

**PENINGKATAN PERFORMANSI DUE DATE DENGAN
PENDEKATAN ALGORITMA AGGREGATE TIME BUFFER
(Studi Kasus pada CV Anugrah Furniture, Klaten)**

TUGAS AKHIR

**Diajukan sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Teknik Industri**



oleh :

Nama : Aditya Purba Kesuma

No. Mahasiswa : 02 522 046

**JURUSAN TEKNIK INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGJAKARTA**

2007

LEMBAR PENGESAHAN PEMBIMBING

**PENINGKATAN PERFORMANSI DUE DATE DENGAN
PENDEKATAN ALGORITMA AGGREGATE TIME BUFFER
(Studi Kasus pada CV Anugrah Furniture, Klaten)**

TUGAS AKHIR

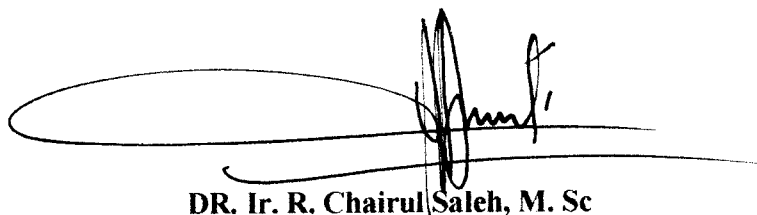
oleh :

Nama : Aditya Purba Kesuma

No. Mahasiswa : 02 522 046

Yogyakarta, Januari 2007

Pembimbing



DR. Ir. R. Chairul Saleh, M. Sc

HALAMAN PERSEMBAHAN

*Kupersembahkan karya ini untuk ibuku, ayahku,
Maz bayu, dimas, dan dio
Sahabat – sahabatku yang telah hadir dalam perjalanan hidupku
Serta Eneng Ristiawati yang selalu memotivasi dan mendoakanku.
Terima kasih untukmu atas kasih sayang, pengertian dan perhatian yang
telah engkau berikan sampai saat ini.*

MOTTO

*“ Aku mendengar Rasulullah Shallahu’alaihi wa sallam bersabda,
“ Sesungguhnya yang pertama kali dihisap dari seorang hamba adalah sholatnya.
Apabila (sholatnya) baik, maka dia pasti beruntung dan selamat,
dan apabila (sholatnya) rusak, maka dia pasti kecewa dan rugi.
(Abu Hurairah)*

*“ Allah turun ke langit dunia ketika sepertiga malam yang pertama berlalu.
Kemudian Dia berfirman, “ Aku adalah Al – Malik, “ Aku adalah Al – Malik,
“ Siapa yang mau berdoa kepadaku – Ku, lalu Aku mengabulkannya?
“ Siapa yang mau memohon ampunan kepadaku – Ku,
lalu Aku memberi ampunan kepadanya?
' Dia tidak henti – hentinya berfirman seperti itu hingga fajar menyingsing.”
(Abu “Isa At-Tarmudzi)*

*“Allah meninggikan orang yang beriman diantara kamu dan orang-orang yang diberi
ilmu pengetahuan, beberapa derajat
(Al-Mujadilah : 11)*

KATA PENGANTAR



Assalamu'alaikum Wr. Wb.

Segala puji syukur penulis panjatkan Kehadirat Allah SWT yang telah melimpahkan rahmat, karunia serta hidayah-Nya kepada hamba-Nya selama masih dalam iman dan ikhsan.

Atas petunjuk dan ridho-Nya jualah Tugas Akhir ini dapat diselesaikan dengan baik dan lancar. Tugas Akhir ini wajib ditempuh oleh mahasiswa Jurusan Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia, sebagai salah satu syarat untuk menyelesaikan jenjang studi Strata I.

Kelancaran dalam mempersiapkan dan menyelesaikan Tugas Akhir ini tidak terlepas dari bantuan berbagai pihak. Oleh karena itu dengan rasa hormat dan terima kasih yang sebesar-besarnya Penulis haturkan kepada:

1. Dekan Fakultas Teknologi Industri, Universitas Islam Indonesia
2. Ketua Jurusan Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia.
3. Bapak DR. Ir. Chairul Saleh, M. Sc., selaku Dosen Pembimbing, yang telah memberikan bimbingan dalam penyusunan laporan Tugas Akhir ini.
4. Aditya Gatot P, yang telah memberikan izin untuk menggunakan data penetiannya.
5. Ayah & Bunda, serta keluargaku , terimakasih atas do'a dan kasih sayangmu.
6. Eneng Ristiawati, terimakasih atas do'a dan kasih sayangmu.
7. Seluruh Asisten Laboratorium Sistem Produksi FTI UII.
8. Semua pihak yang telah memberikan masukan, dorongan dan semangat dalam penyelesaian Tugas Akhir ini.

Wassalamu'alaikum Wr. Wb.

Yogyakarta, 11 Januari 2007

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING.....	ii
LEMBAR PENGESAHAN PENGUJI.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	v
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	xi
ABSTRAKSI	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	6
2.1 Pendahuluan.....	6
2.2 Konsep penjadwalan.....	7
2.2.1 Definisi Penjadwalan	7
2.2.2 Penjadwalan Flow Shop	8
2.2.3 Tujuan Penjadwalan.....	9
2.3 Theory of constraints (TOC).....	9
2.3.1 Definisi Theory Of Constraints.....	9

2.3.2	Bottleneck dan Non Bottleneck	11
2.3.3	Langkah Utama Dalam Toc.....	12
2.3.4	Drum Buffer Rope (DBR).....	14
2.4	Algoritma Zijm	16
2.5	Aggregate Time Buffer.....	18
2.5.1	Aggregate Time Buffer Control.....	18
BAB III METODOLOGI PENELITIAN		23
3.1	Studi Pustaka	24
3.2	Penentuan Obyek Penelitian	24
3.3	Analisis Model	24
3.4	Identifikasi dan Perumusan Masalah.....	25
3.5	Pengumpulan Data	26
3.6	Pengolahan dan Analisis Hasil.....	26
3.7	Hasil Penelitian	30
BAB IV PENGUMPULAN DAN PENGOLAHAN DATA.....		31
4.1	Pengumpulan Data	31
4.1.1	Struktur Produk	31
4.1.2	Due date Setiap Part.....	32
4.1.3	Peta Proