.

6. Validasi Hasil *Solver*

Validasi ini dilakukan dengan membandingkan nilai *makespan* sebelum dan sesudah optimasi lebih kecil dibandingkan nilai *makespan* sebelum optimasi.

3.8 Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Studi Literatur

Pengumpulan referensi dari buku, jurnal, dan artikel yang relevan mengenai penjadwalan produksi, *flow shop*, *flexible flow shop*, dan *genetic algorithm*. Studi literatur ini digunakan untuk mendalami teori-teori dan metode yang sudah ada serta sebagai dasar pembentukan model penjadwalan yang akan digunakan.

2. Studi Dokumentasi

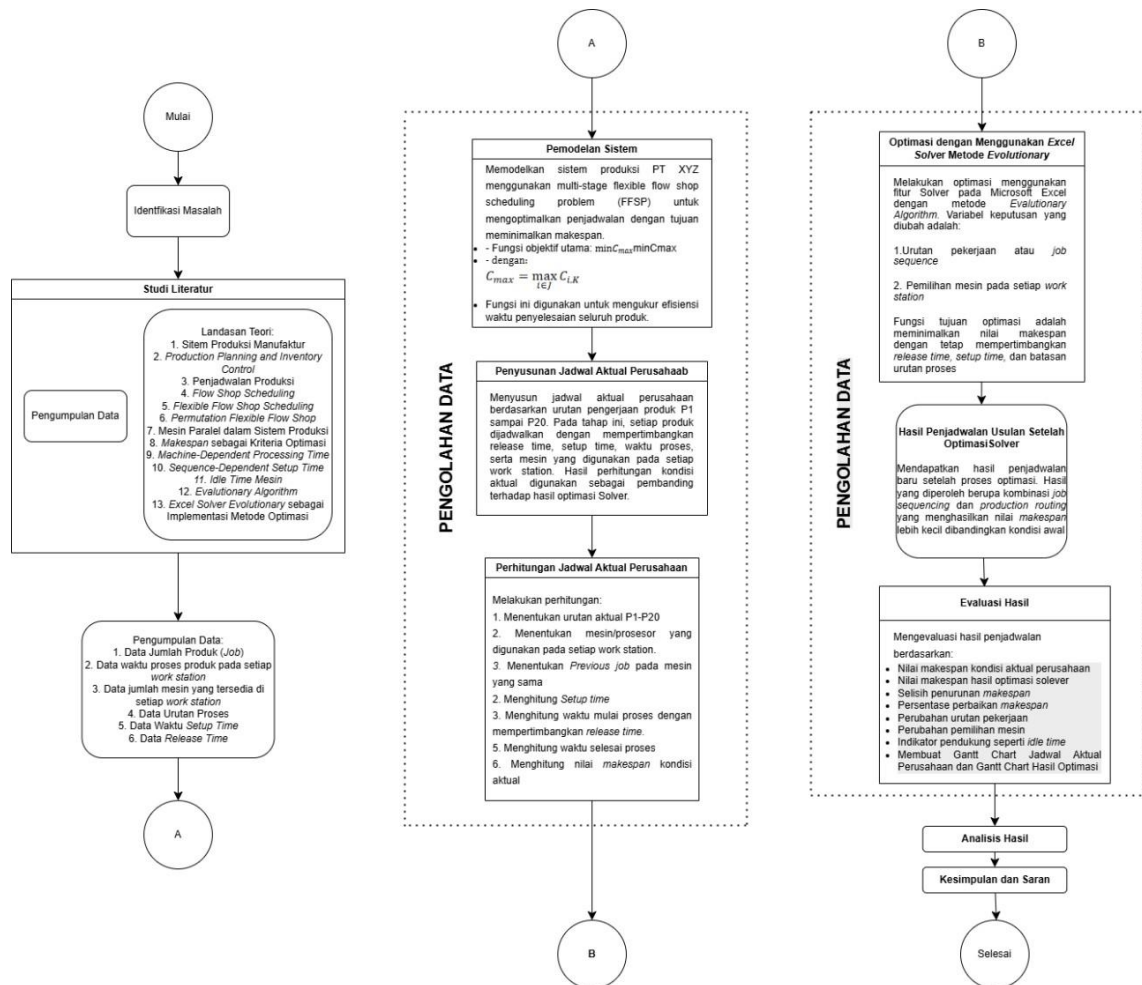
Mengumpulkan data terkait produksi di PT XYZ, termasuk data jumlah produk, waktu proses, jumlah mesin, dan urutan proses yang digunakan dalam produksi modul *transmitter* dan receiver elektronik.

3. Pengolahan Data Sekunder

Mengolah data yang telah dikumpulkan untuk disesuaikan dengan format yang digunakan dalam model penjadwalan dan analisis.

3.9 Alur Penelitian

Berikut ini merupakan alur penelitian yang akan digunakan.



Gambar 3. 1 Alur Penelitian

Berdasarkan alur penelitian pada Gambar 3.1, berikut ini merupakan penjelasan untuk masing-masing tahap yang dilakukan.

1. Identifikasi Masalah

Tahap pertama adalah identifikasi masalah utama dalam penelitian, yaitu alokasi produk (*job*) ke mesin dan urutan pengerjaan pada setiap *work station*. Permasalahan utama adalah bagaimana menentukan kombinasi urutan pengerjaan (*Job assignment*) dan pemilihan mesin (*production routing*) yang optimal untuk meminimalkan *makespan*, mengurangi *idle time*, serta menyeimbangkan beban kerja mesin pada setiap *work station*. Pada kondisi aktual perusahaan, proses penjadwalan masih dilakukan berdasarkan urutan produk secara berurutan dari P1 sampai P20. Pola tersebut belum mempertimbangkan kombinasi urutan pekerjaan, pemilihan mesin, *setup time* antarproduk, serta *release time* masing-masing produk secara optimal.

Oleh karena itu, penelitian ini difokuskan untuk menyusun jadwal produksi usulan pada sistem *multi-stage flexible flow shop* dengan tujuan meminimalkan *makespan*.

2. Studi Literatur

Tahap studi literatur dilakukan untuk memperoleh dasar teori yang mendukung penelitian. Literatur yang dikaji berkaitan dengan sistem produksi manufaktur, *Production Planning and Inventory Control*, penjadwalan produksi, *flow shop scheduling*, *flexible flow shop scheduling*, *multi-stage flexible flow shop*, mesin paralel, *machine-dependent processing time*, *sequence-dependent setup time*, *release time*, *makespan*, *idle time* mesin, *Evolutionary Algorithm*, serta *Excel Solver Evolutionary* sebagai implementasi metode optimasi. Studi literatur ini digunakan untuk memahami konsep dasar penjadwalan produksi, menentukan pendekatan penyelesaian yang sesuai, serta memperkuat dasar pemodelan sistem yang digunakan dalam penelitian.

3. Pengumpulan Data

Tahap ini melibatkan pengumpulan data yang diperlukan untuk penelitian. Data yang dikumpulkan meliputi:

- Data Jumlah Produk (*Job*)
- Data Waktu Proses
- Data Waktu *Setup*
- Data Jumlah Mesin
- Data Urutan Proses
- Data *Work station*
- Data *Release Time*

4. Pemodelan Sistem

Setelah data, pemodelan sistem dilakukan dengan menggunakan *multi-stage flexible flow shop scheduling problem* (FFSS). Dalam model ini, setiap produk harus diproses pada beberapa tahap produksi dengan mesin paralel pada setiap tahap. Fungsi objektif yang digunakan adalah meminimalkan *makespan*, yang dihitung menggunakan rumus berikut:

$$\min C_{max}$$

di mana:

$$C_{max} = \max_{i \in J} C_{i,K}$$

Fungsi objektif tersebut digunakan untuk mengukur efisiensi waktu penyelesaian seluruh produk. Semakin kecil nilai *makespan*, maka semakin cepat seluruh produk dapat diselesaikan.

5. Penyusunan Jadwal Aktual Perusahaan

Tahap penyusunan jadwal aktual perusahaan dilakukan untuk menggambarkan kondisi penjadwalan yang saat ini digunakan oleh perusahaan. Pada kondisi aktual, urutan pengerjaan produk dilakukan secara berurutan dari P1 sampai P20. Penyusunan jadwal aktual dilakukan dengan mempertimbangkan *release time*, waktu proses, *setup time*, urutan proses, serta mesin yang digunakan pada setiap *work station*. Jadwal aktual ini digunakan sebagai dasar pembanding terhadap jadwal usulan yang diperoleh dari hasil optimasi menggunakan *Excel Solver* metode *Evolutionary*. Pendekatan ini digunakan untuk menghasilkan jadwal awal yang layak dan sederhana sebelum dilakukan proses optimasi. Dengan demikian, solusi awal yang terbentuk terdiri dari:

- a. Urutan pengerjaan produk awal
- b. Pemilihan mesin awal pada setiap *work station*,
- c. Waktu *setup* awal
- d. Waktu mulai dan waktu selesai setiap produk
- e. Nilai *makespan* before *Solver*

Solusi awal ini kemudian menjadi dasar pembanding terhadap hasil optimasi setelah *Solver* dijalankan.

6. Optimasi Menggunakan Excel *Solver Evolutionary*

Setelah jadwal awal dihitung, tahap selanjutnya adalah melakukan optimasi menggunakan fitur *Solver* pada Microsoft Excel dengan metode *Evolutionary Algorithm*. Optimasi dilakukan untuk mencari kombinasi urutan pekerjaan dan pemilihan mesin yang menghasilkan nilai *makespan* paling minimum.

Pada proses optimasi, sel objektif yang digunakan adalah nilai *makespan*. *Solver* diarahkan untuk meminimalkan nilai tersebut dengan mengubah variabel keputusan yang terdiri dari:

- a. Urutan pekerjaan (*job sequencing*)

Variabel ini menentukan urutan produk yang akan diproses dalam sistem produksi.

- b. Pemilihan mesin (*production routing*)

Variabel ini menentukan mesin yang digunakan oleh setiap produk pada masing-masing *work station*.

Proses optimasi dilakukan dengan memperhatikan beberapa batasan, yaitu setiap *job* hanya boleh muncul satu kali dalam urutan penjadwalan, setiap *job* hanya diproses oleh satu mesin pada setiap *work station*, setiap mesin hanya dapat memproses satu *job* dalam satu waktu, dan proses produksi bersifat *non-preemptive* atau tidak dapat disela ketika sedang berlangsung.

7. Hasil Jadwal Usulan Setelah Optimasi Solver

Hasil dari proses optimasi berupa jadwal produksi usulan yang terdiri dari urutan pengerjaan produk, pemilihan mesin pada setiap *work station*, waktu setup, waktu mulai proses, waktu selesai proses, *idle time*, dan nilai *makespan* hasil optimasi. Jadwal usulan ini menunjukkan kombinasi penjadwalan yang diperoleh dari Excel Solver metode Evolutionary. Hasil tersebut kemudian digunakan untuk mengetahui apakah jadwal usulan mampu menghasilkan nilai *makespan* yang lebih kecil dibandingkan kondisi aktual perusahaan.

8. Evaluasi Hasil

Evaluasi hasil dilakukan dengan membandingkan kondisi sebelum dan sesudah optimasi. Perbandingan dilakukan untuk mengetahui apakah hasil penjadwalan setelah Solver mampu memberikan perbaikan terhadap sistem produksi.

Indikator utama yang digunakan dalam evaluasi adalah *makespan* karena tujuan utama penelitian ini adalah meminimalkan waktu penyelesaian seluruh produk. Selain itu, evaluasi juga dilakukan terhadap perubahan urutan pekerjaan, perubahan pemilihan mesin, serta *idle time* sebagai indikator pendukung.

Selisih penurunan *makespan* dihitung menggunakan rumus:

$$\Delta C_{max} = C_{max}^{aktual} - C_{max}^{usulan} \quad (2.13)$$

Keterangan:

ΔC_{max} : selisih penurunan *makespan*

C_{max}^{before} : nilai *makespan* pada kondisi aktual perusahaan

C_{max}^{after} : nilai *makespan* setelah optimasi Solver

Persentase perbaikan *makespan* dihitung menggunakan rumus pada 2.12. Selain *makespan*, hasil penjadwalan juga dianalisis berdasarkan indikator pendukung, seperti perubahan urutan pekerjaan, perubahan pemilihan mesin, dan *idle time*.

Namun, indikator tersebut digunakan sebagai analisis pendukung, sedangkan fungsi tujuan utama dalam penelitian ini tetap berfokus pada minimasi *makespan*.

9. Analisis Hasil

Tahap analisis dilakukan untuk menjelaskan hasil perbandingan antara kondisi jadwal aktual perusahaan dan jadwal usulan hasil optimasi *Solver*. Analisis difokuskan pada bagaimana perubahan urutan pekerjaan dan pemilihan mesin dapat memengaruhi nilai *makespan*. Apabila nilai *makespan* hasil optimasi lebih kecil dibandingkan *makespan* kondisi aktual, maka hasil optimasi menunjukkan adanya perbaikan jadwal produksi.

Selain itu, analisis juga dilakukan terhadap *idle time* sebagai indikator pendukung untuk melihat waktu menganggur mesin pada jadwal aktual dan jadwal usulan. Namun, *idle time* tidak digunakan sebagai fungsi tujuan utama, melainkan hanya sebagai informasi tambahan untuk memperkuat pembahasan hasil penjadwalan. Dengan demikian, hasil analisis dapat digunakan sebagai dasar dalam memberikan usulan penjadwalan produksi yang lebih efektif bagi perusahaan.

10. Kesimpulan dan Saran

Tahap akhir penelitian ini adalah menarik kesimpulan dari hasil penelitian yang telah dilakukan dan memberikan saran untuk penelitian lebih lanjut atau penerapan metode yang digunakan di perusahaan lain. Hasil penelitian ini diharapkan dapat memberikan wawasan bagi PT XYZ mengenai penerapan *Genetic Algorithm* (GA) untuk penjadwalan produksi dalam sistem *multi-stage flexible flow shop*. Selain itu, saran juga diberikan terkait bagaimana mengimplementasikan metode penjadwalan ini pada sistem produksi lain yang memiliki karakteristik serupa.

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam proses pemodelan dan optimasi penjadwalan produksi pada sistem *multi-stage flexible flow shop* di PT XYZ. Data yang dikumpulkan digunakan sebagai dasar dalam penyusunan model penjadwalan, pembentukan solusi awal, serta proses optimasi menggunakan *Evolutionary Algorithm (EA)*. Data tersebut meliputi data produk (*job*), data waktu proses pada setiap mesin, data jumlah mesin pada setiap *work station*, serta data urutan proses produksi. Seluruh data diperoleh melalui studi dokumentasi perusahaan dan pengolahan data sekunder yang berkaitan dengan aktivitas produksi modul *transmitter* dan *receiver* elektronik di PT XYZ. Selanjutnya, data yang telah dikumpulkan akan digunakan dalam proses pengolahan data untuk menentukan kombinasi *Job assignment* dan *machine assignment* yang optimal dalam meminimalkan *makespan*.

4.1.1 Data Produk

Berikut ini merupakan data produk yang digunakan dalam penelitian dan dapat dilihat pada Tabel 4.1. Data ini berisi ID produk, nama produk, dan aplikasi masing-masing produk yang akan dijadwalkan pada sistem produksi PT XYZ.

Tabel 4. 1 Nama Produk yang di Produksi PT XYZ

No.	Product ID	Nama Produk	Aplikasi
1.	P1	TV Control Board	Smart TV
2.	P2	Microwave Logic Unit	Microwave Oven
3.	P3	AC Display Board	Split Air Conditioner
4.	P4	Refrigerator Mainboard	Refrigerator
5.	P5	Washing Machine Control Panel	Washing Machine
6.	P6	Dishwasher Control Unit	Dishwasher
7.	P7	Induction Cooker Board	Induction Stove
8.	P8	Dryer Display Module	Clothes Dryer
9.	P9	Vacuum Cleaner Motor Control	Vacuum Cleaner
10.	P10	Water Purifier Sensor Board	Water Purifier
11.	P11	Gaming Console Power Regulator	Gaming Console
12.	P12	Electric Oven Temperature Controller	Electric Oven
13.	P13	Fan Speed Control PCB	Table Fan
14.	P14	LED Lamp Driver	LED Lamp
15.	P15	Air Purifier Control Module	Air Purifier
16.	P16	Smart Lock Motor Driver	Smart Door Lock
17.	P17	Coffee Maker Brewing Controller	Coffee Machine
18.	P18	Toaster Timer Board	Electric Toaster
19.	P19	Home Theater Audio Processor	Home Theater System
20.	P20	Thermostat Sensor Interface	Smart Thermostat

4.1.2 Data Work station dan Mesin

Data *work station* dan mesin digunakan untuk menggambarkan struktur sistem produksi pada PT XYZ. Sistem produksi terdiri dari lima *work station* yang tersusun secara berurutan dalam sistem *multi-stage flexible flow shop*. Pada setiap *work station* terdapat mesin paralel yang dapat digunakan untuk memproses *job*. Secara keseluruhan, terdapat 12 mesin yang digunakan dalam proses produksi, dengan jumlah mesin yang berbeda pada setiap *work station*. Pembagian mesin pada masing-masing *work station* disajikan pada Tabel 4.2.

Tabel 4. 2 Pembagian Mesin Pada Work station

Work station	Mesin
WS1	MA, MB, MC
WS2	MD, ME
WS3	MF, MG

<i>Work station</i>	<i>Mesin</i>
WS4	MH, MI, MJ
WS5	MK, ML

4.1.3 Data Waktu Proses

Data waktu proses dari setiap produk pada masing-masing mesin di setiap *work station* disajikan pada Tabel 4.3. Data ini digunakan sebagai parameter utama dalam proses penjadwalan produksi karena waktu proses setiap produk dapat berbeda pada mesin yang berbeda. Oleh karena itu, data waktu proses menjadi dasar dalam menentukan *job sequencing* dan *machine assignment* yang dapat meminimalkan *makespan*.

Tabel 4. 3 Waktu Proses Produk

	WS1			WS2		WS3		WS4			WS5	
	MA	MB	MC	MD	ME	MF	MG	MH	MI	MJ	MK	ML
P1	3	3	3	5	4	7	7	6	10	10	7	4
P2	4	7	7	6	10	5	9	7	7	12	3	3
P3	3	5	5	5	9	6	5	6	10	6	9	5
P4	5	9	5	6	6	8	8	7	12	7	10	6
P5	4	7	4	4	3	5	4	5	9	9	4	4
P6	3	5	3	5	5	7	12	6	6	10	7	4
P7	5	9	9	6	5	6	6	7	7	7	9	5
P8	4	7	3	5	9	6	6	6	10	10	7	4
P9	4	7	7	6	6	7	7	5	9	5	5	3
P10	3	3	3	4	7	5	9	5	4	4	7	4
P11	4	4	7	6	10	8	8	6	6	6	4	4
P12	3	3	5	5	5	6	10	6	5	10	5	3
P13	2	3	2	4	4	5	9	4	3	7	3	3
P14	2	2	2	3	3	4	4	3	5	5	3	2
P15	3	3	3	5	9	6	10	5	4	5	3	3
P16	2	2	2	4	3	5	9	5	9	9	5	3
P17	4	4	4	6	10	7	6	6	5	6	4	4
P18	2	3	2	3	3	4	7	3	3	3	2	2
P19	5	5	5	7	7	9	8	8	8	8	6	6
P20	3	3	5	5	9	6	5	5	9	4	7	4

4.1.4 Data Release Time

. Data *release time* dari setiap produk disajikan pada Tabel 4.4. *Release time* menunjukkan waktu paling awal suatu *job* tersedia dan dapat mulai diproses dalam sistem produksi. Data ini digunakan sebagai salah satu parameter dalam penyusunan jadwal agar setiap produk hanya dapat diproses setelah waktu ketersediaannya terpenuhi.

Tabel 4. 4 <i>Release Time</i>	
Kode Produk	<i>Release Time</i>
P1	12
P2	10
P3	11
P4	15
P5	18
P6	15
P7	6
P8	9
P9	6
P10	6
P11	20
P12	13
P13	12
P14	5
P15	8
P16	9
P17	20
P18	19
P19	16
P20	13

4.1.5 Data Waktu *Setup Time*

Data waktu proses dari setiap produk pada masing-masing mesin di setiap *work station* disajikan pada Tabel 4.3. Data ini digunakan sebagai parameter utama dalam proses penjadwalan produksi karena waktu proses setiap produk dapat berbeda pada mesin yang berbeda. Oleh karena itu, data waktu proses menjadi dasar dalam menentukan *job sequencing* dan *machine assignment* yang dapat meminimalkan *makespan*.

Tabel 4. 5 *Sequence-Dependent Setup Time*

	<i>Sequence-Dependent Setup Time</i>																			
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
P1	0	4	1	3	3	4	1	1	3	2	3	1	4	3	4	3	3	1	1	3
P2	2	0	2	4	4	2	2	2	4	4	3	1	4	2	1	3	4	1	1	3
P3	3	3	0	4	4	4	4	3	2	2	3	2	3	4	3	4	4	1	3	1
P4	3	3	1	0	3	2	4	1	4	2	2	2	1	2	1	2	4	4	3	4
P5	3	4	1	4	0	3	2	1	4	2	4	4	2	2	2	2	2	4	2	1
P6	3	2	2	4	2	0	2	4	2	3	4	3	4	2	3	4	1	2	4	1
P7	4	1	2	3	1	4	0	1	4	4	1	1	1	1	3	1	1	1	3	1
P8	4	1	4	4	4	3	3	0	1	4	3	3	1	3	1	2	3	2	1	4
P9	3	1	4	4	2	1	4	3	0	2	4	1	3	4	4	2	3	3	1	3
P10	1	3	2	3	1	1	2	3	3	0	3	3	1	3	3	2	2	4	1	3
P11	3	4	3	1	4	1	4	4	2	1	0	3	1	3	1	4	4	1	4	3
P12	3	3	2	4	2	2	1	1	1	1	2	0	4	4	4	4	2	2	3	4
P13	2	3	4	1	3	2	1	1	1	3	2	1	0	1	1	3	3	2	4	3
P14	1	2	1	1	3	2	4	4	3	3	4	4	4	0	1	2	1	2	1	2
P15	4	4	2	3	2	3	1	1	1	4	1	3	1	2	0	4	2	3	1	4
P16	1	1	3	2	2	1	4	2	4	2	4	3	4	3	3	0	3	1	3	2
P17	4	1	4	2	2	2	3	3	4	4	1	1	4	3	4	4	0	3	4	4
P18	3	4	3	2	3	3	3	4	4	3	4	1	1	4	4	3	2	0	1	3
P19	4	1	1	2	3	3	2	2	3	3	4	4	2	4	1	4	4	0	1	1
P20	4	2	4	1	1	4	3	3	1	1	3	3	4	4	3	4	4	2	4	0

4.1.6 Data Urutan Proses

Data urutan proses produksi yang harus dilalui oleh setiap produk disajikan pada Tabel 4.6. Data ini menunjukkan tahapan *work station* yang dilalui oleh setiap *job* secara berurutan dalam sistem *multi-stage flexible flow shop*. Berdasarkan data tersebut, seluruh produk harus melewati lima tahap proses, yaitu WS1, WS2, WS3, WS4, dan WS5.

Tabel 4. 6 Urutan Proses

Tahap	<i>Work station</i>
1	WS1
2	WS2
3	WS3
4	WS4
5	WS5

4.2 Pengolahan Data

Pada bagian ini diuraikan hasil dari pengolahan data sesuai dengan tahapan yang telah diuraikan pada Bab III Metode Penelitian. Gambar atau tabel yang dipaparkan diberikan penjelasan. Pada bagian ini belum dilakukan analisa terhadap hasil dari pengolahan data.

4.2.1 Pengolahan Jadwal Aktual Perusahaan

Pengolahan jadwal aktual dilakukan berdasarkan urutan produksi yang digunakan pada kondisi perusahaan. Pada kondisi aktual, setiap produk diproses melalui lima *work station* secara berurutan.

A. Menentukan Job assignment

Job assignment merupakan urutan pengerjaan produk dalam proses produksi. Pada kondisi aktual, *Job assignment* mengikuti urutan aktual perusahaan Dimana pada penelitian ini terdapat 20 produk. Pada kondisi usulan, *Job assignment* menjadi salah satu variabel keputusan yang diubah oleh *Solver*.

Secara umum, *Job assignment* dituliskan sebagai:

$$J_i = \{1, 2, 3, \dots, n\}$$

J_i : urutan *job* ke- i

n : Produk yang di produksi

B. Menentukan Processor pada Setiap Work station

Processor merupakan mesin yang digunakan untuk memproses produk pada masing-masing *work station*. Karena setiap *work station* memiliki mesin paralel, maka setiap

produk perlu dialokasikan ke processor tertentu. *Processor* yang digunakan dapat dilihat pada Tabel 4.7.

Tabel 4. 7 *Processor yang Digunakan Pada Setiap Work station*

<i>Work station</i>	<i>Mesin</i>	<i>Kode Processor</i>
WS 1	MA	1
	MB	2
	MC	3
WS 2	MD	1
	ME	2
WS 3	MF	1
	MG	2
	MH	1
WS 4	MI	2
	MJ	3
WS 5	MK	1
	ML	2

Secara umum:

M_{ij} = processor yang digunakan *job i* pada *work station j*

C. Menentukan *Previous Job*

Previous job merupakan *job* sebelumnya yang diproses pada *predecessor* yang sama. Apabila suatu *job* merupakan *job* pertama pada processor tersebut, maka *previous job* bernilai 0.

Secara umum:

$$PJ_{i,k} = \begin{cases} 0, & \text{jika belum ada job sebelumnya pada processor yang sama} \\ j, & \text{jika job } j \text{ merupakan job terakhir pada mesin yang sama} \end{cases}$$

Keterangan:

$PJ_{i,k}$: *Previous job* untuk *job* ke-*i* pada *work station* ke-*k*

j : *Job* terakhir sebelumnya yang menggunakan *processor* yang sama

Previous job ditentukan dengan mencari *job* terakhir pada baris sebelumnya yang menggunakan *processor* yang sama dengan *job* saat ini. Apabila belum terdapat *job* sebelumnya pada *processor* tersebut, *previous job* diberikan nilai 0. Penentuan *previous job* pada *Microsoft Excel* dilakukan menggunakan fungsi LOOKUP, sebagaimana ditunjukkan pada persamaan berikut.

$$=IFERROR(LOOKUP(2,1/(rentang_mesin_sebelumnya=mesin_job_saat_ini),rentang_job_sebelumnya),0)$$

Berikut ini contoh penggunaanya di pengolahan data pada penelitian ini, dapat dilihat pada Gambar 4.1.

	A	B	C	D	E	F	G
1	Work Station 1						
2	Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration
3	P1	1	2	0	0	12	
4	P2	2	1	0	0	10	
5	P3	3	1	2	2	16	
6	P4	4	3	=IFERROR(LOOKUP(2,1/(\$C\$3:C5=C6),\$B\$3:B5),0)	0	15	
7	P5	5	1	3	4	23	

Gambar 4. 1 Penentuan *Previous Job*

Selanjutnya, kolom *Job assignment* digunakan untuk mengambil nomor *job* terakhir yang memakai mesin yang sama. Jika belum ada *job* sebelumnya pada mesin tersebut, fungsi IFERROR akan memberikan nilai 0. Perhitungan ini digunakan pada semua *work station* dengan menyesuaikan kolom mesin pada masing-masing *work station*.

D. Menghitung *Setup Time*

Setup time merupakan waktu persiapan yang dibutuhkan ketika *processor* berganti dari *previous job* ke *job* saat ini. Apabila *previous job* bernilai 0, maka *setup time* bernilai 0.

Secara umum:

$$s_{j,i} = \begin{cases} 0, & \text{jika } PJ_{i,k} = 0 \\ s(j,i), & \text{jika terdapat previous job } j \end{cases}$$

$s_{j,i}$: *Setup time* dari *job* sebelumnya j menuju *job* i

$PJ_{i,k}$: *Previous job* i pada *work station* k

Berikut ini contoh formula *excel* dan digunakan dengan cara yang sama pada semua *work station*.

$$=IF(previous\ job=0,0,INDEX(matriks_setup,previous\ job,job_saat_ini))$$

Berikut ini contoh penggunaanya di pengolahan data pada penelitian ini, dapat dilihat pada Gambar 4.2.

	A	B	C	D	E	F	G
1							
2	Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration
3	P1	1	2	0	Up!\$C\$3:	12	3

Gambar 4. 2 Perhitungan *Setup Time*

Fungsi IF digunakan untuk memeriksa apakah *job* memiliki *previous job*. Jika *previous job* bernilai 0, setup time juga bernilai 0. Jika terdapat *previous job*, fungsi INDEX mengambil waktu *setup* dari matriks berdasarkan perpindahan dari *job* sebelumnya menuju *job* saat ini.

E. Menghitung *Start Time*

Dalam melakukan perhitungan *start time* terbagi menjadi *start time work station* pertama dan *work station* kedua hingga lima.

i. Perhitungan *Work station* Pertama

Pada *work station* pertama, *job* hanya dapat mulai ketika:

1. *job* sudah tersedia berdasarkan *release time*.
2. mesin sudah selesai digunakan *predecessor*.
3. proses *setup* telah selesai.

Secara umum:

$$S_{i,1} = \max(r_i, C_{j,i} + s_{j,i})$$

$S_{i,1}$: Waktu mulai *job i* pada *work station* 1

r_i : *Release time job i*

$C_{j,i}$: Waktu selesai *previous job* pada *processor* yang sama di WS 1

$s_{j,i}$: *Setup time* di *previous job j* menuju *job i*

Jika *job* belum memiliki *previous job*, maka:

$$C_{k1} = 0$$

Berikut ini contoh formula *excel* yang digunakan untuk *start time* pada WS 1 untuk produk pertama. Dapat dilihat pada Gambar 4.3.

Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Work Station
P1	1	2	0	0	Proses!	3	15	2	0	0
P2	2	1	0	0	10	4	14	1	0	0

Gambar 4. 3 Perhitungan *Start Time* WS 1

Fungsi MAX memilih waktu yang paling akhir. Jika *job* sudah tersedia tetapi mesin belum siap, *job* harus menunggu mesin. Sebaliknya, jika mesin sudah siap tetapi *job* belum mencapai *release time*, proses harus menunggu *job* tersedia.

Berikut ini contoh formula *excel* yang digunakan untuk *start time* pada WS 1 untuk produk pertama hingga terakhir. Dapat dilihat pada Gambar 4.4.

Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job
P1	1	2	0	0	12	3	15	2	0
P2	2	1	0	0	\$C\$3:C3	4	14	1	0

Gambar 4. 4 Perhitungan *Start Time* WS 1 untuk P2 hingga P20

ii. Perhitungan *Start Time* pada *Work station* Dua hingga Lima

Pada *work station* kedua sampai *work station* terakhir, *job* hanya dapat mulai apabila:

1. *job* sudah selesai dari *work station* sebelumnya.
2. mesin pada *work station* saat ini sudah selesai digunakan oleh *previous job*.
3. proses setup telah selesai.

$$S_{i,k} = \max(C_{i,k-1}, C_{j,k} + s_{j,i}), k = 2, 3, \dots, K$$

Secara umum:

$S_{i,k}$: Waktu mulai *job i* pada *work station k*

$C_{i,k-1}$: Waktu selesai *job i* pada *work station* sebelumnya

$C_{j,k}$: Waktu selesai *previous job* pada *processor* yang sama di WS *k*

$s_{j,i}$: *Setup time* di *previous job j* menuju *job i*

Fungsi maksimum digunakan karena *job* tidak boleh berpindah sebelum proses pada *work station* sebelumnya selesai dan tidak boleh menggunakan mesin yang masih digunakan oleh *job* lain.

Berikut ini contoh formula *excel* yang digunakan untuk *start time* pada WS 2 hingga WS 5. Dapat dilihat pada Gambar 4.5.

AVERAGE fx =MAX(H4,IFERROR(LOOKUP(2,1/(\$I\$3:I3=I4),\$N\$3:N3),0)+K4)													
	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Work Station 1							Work Station 2					
2	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	Finish
3	1	2		0	12	3	15	2	0	0	15	4	19
4	2	1	0	0	10	4	14	1	0	0	14	6	20
5	3	1	2	2	16	3	19	1	2	2	22	5	27

Gambar 4. 5 *Start Time* WS 2 hingga WS 5

F. Menghitung Waktu Proses (*Duration*)

Waktu proses menunjukkan lama pengerjaan suatu *job* i pada mesin yang dipilih.

$$p_{i,k} = p_{i,k,m_{i,k}}$$

Artinya, waktu proses ditentukan berdasarkan *job* yang dikerjakan dan mesin yang digunakan.

$p_{i,k}$: Waktu proses *job i* pada *work station k*

$p_{i,k,m_{i,k}}$: Waktu proses *job i* pada mesin yang dipilih

$m_{i,k}$: Mesin terpilih untuk *job I* pada *work station k*

Berikut ini formula *excel* yang digunakan untuk menghitung waktu proses:

$$= \text{INDEX}(\text{tabel waktu proses}, \text{job saat ini}, \text{mesin terpilih})$$

Berikut ini contoh penggunaanya di pengolahan data pada penelitian ini, dapat dilihat pada Gambar 4.6.

AVERAGE =@INDEX('Data Proses'!\$E\$3:\$F\$22,'Jadwal Aktual Perusahaan'!B3,'Jadwal Aktual Perusahaan'!I3)														
	A	B	C	D	E	F	G	H	I	J	K	L	M	
1	Work Station 1								Work Station 2					
2	Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	
3	P1	1	2	0	0	12	3	15	2	0	0	15	{3}	
4	P2	2		0	0	10	4	14	1	0	0	14	6	

Gambar 4. 6 Perhitungan *Duration*

Fungsi INDEX dipakai untuk mencari dan mengambil waktu proses dari tabel data proses. Sederhananya, Excel akan melihat *job* mana yang sedang dijadwalkan, lalu mencocokkannya dengan mesin(*processor*) yang digunakan. Nomor *job* digunakan untuk menentukan baris data yang dicari, sedangkan nomor mesin digunakan untuk menentukan kolom waktu proses yang akan diambil. Dengan cara ini, waktu proses yang muncul akan selalu sesuai dengan kombinasi *job* dan mesin yang dipilih.

G. Menghitung *Finish Time*

Finish time diperoleh dari penjumlahan *start time* dan waktu proses (*duration*).

$$C_{i,k} = S_{i,k} + p_{i,k,m_{j,k}}$$

Berikut ini contoh formula *excel* yang digunakan untuk menghitung *finish time*, dapat dilihat pada Gambar 4.7.

AVERAGE

⌵

:

✖

✓

fx

⌵

=F3+G3

	A	B	C	D	E	F	G	H
1	Work Station 1							
2	Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish
3	P1	1	2	0	0	12	3	=F3+G3

Gambar 4. 7 Perhitungan *Finish Time*

H. Menghitung *Makespan*

Makespan merupakan waktu selesai paling akhir dari seluruh *job* pada *work station* terakhir.

$$C_{max} = \max_{i \in J} C_{i,K}$$

Berikut ini contoh formula *excel* yang digunakan untuk menghitung *makespan*. Dapat dilihat pada Gambar 4.8.

=MAX(AF3:AF22)

AA	AB	AC	AD	AE	AF
Processor	Previous Job	Setup	Start	Duration	Finish
2	0	0	32	4	36
1	0	0	36	3	39
2	1	1	42	5	47
2	3	4	51	6	57
1	2	4	52	4	56
2	4	2	59	4	63
2	6	2	65	5	70
2	7	1	71	4	75
2	8	1	76	3	79
2	9	2	81	4	85
1	5	4	81	4	85
2	10	3	88	3	91
1	11	1	92	3	95
2	12	4	98	2	100
1	13	1	101	3	104
2	14	4	114	3	117
2	16	3	122	4	126
1	15	3	107	2	109
1	18	1	120	6	126
2	17	4	130	4	134
Makespan					AF22)

Gambar 4. 8 Perhitungan *Makespan*

Fungsi maksimum digunakan karena seluruh proses produksi baru dianggap selesai ketika *job* dengan waktu selesai paling akhir pada *work station* terakhir telah selesai diproses.

I. Menghitung *Idle time*

Idle time merupakan waktu menganggur mesin setelah *previous job* selesai dan *setup* sudah dilakukan, tetapi *job* berikutnya belum mulai diproses

$$IT_{i,k} = S_{i,k} - (C_{j,k} + s_{j,i})$$

$S_{i,k}$: Waktu mulai *job i* pada *work station k*

$IT_{i,k}$: *Idle time job i* pada *work station k*

$C_{j,k}$: Waktu selesai *previous job* pada mesin yang sama

$s_{j,i}$: *setup time* dari *previous job* menuju *job* saat ini

Waktu kesiapan mesin diperoleh dari waktu selesai *previous job* ditambah *setup time*. *Idle time* kemudian dihitung dari selisih antara waktu mulai *job* saat ini dan waktu kesiapan mesin. Fungsi maksimum dengan nilai 0 digunakan agar *idle time* tidak bernilai negatif.

Berikut ini contoh formula *excel* yang digunakan untuk menghitung *idle time*. Dapat dilihat pada Gambar 4.9.

AVERAGE

▼

:

✖

✓

f_x

=IF(D4=0,0,MAX(0,F4-(IFERROR(LOOKUP(2,1/(\$C\$3:C3=C4), \$H\$3:H3),0)+E4)))

	B	C	D	MAX(number1, [number2], [number3], ...)				I	J	K	L	
1	Work Station 1							Work Station 2				
2	D	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start
3		1	2	0	0	12	3	15	2	0	0	15
4		2	1	0	0	10	4	14	1	0	0	14

Gambar 4. 9 *Idle time* pada Setiap *Job*

Idle time job pertama pada setiap mesin diberikan nilai 0. Artinya, *idle time* yang dihitung merupakan *idle time* antar*job* dan tidak memasukkan waktu menganggur awal sebelum *job* pertama diproses.

J. Menghitung *Total Idle time* pada Setiap *Work station*

Total *idle time* pada setiap *work station* diperoleh dengan menjumlahkan *idle time* seluruh *job*.

$$TIT_k = \sum_{i \in j} IT_{i,k}$$

Keterangan:

TIT_k : Total *idle time* pada *work station k*

$IT_{i,k}$: *Idle time job i* pada *work station k*

AG	AH	AI	AJ	AK
Idle Time				
WS 1	WS 2	WS 3	WS 4	WS 5
0	0	0	0	0
0	0	0	0	0
0	0	0	0	5
0	0	0	2	0
0	0	0	0	9
0	0	0	8	0
0	0	0	6	0
0	0	0	3	0
0	0	0	1	0
0	0	0	16	0
0	0	11	2	21
0	0	0	15	0
0	0	0	13	6
0	0	0	0	3
0	0	0	6	5
0	0	0	3	10
0	0	4	0	2
0	0	0	0	0
0	0	0	27	10
0	0	0	0	0
0	0	15	102	AK22

Gambar 4. 10 Total Idle time

K. Hasil Perhitungan Jadwal Aktual Perusahaan

Berikut ini merupakan hasil perhitungan penjadwalan produksi aktual berdasarkan kondisi nyata yang diterapkan oleh perusahaan saat ini. Tabel di bawah ini menyajikan rekapitulasi aliran proses untuk 20 produk (P1 hingga P20) beserta urutan pengerjaannya (*Job assignment*) yang melewati lima stasiun kerja (*Work station 1* hingga *Work station 5*).

- Jadwal Aktual *Work station 1*

Tabel 4. 8 Perhitungan Jadwal Aktual *Work station 1*

Work station 1							
Product ID	Job assignment	Processor	Previous Job	Setup	Start	Duration	Finish
P1	1	2	0	0	12	3	15
P2	2	1	0	0	10	4	14
P3	3	1	2	2	16	3	19
P4	4	3	0	0	15	5	20
P5	5	1	3	4	23	4	27
P6	6	3	4	2	22	3	25
P7	7	1	5	2	29	5	34
P8	8	3	6	4	29	3	32
P9	9	1	7	4	38	4	42
P10	10	2	1	2	17	3	20
P11	11	1	9	4	46	4	50

<i>Work station 1</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Processor</i>	<i>Previous Job</i>	<i>Setup</i>	<i>Start</i>	<i>Duration</i>	<i>Finish</i>
P12	12	2	10	3	23	3	26
P13	13	3	8	1	33	2	35
P14	14	3	13	1	36	2	38
P15	15	2	12	4	30	3	33
P16	16	1	11	1	51	2	53
P17	17	2	15	2	35	4	39
P18	18	3	14	2	40	2	42
P19	19	1	16	3	56	5	61
P20	20	2	17	4	43	3	46

- Jadwal Aktual Perusahaan *Work station 2*

Tabel 4. 9 Perhitungan Jadwal Aktual *Work station 2*

<i>Work station 2</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Processor</i>	<i>Previous Job</i>	<i>Setup</i>	<i>Start</i>	<i>Duration</i>	<i>Finish</i>
P1	1	2	0	0	15	4	19
P2	2	1	0	0	14	6	20
P3	3	1	2	2	22	5	27
P4	4	2	1	3	22	6	28
P5	5	2	4	3	31	3	34
P6	6	1	3	4	31	5	36
P7	7	2	5	2	36	5	41
P8	8	1	6	4	40	5	45
P9	9	1	8	1	46	6	52
P10	10	1	9	2	54	4	58
P11	11	1	10	3	61	6	67
P12	12	2	7	1	42	5	47
P13	13	2	12	4	51	4	55
P14	14	1	11	3	70	3	73
P15	15	1	14	1	74	5	79
P16	16	2	13	3	58	3	61
P17	17	1	15	2	81	6	87
P18	18	2	16	1	62	3	65
P19	19	2	18	1	66	7	73
P20	20	1	17	4	91	5	96

- Jadwal Aktual Perusahaan *Work station 3*

Tabel 4. 10 Perhitungan Jadwal Aktual *Work station 3*

<i>Work station 3</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setup</i>	<i>Start</i>	<i>Duration</i>	<i>Finish</i>
P1	1	1	0	0	19	7	26
P2	2	2	0	0	20	9	29

<i>Work station 3</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setu p</i>	<i>Star t</i>	<i>Duratio n</i>	<i>Finis h</i>
P3	3	2	2	2	31	5	36
P4	4	1	1	3	29	8	37
P5	5	2	3	4	40	4	44
P6	6	1	4	2	39	7	46
P7	7	2	5	2	46	6	52
P8	8	1	6	4	50	6	56
P9	9	1	8	1	57	7	64
P10	10	1	9	2	66	5	71
P11	11	2	7	4	67	8	75
P12	12	1	10	3	74	6	80
P13	13	1	12	4	84	5	89
P14	14	2	11	3	78	4	82
P15	15	1	13	1	90	6	96
P16	16	1	15	4	100	5	105
P17	17	2	14	1	87	6	93
P18	18	2	17	3	96	7	103
P19	19	2	18	1	104	8	112
P20	20	1	16	2	107	6	113

- Jadwal Aktual Perusahaan *Work station 4*

Tabel 4. 11 Perhitungan Jadwal Aktual *Work station 4*

<i>Work station 4</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setu p</i>	<i>Star t</i>	<i>Duratio n</i>	<i>Finis h</i>
P1	1	1	0	0	26	6	32
P2	2	2	0	0	29	7	36
P3	3	3	0	0	36	6	42
P4	4	1	1	3	37	7	44
P5	5	1	4	3	47	5	52
P6	6	2	2	2	46	6	52
P7	7	3	3	4	52	7	59
P8	8	1	5	1	56	6	62
P9	9	3	7	4	64	5	69
P10	10	2	6	3	71	4	75
P11	11	3	9	4	75	6	81
P12	12	1	8	3	80	6	86
P13	13	2	10	1	89	3	92
P14	14	2	13	1	93	5	98
P15	15	1	12	4	96	5	101
P16	16	2	14	4	105	9	114

Work station 4							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finish h
P17	17	2	16	3	117	5	122
P18	18	1	15	3	104	3	107
P19	19	3	11	4	112	8	120
P20	20	3	19	1	121	4	125

- Jadwal Aktual Perusahaan *Work station 5*

Tabel 4. 12 Perhitungan Jadwal Aktual *Work station 5*

Work station 5							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finish h
P1	1	2	0	0	32	4	36
P2	2	1	0	0	36	3	39
P3	3	2	1	1	42	5	47
P4	4	2	3	4	51	6	57
P5	5	1	2	4	52	4	56
P6	6	2	4	2	59	4	63
P7	7	2	6	2	65	5	70
P8	8	2	7	1	71	4	75
P9	9	2	8	1	76	3	79
P10	10	2	9	2	81	4	85
P11	11	1	5	4	81	4	85
P12	12	2	10	3	88	3	91
P13	13	1	11	1	92	3	95
P14	14	2	12	4	98	2	100
P15	15	1	13	1	101	3	104
P16	16	2	14	4	114	3	117
P17	17	2	16	3	122	4	126
P18	18	1	15	3	107	2	109
P19	19	1	18	1	120	6	126
P20	20	2	17	4	130	4	134
<i>Makespan</i>							134

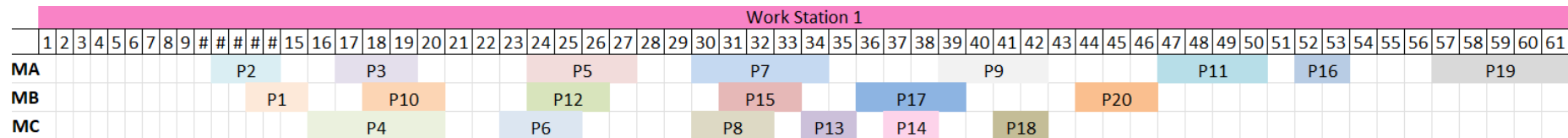
- *Idle time* Jadwal Aktual Perusahaan

Tabel 4. 13 *Idle time* Jadwal Aktual Perusahaan

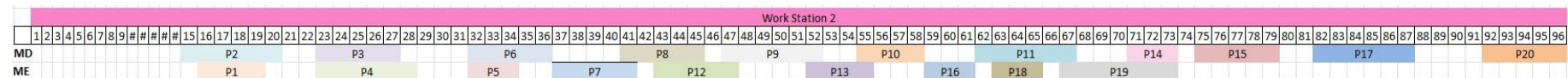
Idle time				
WS 1	WS 2	WS 3	WS 4	WS 5
0	0	0	0	0
0	0	0	0	0
0	0	0	0	5
0	0	0	2	0
0	0	0	0	9

<i>Idle time</i>				
WS 1	WS 2	WS 3	WS 4	WS 5
0	0	0	8	0
0	0	0	6	0
0	0	0	3	0
0	0	0	1	0
0	0	0	16	0
0	0	11	2	21
0	0	0	15	0
0	0	0	13	6
0	0	0	0	3
0	0	0	6	5
0	0	0	3	10
0	0	4	0	2
0	0	0	0	0
0	0	0	27	10
0	0	0	0	0
0	0	15	102	71

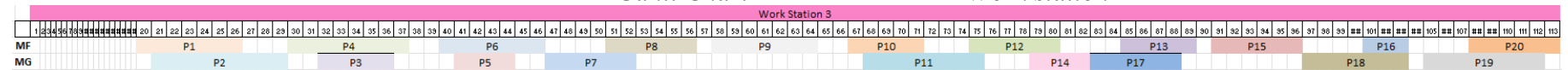
- *Gantt Chart* Jadwal Aktual Perusahaan



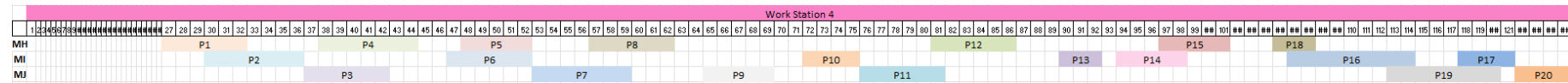
Gambar 4. 11 *Gantt Chart* Aktual Perusahaan *Work station 1*



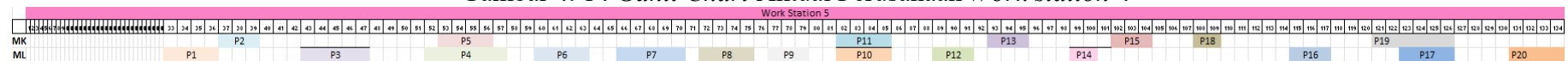
Gambar 4. 12 *Gantt Chart* Aktual Perusahaan *Work station 2*



Gambar 4. 13 *Gantt Chart* Aktual Perusahaan *Work station 3*



Gambar 4. 14 *Gantt Chart* Aktual Perusahaan *Work station 4*



Gambar 4. 15 *Gantt Chart* Aktual Perusahaan *Work station 5*

4.2.2 Pengolahan Jadwal Optimasi *Solver*

A. Menentukan Variabel Keputusan *Solver*

Variabel keputusan merupakan nilai yang dapat diubah oleh *Solver* untuk memperoleh jadwal usulan.

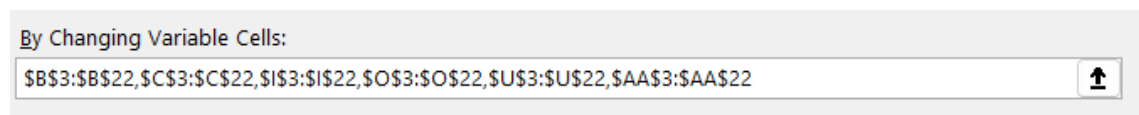
$$X = \{Z_{i,r}, m_{i,k}\}$$

Variabel keputusan terdiri dari *Job assignment* dan mesin (*processor*) yang dipilih pada setiap *work station*.

Tabel 4. 14 Variabel Keputusan *Solver*

<i>Job assignment</i>	<i>Work station 1</i>	<i>Work station 2</i>	<i>Work station 3</i>	<i>Work station 4</i>	<i>Work station 5</i>
	<i>Processor</i>	<i>Processor</i>	<i>Processor</i>	<i>Processor</i>	<i>Processor</i>
1	MB	ME	MF	MH	ML
2	MA	MD	MG	MI	MK
3	MA	MD	MG	MJ	ML
4	MC	ME	MF	MH	ML
5	MA	ME	MG	MH	MK
6	MC	MD	MF	MI	ML
7	MA	ME	MG	MJ	ML
8	MC	MD	MF	MH	ML
9	MA	MD	MF	MJ	ML
10	MB	MD	MF	MI	ML
11	MA	MD	MG	MJ	MK
12	MB	ME	MF	MH	ML
13	MC	ME	MF	MI	MK
14	MC	MD	MG	MI	ML
15	MB	MD	MF	MH	MK
16	MA	ME	MF	MI	ML
17	MB	MD	MG	MI	ML
18	MC	ME	MG	MH	MK
19	MA	ME	MG	MJ	MK
20	MB	MD	MF	MJ	ML

Berikut ini konfigurasi *Solver* untuk variabel keputusan yang akan diubah pada pengolahan data penelitian ini.



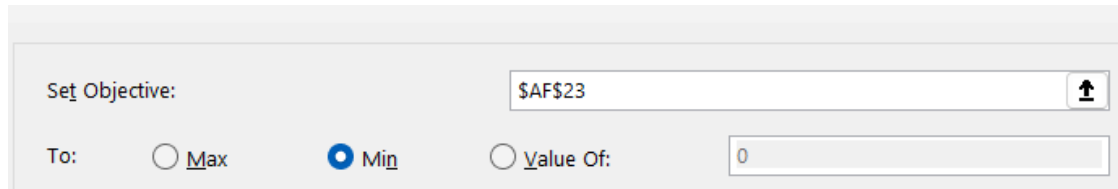
Gambar 4. 16 Variabel Keputusan yang Akan Diubah *Solver*

B. Menentukan Fungsi Tujuan *Solver*

Fungsi tujuan minimasi Adalah meminimalkan *makespan* sesuai dengan rumus pada (2.1), Secara matematis, fungsi tujuan dituliskan sebagai berikut:

$$\min C_{\max}$$

Pada pengolahan data yang dilakukan, *makespan* berada di rentang AF23, maka dari itu nanti pada saat memasukkan data di *Solver Parameters* bagian *set objective* akan memasukkan *Cell* AF23 dengan fungsi tujuannya minimasi. Berikut ini konfigurasi *Solver* untuk fungsi tujuan.



Gambar 4. 17 Fungsi Tujuan *Solver*

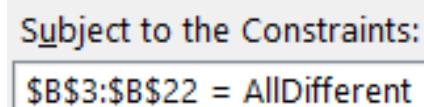
C. Menentukan Batasan *Solver*

Batasan *Solver* digunakan agar hasil optimasi tetap sesuai dengan kondisi sistem produksi. Batasan yang dimasukkan meliputi ketentuan urutan *job*, pemilihan mesin, release time, urutan proses antar-*work station*, dan kapasitas mesin. Batasan yang digunakan adalah sebagai berikut.

- **Setiap *job* hanya dijadwalkan satu kali**

Setiap *job* harus ditempatkan tepat satu kali dalam urutan penjadwalan. Batasan ini digunakan untuk mencegah adanya *job* yang muncul lebih dari satu kali atau tidak ikut dijadwalkan. Selain itu, setiap posisi urutan hanya boleh ditempati oleh satu *job*, sehingga tidak terdapat dua *job* yang menempati posisi yang sama.

Pada pengolahan data menggunakan *Microsoft Excel*, batasan tersebut diterapkan dengan memberikan ketentuan AllDifferent pada rentang *Job assignment*.



Gambar 4. 18 *Constraints* untuk *Job assignment*

Batasan AllDifferent memastikan seluruh nilai *Job assignment* berbeda, sehingga setiap *job* hanya muncul satu kali dalam urutan penjadwalan.

- **Mesin yang dipilih sesuai dengan mesin yang tersedia**

Setiap *job* hanya boleh dialokasikan pada mesin yang tersedia di masing-masing *work station*. Nomor mesin juga harus berupa bilangan bulat karena menunjukkan identitas mesin yang digunakan.

$$m_{i,k} \in M_k$$

$$m_{i,k} \in Z$$

Jumlah mesin pada setiap *work station* berbeda. Oleh karena itu, batas bawah dan batas atas nomor mesin disesuaikan dengan jumlah mesin yang tersedia.

Tabel 4. 15 *Constraints* pada *Excel Solver*

<i>Work station</i>	<i>Batasan Mesin</i>
WS 1	C3:C22 $\geq$ 1,
	C3:C22 $\leq$ 3,
	C3:C22 integer
WS 2	I3:I22 $\geq$ 1,
	I3:I22 $\leq$ 2,
	I3:I22 integer
WS 3	O3:O22 $\geq$ 1,
	O3:O22 $\leq$ 2,
	O3:O22 integer
WS 4	U3:U22 $\geq$ 1,
	U3:U22 $\leq$ 3,
	U3:U22 integer
WS 5	AA3:AA22 $\geq$ 1,
	AA3:AA22 $\leq$ 2,
	AA3:AA22 integer

Sebagai contoh, WS 1 memiliki tiga mesin, sehingga nomor mesin yang dapat dipilih hanya 1, 2, atau 3. Sementara itu, WS 2 hanya memiliki dua mesin, sehingga nomor mesin yang dapat dipilih hanya 1 atau 2.

D. Menjalankan *Solver*

Optimasi dilakukan menggunakan *Microsoft Excel Solver* dengan metode *Evolutionary Algorithm*. Metode tersebut digunakan karena model penjadwalan melibatkan kombinasi urutan *job*, pemilihan mesin, variabel bilangan bulat, dan formula yang menggunakan fungsi MAX, INDEX, serta LOOKUP. *Solver* akan mencoba berbagai kombinasi *Job assignment* dan pemilihan mesin untuk memperoleh nilai *makespan* minimum. Berikut ini merupakan *Solver parameter* akhir yang digunakan untuk mengoptimasi jadwal aktual perusahaan.

Gambar 4. 19 *Solver Parameters* Untuk Menjalankan *Solver*

E. Hasil Optimasi *Solver* Jadwal Aktual

Berikut ini merupakan hasil perhitungan optimasi penjadwalan produksi yang didapatkan melalui pendekatan *Solver*. Proses optimasi ini dilakukan terhadap data jadwal aktual perusahaan dengan tujuan untuk mengurutkan kembali pengerjaan (*Job assignment*) 20 produk (P1 hingga P20) agar menghasilkan aliran proses yang lebih efisien melintasi kelima stasiun kerja (*Work station 1* hingga *Work station 5*).

- Hasil Optimasi *Solver* Jadwal Aktual *Work station 1*

Tabel 4. 16 Optimasi *Solver* Jadwal Aktual *Work station 1*

<i>Work station 1</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setu p</i>	<i>Star t</i>	<i>Duratio n</i>	<i>Finis h</i>
P14	14	1	0	0	5	2	7
P10	10	2	0	0	6	3	9
P15	15	3	0	0	8	3	11
P16	16	1	14	4	11	2	13
P7	7	2	10	2	11	9	20
P2	2	1	16	1	14	4	18
P5	5	3	15	2	18	4	22
P17	17	3	5	2	24	4	28
P19	19	1	2	1	19	5	24
P1	1	2	7	0	20	3	23
P20	20	2	1	3	26	3	29

<i>Work station 1</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setup p</i>	<i>Start t</i>	<i>Duration n</i>	<i>Finis h</i>
P11	11	3	17	1	29	7	36
P3	3	3	11	3	39	5	44
P18	18	1	19	1	25	2	27
P4	4	2	20	1	30	9	39
P13	13	1	18	1	28	2	30
P12	12	3	3	2	46	5	51
P9	9	2	4	4	43	7	50
P8	8	3	12	2	53	3	56
P6	6	2	9	1	51	5	56

- Hasil Optimasi *Solver* Jadwal Aktual *Work station 2*

Tabel 4. 17 Optimasi *Solver* Jadwal Aktual *Work station 2*

<i>Work station 2</i>							
<i>Product ID</i>	<i>Job assignment</i>	<i>Process or</i>	<i>Previous Job</i>	<i>Setup p</i>	<i>Start t</i>	<i>Duration n</i>	<i>Finis h</i>
P14	14	1	0	0	7	3	10
P10	10	2	0	0	9	7	16
P15	15	1	14	1	11	5	16
P16	16	2	10	2	18	3	21
P7	7	1	15	1	20	6	26
P2	2	2	16	1	22	10	32
P5	5	1	7	1	27	4	31
P17	17	1	5	2	33	6	39
P19	19	2	2	1	33	7	40
P1	1	1	17	4	43	5	48
P20	20	2	19	1	41	9	50
P11	11	1	1	3	51	6	57
P3	3	1	11	3	60	5	65
P18	18	2	20	2	52	3	55
P4	4	2	18	2	57	6	63
P13	13	2	4	1	64	4	68
P12	12	2	13	1	69	5	74
P9	9	1	3	2	67	6	73
P8	8	1	9	3	76	5	81
P6	6	1	8	3	84	5	89

- Hasil Optimasi *Solver* Jadwal Aktual *Work station 3*

Tabel 4. 18 Optimasi *Solver* Jadwal Aktual *Work station 3*

Work station 3							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finish h
P14	14	1	0	0	10	4	14
P10	10	2	0	0	16	9	25
P15	15	1	14	1	16	6	22
P16	16	2	10	2	27	9	36
P7	7	1	15	1	26	6	32
P2	2	1	7	1	33	5	38
P5	5	2	16	2	38	4	42
P17	17	2	5	2	44	6	50
P19	19	1	2	1	40	9	49
P1	1	2	17	4	54	7	61
P20	20	1	19	1	50	6	56
P11	11	1	20	3	59	8	67
P3	3	2	1	1	65	5	70
P18	18	2	3	1	71	7	78
P4	4	1	11	1	68	8	76
P13	13	1	4	1	77	5	82
P12	12	2	18	1	79	10	89
P9	9	2	12	1	90	7	97
P8	8	1	13	1	83	6	89
P6	6	1	8	3	92	7	99

- Hasil Optimasi *Solver* Jadwal Aktual *Work station 4*

Tabel 4. 19 Optimasi *Solver* Jadwal Aktual *Work station 4*

Work station 4							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finish h
P14	14	3	0	0	14	5	19
P10	10	2	0	0	25	4	29
P15	15	3	14	1	22	5	27
P16	16	1	0	0	36	5	41
P7	7	1	16	4	45	7	52
P2	2	2	10	3	38	7	45
P5	5	3	15	2	42	9	51
P17	17	1	7	1	53	6	59
P19	19	2	2	1	49	8	57
P1	1	2	19	4	61	10	71
P20	20	3	5	1	56	4	60
P11	11	1	17	1	67	6	73
P3	3	3	20	4	70	6	76

Work station 4							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finis h
P18	18	2	1	4	78	3	81
P4	4	1	11	1	76	7	83
P13	13	1	4	1	84	4	88
P12	12	2	18	1	89	5	94
P9	9	1	13	1	97	5	102
P8	8	3	3	3	89	10	99
P6	6	2	12	2	99	6	105

- Hasil Optimasi Solver Jadwal Aktual *Work station 5*

Tabel 4. 20 Optimasi Solver Jadwal Aktual *Work station 5*

Work station 5							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finis h
P14	14	2	0	0	19	2	21
P10	10	1	0	0	29	7	36
P15	15	2	14	1	27	3	30
P16	16	2	15	4	41	3	44
P7	7	1	10	2	52	9	61
P2	2	1	7	1	62	3	65
P5	5	1	2	4	69	4	73
P17	17	2	16	3	59	4	63
P19	19	2	17	4	67	6	73
P1	1	1	5	3	76	7	83
P20	20	2	19	1	74	4	78
P11	11	2	20	3	81	4	85
P3	3	1	1	1	84	9	93
P18	18	2	11	1	86	2	88
P4	4	2	18	2	90	6	96
P13	13	2	4	1	97	3	100
P12	12	1	3	2	95	5	100
P9	9	2	13	1	102	3	105
P8	8	1	12	2	102	7	109
P6	6	2	9	1	106	4	110
<i>Makespan</i>							110

- Idle time Optimasi Solver Perusahaan

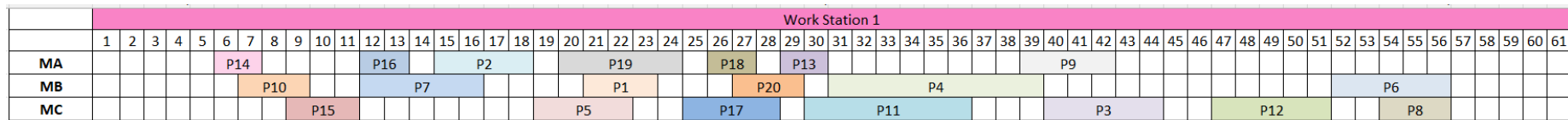
Tabel 4. 21 Idle time Optimasi Solver

Idle time				
WS 1	WS 2	WS 3	WS 4	WS 5
0	0	0	0	0
0	0	0	0	0
0	0	1	2	5

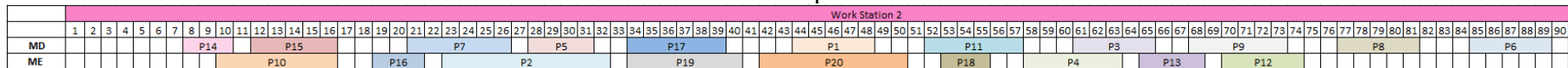
<i>Idle time</i>				
WS 1	WS 2	WS 3	WS 4	WS 5
0	0	0	0	7
0	3	3	0	14
0	0	0	6	0
5	0	0	13	0
0	0	0	0	12
0	0	1	3	0
0	0	0	0	0
0	0	0	4	0
0	0	0	7	0
0	0	3	6	0
0	0	0	3	0
0	0	0	2	0
0	0	0	0	0
0	0	0	7	0
0	0	0	8	1
0	0	0	10	0
0	0	0	3	0
5	3	8	74	39

- *Gantt Chart* Optimasi Solver

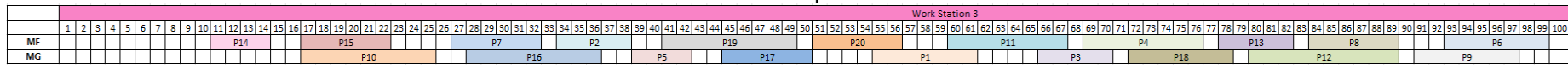
Berikut ini merupakan representasi visual dari hasil perhitungan optimasi penjadwalan *Solver* yang disajikan dalam bentuk bagan Gantt (*Gantt Chart*). Bagan ini memvisualisasikan urutan pengerjaan seluruh produk (P1 hingga P20) beserta alokasi waktunya pada masing-masing prosesor di setiap stasiun kerja.



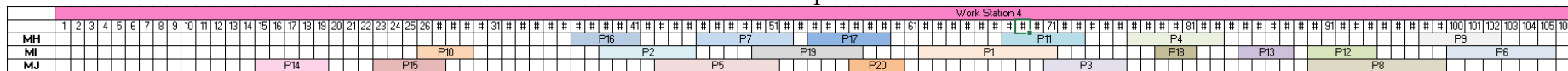
Gambar 4. 20 *Gantt Chart* Hasil Optimasi Solver *Work station 1*



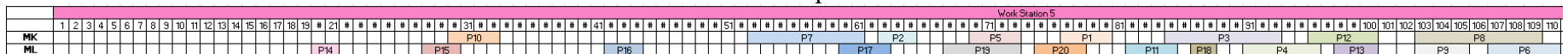
Gambar 4. 21 *Gantt Chart* Hasil Optimasi Solver Work station 2



Gambar 4. 22 *Gantt Chart* Hasil Optimasi Solver Work station 3



Gambar 4. 23 *Gantt Chart* Hasil Optimasi Solver Work station 4



Gambar 4. 24 *Gantt Chart* Hasil Optimasi Solver Work station 5

BAB V

PEMBAHASAN DAN ANALISIS HASIL

5.1 Pengujian dan Validasi Model Penjadwalan

Pengujian dan validasi model penjadwalan dilakukan untuk memastikan bahwa jadwal yang dihasilkan telah mengikuti logika perhitungan dan batasan sistem produksi *multi-stage flexible flow shop*. Pengujian ini diperlukan karena nilai fungsi tujuan yang lebih kecil belum cukup untuk menunjukkan bahwa suatu jadwal dapat diterapkan. Jadwal harus terlebih dahulu memenuhi ketentuan *release time*, urutan proses antar-*work station*, pemilihan mesin, kapasitas mesin, *sequence-dependent setup time*, dan sifat proses *non-preemptive*.

Pengujian dilakukan terhadap dua jadwal, yaitu jadwal aktual perusahaan dan jadwal hasil Optimasi *Hasil optimasi Solver*. Masing-masing jadwal terdiri atas 20 produk yang harus melewati lima *work station* secara berurutan. Dengan demikian, terdapat 100 operasi pada setiap jadwal atau 200 operasi yang diperiksa secara keseluruhan. Setiap operasi memuat informasi Product ID, *processor*, *previous job*, *setup time*, *start time*, *duration*, dan *finish time*.

5.1.1 Pengujian Konsistensi Data dan Perhitungan

Pengujian konsistensi data dan perhitungan dilakukan untuk memastikan bahwa seluruh parameter yang digunakan telah tersedia serta setiap formula dalam model penjadwalan menghasilkan nilai yang sesuai dengan logika sistem produksi. Pengujian meliputi kelengkapan data masukan, perhitungan *duration* dan *finish time*, penentuan *previous job*, perhitungan *setup time*, penentuan *start time*, serta perhitungan *makespan*.

A. Pengujian Kelengkapan dan Konsistensi Data Masukan

Hasil pemeriksaan menunjukkan bahwa produk yang digunakan dalam model terdiri atas P1 sampai P20. Setiap produk memiliki waktu proses yang berbeda pada setiap alternatif mesin. Perbedaan tersebut menunjukkan bahwa model telah merepresentasikan karakteristik *machine-dependent processing time*, yaitu waktu pemrosesan suatu produk bergantung pada mesin yang dipilih.

Waktu proses suatu produk dapat berbeda ketika produk tersebut diproses pada mesin yang berbeda. Kondisi tersebut menunjukkan bahwa data telah merepresentasikan

machine-dependent processing time. Sebagai contoh, P7 membutuhkan waktu proses sebesar 5 menit pada MA, tetapi membutuhkan 9 menit pada MB dan MC. Oleh karena itu, perubahan mesin yang dipilih dapat menyebabkan perubahan duration pada hasil penjadwalan.

Setiap produk juga memiliki *release time* yang menjadi batas waktu paling awal untuk memasuki WS1. Berdasarkan data masukan, *release time* paling awal dimiliki oleh P1 sebesar 12 menit, sedangkan *release time* terbesar dimiliki oleh P11 dan P17 sebesar 20 menit. Sementara itu, P20 memiliki *release time* sebesar 13 menit. Seluruh produk harus melewati urutan proses yang sama, yaitu WS1, WS2, WS3, WS4, dan WS5.

Seluruh produk harus melewati urutan proses yang sama, yaitu WS1, WS2, WS3, WS4, dan WS5. Selain itu, matriks *sequence-dependent setup time* telah memuat waktu persiapan berdasarkan pasangan produk sebelumnya dan produk yang akan diproses. Dengan demikian, nilai setup tidak hanya ditentukan oleh produk saat ini, tetapi juga oleh produk yang sebelumnya menggunakan mesin yang sama.

Berdasarkan pemeriksaan tersebut, data produk, *release time*, waktu proses, pembagian mesin, urutan proses, dan *setup time* telah tersedia untuk mendukung pembentukan model penjadwalan.

B. Pengujian *Duration* dan *Finish Time*

Duration merupakan waktu yang dibutuhkan untuk memproses suatu produk pada processor yang dipilih. Nilai duration diambil dari tabel waktu proses berdasarkan kombinasi Product ID, *work station*, dan *processor* yang digunakan.

Setelah kelengkapan data diperiksa, pengujian dilanjutkan terhadap konsistensi hubungan antara *start time*, *duration*, dan *finish time*. Setiap operasi dinyatakan valid apabila memenuhi hubungan:

$$Finish\ time = Start\ time + Duration$$

Berikut ini merupakan hasil contoh pengujian konsistensi perhitungan ditunjukkan pada Tabel 5.1.

Tabel 5. 1 Contoh Pengujian Konsistensi Perhitungan Waktu

Jadwal	Produk	<i>Work station</i>	<i>Previous Job</i>	<i>Setup</i>	<i>Start</i>	<i>Duration</i>	<i>Finish</i>	<i>Status</i>
Aktual	P1	WS 1	0	0	12	3	15	Valid
Aktual	P3	WS 1	P2	2	16	3	19	Valid
Aktual	P20	WS 5	P17	4	130	4	134	Valid
Optimasi Solver	P14	WS 1	0	0	5	2	7	Valid

Optimasi Solver	P16	WS 1	P14	4	11	2	13	Valid
Optimasi Solver	P2	WS 1	P 16	1	14	4	18	Valid
Optimasi Solver	P6	WS 5	P 9	1	106	4	110	Valid

Sebagai contoh, P14 pada WS1 hasil *Hasil optimasi Solver* memiliki *start time* sebesar 5 menit dan *duration* sebesar 2 menit. Perhitungan *finish time*-nya adalah:

$$\text{Finish P14, WS 1} = \text{Start P14, WS 1} + \text{Duration P14, WS 1}$$

$$\text{Finish P14, WS 1} = 5 + 2 = 7$$

Hasil tersebut menunjukkan bahwa nilai *finish time* pada kedua contoh sesuai dengan penjumlahan *start time* dan *duration*. Pemeriksaan yang sama dilakukan terhadap 100 operasi pada jadwal aktual dan 100 operasi pada *Hasil optimasi Solver*. Hasil pemeriksaan menunjukkan bahwa seluruh operasi memenuhi hubungan tersebut. Dengan demikian, tidak ditemukan ketidaksesuaian antara waktu mulai, waktu proses, dan waktu selesai.

C. Pengujian *Previous Job*

Previous job merupakan produk terakhir yang sebelumnya menggunakan processor yang sama. Penentuan *previous job* diperlukan untuk mengetahui urutan pemakaian mesin dan menentukan *setup time* yang diterima oleh produk berikutnya.

Apabila suatu produk merupakan pekerjaan pertama pada mesin, *previous job* diberikan nilai 0. Sebaliknya, apabila mesin telah digunakan oleh produk lain, *previous job* harus menunjukkan produk terakhir yang sebelumnya menggunakan mesin tersebut.

Sebagai contoh, urutan penggunaan mesin MA pada WS1 hasil *Hasil optimasi Solver* adalah P14 → P16 → P2 → P19 → P18 → P13. Berdasarkan urutan tersebut, *previous job* setiap produk pada mesin MA ditunjukkan pada tabel 5.2 berikut.

Tabel 5. 2 Pengujian *Previous Job* pada Mesin MA di WS1 Hasil Optimasi Solver

Urutan	Produk	<i>Previous Job</i>
1	P14	0
2	P16	P14
3	P2	P16
4	P19	P2
5	P18	P19
6	P13	P18

P14 merupakan produk pertama yang menggunakan mesin MA sehingga *previous job*-nya bernilai 0. Setelah P14 selesai, mesin MA digunakan oleh P16 sehingga *previous*

job P16 adalah P14. Mesin yang sama kemudian digunakan oleh P2 sehingga *previous job* P2 adalah P16.

Hasil pemeriksaan menunjukkan bahwa nilai *previous job* telah sesuai dengan produk terakhir yang sebelumnya menggunakan *processor* yang sama. Produk pertama pada setiap mesin memiliki *previous job* bernilai 0, sedangkan produk *berikutnya* memiliki *previous job* sesuai dengan urutan penggunaan mesin.

D. Pengujian *Setup Time* dan Kesiapan Mesin

Setup time merupakan waktu persiapan yang diperlukan ketika mesin berpindah dari *previous job* menuju *current job*. Karena *setup time* pada penelitian ini bersifat *sequence-dependent*, nilainya ditentukan berdasarkan pasangan produk sebelumnya dan produk yang akan diproses.

Apabila suatu produk menjadi pekerjaan pertama pada mesin, *setup time* bernilai 0. Apabila terdapat *previous job*, nilai *setup* diambil dari matriks *sequence-dependent setup time* berdasarkan baris *previous job* dan kolom *current job*.

Pada WS1 hasil Optimasi *Solver*, P14 merupakan pekerjaan pertama pada mesin MA sehingga *previous job* dan *setup time* P14 bernilai 0. Setelah P14, mesin MA digunakan oleh P16. *Setup time* perpindahan P14 menuju P16 sebesar 4 menit. P14 selesai pada menit ke-7. Contohnya dapat dilihat pada Gambar 5.1.

		Produk Sekarang															
Produk Sebelumnya		P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16
	P1	0	4	1	3	3	4	1	1	3	2	3	3	3	4	1	
	P2	2	0	2	4	4	2	2	2	4	4	1	1	4	2	2	1
	P3	3	3	0	4	4	4	4	3	2	2	3	2	3	4	3	4
	P4	3	3	1	0	3	2	4	1	4	2	2	2	1	2	1	2
	P5	3	4	1	4	0	3	2	1	4	2	4	4	2	2	2	2
	P6	3	2	2	4	2	0	2	4	2	3	4	3	4	2	3	4
	P7	0	1	2	3	1	4	0	1	4	4	4	1	1	1	3	1
	P8	4	1	4	4	4	3	3	0	1	4	3	3	1	3	1	2
	P9	3	1	4	4	2	1	4	3	0	2	4	1	3	4	4	2
	P10	1	3	2	3	1	1	2	3	3	0	3	3	1	3	3	2
	P11	3	4	3	1	4	1	4	4	2	1	0	3	1	3	3	1
	P12	3	3	2	4	2	2	1	2	1	1	2	0	4	4	4	4
	P13	2	3	4	1	3	2	1	1	1	3	2	1	0	1	1	2
	P14	1	2	1	1	3	2	4	4	3	3	2	4	4	0	1	4
	P15	4	4	2	3	2	3	1	1	1	4	1	3	1	2	0	4
	P16	1	1	3	2	2	1	4	2	4	2	4	3	4	3	3	0
	P17	4	1	4	2	2	2	3	3	4	4	1	1	4	3	2	4
	P18	3	4	3	2	3	3	3	4	4	3	4	1	1	4	4	3
	P19	4	1	1	2	3	3	2	3	3	3	4	4	2	4	1	4
	P20	4	2	4	1	1	4	3	3	1	1	3	3	4	4	3	4

Gambar 5. 1 Matriks *Setup Time*

Oleh karena itu, waktu kesiapan mesin MA untuk memproses P16 dihitung sebagai berikut:

$$\text{Waktu kesiapan MA untuk P16} = \text{Finish P14} + \text{Setup P14, P16}$$

$$\text{Waktu kesiapan MA untuk P16} = 7 + 4 = 11$$

Berikut ini contoh pada data penelitian yang diteliti.

Work Station 1							
Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish
P14	14	1	0	0	5	2	7
P10	10	2	0	0	6	3	9
P15	15	3	0	0	8	3	11
P16	16	1	14	4	11	2	13
P7	7	2	10	2	11	9	20
P2	2	1	16	1	14	4	18

Gambar 5. 2 Contoh Pengujian Setup Time pada Mesin MA *Work station 1* Hasil Optimasi Solver

P16 mulai diproses pada menit ke-11. Dengan demikian, waktu mulai P16 sesuai dengan waktu kesiapan mesin. Setelah P16 selesai pada menit ke-13, mesin MA digunakan oleh P2. Setup time perpindahan P16 menuju P2 sebesar 1 menit. Waktu kesiapan mesin untuk P2 adalah:

$$\text{Waktu kesiapan MA untuk P2} = \text{Finish P16} + \text{Setup P16, P2}$$

$$\text{Waktu kesiapan MA untuk P2} = 13 + 1 = 14$$

P2 mulai diproses pada menit ke-14. Dengan demikian, hubungan antara *previous job*, *setup time*, *finish time*, dan waktu kesiapan mesin telah terpenuhi.

E. Pengujian *Start Time*

Start time menunjukkan waktu suatu produk mulai diproses pada *work station* tertentu. Penentuan start time pada WS1 berbeda dengan penentuan start time pada WS2 sampai WS5. Pada WS1, *start time* ditentukan berdasarkan waktu yang paling akhir antara *release time* dan waktu kesiapan mesin. Hubungannya dapat dilihat berdasarkan rumus (2.4).

Sebagai contoh pada jadwal aktual perusahaan, P1 merupakan pekerjaan pertama pada mesin yang digunakan di WS1 sehingga tidak memiliki *previous job* dan *setup time* bernilai 0. *Release time* P1 adalah 12 menit. Oleh karena itu:

$$ST(P1, WS 1) = \max(12, 0 + 0)$$

$$ST(P1, WS 1) = \max(12, 0) = 12$$

Hasil tersebut menunjukkan bahwa P1 mulai diproses pada menit ke-12 sesuai dengan *release time* yang dimilikinya. Berikut ini contoh pada data penelitian yang diteliti.

Work Station 1								Work Station 2			
Product ID	Job Sequence	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start
P1	1	2	0	0	12	3	15	2	0	0	15
P2	2	1	0	0	10	4	14	1	0	0	14

Gambar 5. 3 Contoh Perhitungan *Start Time* pada *Work station 1*

Pada WS2 sampai WS5, *start time* ditentukan berdasarkan waktu yang paling akhir antara *finish time* produk pada *work station* sebelumnya dan waktu kesiapan mesin

pada *work station* yang sedang digunakan. Hubungannya dapat dilihat berdasarkan rumus (2.5).

Sebagai contoh pada jadwal aktual perusahaan, P1 selesai di WS1 pada waktu ke-15. Karena mesin yang digunakan pada WS2 telah siap menerima pekerjaan berikutnya pada waktu tersebut, maka:

$$ST(P1, WS 2) = \max(15, 15)$$

$$ST(P16, WS 2) = 15$$

Dengan demikian, P1 dapat langsung memasuki WS2 setelah menyelesaikan proses di WS1 tanpa mengalami waktu tunggu. Pada kondisi lain, apabila mesin pada *work station* berikutnya belum siap karena masih digunakan oleh produk lain atau masih menjalani *setup*, maka produk harus menunggu hingga mesin tersedia.

Penggunaan nilai maksimum memastikan bahwa produk hanya mulai diproses apabila dua kondisi telah terpenuhi, yaitu produk telah tersedia dan mesin yang dipilih telah siap. Hasil pemeriksaan menunjukkan bahwa seluruh nilai *start time* pada jadwal aktual dan *Hasil optimasi Solver* telah mempertimbangkan *release time*, *finish time* pada tahap sebelumnya, *finish time previous job*, serta *setup time*.

F. Pengujian Makespan

Makespan merupakan waktu penyelesaian seluruh produk pada sistem produksi. Nilai *makespan* diperoleh dari *finish time* terbesar pada *work station* terakhir, yaitu WS5. Hubungannya dapat dilihat berdasarkan rumus (2.1).

Pengujian dilakukan dengan membandingkan nilai *makespan* pada sel objektif dengan *finish time* terbesar pada WS5. Pada jadwal aktual perusahaan, *finish time* terbesar pada WS5 adalah 134 menit dan berasal dari P20. Dengan demikian:

$$C_{\max\text{aktual}} = \max C(i, 5) = 134$$

Berikut ini contoh pada data penelitian yang diteliti.

Tabel 5. 3 Hasil *Makespan* Keseluruhan

Work station 5							
Product ID	Job assignment	Process or	Previous Job	Setu p	Star t	Duratio n	Finis h
P1	1	2	0	0	32	4	36
P2	2	1	0	0	36	3	39
P3	3	2	1	1	42	5	47
P4	4	2	3	4	51	6	57
P5	5	1	2	4	52	4	56
P6	6	2	4	2	59	4	63
P7	7	2	6	2	65	5	70

Work station 5							
Product ID	Job assignment	Process or	Previous Job	Setup p	Start t	Duration n	Finish h
P8	8	2	7	1	71	4	75
P9	9	2	8	1	76	3	79
P10	10	2	9	2	81	4	85
P11	11	1	5	4	81	4	85
P12	12	2	10	3	88	3	91
P13	13	1	11	1	92	3	95
P14	14	2	12	4	98	2	100
P15	15	1	13	1	101	3	104
P16	16	2	14	4	114	3	117
P17	17	2	16	3	122	4	126
P18	18	1	15	3	107	2	109
P19	19	1	18	1	120	6	126
P20	20	2	17	4	130	4	134
Makespan							134

Pada jadwal optimasi *Solver*, *finish time* terbesar pada WS 5 adalah 110 menit dan berasal dari P6. Dengan demikian:

$$C_{\max} \text{optimasi} = \max C(i, 5) = 110$$

Nilai tersebut juga sama dengan *makespan* yang tercantum pada *sheet* Optimasi Hasil optimasi *Solver*. Hasil ini menunjukkan bahwa formula *makespan* telah mengambil waktu penyelesaian maksimum pada WS5 secara tepat.

5.1.2 Pengujian Pemenuhan Batasan Penjadwalan

Pengujian pemenuhan batasan penjadwalan dilakukan untuk memastikan bahwa jadwal aktual perusahaan dan jadwal hasil Hasil optimasi *Solver* merupakan jadwal yang *feasible*. Jadwal dinyatakan *feasible* apabila seluruh produk diproses sesuai waktu ketersediaannya, melewati *work station* secara berurutan, menggunakan mesin yang tersedia, tidak mengalami duplikasi, dan tidak menggunakan mesin yang sama secara bersamaan.

A. Pengujian Kapasitas Mesin dan Non-Overlap

Setiap mesin hanya dapat memproses satu produk pada satu waktu. Pekerjaan berikutnya pada mesin yang sama hanya dapat dimulai setelah *previous job* selesai dan *setup time* telah dipenuhi. Hubungannya dapat dilihat berdasarkan rumus pada (2.9). Contoh pengujian penggunaan mesin MA pada WS1 Optimasi *Solver* ditunjukkan pada tabel 5.4.

Tabel 5. 4 Pengujian Non-Overlap pada Mesin MA di WS1 Solver 2

Urutan	Produk	Previous Job	Setup	Start	Finish	Status
1	P14	0	0	5	7	Valid

2	P16	P14	4	11	13	Valid
3	P2	P16	1	14	18	Valid
4	P19	P2	1	19	24	Valid
5	P18	P19	1	25	27	Valid
6	P13	P18	1	28	30	Valid

P16 mulai pada menit ke-11 setelah P14 selesai pada menit ke-7 dan setup selama 4 menit. P2 mulai pada menit ke-14 setelah P16 selesai pada menit ke-13 dan setup selama 1 menit. Pola yang sama terjadi pada P19, P18, dan P13.

Tidak terdapat dua interval proses yang saling beririsan pada mesin MA. Pemeriksaan terhadap seluruh mesin pada jadwal aktual dan *Hasil optimasi Solver* juga menunjukkan bahwa tidak terdapat overlap pada mesin yang sama. Tidak ditemukannya overlap merupakan kondisi yang benar karena satu mesin tidak dapat memproses dua produk secara bersamaan.

Beberapa produk tetap dapat diproses pada waktu yang sama apabila menggunakan mesin yang berbeda. Proses simultan pada mesin paralel dalam *work station* yang sama diperbolehkan. Proses pada *work station* berbeda juga dapat berjalan bersamaan selama urutan proses masing-masing produk tetap terpenuhi.

B. Pengujian Keunikan *Job*

Setiap produk harus dijadwalkan tepat satu kali. Produk tidak boleh muncul lebih dari satu kali dan tidak boleh ada produk yang tidak dimasukkan ke dalam jadwal. Pada jadwal aktual, urutan pekerjaan terdiri atas P1 sampai P20. Pada hasil optimasi *Solver*, urutan pekerjaan yang dihasilkan adalah P14 → P10 → P15 → P16 → P7 → P2 → P5 → P17 → P19 → P1 → P20 → P11 → P3 → P18 → P4 → P13 → P12 → P9 → P8 → P6.

Hasil pemeriksaan menunjukkan bahwa P1 sampai P20 masing-masing muncul satu kali pada kedua jadwal. Tidak terdapat produk yang mengalami duplikasi maupun produk yang tidak dijadwalkan. Setiap produk juga memiliki satu operasi pada masing-masing *work station* sehingga seluruh produk menjalani lima tahap proses secara lengkap.

Pada proses optimasi, keunikan *job* dikendalikan melalui batasan AllDifferent pada variabel *Job assignment*. Batasan tersebut memastikan bahwa seluruh nilai *job* berbeda sehingga satu produk tidak dapat menempati lebih dari satu posisi urutan.

C. Pengujian Sifat *Non-Preemptive*

Non-preemptive berarti bahwa setiap operasi yang telah dimulai pada suatu mesin tidak boleh dihentikan atau disela sampai operasi tersebut selesai. Sifat *non-preemptive* tidak berarti bahwa produk harus terus diproses tanpa menunggu sejak WS1 sampai WS5.

Produk masih dapat mengalami waktu tunggu sebelum memasuki *work station* berikutnya apabila mesin tujuan belum tersedia.

Setiap operasi pada jadwal aktual dan hasil optimasi *Solver* direpresentasikan oleh satu interval kontinu dari *start time* sampai *finish time*. Tidak terdapat operasi yang dibagi ke dalam dua atau lebih interval pengerjaan. Dengan demikian, seluruh operasi telah memenuhi asumsi

5.1.3 Pengujian Stabilitas Hasil *Solver*

Pengujian stabilitas hasil *Solver* dilakukan karena metode *Evolutionary* memiliki proses pencarian solusi yang bersifat probabilistik. Metode tersebut dapat menghasilkan kombinasi *Job assignment* dan *machine assignment* yang berbeda pada setiap *running* meskipun menggunakan fungsi tujuan dan batasan yang sama.

Perbedaan hasil dapat terjadi karena proses pencarian melibatkan pembentukan populasi awal, evaluasi *fitness*, seleksi, *crossover*, dan mutasi. Populasi awal dan perubahan kandidat solusi selama pencarian dapat mengarahkan *Solver* menuju area solusi yang berbeda. Oleh karena itu, *Solver* perlu dijalankan lebih dari satu kali.

Pada penelitian ini, *Solver* dijalankan sebanyak lima kali menggunakan sel objektif, variabel Keputusan, fungsi tujuan, dan Batasan yang sama. Nilai *makespan* dari lima percobaan ditunjukkan pada Tabel 5.5.

Tabel 5. 5 Hasil Lima Percobaan *Solver Evolutionary*

Percobaan	Makespan	Selisih dari Jadwal Aktual	Persentase Perbaikan
Optimasi <i>Solver</i> 1	122	12	8,96%
Optimasi Hasil optimasi <i>Solver</i>	110	24	17,91%
Optimasi <i>Solver</i> 3	125	9	6,72%
Optimasi <i>Solver</i> 4	127	7	5,22%
Optimasi <i>Solver</i> 5	118	16	11,94%

Berdasarkan hasil lima kali running *Solver*, nilai *makespan* yang diperoleh menunjukkan variasi hasil. Perbedaan ini terjadi karena metode *Evolutionary* pada Microsoft Excel *Solver* bersifat heuristik atau metaheuristik, sehingga proses pencarian solusi dilakukan secara iteratif dan probabilistik. Dengan demikian, metode ini tidak menjamin bahwa solusi optimal global atau global optimum dapat langsung ditemukan hanya dalam satu kali percobaan.

Nilai *makespan* 110 menit dipilih sebagai hasil akhir karena merupakan nilai *makespan* terkecil dari lima kali percobaan *Solver*. Namun, hasil tersebut lebih tepat dinyatakan sebagai solusi terbaik yang berhasil diperoleh dalam ruang lingkup percobaan

penelitian ini, bukan sebagai jaminan global optimum. Oleh karena itu, jumlah running, parameter Solver, dan metode optimasi yang digunakan menjadi aspek yang perlu diperhatikan apabila penelitian selanjutnya ingin memperoleh hasil yang lebih stabil dan mendekati global optimum.

Selisih *makespan* dihitung mengurangi *makespan* jadwal aktual sebesar 134 menit dengan *makespan* setiap hasil Solver. Persentase perbaikan dihitung menggunakan rumus pada (2.12). seluruh percobaan menghasilkan nilai *makespan* yang lebih rendah dibandingkan jadwal aktual Perusahaan. Hasil optimasi Solver 1 menghasilkan *makespan* sebesar 122 menit, optimasi *hasil optimasi Solver* sebesar 110 menit, optimasi Solver 3 sebesar 125 menit, optimasi Solver 4 sebesar 127 menit, optimasi Solver 5 sebesar 118 menit.

Untuk jadwal usulan yang terpilih adalah hasil optimasi *hasil optimasi Solver* karena menghasilkan *makespan* terendah, yaitu 110 menit. Nilai tersebut lebih rendah sebanyak 24 menit atau 17,91% dibandingkan jadwal aktual perusahaan sebesar 134 menit. Selain memiliki nilai fungsi tujuan paling rendah, hasil optimasi *hasil optimasi Solver* juga telah memenuhi seluruh Batasan model. Jadwal tersebut tidak memiliki duplikasi produk, tidak melanggar *release time*, mengikuti urutan WS 1 hingga WS 5, menggunakan *processor* dalam rentang yang tersedia, dan tidak menghasilkan *overlap* pada mesin yang sama.

5.2 Analisis Hasil Penjadwalan Aktual dan Hasil Optimasi Solver

Analisis hasil penjadwalan dilakukan dengan membandingkan jadwal aktual Perusahaan dan jadwal hasil optimasi Solver. Jadwal aktual menggunakan urutan P1 hingga P20, sedangkan hasil optimasi *olver* menghasilkan perubahan urutan pekerjaan dan pemilihan mesin pada setiap *work station*. Perbandingan dilakukan untuk menjelaskan perubahan keputusan penjadwalan serta pengaruhnya terhadap waktu penyelesaian keseluruhan produk.

Jadwal aktual dan jadwal hasil optimasi *olver* telah dinyatakan *feasible* berdasarkan pengujian pada pembahasan sub-bab 5.1. kedua jadwal telah memenuhi Batasan *release time*, urutan proses antar-*work station*, keunikan produk, kapasitas mesin, dan *non-preemptive*. Dengan demikian, perbedaan kinerja kedua jadwal tidak disebabkan oleh pelanggaran Batasan pada jadwal aktual, tetapi oleh perbedaan kombinasi *Job assignment* dan *machine assignment*.

Analisis pada bagian ini dibagi menjadi empat aspek. Aspek pertama membahas perubahan *Job assignment* dan *machine assignment* sebagai variabel keputusan utama. Aspek kedua membahas perubahan *makespan* sebagai fungsi tujuan. Aspek ketiga membahas *idle time* sebagai indikator pendukung. Aspek keempat membahas perbedaan pola jadwal secara visual melalui *Gantt Chart*.

5.2.1 Perubahan *Job assignment* dan *Machine Assignment*

Job assignment menentukan posisi setiap produk dalam urutan pengerjaan, sedangkan *machine assignment* menentukan mesin yang digunakan oleh setiap produk pada masing-masing *work station*. Kedua keputusan tersebut saling berkaitan karena perubahan urutan produk akan mengubah *previous job*, *setup time*, waktu kesiapan mesin, dan waktu mulai proses. Sementara itu, perubahan mesin akan memengaruhi *processing time* karena waktu proses setiap produk bergantung pada mesin yang digunakan.

A. Perubahan *Job assignment* dan Pengaruhnya terhadap Aliran Produk

Pada jadwal aktual perusahaan, urutan pengerjaan produk disusun berdasarkan nomor identitas produk, yaitu P1 sampai P20. Urutan tersebut tidak berubah berdasarkan *release time*, waktu proses, maupun pasangan *setup time*. Setelah dilakukan optimasi, hasil optimasi *Solver* menghasilkan susunan produk yang berbeda. Perbandingan kedua urutan ditunjukkan pada Tabel 5.6.

Tabel 5. 6 Perbandingan *Job assignment* dan *Machine Assignment* Jadwal Aktual

Urutan	Pengerjaan Produk di Jadwal Aktual	WS 1	WS 2	WS 3	WS 4	WS 5
1.	P1	MB	ME	MF	MH	ML
2.	P2	MA	MD	MG	MI	MK
3.	P3	MA	MD	MG	MJ	ML
4.	P4	MC	ME	MF	MH	ML
5.	P5	MA	ME	MG	MH	MK
6.	P6	MC	MD	MF	MI	ML
7.	P7	MA	ME	MG	MJ	ML
8.	P8	MC	MD	MF	MH	ML
9.	P9	MA	MD	MF	MJ	ML
10.	P10	MB	MD	MF	MI	ML
11.	P11	MA	MD	MG	MJ	MK
12.	P12	MB	ME	MF	MH	ML
13.	P13	MC	ME	MF	MI	MK
14.	P14	MC	MD	MG	MI	ML
15.	P15	MB	MD	MF	MH	MK
16.	P16	MA	ME	MF	MI	ML
17.	P17	MB	MD	MG	MI	ML
18.	P18	MC	ME	MG	MH	MK
19.	P19	MA	ME	MG	MJ	MK

Urutan	Pengerjaan Produk di Jadwal Aktual	WS 1	WS 2	WS 3	WS 4	WS 5
20.	P20	MB	MD	MF	MJ	ML

Tabel 5. 7 Perbandingan *Job assignment* dan Machine Assignment Hasil Optimasi Solver

Urutan	Pengerjaan Produk Hasil Optimasi <i>Solver</i>	WS 1	WS 2	WS 3	WS 4	WS 5
1.	P14	MA	MD	MF	MJ	ML
2.	P10	MB	ME	MG	MI	MK
3.	P15	MC	MD	MF	MJ	ML
4.	P16	MA	ME	MG	MH	ML
5.	P7	MB	MD	MF	MH	MK
6.	P2	MA	ME	MF	MI	MK
7.	P5	MC	MD	MG	MJ	MK
8.	P17	MC	MD	MG	MH	ML
9.	P19	MA	ME	MF	MI	ML
10.	P1	MB	MD	MG	MI	MK
11.	P20	MB	ME	MF	MJ	ML
12.	P11	MC	MD	MF	MH	ML
13.	P3	MC	MD	MG	MJ	MK
14.	P18	MA	ME	MG	MI	ML
15.	P4	MB	ME	MF	MH	ML
16.	P13	MA	ME	MF	MH	ML
17.	P12	MC	ME	MG	MI	MK
18.	P9	MB	MD	MG	MH	ML
19.	P8	MC	MD	MF	MJ	MK
20.	P6	MB	MD	MF	MI	ML

Berdasarkan Tabel 5.6 dan Tabel 5.7 dapat dilihat bahwa *Solver* menghasilkan perubahan urutan pengerjaan produk secara menyeluruh. Pada jadwal aktual, urutan pengerjaan mengikuti nomor identitas produk. Perubahan tersebut menunjukkan bahwa *Solver* tidak menggunakan nomor identitas produk sebagai dasar prioritas pengerjaan. Produk yang berada pada posisi belakang dalam jadwal aktual dapat dipindahkan ke posisi awal apabila kombinasi tersebut memberikan nilai *makespan* yang lebih rendah.

Perubahan urutan pada hasil optimasi *Solver* terjadi pada seluruh posisi pekerjaan. P14 berpindah dari posisi ke-14 menjadi posisi pertama, P10 berpindah dari posisi ke-10 menjadi posisi kedua, P15 berpindah dari posisi ke-15 menjadi posisi ketiga, dan P16 berpindah dari posisi ke-16 menjadi posisi keempat. Sebaliknya, P6 berpindah dari posisi keenam menjadi posisi terakhir, sedangkan P8 dan P9 ditempatkan mendekati bagian akhir jadwal.

Penempatan P14 pada posisi pertama berkaitan dengan *release time* produk tersebut yang paling awal, yaitu menit ke-5. Pada jadwal aktual, P14 baru mulai diproses di WS1

pada menit ke-36 karena berada pada posisi ke-14. Setelah ditempatkan pada urutan pertama, P14 dapat langsung mulai pada menit ke-5.

Perubahan urutan produk menyebabkan perubahan hubungan antara *previous job* dan *current job* pada mesin yang sama. Karena *setup time* dalam penelitian ini bersifat *sequence-dependent*, perubahan pasangan produk yang berurutan dapat menghasilkan kebutuhan waktu *setup* yang berbeda. Dengan demikian, perubahan *job assignment* tidak hanya memengaruhi posisi produk dalam antrean, tetapi juga memengaruhi waktu persiapan mesin yang diperlukan sebelum proses berikutnya dapat dimulai.

Selain itu, hasil optimasi tidak selalu menempatkan produk dengan *release time* paling awal pada posisi awal urutan. Beberapa produk yang tersedia lebih awal tetap ditempatkan pada bagian tengah atau akhir jadwal apabila keputusan tersebut memberikan kombinasi yang lebih baik terhadap keseluruhan sistem. Hal ini menunjukkan bahwa *Solver* tidak menerapkan aturan prioritas sederhana seperti *first come first served* atau *shortest processing time*, melainkan mengevaluasi dampak setiap kombinasi keputusan terhadap nilai *makespan* akhir.

Berdasarkan hasil tersebut dapat disimpulkan bahwa perbedaan utama antara jadwal aktual dan hasil optimasi terletak pada kombinasi *job assignment* dan *machine assignment* yang digunakan. Jadwal aktual telah memenuhi seluruh batasan model, namun belum mempertimbangkan interaksi menyeluruh antarproduk dan antar mesin. Hasil optimasi menunjukkan bahwa pengaturan ulang urutan produk serta pemilihan mesin yang berbeda masih mampu menghasilkan aliran produksi yang lebih efisien dan menjadi dasar tercapainya nilai *makespan* yang lebih rendah.

B. Perubahan *Machine Assignment* dan Pemanfaatan Mesin Paralel

Pemilihan mesin pada hasil optimasi *Solver* diperoleh melalui perubahan variabel *processor* pada setiap *work station*. Untuk setiap kombinasi urutan produk dan mesin yang terbentuk, formula Excel menghitung *duration*, *previous job*, *setup time*, *start time*, *finish time*, dan *makespan*. *Solver* kemudian mengevaluasi setiap kombinasi berdasarkan fungsi tujuan minimasi *makespan* dengan tetap memenuhi batasan jumlah mesin, keunikan produk, urutan proses, serta kapasitas mesin.

Dengan demikian, *Solver* tidak memilih mesin hanya berdasarkan waktu proses yang paling pendek. Mesin dengan waktu proses lebih panjang masih dapat dipilih apabila tersedia lebih awal, memiliki antrean lebih pendek, menghasilkan pasangan *setup time*

yang lebih menguntungkan, atau memungkinkan beberapa pekerjaan berlangsung secara paralel.

Data perubahan alokasi mesin diperoleh dengan membandingkan mesin yang digunakan oleh produk yang sama pada jadwal aktual perusahaan dan hasil optimasi *Solver*. Perbandingan dilakukan berdasarkan *Product ID*. Setiap produk memiliki lima keputusan mesin, yaitu satu keputusan pada masing-masing WS1 sampai WS5. Dengan jumlah 20 produk dan lima *work station*, total keputusan alokasi mesin yang dibandingkan adalah:

$$\text{Total keputusan alokasi mesin} = 20 \text{ produk} \times 5 \text{ work station}$$

$$\text{Total keputusan alokasi mesin} = 100 \text{ operasi}$$

Kriteria perbandingan yang digunakan adalah sebagai berikut:

1. Apabila mesin aktual sama dengan mesin pada hasil optimasi *Solver*, keputusan dinyatakan tetap.
2. Apabila mesin aktual berbeda dari mesin pada hasil optimasi *Solver*, keputusan dinyatakan berubah.

Sebagai contoh, perbandingan alokasi mesin untuk P14 ditunjukkan pada Tabel 5.8.

Tabel 5. 8 Contoh Perubahan Alokasi Mesin Produk P14

<i>Work station</i>	Mesin Aktual	Mesin Hasil Optimasi <i>Solver</i>	Status
WS 1	MC	MA	Berubah
WS 2	MD	MD	Tetap
WS 3	MG	MF	Berubah
WS 4	MI	MJ	Berubah
WS 5	ML	ML	Tetap

P14 mengalami perubahan mesin pada WS1, WS3, dan WS4, sedangkan mesin pada WS2 dan WS5 tetap. Dengan demikian, dari lima keputusan mesin untuk P14, tiga keputusan berubah dan dua keputusan tetap. Berdasarkan Tabel 5.8, Prosedur yang sama diterapkan pada seluruh produk P1 sampai P20. Sebagai contoh, pada WS1 terdapat 13 produk yang menggunakan mesin berbeda antara jadwal aktual dan hasil optimasi *Solver*, sedangkan tujuh produk tetap menggunakan mesin yang sama. Persentase perubahan alokasi mesin pada WS1 dihitung sebagai berikut:

$$\begin{aligned}
 &\text{Persentase perubahan WS1} \\
 &= (\text{Jumlah produk dengan mesin berubah} / \text{Jumlah produk}) \times 100\% \\
 &= \left(\frac{13}{20} \right) \times 100\% = 65\%
 \end{aligned}$$

Pada WS2, terdapat enam produk dengan mesin berubah dan 14 produk dengan mesin tetap, sehingga persentase perubahannya adalah:

$$\text{Persentase perubahan WS2} = \left(\frac{6}{20}\right) \times 100\% = 30\%$$

Perhitungan yang sama dilakukan untuk WS3, WS4, dan WS5. Ringkasan hasil perbandingan alokasi mesin ditunjukkan pada Tabel 5.9.

Tabel 5. 9 Ringkasan Perubahan *Machine Assignment* pada Setiap *Work station*

<i>Work station</i>	Mesin Berubah	Mesin Tetap	Persentase Perubahan
WS 1	13 produk	7 produk	65%
WS 2	6 produk	14 produk	30%
WS 3	10 produk	10 produk	50%
WS 4	14 produk	6 produk	70%
WS 5	11 produk	9 produk	55%
Total	54 operasi	46 operasi	54%

Hasil tersebut menunjukkan bahwa lebih dari separuh keputusan alokasi mesin pada hasil optimasi *Solver* berbeda dari jadwal aktual perusahaan. Perubahan terbesar terjadi pada WS4, yaitu sebanyak 14 dari 20 produk atau sebesar 70%, sedangkan perubahan terkecil terjadi pada WS2, yaitu enam produk atau sebesar 30%.

Besarnya perubahan pada WS4 tidak secara langsung menunjukkan bahwa WS4 merupakan *bottleneck*. Data tersebut hanya menunjukkan bahwa *Solver* banyak mengubah alokasi produk pada mesin MH, MI, dan MJ ketika mencari kombinasi dengan *makespan* lebih rendah. Penentuan *bottleneck* memerlukan analisis tambahan terhadap kapasitas, utilisasi, beban kerja, dan antrean pada setiap mesin.

Jumlah perubahan mesin hanya menunjukkan besarnya perbedaan antara jadwal aktual dan hasil optimasi *Solver*. Nilai tersebut belum menunjukkan apakah perubahan yang dilakukan menghasilkan pengaruh positif atau negatif terhadap waktu penyelesaian. Oleh karena itu, analisis dilanjutkan dengan melihat contoh perubahan urutan, mesin, waktu proses, waktu mulai, dan waktu selesai produk.

Contoh perubahan pada P14 dan P7 ditunjukkan pada Tabel 5.10.

Tabel 5. 10 Perbandingan Perubahan Urutan, Mesin, dan Waktu Penyelesaian Produk P14 dan P7

Produk	Kondisi	Proses Urutan	Mesin WS 1	Durasi WS 1	Start Time WS 1	Finish Time WS 5
P14	Aktual	14	MC	2	36	100
P14	Optimasi	1	MA	2	5	21
P7	Aktual	7	MA	5	29	70
P7	Optimasi	5	MB	9	11	61

Pada P14, mesin WS1 berubah dari MC pada jadwal aktual menjadi MA pada hasil optimasi *Solver*, tetapi waktu prosesnya tetap sebesar 2 menit. Meskipun durasi pada WS1

tidak mengalami penurunan, P14 selesai jauh lebih cepat, yaitu dari menit ke-100 menjadi menit ke-21.

Percepatan P14 menunjukkan bahwa hasil tersebut tidak terutama disebabkan oleh pemilihan mesin dengan waktu proses lebih pendek. Faktor yang lebih dominan adalah perpindahan P14 dari urutan ke-14 menjadi urutan pertama. Posisi tersebut memungkinkan P14 mulai tepat pada *release time*-nya, yaitu menit ke-5, dibandingkan menit ke-36 pada jadwal aktual.

Hasil tersebut menunjukkan bahwa perubahan *job assignment* dan alokasi mesin tidak dapat dianalisis secara terpisah. Mesin yang digunakan menentukan durasi dan kesiapan proses, sedangkan posisi urutan menentukan kapan produk memperoleh kesempatan untuk menggunakan mesin tersebut.

Temuan yang berbeda terlihat pada P7. Pada jadwal aktual, P7 menggunakan MA dengan waktu proses 5 menit. Pada hasil optimasi *Solver*, P7 dialokasikan ke MB dengan waktu proses 9 menit. Dengan demikian, waktu proses P7 pada WS1 justru meningkat sebesar 4 menit.

Meskipun menggunakan mesin dengan waktu proses lebih panjang, P7 mulai pada menit ke-11 dan selesai pada WS5 pada menit ke-61. Pada jadwal aktual, P7 mulai pada menit ke-29 dan selesai pada WS5 pada menit ke-70. Dengan demikian, P7 tetap selesai 9 menit lebih cepat setelah dilakukan optimasi.

Kondisi tersebut menunjukkan bahwa keputusan yang terlihat kurang baik pada satu operasi dapat memberikan hasil yang lebih baik pada tingkat sistem. Mesin MA memang memiliki waktu proses lebih pendek untuk P7, tetapi produk harus menunggu lebih lama pada jadwal aktual. Penggunaan MB pada hasil optimasi *Solver* memungkinkan P7 mulai lebih awal dan memasuki tahapan berikutnya pada waktu yang lebih sesuai.

Temuan tersebut memperlihatkan bahwa mesin dengan waktu proses paling pendek belum tentu menjadi mesin terbaik bagi keseluruhan jadwal. Pemilihan mesin juga perlu mempertimbangkan waktu kesiapan mesin, antrean produk, *setup time*, dan pengaruhnya terhadap waktu mulai pada *work station* berikutnya.

Pada pemanfaatan mesin paralel Perubahan alokasi mesin memungkinkan produk didistribusikan pada beberapa mesin paralel dalam *work station* yang sama. Pemanfaatan mesin paralel terlihat pada bagian awal jadwal hasil optimasi *Solver* pada Tabel 5.11 berikut.

Tabel 5. 11 Contoh Pemanfaatan Mesin Paralel pada *Work station* 1 Hasil Optimasi Solver

Work station 1							
Product ID	Job assignment	Prosesor	Previous Job	Setup	Start	Duration	Finish
P14	14	1	0	0	5	2	7
P10	10	2	0	0	6	3	9
P15	15	3	0	0	8	3	11

P14 mulai diproses pada MA pada menit ke-5, P10 mulai pada MB pada menit ke-6, dan P15 mulai pada MC pada menit ke-8. Ketiga produk tersebut dapat diproses dalam interval waktu yang berdekatan karena menggunakan tiga mesin berbeda pada WS1.

Pemrosesan tersebut tidak termasuk *overlap* yang melanggar batasan karena setiap produk menggunakan mesin yang berbeda. Dalam sistem *flexible flow shop*, beberapa pekerjaan diperbolehkan berlangsung secara bersamaan selama tidak menggunakan mesin yang sama dan urutan proses masing-masing produk tetap terpenuhi. Pemanfaatan proses paralel menyebabkan panjang horizon produksi tidak ditentukan oleh penjumlahan seluruh durasi pekerjaan. Beberapa durasi proses dapat berlangsung pada waktu yang sama, sehingga penyelesaian keseluruhan produk dapat menjadi lebih pendek.

Perubahan alokasi mesin juga memengaruhi distribusi beban. Pada WS2, total waktu proses MD dan ME pada jadwal aktual masing-masing sebesar 56 dan 40 menit. Pada *Hasil optimasi Solver*, nilainya menjadi 56 dan 54 menit. Selisih beban berkurang dari 16 menjadi 2 menit.

Pada WS5, waktu proses MK dan ML pada jadwal aktual masing-masing sebesar 25 dan 51 menit. Pada *Hasil optimasi Solver*, nilainya menjadi 51 dan 44 menit. Pekerjaan pada tahap akhir menjadi lebih tersebar dan tidak lagi terlalu terkonsentrasi pada ML. Namun, pemerataan tidak terjadi pada seluruh *work station*. Pada WS1, total waktu proses MA, MB, dan MC pada jadwal aktual masing-masing sebesar 31, 19, dan 17 menit. Setelah optimasi, nilainya menjadi 17, 39, dan 31 menit. *Solver* lebih banyak menggunakan MB dan MC.

Hasil tersebut menunjukkan bahwa pemerataan beban bukan fungsi tujuan model. Pemerataan yang terjadi pada WS2 dan WS5 merupakan konsekuensi dari pencarian *makespan* yang rendah, bukan karena *Solver* secara langsung meminimalkan ketidakseimbangan beban. Oleh karena itu, hasil *Solver* tidak dapat diklaim telah menyeimbangkan seluruh mesin.

C. Dampak perubahan Urutan dan Alokasi Mesin terhadap Waktu Produksi

Setelah perubahan urutan dan alokasi mesin dianalisis pada tingkat produk dan mesin, pembahasan selanjutnya diarahkan pada dampak agregat keputusan tersebut terhadap waktu produksi. Komponen yang dibandingkan adalah total beban waktu proses agregat dan total *setup time*.

Tabel 5. 12 Perbandingan Total Duration dan *Setup Time* antara Jadwal Aktual dan Hasil Optimasi *Solver*

<i>Work station</i>	Σ duration aktual	Σ duration solver	Perubahan	<i>Setup Time</i> Aktual	<i>Setup Time</i> Optimal Solver	Perubahan
WS 1	67	87	Naik 20	45	31	Turun 14
WS 2	96	110	Naik 14	44	34	Turun 10
WS 3	125	134	Naik 9	46	28	Turun 18
WS 4	113	122	Naik 9	48	35	Turun 13
WS 5	76	95	Naik 19	45	37	Turun 8
Total	477	548	Naik 71	228	165	Turun 63

Berdasarkan tabel 5.12 total beban waktu proses meningkat sebesar 71 menit. Peningkatan tersebut terjadi karena waktu proses bersifat *machine-dependent*. Ketika *Solver* mengalihkan produk ke mesin lain, waktu proses suatu produk dapat menjadi lebih panjang daripada waktu pada mesin aktual.

Sebaliknya, total *setup time* turun dari 228 menjadi 165 menit. Besarnya penurunan sebanyak 63 menit. Penurunan *setup time* terjadi pada seluruh *work station*. Penurunan terbesar terjadi di WS3, yaitu dari 46 menjadi 28 menit. Kondisi tersebut menunjukkan bahwa urutan hasil *Solver* menghasilkan pasangan *previous job–current job* yang secara agregat membutuhkan waktu persiapan lebih kecil.

Penurunan *setup time* terjadi pada seluruh *work station*. Penurunan terbesar terjadi di WS3, yaitu dari 46 menjadi 28 menit. Kondisi tersebut menunjukkan bahwa urutan hasil *Solver* menghasilkan pasangan *previous job–current job* yang secara agregat membutuhkan waktu persiapan lebih kecil.

Temuan yang paling penting adalah meningkatnya total beban waktu proses ketika *makespan* justru menurun. Sekilas, kondisi tersebut terlihat tidak normal. Namun, total beban waktu proses merupakan penjumlahan durasi seluruh operasi, sedangkan *makespan* menunjukkan panjang waktu kalender sejak proses dimulai sampai produk terakhir selesai.

Pada sistem dengan mesin paralel, beberapa operasi dapat berlangsung secara bersamaan. Oleh karena itu, total durasi seluruh operasi dapat meningkat tanpa

memperpanjang waktu kalender penyelesaian apabila aktivitas dapat dilakukan secara lebih paralel. Hasil tersebut menunjukkan bahwa minimasi *makespan* tidak identik dengan minimasi total waktu proses. *Hasil optimasi Solver* bukan jadwal dengan jumlah waktu proses terkecil, tetapi merupakan solusi terbaik dari lima percobaan berdasarkan waktu selesai seluruh batch. Penurunan *setup time* turut membantu memperpendek jadwal karena waktu yang digunakan untuk persiapan antarproduk menjadi lebih kecil. Dengan demikian, kualitas jadwal ditentukan oleh interaksi antara urutan produk, alokasi mesin, *setup time*, kesiapan mesin, dan pemanfaatan proses paralel.

Temuan tersebut juga menunjukkan keterbatasan fungsi tujuan tunggal. Model belum memberikan penalti terhadap total waktu proses yang lebih besar, biaya penggunaan mesin, konsumsi energi, atau ketidakseimbangan beban. Oleh karena itu, jadwal *Hasil optimasi Solver* lebih baik berdasarkan *makespan*, tetapi belum tentu lebih baik berdasarkan seluruh indikator produksi.

5.2.2 Perbandingan *Makespan*

Makespan merupakan fungsi tujuan utama dalam penelitian dan menunjukkan waktu penyelesaian seluruh produk pada WS5. Semakin kecil *makespan*, semakin singkat total waktu yang diperlukan untuk menyelesaikan satu batch produksi. Perbandingan *makespan* dari jadwal aktual perusahaan dan hasil optimasi *Solver* dapat dilihat pada tabel 5.13 berikut.

Tabel 5. 13 Perbandingan *Makespan* Jadwal Aktual dan Hasil Optimasi Solver

Kondisi	<i>Makespan</i>
Jadwal Aktual	134 menit
<i>Hasil optimasi Solver</i>	110 menit
Penurunan	24 menit
Persentase Penurunan	17,91%

Hasil tersebut menunjukkan bahwa seluruh 20 produk dapat diselesaikan 24 menit lebih cepat berdasarkan jadwal optimasi *Solver*. Pada jadwal aktual, produk yang menentukan *makespan* adalah P20. Setelah optimasi, produk terakhir berubah menjadi P6. Perubahan tersebut menunjukkan bahwa *Solver* mengatur ulang posisi dan aliran seluruh produk, bukan hanya mempercepat produk yang sebelumnya selesai paling akhir.

Penurunan *makespan* merupakan hasil gabungan dari perubahan urutan, perubahan alokasi mesin, pengurangan *setup time*, pemanfaatan mesin paralel, serta perubahan jeda antarpekerjaan.

A. Perubahan Waktu Penyelesaian Setiap *Work station*

Perbandingan waktu penyelesaian terakhir setiap *work station* dapat dilihat pada Tabel 5.14 Gambar. Nilai pada kolom jadwal aktual diperoleh dari waktu *finish time* terbesar pada masing-masing *work station* dalam jadwal aktual perusahaan. Dengan demikian, nilai tersebut merepresentasikan waktu ketika seluruh pekerjaan pada *work station* terkait telah selesai diproses. Sementara itu, nilai pada kolom hasil optimasi *Solver* diperoleh dari waktu *finish time* terbesar pada masing-masing *work station* hasil optimasi.

Tabel 5. 14 Perbandingan Waktu Penyelesaian Terakhir pada Setiap *Work station*

<i>Work station</i>	Jadwal Aktual	Hasil Optimasi <i>Solver</i>	Penurunan
WS 1	61	56	5
WS 2	96	89	7
WS 3	113	99	14
WS 4	125	105	20
WS 5	134	110	24

Selisih waktu penyelesaian meningkat dari 5 menit di WS1 menjadi 24 menit di WS5. Pola tersebut menunjukkan bahwa pengaruh optimasi bersifat kumulatif dan tidak hanya terjadi pada satu tahap proses.

Pada WS1, perbaikan yang diperoleh masih relatif kecil karena perubahan yang terjadi terutama berupa percepatan waktu mulai dan penyesuaian alokasi mesin. Namun, percepatan tersebut menyebabkan produk memasuki WS2 lebih awal sehingga waktu tunggu sebelum proses berikutnya dapat berkurang. Efek yang sama kemudian berlanjut ke WS3, WS4, dan WS5.

Dalam sistem *flow shop*, keterlambatan yang terjadi pada satu tahap akan terbawa ke tahap berikutnya karena setiap operasi harus menunggu operasi sebelumnya selesai. Sebaliknya, ketika suatu produk dapat menyelesaikan proses lebih cepat pada tahap awal, keuntungan waktu tersebut dapat diwariskan ke seluruh tahapan berikutnya. Akibatnya, selisih waktu penyelesaian antara jadwal aktual dan hasil optimasi cenderung semakin besar pada tahap-tahap akhir.

Peningkatan selisih dari 5 menit di WS1 menjadi 24 menit di WS5 juga menunjukkan bahwa manfaat optimasi tidak hanya berasal dari pengurangan durasi proses pada satu mesin tertentu. Perbaikan terutama muncul dari kombinasi perubahan urutan produk, pemilihan mesin, pengurangan waktu tunggu, serta pemanfaatan mesin paralel yang lebih efektif. Ketika beberapa keputusan tersebut bekerja secara bersamaan, akumulasi penghematan waktu menjadi semakin besar pada tahap akhir produksi.

Dengan kata lain, hasil optimasi menghasilkan aliran produksi yang lebih lancar (*smoother flow*). Produk tidak hanya diproses lebih cepat pada satu operasi, tetapi juga mengalami lebih sedikit waktu menganggur (*idle waiting time*) di antara *work station*. Kondisi inilah yang menyebabkan perbedaan waktu penyelesaian semakin terlihat ketika produk mencapai WS4 dan WS5.

B. Perubahan Waktu Selesai Produk

Penurunan *makespan* tidak berarti seluruh produk selesai lebih cepat. Contoh perubahan *finish time* di WS5 ditunjukkan pada Tabel 5.15.

Tabel 5. 15 Perbandingan Waktu Selesai Produk pada Jadwal Aktual dan Hasil Optimasi *Solver*

Produk	Jadwal Aktual	Hail Optimasi <i>Solver</i>	Perubahan
P14	100	21	Lebih cepat 79 menit
P15	104	30	Lebih cepat 74 menit
P16	117	44	Lebih cepat 73 menit
P17	126	63	Lebih cepat 63 menit
P20	134	78	Lebih cepat 56 menit
P11	85	85	Tetap
P1	83	83	Lebih lambat 47 menit
P3	93	93	Lebih lambat 46 menit
P6	110	110	Lebih lambat 47 menit

Dari 20 produk, sembilan produk selesai lebih cepat, satu produk tetap, dan sepuluh produk selesai lebih lambat. Kondisi tersebut bukan kesalahan hasil optimasi. Fungsi tujuan hanya meminimalkan waktu selesai terbesar dan tidak meminimalkan waktu selesai setiap produk. *Solver* dapat menunda beberapa produk selama penundaan tersebut memperkecil waktu selesai seluruh batch.

Temuan ini menunjukkan konsekuensi penggunaan fungsi tujuan tunggal. Apabila perusahaan memiliki *due date* atau prioritas produk, minimasi *makespan* saja belum cukup.

Penurunan *makespan* sebesar 17,91% menunjukkan bahwa Hasil optimasi *Solver* menghasilkan jadwal yang lebih baik berdasarkan fungsi tujuan penelitian. Dalam sistem *make to stock*, penyelesaian batch yang lebih cepat berpotensi membuat mesin tersedia lebih awal untuk produksi berikutnya. Namun, penelitian belum menerapkan jadwal tersebut secara langsung dan belum mengukur peningkatan *output* atau produktivitas aktual.

Hasil perbandingan waktu selesai produk menunjukkan bahwa penurunan *makespan* tidak selalu menyebabkan seluruh produk selesai lebih cepat. Beberapa produk memang mengalami percepatan waktu selesai, tetapi beberapa produk lainnya dapat selesai lebih

lambat dibandingkan jadwal aktual. Kondisi ini terjadi karena fungsi tujuan yang digunakan dalam penelitian ini hanya berfokus pada minimasi *makespan*, yaitu waktu penyelesaian paling akhir dari seluruh produk.

Dengan demikian, *Solver* akan mencari kombinasi jadwal yang mampu memperkecil waktu selesai keseluruhan batch, meskipun terdapat produk tertentu yang harus ditempatkan pada urutan lebih akhir. Apabila perusahaan memiliki batas waktu penyelesaian atau *due date* untuk masing-masing produk, maka fungsi tujuan lain seperti total *tardiness*, *maximum lateness*, jumlah produk terlambat, atau *total flow time* perlu dipertimbangkan. Indikator tersebut penting apabila perusahaan tidak hanya ingin mempercepat penyelesaian *batch* secara keseluruhan, tetapi juga ingin memastikan ketepatan penyelesaian setiap produk terhadap target waktunya.

5.2.3 Analisis *Idle time*

Idle time digunakan sebagai indikator pendukung untuk melihat jeda antarpekerjaan pada mesin yang sama. Berikut ini merupakan perbandingan *idle time* antara jadwal aktual perusahaan dan hasil optimasi *Solver*.

Tabel 5. 16 Perbandingan *Idle time* Jadwal Aktual dan Hasil Optimasi *Solver*

<i>Work station</i>	Jadwal Aktual	Hasil Optimasi <i>Solver</i>	Perubahan
WS 1	0	5	Naik 5
WS 2	0	3	Naik 3
WS 3	15	8	Turun 7
WS 4	102	74	Turun 28
WS 5	71	39	Turun 32
Total	188	129	Turun 59

Jadi, secara keseluruhan total *idle time* turun sebesar 59 menit. Lalu, Penurunan tidak terjadi pada seluruh *work station*. WS1 dan WS2 mengalami peningkatan kecil, sedangkan WS3, WS4, dan WS5 mengalami penurunan.

Peningkatan di WS1 dan WS2 menunjukkan bahwa *Solver* tidak berusaha membuat setiap mesin selalu aktif. Sedikit jeda pada tahap awal masih dapat diterima apabila menghasilkan aliran yang lebih baik di tahap berikutnya.

Pada WS1, *idle time* meningkat, tetapi waktu penyelesaian tahap turun dari 61 menjadi 56 menit. Pada WS2, *idle time* meningkat, tetapi tahap selesai lebih cepat dari 96 menjadi 89 menit.

Hal tersebut menunjukkan bahwa nilai *idle time* yang lebih kecil pada satu tahap tidak selalu menghasilkan *makespan* lebih kecil. Posisi dan distribusi jeda lebih penting daripada sekadar menghilangkan seluruh waktu menganggur. Penurunan terbesar terjadi

di WS5 dan WS4. Hal ini menunjukkan bahwa pola kedatangan produk ke tahap akhir menjadi lebih sesuai dengan kesiapan mesin.

WS4 masih memiliki *idle time* terbesar pada *Hasil optimasi Solver*, yaitu 74 menit. Nilai tersebut tidak dapat langsung digunakan untuk menyebut WS4 sebagai *bottleneck*, karena mesin dapat menganggur akibat menunggu produk dari tahap sebelumnya.

Idle time dalam penelitian ini digunakan sebagai indikator pendukung untuk melihat jeda antarpekerjaan pada mesin yang sama. Namun, *idle time* yang dihitung masih terbatas pada *internal idle time*, yaitu waktu menganggur setelah pekerjaan sebelumnya selesai dan *setup* telah terpenuhi sampai pekerjaan berikutnya mulai diproses. Perhitungan ini belum mencakup seluruh aspek utilisasi mesin, seperti waktu menganggur sebelum pekerjaan pertama, waktu menganggur setelah pekerjaan terakhir, total waktu tersedia, waktu proses, dan waktu *setup*.

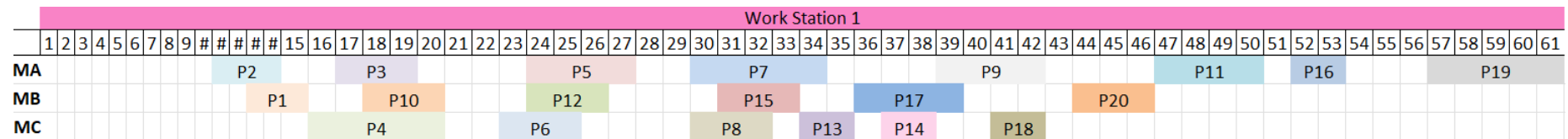
Oleh karena itu, meskipun hasil optimasi *Solver* menunjukkan penurunan *internal idle time*, hasil tersebut belum dapat sepenuhnya menggambarkan tingkat utilisasi mesin secara menyeluruh. Apabila perusahaan ingin mengevaluasi pemanfaatan mesin secara lebih lengkap, maka penelitian selanjutnya perlu menambahkan perhitungan utilisasi mesin yang mempertimbangkan seluruh horizon waktu produksi.

5.2.4 Analisis Gantt Chart

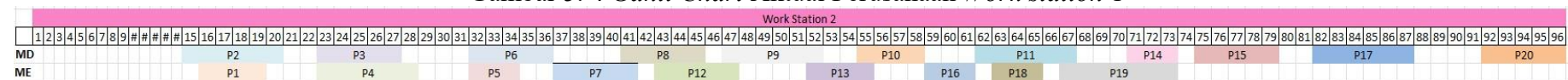
Gantt Chart digunakan untuk menunjukkan kapan setiap pekerjaan dikerjakan, mesin yang digunakan, serta proses yang berlangsung secara paralel.

A. Analisis *Gantt Chart* Jadwal Aktual Perusahaan

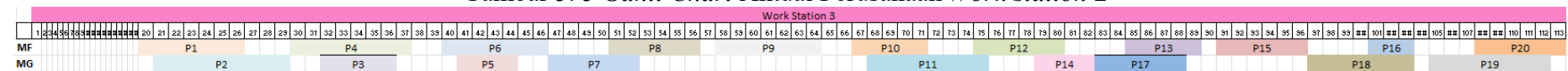
Berikut ini merupakan *Gantt Chart* untuk jadwal aktual perusahaan, dapat dilihat pada Gambar 5.4 hingga 5.8.



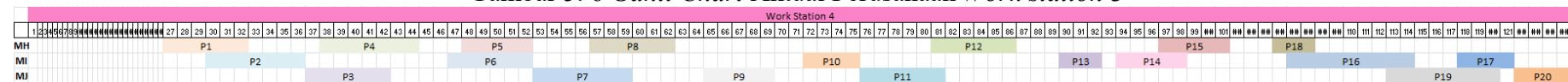
Gambar 5. 4 *Gantt Chart* Aktual Perusahaan *Work station 1*



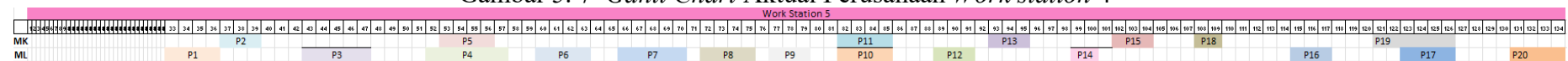
Gambar 5. 5 *Gantt Chart* Aktual Perusahaan *Work station 2*



Gambar 5. 6 *Gantt Chart* Aktual Perusahaan *Work station 3*



Gambar 5. 7 *Gantt Chart* Aktual Perusahaan *Work station* 4



Gambar 5. 8 *Gantt Chart* Aktual Perusahaan *Work station 5*

Gantt Chart aktual menunjukkan bahwa produksi berlangsung sampai menit ke-134. WS1 selesai pada menit ke-61, WS2 pada menit ke-96, WS3 pada menit ke-113, WS4 pada menit ke-125, dan WS5 pada menit ke-134.

Urutan produksi mengikuti P1 sampai P20 sehingga produk dengan nomor besar tetap berada di belakang meskipun sebagian memiliki *release time* yang lebih awal. Contohnya, P14 sudah tersedia pada menit ke-5, tetapi baru mulai diproses pada menit ke-36 karena harus menunggu seluruh produk yang berada di depannya dalam urutan produksi. Kondisi ini menunjukkan bahwa jadwal aktual lebih dipengaruhi oleh urutan tetap yang digunakan perusahaan dibandingkan oleh kesiapan produk. Akibatnya, beberapa produk mengalami waktu tunggu yang cukup panjang sebelum memasuki proses pertama, meskipun secara teoritis sudah dapat diproses lebih awal.

Sebagai ilustrasi yang lebih rinci, pada WS1 mesin MA memproses P2 mulai menit ke-11 hingga menit ke-14. Setelah P2 selesai, mesin tidak dapat langsung memproses produk berikutnya apabila terdapat kebutuhan *setup* akibat pergantian jenis produk. Misalnya, jika produk berikutnya adalah P3 dan pasangan P2–P3 memerlukan *setup time* selama 2 menit, maka mesin terlebih dahulu melakukan *setup* dari menit ke-14 hingga menit ke-16. Akibatnya, meskipun mesin telah menyelesaikan P2 pada menit ke-14, P3 baru dapat mulai diproses pada menit ke-16. Dengan demikian, waktu mulai suatu produk tidak hanya dipengaruhi oleh selesainya produk sebelumnya, tetapi juga oleh kebutuhan *setup* yang muncul akibat perubahan jenis produk.

Analisis ini menunjukkan bahwa *setup time* berkontribusi terhadap bertambahnya waktu penyelesaian produksi karena menciptakan periode ketika mesin tidak menghasilkan *output*. Semakin sering terjadi pergantian jenis produk yang membutuhkan *setup*, semakin besar pula akumulasi waktu yang tidak digunakan untuk proses produksi langsung. Oleh karena itu, urutan produk memiliki pengaruh penting terhadap efisiensi jadwal karena dapat memperbesar atau memperkecil total *setup time* yang terjadi.

Setelah P3 selesai diproses di WS1, produk tersebut tidak otomatis langsung masuk ke WS2. Produk harus menunggu hingga dua kondisi terpenuhi, yaitu proses pada WS1 telah selesai dan mesin yang dituju pada WS2 telah tersedia. Sebagai contoh, apabila P3 selesai di WS1 pada menit ke-19 tetapi mesin yang akan digunakan di WS2 masih memproses produk lain hingga menit ke-20 maka P3 harus menunggu selama 3 menit sebelum dapat memulai proses berikutnya. Waktu tunggu antar-*work station* seperti ini juga terlihat pada *Gantt Chart* sebagai bagian dari jeda aliran produksi.

Keberadaan waktu tunggu antar-*work station* menunjukkan bahwa aliran produksi belum sepenuhnya sinkron. Meskipun suatu produk telah selesai pada tahap sebelumnya, produk tersebut belum tentu dapat langsung diproses pada tahap berikutnya karena keterbatasan kapasitas atau antrean mesin. Akumulasi waktu tunggu seperti ini menyebabkan *flow time* produk menjadi lebih panjang dan pada akhirnya meningkatkan *makespan* keseluruhan sistem.

Pada WS4 dan WS5 terlihat interval kosong yang relatif panjang. Ruang kosong tersebut dapat terdiri atas waktu *setup*, waktu menunggu produk dari tahap sebelumnya, atau waktu mesin benar-benar menganggur karena belum ada produk yang siap diproses.

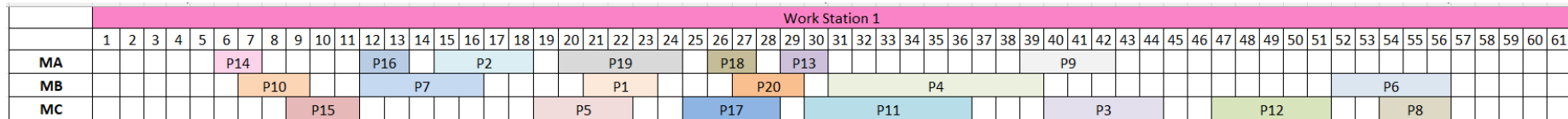
Keberadaan interval kosong pada WS4 dan WS5 mengindikasikan bahwa kapasitas mesin pada tahap tersebut belum dimanfaatkan secara optimal. Jika sebagian besar ruang kosong disebabkan oleh keterlambatan kedatangan produk dari tahap sebelumnya, maka masalah utama bukan terletak pada kapasitas WS4 dan WS5, melainkan pada ketidakseimbangan aliran produksi antar-*work station*. Kondisi ini menunjukkan bahwa peningkatan kinerja sistem tidak cukup dilakukan dengan mempercepat satu tahap saja, tetapi perlu mempertimbangkan keterkaitan seluruh tahapan proses.

Jadwal aktual tidak menunjukkan *overlap* pada mesin yang sama. Dengan demikian, kelemahan jadwal aktual bukan berasal dari pelanggaran kapasitas mesin, melainkan dari panjangnya horizon produksi serta adanya waktu tunggu dan ketidaksinkronan aliran antar-*work station* yang masih dapat diperbaiki.

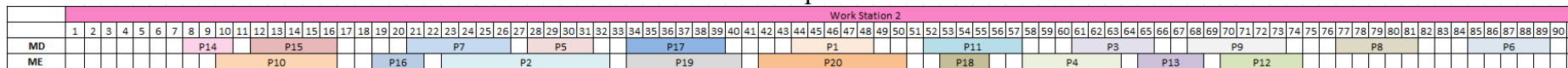
Secara keseluruhan, *Gantt Chart* aktual menunjukkan bahwa sistem produksi telah memenuhi batasan kapasitas mesin, tetapi belum menghasilkan aliran produksi yang efisien. Indikasinya terlihat dari adanya produk yang menunggu lama sebelum diproses, akumulasi *setup time*, waktu tunggu antar-*work station*, serta interval menganggur pada beberapa mesin. Faktor-faktor tersebut menyebabkan *makespan* mencapai 134 menit dan menjadi peluang utama yang dapat diperbaiki melalui optimasi urutan produk dan alokasi mesin.

B. Analisis *Gantt Chart* Hasil Optimasi Solver

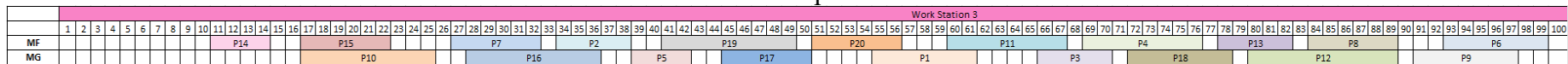
Berikut ini merupakan *Gantt Chart* hasil optimasi *Solver* untuk usulan perusahaan, dapat dilihat pada Gambar 5.9 hingga 5.13.



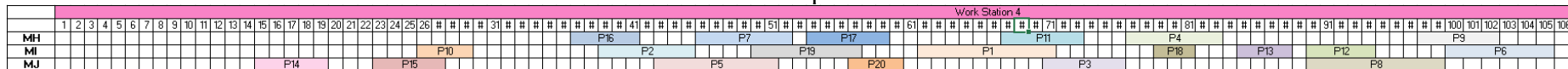
Gambar 5. 9 *Gantt Chart* Hasil Optimasi Solver *Work station 1*



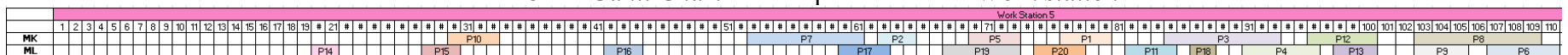
Gambar 5. 10 *Gantt Chart* Hasil Optimasi Solver *Work station 2*



Gambar 5. 11 *Gantt Chart* Hasil Optimasi Solver *Work station 3*



Gambar 5. 12 *Gantt Chart* Hasil Optimasi Solver *Work station 4*



Gambar 5. 13 *Gantt Chart* Hasil Optimasi Solver *Work station* 5

Gantt Chart hasil optimasi *Solver* menunjukkan perubahan pola penjadwalan dibandingkan jadwal aktual. Pada jadwal aktual, produk diproses mengikuti urutan P1 sampai P20, sedangkan pada hasil optimasi *Solver* urutan produk berubah menjadi kombinasi yang dipilih untuk meminimalkan *makespan*. Produk pertama yang diproses adalah P14 pada menit ke-5 sesuai *release time*-nya, kemudian diikuti oleh P10 pada menit ke-6 dan P15 pada menit ke-8.

Pada WS1 terlihat bahwa beberapa produk dapat diproses hampir bersamaan karena dialokasikan pada mesin paralel yang berbeda. P14 diproses pada MA, P10 pada MB, dan P15 pada MC. Kondisi ini menunjukkan bahwa kapasitas mesin paralel dimanfaatkan untuk mengurangi waktu tunggu produk pada tahap awal produksi. Berbeda dengan jadwal aktual yang lebih banyak mengikuti urutan kedatangan produk, hasil optimasi *Solver* mendistribusikan pekerjaan ke mesin yang tersedia sehingga beberapa operasi dapat berlangsung secara paralel.

Perbedaan pola tersebut dapat dijelaskan melalui contoh beberapa produk pada awal jadwal. Pada hasil optimasi *Solver*, P14 mulai diproses pada menit ke-5 di WS1 dan langsung dilanjutkan ke *work station* berikutnya ketika mesin tujuan tersedia. Setelah P14, produk lain seperti P10 dan P15 juga segera memasuki mesin paralel yang berbeda tanpa harus menunggu produk sebelumnya selesai seluruh prosesnya. Akibatnya, jarak waktu antarproduk pada awal jadwal menjadi lebih rapat dibandingkan jadwal aktual.

Sebagai contoh, pada jadwal aktual sering ditemukan jeda yang cukup besar antara penyelesaian suatu produk dan dimulainya produk berikutnya pada mesin tertentu karena produk masih mengikuti urutan kedatangan. Selain itu, waktu setup yang muncul akibat pergantian jenis produk dapat menyebabkan mesin menunggu lebih lama sebelum memulai pekerjaan berikutnya. Pada hasil optimasi *Solver*, urutan produk dipilih sedemikian rupa sehingga kombinasi setup time dan ketersediaan mesin menjadi lebih menguntungkan. Dengan demikian, meskipun setup time tetap ada, dampaknya terhadap keterlambatan jadwal dapat dikurangi.

Contoh lain dapat dilihat pada P14 dan P10. Kedua produk tersebut diproses pada mesin yang berbeda di WS1 sehingga dapat berjalan hampir bersamaan. Setelah menyelesaikan proses di WS1, keduanya juga dapat berpindah ke WS2 tanpa menunggu terlalu lama karena *Solver* telah mengatur kombinasi urutan dan alokasi mesin yang sesuai. Kondisi ini berbeda dengan jadwal aktual yang menunjukkan lebih banyak celah waktu antarproduk akibat antrean pada mesin tertentu.

Pergerakan produk dari WS1 menuju WS2, WS3, WS4, dan WS5 juga menunjukkan pola yang lebih terkoordinasi. Produk yang telah menyelesaikan proses pada suatu *work station* dapat segera memasuki *work station* berikutnya apabila mesin tujuan tersedia. Akibatnya, antrean pada beberapa titik proses menjadi lebih pendek dibandingkan jadwal aktual.

Jika pada jadwal aktual perpindahan antar *work station* masih menunjukkan banyak ruang kosong akibat produk harus menunggu mesin tersedia, pada hasil optimasi *Solver* pola perpindahan produk terlihat lebih padat. Produk yang keluar dari suatu *work station* cenderung lebih cepat memasuki *work station* berikutnya sehingga aliran produksi menjadi lebih kontinu. Dengan kata lain, bukan hanya urutan produknya yang berubah, tetapi juga sinkronisasi antar *work station* menjadi lebih baik.

Berdasarkan *Gantt Chart*, penyelesaian setiap *work station* terjadi pada waktu yang berbeda. WS1 selesai pada menit ke-56, WS2 pada menit ke-89, WS3 pada menit ke-99, WS4 pada menit ke-105, dan WS5 pada menit ke-110. Dengan demikian, *makespan* keseluruhan jadwal hasil optimasi *Solver* adalah 110 menit.

Meskipun terjadi perbaikan jadwal, *Gantt Chart* masih menunjukkan adanya interval menganggur pada beberapa mesin. Waktu menganggur paling terlihat pada WS4, terutama ketika mesin harus menunggu kedatangan produk dari proses sebelumnya. Kondisi tersebut menunjukkan bahwa optimasi belum mampu menghilangkan seluruh ketidaksinkronan antar *work station*. Namun, keberadaan waktu menganggur tersebut tidak selalu menunjukkan jadwal yang buruk karena *Solver* berfokus pada minimasi *makespan*, bukan pada eliminasi seluruh *idle time*.

Jika dibandingkan dengan *Gantt Chart* jadwal aktual, bagian akhir jadwal hasil optimasi *Solver* terlihat lebih padat. Produk memasuki WS4 dan WS5 dengan pola yang lebih teratur sehingga penyelesaian produk terakhir dapat dicapai lebih cepat. Selain itu, distribusi pekerjaan pada mesin paralel menjadi lebih fleksibel karena *Solver* mempertimbangkan kombinasi urutan produk, waktu proses, *setup time*, dan ketersediaan mesin secara bersamaan.

Kepadatan jadwal tersebut terlihat dari semakin kecilnya jarak antarproduk pada mesin-mesin yang aktif. Pada jadwal aktual, masih terdapat beberapa interval kosong yang cukup panjang sebelum produk berikutnya diproses. Sebaliknya, pada hasil optimasi *Solver* banyak mesin yang langsung menerima pekerjaan berikutnya setelah *setup* selesai atau setelah produk sebelumnya keluar dari mesin. Hal ini menunjukkan bahwa kapasitas

mesin dimanfaatkan secara lebih intensif sehingga total waktu penyelesaian produksi dapat dipersingkat.

Secara keseluruhan, *Gantt Chart* hasil optimasi *Solver* menunjukkan bahwa perubahan alokasi mesin berhasil memperpendek total waktu penyelesaian produksi. Pemanfaatan mesin paralel yang lebih efektif, pengurangan waktu tunggu pada beberapa tahap proses, pengaturan *setup time* yang lebih baik, serta koordinasi perpindahan produk antar *work station* menjadi faktor utama yang mendukung penurunan *makespan* pada jadwal hasil optimasi.

C. Perbandingan Visual Kedua *Gantt Chart*

Perbedaan utama kedua *Gantt Chart* adalah total waktu penyelesaian produksi. Jadwal aktual berakhir pada menit ke-134, sedangkan pada hasil optimasi *Solver* berakhir pada menit ke-110. *Solver* mengubah urutan produk dan lebih dari setengah keputusan pemilihan mesin. Perubahan tersebut menghasilkan pola pemrosesan paralel dan waktu kesiapan mesin yang berbeda.

Penurunan *makespan* bukan karena *Solver* menghilangkan *overlap* pada jadwal aktual. Kedua jadwal sama-sama tidak memiliki *overlap* pada mesin yang sama. Perbaikan berasal dari kombinasi urutan, alokasi mesin, pengurangan *setup time*, dan penurunan jeda pada tahap akhir.

Gantt Chart Solver juga tidak selalu lebih padat pada setiap tahap. Beberapa jeda muncul di WS1 dan WS2, tetapi tidak memperpanjang waktu selesai keseluruhan karena diimbangi oleh perbaikan yang lebih besar di WS3 sampai WS5.

Untuk memperjelas visualisasi, kedua *Gantt Chart* sebaiknya menggunakan skala waktu yang sama. Garis vertikal penanda *makespan* dapat ditambahkan pada menit ke-134 dan ke-110. Blok *setup time* juga sebaiknya dibedakan dari blok *processing time*.

Berdasarkan analisis tersebut, *Hasil optimasi Solver* menghasilkan kombinasi *job assignment* dan *machine assignment* yang memperpendek *makespan* dari 134 menjadi 110 menit. Perubahan tersebut disertai penurunan *setup time* dan *internal idle time*, tetapi meningkatkan total beban waktu proses agregat dan tidak mempercepat seluruh produk. Oleh karena itu, *Hasil optimasi Solver* dapat dinyatakan lebih baik berdasarkan fungsi tujuan *makespan*, tetapi belum dapat dinyatakan lebih baik berdasarkan seluruh indikator penjadwalan.

Berdasarkan perbandingan visual *Gantt chart*, jadwal hasil optimasi *Solver* dapat dinyatakan valid karena tidak terdapat *overlap* pada mesin yang sama. Perbaikan yang

terjadi bukan disebabkan oleh penghilangan *overlap*, tetapi oleh perubahan kombinasi *job assignment*, *machine assignment*, pengurangan *setup time*, dan penurunan jeda pada beberapa tahap proses. Namun, hasil optimasi tersebut tetap perlu dipahami sesuai dengan fungsi tujuan yang digunakan. Jadwal *Solver* lebih baik berdasarkan indikator *makespan*, tetapi belum tentu menjadi jadwal terbaik apabila perusahaan juga mempertimbangkan *due date*, prioritas produk, biaya produksi, atau utilisasi mesin secara menyeluruh.

5.3 Temuan Utama dan Interpretasi Hasil

Berdasarkan analisis jadwal aktual perusahaan dan hasil optimasi *Solver*, terdapat beberapa temuan utama yang perlu dibahas lebih lanjut. Temuan tersebut tidak hanya berkaitan dengan penurunan nilai *makespan*, tetapi juga menunjukkan bagaimana perubahan urutan produk dan alokasi mesin memengaruhi komponen waktu produksi secara keseluruhan. Berikut ini ringkasan temuan utama hasil penjadwalan ditunjukkan pada Tabel 5.17.

Tabel 5. 17 Ringkasan Temuan Utama Hasil Penjadwalan

Aspek Temuan	Jadwal Aktual	Hasl Optimasi <i>Solver</i>	Interpretasi Utama
<i>Makespan</i>	134 menit	110 menit	Turun 24 menit atau 17,91%
Total beban waktu proses agregat	477 menit	548 menit	Naik 71 menit, tetapi <i>makespan</i> turun
Total <i>setup time</i>	228 menit	165 menit	Turun 63 menit atau 27,63%
<i>Internal idle time</i>	188 menit	129 menit	Turun 59 menit atau 31,38%
Produk selesai lebih cepat	-	9 produk	Tidak semua produk dipercepat
Produk selesai tetap	-	1 produk	Tidak berubah
Produk selesai lebih lambat	-	10 produk	Konsekuensi fungsi tujuan Tunggal
Hasil lima <i>running Solver</i>	134 menit	122 menit, 110 menit, 125 menit, 127 menit, dan menit.	Nilai 110 merupakan hasil terbaik dari lima percobaan.

Tabel 5.17 menunjukkan bahwa hasil optimasi *Solver* memberikan perbaikan terhadap *makespan*, *setup time*, dan *internal idle time*. Di sisi lain, hasil optimasi juga menunjukkan beberapa karakteristik yang perlu diinterpretasikan lebih lanjut, terutama terkait peningkatan total beban waktu proses agregat serta perubahan waktu penyelesaian produk, di mana Sebagian produk selesai lebih lambat dibandingkan jadwal aktual sebagai konsekuensi dari fungsi tujuan minimasi *makespan*.

Berdasarkan ringkasan temuan pada Tabel 5.17, terdapat beberapa aspek yang memberikan kontribusi penting terhadap perubahan kinerja jadwal setelah dilakukan optimasi. Setiap temuan perlu dianalisis lebih lanjut untuk memahami mekanisme yang menyebabkan penurunan *makespan* serta implikasinya terhadap sistem produksi secara keseluruhan. Pembahasan berikut menguraikan masing-masing temuan secara lebih rinci.

a. Penurunan *Makespan* Terjadi karena Kombinasi Urutan Produk dan Alokasi Mesin

Temuan utama pertama adalah bahwa penurunan *makespan* tidak disebabkan oleh satu faktor tunggal. Nilai *makespan* turun dari 134 menit menjadi 110 menit karena adanya kombinasi antara perubahan *job assignment*, perubahan alokasi mesin, penurunan *setup time*, dan pemanfaatan mesin paralel.

Jadwal aktual perusahaan menggunakan urutan P1 sampai P20. Urutan tersebut telah dinyatakan layak karena memenuhi batasan model, tetapi belum mempertimbangkan interaksi menyeluruh antara *release time*, waktu proses, *setup time*, dan ketersediaan mesin. Setelah optimasi, urutan produk berubah menjadi P14–P10–P15–P16–P7–P2–P5–P17–P19–P1–P20–P11–P3–P18–P4–P13–P12–P9–P8–P6. Perubahan urutan tersebut membuat beberapa produk yang sebelumnya berada di bagian belakang dapat diproses lebih awal. Sebagai contoh, P14 berpindah dari urutan ke-14 pada jadwal aktual menjadi urutan pertama pada hasil optimasi *Solver*. Produk tersebut dapat mulai pada menit ke-5 dan selesai pada WS5 pada menit ke-21, sedangkan pada jadwal aktual P14 baru selesai pada menit ke-100.

Namun, percepatan tersebut tidak dapat dijelaskan hanya dari perubahan urutan. Hasil optimasi juga mengubah alokasi mesin pada 54 dari 100 operasi. Hal ini menunjukkan bahwa *Solver* tidak hanya mengatur ulang posisi produk, tetapi juga menentukan mesin yang digunakan pada setiap *work station* agar kombinasi yang terbentuk menghasilkan *makespan* lebih rendah.

Selain itu, total *setup time* turun dari 228 menit menjadi 165 menit. Karena *setup time* bersifat *sequence-dependent*, perubahan urutan produk menghasilkan pasangan *previous job* dan *current job* yang berbeda. Kombinasi urutan yang baru membuat waktu yang dibutuhkan untuk menyiapkan mesin menjadi lebih singkat secara keseluruhan.

Dengan demikian, penurunan *makespan* merupakan hasil interaksi dari beberapa keputusan sekaligus. Jadwal tidak menjadi lebih pendek hanya karena produk tertentu dipindahkan ke awal, atau karena mesin tertentu diganti, tetapi karena kombinasi

keputusan tersebut membuat aliran produksi menjadi lebih sinkron dari WS1 sampai WS5.

Temuan ini menunjukkan bahwa pada sistem *multi-stage flexible flow shop*, penjadwalan tidak cukup dilakukan dengan aturan sederhana seperti mengikuti nomor produk, mendahulukan produk dengan *release time* awal, atau memilih mesin tercepat pada setiap operasi. Sistem seperti ini memerlukan evaluasi kombinasi keputusan secara menyeluruh karena setiap keputusan pada satu tahap dapat memengaruhi tahap berikutnya.

b. Mesin dengan Waktu Proses Terpendek Tidak Selalu Menghasilkan Jadwal Terbaik

Temuan kedua adalah bahwa mesin dengan waktu proses paling pendek tidak selalu menghasilkan jadwal terbaik pada tingkat sistem. Hal ini terlihat pada P7. Pada jadwal aktual, P7 menggunakan mesin MA pada WS1 dengan waktu proses 5 menit. Pada hasil optimasi *Solver*, P7 dialokasikan ke mesin MB dengan waktu proses 9 menit. Jika hanya dilihat dari satu operasi di WS1, keputusan tersebut terlihat kurang baik karena waktu proses P7 meningkat 4 menit.

Namun, jika dilihat sampai tahap akhir, P7 justru selesai lebih cepat. Pada jadwal aktual, P7 selesai di WS5 pada menit ke-70. Setelah optimasi, P7 selesai di WS5 pada menit ke-61. Artinya, meskipun durasi WS1 menjadi lebih panjang, waktu selesai akhir P7 justru lebih cepat 9 menit. Kondisi ini menunjukkan bahwa keputusan lokal tidak selalu sejalan dengan kinerja sistem. Mesin MA memang lebih cepat untuk P7, tetapi dalam jadwal aktual produk tersebut baru mulai pada menit ke-29. Pada hasil optimasi *Solver*, P7 menggunakan mesin MB yang lebih lama, tetapi dapat mulai lebih awal pada menit ke-11. Perbedaan waktu mulai tersebut membuat P7 tetap selesai lebih cepat pada tahap akhir.

Temuan ini penting karena menunjukkan bahwa pemilihan mesin tidak dapat dinilai hanya dari waktu proses satu operasi. Ketersediaan mesin, antrean pekerjaan, *setup time*, urutan produk, dan waktu perpindahan ke *work station* berikutnya juga menentukan apakah suatu pilihan mesin benar-benar menguntungkan bagi sistem. Dengan kata lain, mesin tercepat secara individual belum tentu menjadi pilihan terbaik apabila menyebabkan antrean atau tidak selaras dengan aliran proses berikutnya. Sebaliknya, mesin yang lebih lambat dapat menjadi pilihan yang lebih baik apabila memungkinkan produk diproses lebih awal dan mengurangi waktu tunggu pada tahap berikutnya.

Temuan ini juga menjelaskan mengapa total beban waktu proses agregat meningkat dari 477 menit menjadi 548 menit, tetapi *makespan* tetap menurun. Beberapa produk memang dialokasikan ke mesin dengan durasi lebih panjang, tetapi pekerjaan dapat berjalan lebih paralel dan aliran produk menjadi lebih baik sehingga waktu selesai seluruh batch tetap lebih pendek.

c. Penurunan *Makespan* Tetap Terjadi meskipun Total Beban Waktu Proses Meningkat

Temuan ketiga adalah total beban waktu proses agregat meningkat dari 477 menit menjadi 548 menit, tetapi *makespan* justru turun dari 134 menit menjadi 110 menit. Sekilas kondisi ini terlihat bertentangan karena jika total waktu proses bertambah, jadwal seharusnya menjadi lebih panjang. Namun, total beban waktu proses agregat berbeda dengan *makespan*. Total beban waktu proses agregat merupakan jumlah seluruh durasi proses dari semua operasi pada lima *work station*. Sementara itu, *makespan* adalah total waktu yang dibutuhkan sejak proses pertama dimulai hingga produk terakhir selesai di WS5.

Pada sistem yang memiliki mesin paralel, beberapa operasi dapat dikerjakan secara bersamaan pada mesin yang berbeda. Karena itu, jumlah seluruh durasi operasi bisa saja lebih besar daripada total waktu penyelesaian produksi. Dengan kata lain, meskipun total waktu proses meningkat, *makespan* belum tentu ikut meningkat jika pekerjaan dapat dibagi dan dijalankan secara lebih bersamaan.

Hasil optimasi *Solver* menunjukkan bahwa beberapa produk memang dialokasikan ke mesin dengan waktu proses yang lebih lama. Namun, keputusan tersebut memungkinkan pekerjaan dimulai lebih cepat atau mengurangi antrean pada mesin lain. Akibatnya, tambahan waktu pada beberapa operasi dapat tertutupi oleh berkurangnya waktu tunggu, berkurangnya *setup time*, dan penggunaan mesin paralel yang lebih efektif. Temuan ini menunjukkan bahwa minimasi *makespan* tidak sama dengan meminimalkan total waktu proses. Jadwal hasil optimasi *Solver* bukan jadwal dengan total durasi proses paling kecil, tetapi jadwal yang mampu menyelesaikan seluruh produk dalam waktu paling singkat dari lima percobaan *Solver* yang dilakukan.

Secara praktis, hasil ini menunjukkan bahwa perusahaan tidak cukup hanya memilih mesin tercepat untuk setiap produk. Keputusan yang terlihat paling cepat pada satu operasi belum tentu menghasilkan jadwal terbaik secara keseluruhan. Pada sistem

produksi bertahap dengan mesin paralel, yang lebih penting adalah bagaimana pekerjaan diatur agar aliran produk dari WS1 sampai WS5 berjalan lebih lancar.

d. Dampak Optimasi Semakin Terlihat pada *Work station* Akhir

Temuan keempat adalah bahwa dampak optimasi semakin besar pada *work station* bagian akhir. Pada WS1, waktu penyelesaian terakhir turun dari 61 menit menjadi 56 menit atau berkurang 5 menit. Pada WS2, penurunannya menjadi 7 menit. Selanjutnya, penurunan meningkat menjadi 14 menit pada WS3, 20 menit pada WS4, dan 24 menit pada WS5. Pola ini menunjukkan bahwa manfaat perbaikan jadwal tidak hanya dirasakan pada tahap awal, tetapi juga terbawa hingga tahap berikutnya. Ketika suatu produk dapat diproses lebih awal di WS1, produk tersebut juga berpeluang masuk lebih cepat ke WS2, WS3, WS4, dan WS5.

Pada jadwal aktual, beberapa produk yang sebenarnya sudah siap diproses masih harus menunggu karena posisinya berada di urutan belakang. Kondisi ini menyebabkan keterlambatan yang terus terbawa ke tahap berikutnya. Setelah optimasi, beberapa produk ditempatkan pada urutan yang lebih sesuai sehingga aliran menuju *work station* akhir menjadi lebih lancar.

Penurunan *internal idle time* pada WS4 dan WS5 juga mendukung temuan ini. *Idle time* WS4 turun dari 102 menjadi 74 menit, sedangkan WS5 turun dari 71 menjadi 39 menit. Hal ini menunjukkan bahwa kedatangan produk ke tahap akhir menjadi lebih sesuai dengan ketersediaan mesin sehingga waktu menganggur mesin dapat berkurang. Namun, hasil ini perlu dipahami dengan hati-hati. Penurunan yang lebih besar pada WS4 dan WS5 tidak otomatis berarti kedua *work station* tersebut merupakan *bottleneck*. Untuk memastikan adanya *bottleneck*, diperlukan analisis tambahan seperti kapasitas mesin, tingkat utilisasi, beban kerja, dan antrean. Data pada penelitian ini hanya menunjukkan bahwa manfaat optimasi semakin terlihat pada tahap akhir proses produksi.

Dengan demikian, temuan ini menunjukkan bahwa perbaikan jadwal pada sistem bertahap tidak dapat dilihat per *work station* secara terpisah. Keputusan pada WS1 dapat memengaruhi kondisi di WS5. Oleh karena itu, penyusunan jadwal perlu mempertimbangkan aliran produk secara keseluruhan dari awal hingga akhir proses.

e. Penurunan *Makespan* Tidak Berarti Semua Produk Selesai Lebih Cepat

Temuan kelima adalah bahwa *makespan* yang lebih kecil tidak berarti semua produk selesai lebih cepat. Berdasarkan perbandingan *finish time* pada WS5, terdapat sembilan

produk yang selesai lebih cepat, satu produk memiliki waktu selesai yang sama, dan sepuluh produk selesai lebih lambat dibandingkan jadwal aktual.

Produk seperti P14, P15, P16, P17, dan P20 mengalami percepatan yang cukup besar. P14 selesai lebih cepat dari menit ke-100 menjadi menit ke-21. P15 selesai lebih cepat dari menit ke-104 menjadi menit ke-30. P16 selesai lebih cepat dari menit ke-117 menjadi menit ke-44. P17 selesai lebih cepat dari menit ke-126 menjadi menit ke-63, sedangkan P20 selesai lebih cepat dari menit ke-134 menjadi menit ke-78.

Sebaliknya, beberapa produk yang sebelumnya berada di awal jadwal justru selesai lebih lambat. P1 yang semula selesai pada menit ke-36 menjadi selesai pada menit ke-83. P3 yang semula selesai pada menit ke-47 menjadi selesai pada menit ke-93. P6 yang semula selesai pada menit ke-63 menjadi produk terakhir dengan waktu selesai menit ke-110.

Kondisi ini bukan menunjukkan adanya kesalahan pada model. Fungsi tujuan yang digunakan dalam penelitian adalah minimasi *makespan*, yaitu memperkecil waktu selesai produk terakhir. Model tidak dirancang untuk mempercepat seluruh produk secara bersamaan atau meminimalkan rata-rata waktu selesai setiap produk. Karena tujuan utamanya adalah memperkecil *makespan*, *Solver* dapat menempatkan sebagian produk pada urutan yang lebih belakang jika hal tersebut membantu mempercepat penyelesaian seluruh kelompok produk secara keseluruhan. Oleh karena itu, jadwal hasil optimasi dapat lebih baik dari sisi *makespan*, tetapi belum tentu lebih baik untuk setiap produk secara individu.

Temuan ini menunjukkan keterbatasan penggunaan satu fungsi tujuan saja. Jika perusahaan memiliki produk prioritas, target pengiriman tertentu, atau batas waktu pengiriman (*due date*), maka minimasi *makespan* saja mungkin belum cukup. Penelitian selanjutnya dapat mempertimbangkan indikator lain seperti *total tardiness*, *maximum lateness*, jumlah produk terlambat, atau tingkat prioritas produk agar hasil penjadwalan lebih sesuai dengan kebutuhan perusahaan.

f. Hasil Optimasi *Solver* Merupakan Solusi Terbaik dari Percobaan, bukan Global Optimum

Temuan keenam berkaitan dengan konsistensi hasil *Solver*. Pada Subbab 5.1 telah ditunjukkan bahwa *Solver* dijalankan sebanyak lima kali dengan fungsi tujuan dan batasan yang sama. Hasilnya menghasilkan nilai *makespan* yang berbeda, yaitu 122, 110, 125, 127, dan 118 menit.

Perbedaan hasil tersebut menunjukkan bahwa metode *Evolutionary* pada *Solver* memiliki unsur pencarian acak. Meskipun fungsi tujuan dan batasannya sama, kombinasi *job assignment*

dan alokasi mesin yang ditemukan pada setiap percobaan dapat berbeda. Oleh karena itu, satu kali percobaan belum tentu menghasilkan solusi terbaik.

Nilai *makespan* terkecil dari lima percobaan adalah 110 menit, sedangkan nilai terbesar adalah 127 menit. Dengan demikian, selisih hasil antarrunning mencapai 17 menit. Rata-rata *makespan* dari lima percobaan tersebut adalah 120,40 menit.

Hasil ini menunjukkan bahwa pemilihan jadwal usulan dengan *makespan* 110 menit sudah tepat karena merupakan hasil terbaik dari lima percobaan yang dilakukan. Namun, hasil tersebut belum dapat dipastikan sebagai *global optimum*. Penelitian ini belum menggunakan metode yang dapat membuktikan secara matematis bahwa tidak ada solusi lain dengan *makespan* yang lebih kecil.

Oleh karena itu, pernyataan yang lebih tepat adalah bahwa hasil optimasi *Solver* merupakan solusi terbaik yang berhasil diperoleh dari lima kali percobaan *Solver*. Pernyataan ini lebih sesuai dengan ruang lingkup penelitian yang dilakukan.

Temuan ini juga membuka peluang untuk penelitian selanjutnya. Jumlah percobaan *Solver* dapat ditambah, parameter *Solver* dapat diuji lebih lanjut, atau hasilnya dapat dibandingkan dengan metode lain seperti *Genetic Algorithm*, *Simulated Annealing*, *Tabu Search*, *Mixed Integer Linear Programming*, atau *Constraint Programming*.

g. Arti Penting Temuan Terhadap Sistem Produksi

Secara keseluruhan, hasil penelitian menunjukkan bahwa jadwal aktual perusahaan sudah layak digunakan, tetapi belum menghasilkan aliran produksi yang paling efisien berdasarkan indikator *makespan*. Jadwal aktual tidak melanggar batasan mesin dan tidak memiliki *overlap* pada mesin yang sama. Dengan demikian, masalah utama bukan pada kelayakan jadwal, melainkan pada lamanya waktu penyelesaian produksi, besarnya *setup time*, dan kurang lancarnya aliran produk antar-*work station*.

Hasil optimasi *Solver* memberikan alternatif jadwal yang mampu menurunkan *makespan* dari 134 menit menjadi 110 menit. Hasil ini menunjukkan bahwa perusahaan masih memiliki peluang untuk meningkatkan kinerja penjadwalan tanpa harus menambah jumlah mesin. Perbaikan tersebut dapat dicapai melalui pengaturan ulang urutan produk dan pemilihan mesin yang lebih sesuai.

Namun, hasil optimasi juga menunjukkan bahwa penurunan *makespan* tidak selalu diikuti oleh perbaikan pada semua indikator. Total beban waktu proses agregat meningkat, sebagian produk selesai lebih lambat, dan hasil terbaik Solver belum dapat dipastikan sebagai *global optimum*. Oleh karena itu, hasil optimasi perlu dipahami sesuai dengan tujuan yang digunakan dalam penelitian.

Jadwal hasil optimasi Solver dapat dinyatakan lebih efisien berdasarkan indikator *makespan*. Akan tetapi, jadwal tersebut belum tentu menjadi jadwal terbaik jika perusahaan juga mempertimbangkan faktor lain seperti prioritas produk, biaya produksi, utilisasi mesin, atau target pengiriman. Jika faktor-faktor tersebut dianggap penting, maka model penjadwalan perlu dikembangkan dengan fungsi tujuan tambahan.

Dengan demikian, temuan utama penelitian ini bukan hanya bahwa *makespan* berhasil diturunkan, tetapi juga bahwa penurunan tersebut terjadi karena kombinasi perubahan urutan produk, pemilihan mesin, pengurangan *setup time*, dan pemanfaatan mesin paralel yang lebih baik. Temuan ini dapat menjadi dasar untuk menyusun implikasi praktis, menjelaskan keterbatasan penelitian, serta memberikan arahan untuk penelitian selanjutnya.

Berdasarkan temuan tersebut, hasil optimasi Solver menunjukkan bahwa minimasi *makespan* mampu mempercepat penyelesaian keseluruhan batch produksi dari 134 menit menjadi 110 menit. Akan tetapi, penelitian ini masih memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, fungsi tujuan yang digunakan masih tunggal, yaitu minimasi *makespan*, sehingga belum mempertimbangkan indikator lain seperti total *tardiness*, *maximum lateness*, jumlah produk terlambat, *total flow time*, biaya produksi, maupun konsumsi energi. Kedua, hasil optimasi Solver belum dapat dipastikan sebagai *global optimum* karena metode *Evolutionary* bersifat heuristik atau metaheuristik dan menghasilkan nilai yang dapat berbeda pada setiap running. Ketiga, data yang digunakan masih terbatas pada parameter waktu proses, *setup time*, *release time*, dan alokasi mesin, sehingga belum memasukkan faktor produksi lain seperti *due date*, prioritas produk, keterbatasan operator, kapasitas *buffer*, waktu perpindahan material, gangguan mesin, atau perubahan permintaan.

Dengan demikian, hasil penelitian ini dapat menjadi dasar untuk pengembangan penelitian selanjutnya. Pengembangan dapat dilakukan dengan menambahkan fungsi tujuan lain, memperluas parameter data produksi, melakukan pengujian running Solver yang lebih banyak, serta membandingkan hasil *Solver Evolutionary* dengan metode

metaheuristik lain seperti *Grey Wolf Algorithm*, *NSGA II*, dan *Particle Swarm Optimization* (PSO).

BAB VI PENUTUP

6.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai optimasi penjadwalan produksi pada sistem *multi-stage flexible flow shop* menggunakan pendekatan Evolutionary Algorithm melalui Solver, diperoleh kesimpulan sebagai berikut.

1. Kombinasi urutan pekerjaan (*job assignment*) dan pemilihan mesin (*machine assignment*) yang diperoleh dari hasil optimasi Solver adalah urutan P14–P10–P15–P16–P7–P2–P5–P17–P19–P1–P20–P11–P3–P18–P4–P13–P12–P9–P8–P6.

Kombinasi alokasi mesin yang digunakan pada masing-masing produk adalah sebagai berikut.

Tabel 6. 1 Kombinasi Alokasi Mesin di Maing-Masing Produk

Produk	Alokasi Mesin
P14	MA-MD-MF-MJ-ML
P10	MB-ME-MG-MI-MK
P15	MC-MD-MF-MJ-ML
P16	MA-ME-MG-MH-ML
P7	MB-MD-MF-MH-MK
P2	MA-ME-MF-MI-MK
P5	MC-MD-MG-MJ-MK
P17	MC-MD-MG-MH-ML
P19	MA-ME-MF-MI-ML
P1	MB-MD-MG-MI-MK
P20	MB-ME-MF-MJ-ML
P11	MC-MD-MF-MH-ML
P3	MC-MD-MG-MJ-MK
P18	MA-ME-MG-MI-ML
P4	MB-ME-MF-MH-ML
P13	MA-ME-MF-MH-ML
P12	MC-ME-MG-MI-MK
P9	MB-MD-MG-MH-ML
P8	MC-MD-MF-MJ-MK
P6	MB-MD-MF-MI-ML

Kombinasi tersebut menunjukkan bahwa hasil optimasi tidak hanya mengubah urutan produk, tetapi juga mengubah alokasi mesin pada setiap *work station*. Sebanyak 54 dari 100 keputusan alokasi mesin berbeda dari jadwal aktual perusahaan. Hasil ini menunjukkan bahwa penurunan *makespan* diperoleh dari interaksi antara perubahan

urutan produk, pemilihan mesin, penurunan *setup time*, dan pemanfaatan mesin paralel, bukan hanya dari pemilihan mesin tercepat pada setiap operasi.

2. Nilai *makespan* jadwal aktual perusahaan adalah 134 menit, sedangkan nilai *makespan* hasil optimasi Solver adalah 110 menit. Dengan demikian, penjadwalan ulang menggunakan Solver mampu menurunkan *makespan* sebesar 24 menit atau 17,91%. Penurunan tersebut menunjukkan bahwa hasil optimasi Solver menghasilkan jadwal yang lebih baik berdasarkan fungsi tujuan minimasi *makespan*. Selain itu, total *setup time* menurun dari 228 menit menjadi 165 menit dan *internal idle time* menurun dari 188 menit menjadi 129 menit. Namun, hasil optimasi juga menunjukkan bahwa total beban waktu proses agregat meningkat dari 477 menit menjadi 548 menit dan tidak semua produk selesai lebih cepat. Oleh karena itu, hasil optimasi Solver dapat dinyatakan lebih baik berdasarkan indikator *makespan*, tetapi belum dapat dinyatakan lebih baik berdasarkan seluruh indikator penjadwalan produksi.

6.2 Saran

Berdasarkan hasil penelitian, pembahasan, serta keterbatasan yang ditemukan selama proses penelitian, saran yang dapat diberikan adalah sebagai berikut.

1. Penelitian selanjutnya disarankan untuk menambahkan fungsi tujuan selain minimasi *makespan*, seperti *total tardiness*, *maximum lateness*, jumlah produk terlambat, total *flow time*, atau rata-rata waktu penyelesaian produk. Hal ini diperlukan karena hasil penelitian menunjukkan bahwa penurunan *makespan* tidak selalu menyebabkan seluruh produk selesai lebih cepat.
2. Penelitian selanjutnya dapat mempertimbangkan *due date* dan prioritas produk dalam model penjadwalan. Penambahan parameter tersebut penting apabila perusahaan memiliki produk tertentu yang harus diselesaikan lebih awal atau memiliki tingkat urgensi lebih tinggi.
3. Model penjadwalan dapat dikembangkan dengan mempertimbangkan indikator biaya dan penggunaan sumber daya, seperti biaya penggunaan mesin, biaya *setup*, biaya keterlambatan, biaya lembur, atau konsumsi energi. Hal ini diperlukan karena hasil optimasi menunjukkan adanya peningkatan total beban waktu proses agregat, sehingga evaluasi berdasarkan waktu saja belum cukup untuk menggambarkan efisiensi produksi secara menyeluruh.

4. Penelitian selanjutnya disarankan untuk menghitung utilisasi mesin secara lebih lengkap. Pada penelitian ini, *idle time* yang digunakan masih berupa *internal idle time*, yaitu jeda antarpekerjaan pada mesin yang sama. Perhitungan utilisasi yang lebih lengkap dapat mempertimbangkan total waktu tersedia, waktu proses, waktu *setup*, waktu menganggur sebelum pekerjaan pertama, dan waktu menganggur setelah pekerjaan terakhir.
5. Penelitian selanjutnya dapat memasukkan kondisi produksi yang lebih dinamis, seperti gangguan mesin, keterbatasan operator, waktu perpindahan material antar-*work station*, kapasitas *buffer*, keterlambatan material, atau perubahan permintaan produksi. Penambahan faktor tersebut dapat membuat model lebih mendekati kondisi nyata di rantai produksi.
6. Penelitian selanjutnya disarankan untuk membandingkan hasil Solver dengan metode optimasi lain, seperti *Grey Wolf Algorithm*, *NSGA II*, *Particle Swarm Optimization* (PSO) Perbandingan metode diperlukan karena hasil Solver bersifat probabilistik dan nilai terbaik yang diperoleh belum dapat diklaim sebagai *global optimum*.
7. Pengujian hasil optimasi sebaiknya dilakukan pada data produksi aktual dengan periode yang lebih panjang atau pada beberapa skenario produksi yang berbeda. Hal ini bertujuan untuk mengetahui apakah kombinasi *job assignment* dan *machine assignment* yang dihasilkan tetap memberikan perbaikan pada kondisi produksi yang lebih bervariasi.
8. Apabila memungkinkan, perusahaan dapat melakukan uji coba terbatas terhadap jadwal hasil optimasi sebelum diterapkan secara penuh. Uji coba tersebut diperlukan untuk menilai kesesuaian jadwal dengan kondisi operasional aktual, seperti kesiapan operator, kebijakan produksi, ketersediaan mesin, dan aliran material.

DAFTAR PUSTAKA

- Allahverdi, A. (2015). The third comprehensive survey on scheduling problems with setup times/costs. *European Journal of Operational Research*, 246(2), 345–378. <https://doi.org/10.1016/j.ejor.2015.04.004>
- Boylestad, R. L., & Nashelsky, L. (2019). *Electronic Devices and Circuit Theory* (11th ed.). Pearson Education.
- Brucker, P. (2007). *Scheduling Algorithms* (5th ed.). Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-540-69516-5>
- Chen, N., Xie, N., & Wang, Y. (2022). An elite genetic algorithm for flexible job shop scheduling problem with extracted grey processing time. *Applied Soft Computing*, 131, 109783. <https://doi.org/10.1016/j.asoc.2022.109783>
- Chen, Y.-Y., Huang, P.-Y., Huang, C.-J., Huang, S.-Q., & Chou, F.-D. (2021). Makespan minimization for scheduling on two identical parallel machines with flexible maintenance and nonresumable jobs. *Journal of Industrial and Production Engineering*, 38(4), 271–284. <https://doi.org/10.1080/21681015.2021.1883131>
- Cheng, L., Tang, Q., Zhang, L., & Yu, C. (2022). Computers & Industrial Engineering Scheduling flexible manufacturing cell with no-idle flow-lines and job-shop via Q-learning-based genetic algorithm. *Computers & Industrial Engineering*, 169(April), 108293. <https://doi.org/10.1016/j.cie.2022.108293>
- Dasgupta, D., & Michalewicz, Z. (1997). *Evolutionary Algorithms in Engineering Applications*. Springer.
- Eiben, A. E., & Smith, J. E. (2015). *Introduction to Evolutionary Computing* (2nd ed.). Springer. <https://doi.org/10.1007/978-3-662-44874-8>
- Fan, J., Zhang, C., Liu, Q., Shen, W., & Gao, L. (2022). An improved genetic algorithm for flexible job shop scheduling problem considering reconfigurable machine tools with limited auxiliary modules. *Journal of Manufacturing Systems*, 62(February), 650–667. <https://doi.org/10.1016/j.jmsy.2022.01.014>
- Fortune Business Insights. (2026). *Consumer Electronics Market Size, Share & Industry Analysis, By Product Type, Distribution Channel, and Regional Forecast, 2025--2032* (Number FBI104693).
- Fylstra, D., Lasdon, L., Watson, J., & Waren, A. (1998). Design and use of the Microsoft Excel Solver. *Interfaces*, 28(5), 29–55. <https://doi.org/10.1287/inte.28.5.29>
- Goldberg, D. E. (2013). *Genetic Algorithm* (pp. 1–432). Pearson Education India.
- Gong, H., Xu, W., Sun, W., & Xu, K. (2023). Multi-Objective Flexible Flow Shop Production Scheduling Problem Based on the Double Deep Q-Network Algorithm. *Processes*, 11(12), 1–17. <https://doi.org/10.3390/pr11123321>
- Groover, M. P. (2019). *Automation, Production Systems, and Computer-Integrated Manufacturing* (4th ed.). Pearson.
- Gurusinga, M. O., Ginting, R., & Sinulingga, S. (2025). *Jurnal Sistem Teknik Industri Parallel Scheduling using Genetic Algorithm and Knowledge Based Approach*. 27(2), 127–136.
- Han, Z., Han, C., Lin, S., Dong, X., & Shi, H. (2019). Flexible flow shop scheduling method with public buffer. *Processes*, 7(10), 1–23. <https://doi.org/10.3390/pr7100681>
- Hasani, A., Mohammad, S., Hosseini, H., & Sankar, S. (2022a). Sustainability Analytics and Modeling Scheduling in a flexible flow shop with unrelated parallel machines and machine-dependent process stages: Trade-off between Makespan and

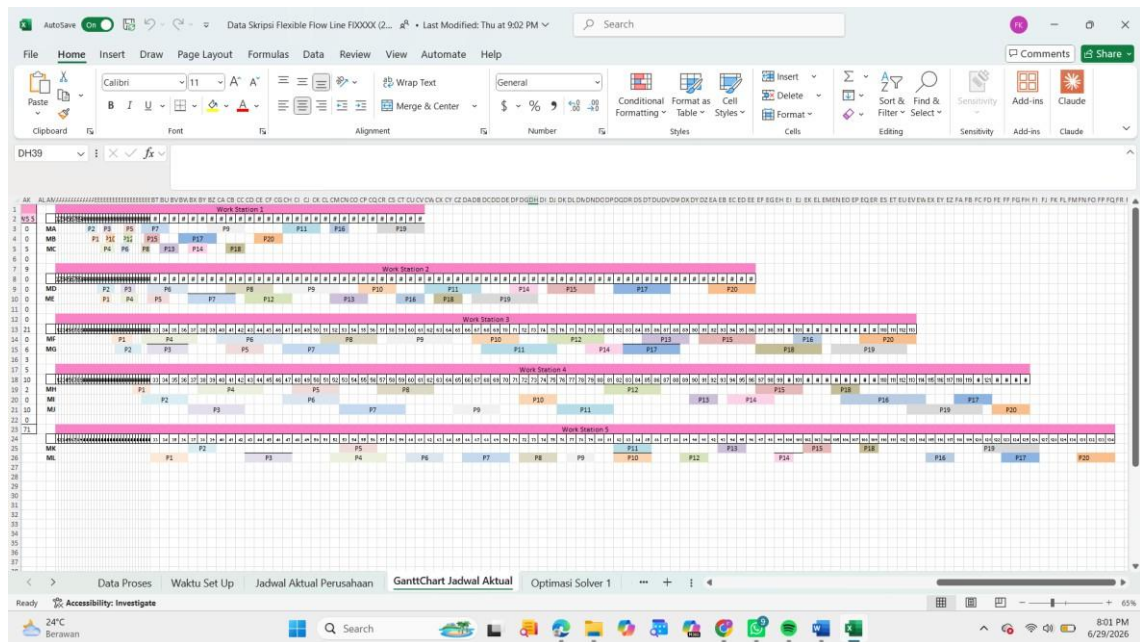
- production costs. *Sustainability Analytics and Modeling*, 2(September), 100010. <https://doi.org/10.1016/j.samod.2022.100010>
- Hasani, A., Mohammad, S., Hosseini, H., & Sankar, S. (2022b). Sustainability Analytics and Modeling Scheduling in a flexible flow shop with unrelated parallel machines and machine-dependent process stages : Trade-off between Makespan and production costs. *Sustainability Analytics and Modeling*, 2(September), 100010. <https://doi.org/10.1016/j.samod.2022.100010>
- Hatim, H. A., & Ahmad, F. (2022). Pendekatan Algoritma Genetika Dalam Upaya Optimalisasi Penjadwalan Di Pt. Nuansa Indah. *JISI: Jurnal Integrasi Sistem Industri*, 9(2), 145–154. <https://doi.org/10.24853/jisi.9.2.145-154>
- Helfrick, A. D., & Cooper, W. D. (1990). *Modern Electronic Instrumentation and Measurement Techniques*. Prentice Hall.
- Hidri, L. (2023). Effective Two-Phase Heuristic and Lower Bounds for Multi-Stage Flexible Flow Shop Scheduling Problem with Unloading Times. *Symmetry*, 15(11). <https://doi.org/10.3390/sym15112005>
- Huang, Z., Huang, L., & Li, D. (2024). *Co-Evolutionary Algorithm for Two-Stage Hybrid Flow Shop Scheduling Problem with Suspension Shifts*.
- Jacobs, F. R., Berry, W. L., Whybark, D. C., & Vollmann, T. E. (2024). *Manufacturing Planning and Control for Supply Chain Management: The CPIM Reference* (3rd ed.). McGraw Hill.
- Kanet, J., & Sridharan, V. (2000). *Scheduling with Inserted Idle Time: Problem Taxonomy and Literature Review* (pp. 99–110). Operations Research.
- Khurshid, B., Maqsood, S., Omair, M., Sarkar, B., Saad, M., & Asad, U. (2021a). *Fast Evolutionary algorithm for Flow Shop Scheduling Problems*. 1–17. <https://doi.org/10.1109/ACCESS.2021.3066446>
- Khurshid, B., Maqsood, S., Omair, M., Sarkar, B., Saad, M., & Asad, U. (2021b). *Fast Evolutionary algorithm for Flow Shop Scheduling Problems*. 1–17. <https://doi.org/10.1109/ACCESS.2021.3066446>
- Kim, T., & Glock, C. H. (2018). International Journal of Production Economics Production planning for a two-stage production system with multiple parallel machines and variable production rates. *International Journal of Production Economics*, 196(April 2016), 284–292. <https://doi.org/10.1016/j.ijpe.2017.11.018>
- Larasati, K. A., & Akbar, M. (2024). *A Dual Resource Constrained Unrelated Parallel Machine Scheduling Model Considering Tardiness and Workload Balance*. 26(2), 87–102.
- Madraki, G., Author, C., & Judd, R. P. (n.d.). *Accelerating the Calculation of Makespan Used in Scheduling Improvement Heuristics*. (315).
- Nasution, A. H., & Prasetyawan, Y. (2008). Perencanaan dan pengendalian produksi. *Yogyakarta: Graha Ilmu*.
- Pallas-Areny, R., & Webster, J. G. (2012). *Sensors and Signal Conditioning* (3rd ed.). John Wiley & Sons.
- Pan, Q. (2013). *Computers & Operations Research A comprehensive review and evaluation of permutation flowshop heuristics to minimize flowtime*. 40, 117–128. <https://doi.org/10.1016/j.cor.2012.05.018>
- Pinedo, M. L. (2016). *Scheduling: Theory, Algorithms, and Systems* (5th ed.). Springer. <https://doi.org/10.1007/978-3-662-44874-8>
- Pinedo, M. L. (2022a). Flow Shops and Flexible Flow Shops (Deterministic). In *Scheduling: Theory, Algorithms, and Systems* (pp. 153–186). Springer International Publishing. https://doi.org/10.1007/978-3-031-05921-6_6

- Pinedo, M. L. (2022b). Flow Shops, Job Shops, and Open Shops (Stochastic). In *Scheduling: Theory, Algorithms, and Systems* (pp. 357–380). Springer International Publishing. https://doi.org/10.1007/978-3-031-05921-6_13
- Risma, Y. M., & Utama, D. M. (2023). *AVOA and ALO Algorithm for Energy-Efficient No-Idle Permutation Flow Shop Scheduling Problem : A Comparison Study*. <https://doi.org/10.25077/josi.v22.n2.p126-141.2023>
- Ruiz, R., & Vázquez-Rodríguez, J. A. (2010). The hybrid flow shop scheduling problem. *European Journal of Operational Research*, 205(1), 1–18. <https://doi.org/https://doi.org/10.1016/j.ejor.2009.09.024>
- Saiful, M., Mustafid, M., & Suryono, S. (2022a). A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem. *Journal of King Saud University - Computer and Information Sciences*, 34(9), 7459–7467. <https://doi.org/10.1016/j.jksuci.2021.08.025>
- Saiful, M., Mustafid, M., & Suryono, S. (2022b). A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem. *Journal of King Saud University - Computer and Information Sciences*, 34(9), 7459–7467. <https://doi.org/10.1016/j.jksuci.2021.08.025>
- Slack, N., & Brandon-Jones, A. (2019a). *Operations Management* (9th ed.). Pearson. https://www.pearson.com/nl/en_NL/higher-education/subject-catalogue/decision-science/slack-operations-management-9thedition.html
- Slack, N., & Brandon-Jones, A. (2019b). *Operations Management* (9th ed.). Pearson.
- Song, J., Shen, Y., Wang, L., Liu, C., Tang, D., & Nie, Q. (2025). *An Efficient Job Insertion Algorithm for Hybrid Human – Machine Collaborative Flexible Job Shop Scheduling with Random Job Arrivals*. 1–22.
- Tremblet, D., Thevenin, S., Alexandre, T., Dolgui, A., Thevenin, S., Dolgui, A., Tremblet, D., Thevenin, S., Dolgui, A., Alfred, R., Kastler, A., & Chantreterie, L. (2022). Predicting makespan in Flexible Job Shop Predicting Predicting makespan makespan in in Flexible Flexible Job Job Shop Shop Scheduling Problem Machine Predicting makespan in using Flexible Job Shop Scheduling Problem using Machine Scheduling using Scheduli. *IFAC PapersOnLine*, 55(10), 1–6. <https://doi.org/10.1016/j.ifacol.2022.09.305>
- Wang, B., Huang, K., & Li, T. (2018). Computers & Industrial Engineering Permutation flowshop scheduling with time lag constraints and makespan criterion. *Computers & Industrial Engineering*, 120(February), 1–14. <https://doi.org/10.1016/j.cie.2018.04.021>
- Wang, Y., Jia, Z., & Zhang, X. (2022). A hybrid meta-heuristic for the flexible flow shop scheduling with blocking. *Swarm and Evolutionary Computation*, 75(February), 101195. <https://doi.org/10.1016/j.swevo.2022.101195>
- Xu, R., Xu, Y., & Xiao, W. (2026). A Q-learning based memetic algorithm for flexible flow shop scheduling with active waiting and job switching. *Swarm and Evolutionary Computation*, 100(July 2025), 102227. <https://doi.org/10.1016/j.swevo.2025.102227>
- Xue, L., Zhao, S., Mahmoudi, A., & Reza, M. (2024). Flexible job-shop scheduling problem with parallel batch machines based on an enhanced multi-population genetic algorithm. *Complex & Intelligent Systems*, 10(3), 4083–4101. <https://doi.org/10.1007/s40747-024-01374-7>
- Yu, X., & Gen, M. (2021). *Introduction to Evolutionary Algorithms* (R. Roy, Ed.; Vol. 1, Number 2020). Springer London Dordrecht Heidelberg New York.

- Yusriski, R., Rachmat, A., Nasution, K., Wijayanti, L., & Octaviani, S. (2023). *A Two-Machine Flow Shop Batch Scheduling Model to Minimize Total Actual Flow Time*. 25(2), 179–193.
- Zeschke, M., & Zacher, H. (2023). *Effects of Idle Time on Outcomes*. European Journal of Work and Organizational Psychology.
- Zhang, Z., Chen, H., Wang, Z., & Han, J. (2025). *Dynamic Flexible Job Shop Scheduling via an Adaptive Genetic Algorithm and Deep Learning*. 1–22.

The screenshot displays the Microsoft Excel application window. The title bar indicates the file is 'Data Skripsi Flexible Flow Line FKKK (2...)' and was last modified on Thursday at 9:02 PM. The ribbon is set to the 'Home' tab, showing options for Font, Paragraph, Styles, and Cells. The main workspace contains a Gantt chart titled 'Jadwal Aktual Perusahaan'. The chart uses a color-coded system where pink bars represent tasks and blue bars represent dependencies. The timeline spans from 19 to 24. The status bar at the bottom shows 'Ready', 'Accessibility: Investigate', and a zoom level of 70%.

A-Pengolahan Data Jadwal Aktual Perusahaan (Lanjutan)



B-Pengolahan Data Hasil Optimasi Solver

AutoSave Data Skripsi Flexible Flow Line F000X (2... Last Modified: Thu at 9:02 PM

File Home Insert Draw Page Layout Formulas Data Review View Automate Help

Clipboard Font Alignment Number Styles Cells Editing Sensitivity Add-ins Claude

P34

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1																											
2	Product ID	Job ID	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	Finish	Processor #
3	P14	14	1	0	0	5	2	7	1	0	0	7	5	10	1	0	0	10	4	14	1	0	0	14	5	19	2
4	P10	10	2	0	0	6	3	9	2	0	0	9	7	16	2	0	0	16	9	25	2	0	0	25	4	29	1
5	P15	15	3	0	0	8	3	11	1	14	1	11	5	16	1	14	1	16	5	22	3	14	1	22	5	27	2
6	P16	16	1	14	4	11	2	13	2	10	2	18	3	21	2	10	2	27	9	36	1	0	0	36	5	41	2
7	P7	7	2	10	2	11	9	20	1	15	1	20	6	26	1	15	1	26	6	32	1	16	4	45	7	52	1
8	P2	2	1	16	1	14	4	18	2	16	1	22	10	32	1	7	1	33	5	38	2	10	3	38	7	45	1
9	P3	3	5	15	3	18	4	22	1	7	1	27	4	31	2	16	2	38	4	42	3	15	2	42	9	51	1
10	P17	17	3	5	2	24	4	28	1	5	2	33	6	39	2	5	2	44	6	50	1	7	1	53	6	59	2
11	P19	19	1	2	1	19	5	24	2	2	1	33	7	40	1	2	1	40	9	49	2	2	1	49	8	57	2
12	P1	1	2	7	0	20	5	23	1	17	4	45	5	48	2	17	4	54	7	61	2	19	4	61	10	71	1
13	P20	20	2	1	3	26	3	29	2	19	1	41	9	50	1	19	1	50	6	56	3	5	1	56	4	60	2
14	P11	11	3	17	1	29	7	36	1	1	3	51	6	57	1	20	3	59	8	67	1	17	1	67	6	73	2
15	P3	3	3	11	3	39	5	44	1	11	3	60	5	65	2	1	1	65	5	70	3	20	4	70	6	76	1
16	P18	18	1	19	1	25	2	27	2	20	2	52	3	55	2	3	1	71	7	78	2	1	4	78	3	81	2
17	P4	4	2	20	1	30	9	39	2	18	2	57	6	63	1	11	1	68	8	76	1	11	1	76	7	83	2
18	P13	13	1	18	1	28	2	30	2	4	1	64	4	68	1	4	1	77	5	82	1	4	1	84	4	88	2
19	P12	12	3	3	2	46	5	51	2	13	1	69	5	74	2	18	1	79	10	89	2	18	1	89	5	94	1
20	P9	9	3	4	4	43	7	50	1	3	2	67	8	73	2	12	1	90	7	97	1	13	1	97	5	103	2
21	P8	8	3	12	2	53	3	56	1	9	3	76	5	81	1	13	1	83	6	89	3	3	3	89	10	99	1
22	P6	6	2	9	1	51	5	56	1	8	3	84	5	89	1	8	3	92	7	99	2	12	2	99	6	105	2
23																											
24																											
25																											
26	MMA	MD	MP	MI	ML																						
27	MB	ME	MG	MI	ML																						
28	MC	MD	MP	MI	ML																						
29	MA	ME	MG	MI	ML																						
30	MB	ME	MG	MI	ML																						
31	MA	ME	MP	MI	ML																						
32	MC	MD	MG	MI	ML																						
33	MA	ME	MP	MI	ML																						
34	MB	ME	MG	MI	ML																						
35	MA	ME	MP	MI	ML																						
36	MB	ME	MG	MI	ML																						

Jadwal Aktual Perusahaan GanttChart Jadwal Aktual Optimasi Solver 1 Optimasi Solver 2 GanttChart

Ready Accessibility: Investigate 24°C Berawan 8:03 PM 6/29/2026

AutoSave

File Home Insert Draw Page Layout Formulas Data Review View Automate Help

Clipboard

Font

Alignment

Number

Styles

Cells

Editing

Sensitivity

Add-ins

Claude

Comments Share

Calibri 11

General

Conditional Formatting

Format as Table

Cell Styles

Insert

Delete

Format

Cells

Σ

Sort & Filter

Find & Select

Security

Add-ins

Claude

P34

fx

	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	
1				Work Station 4						Work Station 5						Idle Time														
2	Finish	Processor	Previous Job	Setup	Start	Duration	Finish	Processor	Previous Job	Setup	Start	Duration	Finish	WS 1	WS 2	WS 3	WS 4	WS 5												
3	14	3	0	0	14	5	19	2	0	0	19	2	21	0	0	0	0	0												
4	25	2	0	0	25	4	29	1	0	0	29	7	36	0	0	0	0	0												
5	22	3	14	1	22	5	27	2	14	1	27	3	30	0	0	0	1	2	5											
6	36	1	0	0	36	5	41	2	15	4	41	3	44	0	0	0	0	0	7											
7	32	1	16	4	45	7	52	1	10	2	52	9	61	0	0	0	3	0	14											
8	38	2	10	3	38	7	45	1	7	1	62	3	65	0	0	0	0	6	0											
9	42	3	15	2	42	9	51	1	2	4	69	4	73	0	0	0	0	13	0											
10	50	1	7	1	53	6	59	2	16	3	59	4	63	0	0	0	0	12												
11	49	2	2	1	49	8	57	2	17	4	67	6	73	0	0	0	1	3	0											
12	61	2	19	4	61	10	71	1	5	3	76	7	83	0	0	0	0	0												
13	56	3	5	1	56	4	60	2	19	1	74	4	78	0	0	0	0	4	0											
14	67	1	17	1	67	6	73	2	20	3	81	4	85	0	0	0	0	7	0											
15	70	3	20	4	70	6	76	1	1	1	84	9	93	0	0	0	0	8	6	0										
16	78	2	1	4	78	3	81	2	11	1	86	2	88	0	0	0	0	5	0											
17	76	1	11	1	76	7	83	2	18	2	90	6	96	0	0	0	0	2	0											
18	82	1	4	1	84	4	88	2	4	1	97	5	100	0	0	0	0	0	0											
19	89	2	18	1	89	5	94	1	3	2	95	5	100	0	0	0	0	7	0											
20	97	1	13	1	97	5	102	2	13	1	102	3	105	0	0	0	0	8	1											
21	89	3	3	3	89	10	99	1	12	2	102	7	109	0	0	0	0	10	0											
22	99	2	12	2	99	6	105	2	9	1	106	4	110	0	0	0	0	3	0											
23																														
24																														
25																														
26																														
27																														
28																														
29																														
30																														
31																														
32																														
33																														
34																														
35																														

Jadwal Aktual Perusahaan

GanttChart Jadwal Aktual

Optimasi Solver 1

Optimasi Solver 2

GanttChart

Ready

Accessibility: Investigate

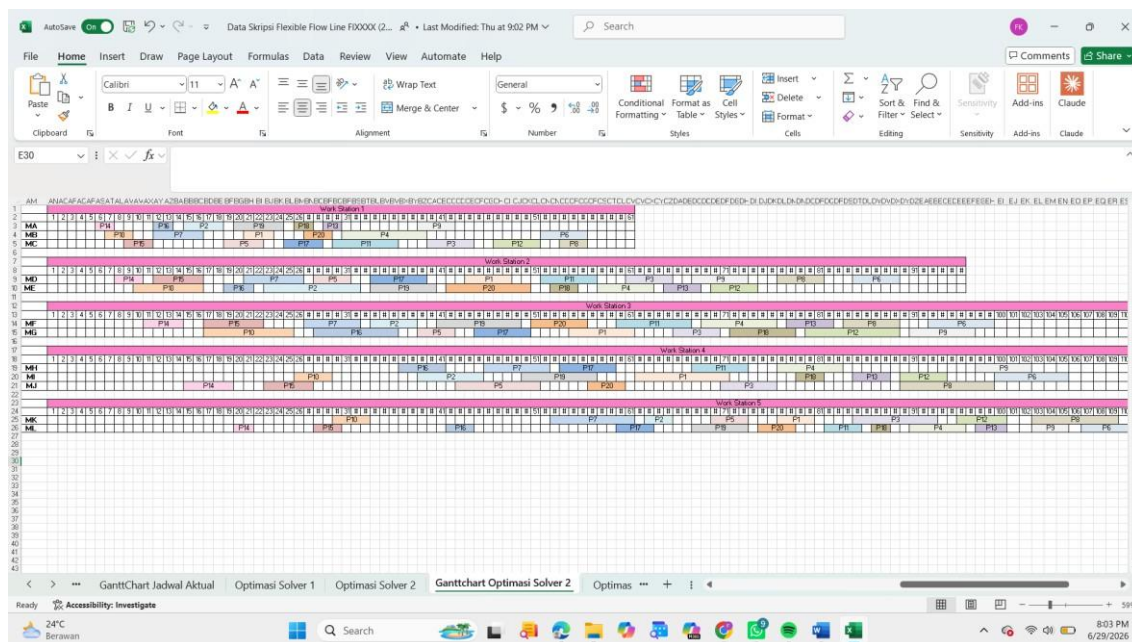
24°C

Berawan

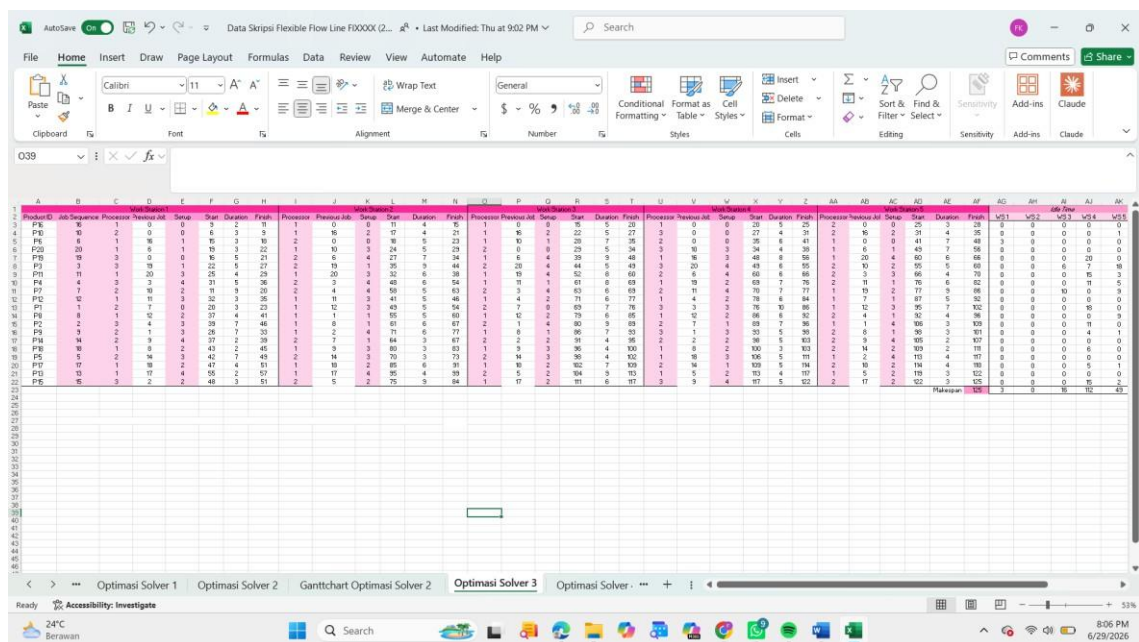
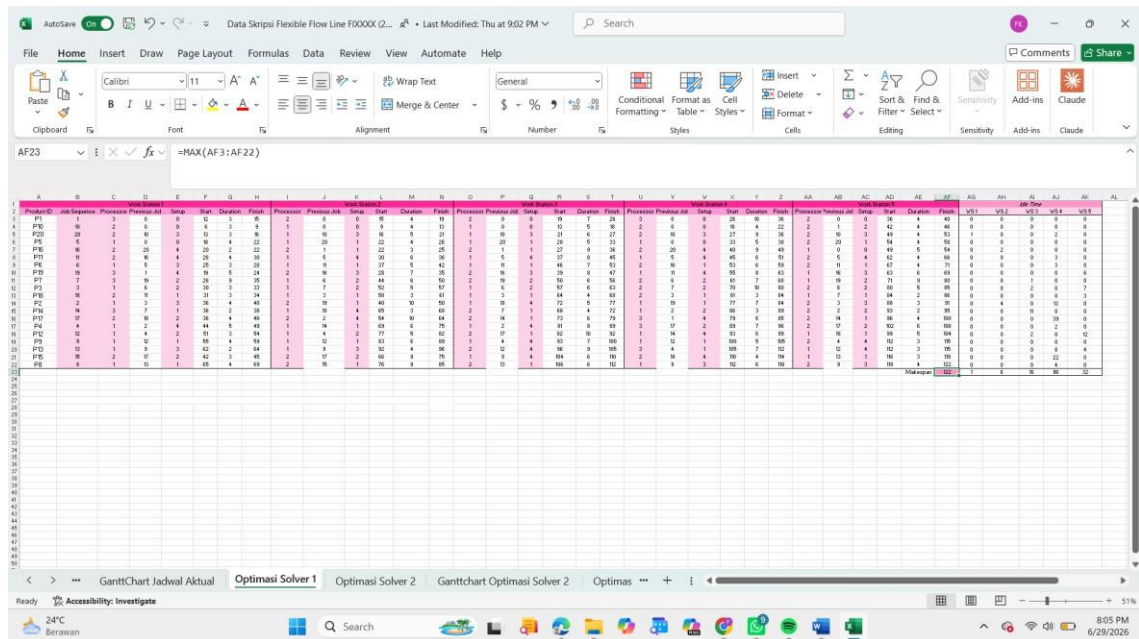
8:03 PM

6/29/2026

B-Pengolahan Data Hasil Optimasi Solver (Lanjutan)



C-Hasil Percobaan *Running Excel Solver*



C-Hasil Percobaan *Running Excel Solver* (Lanjutan)

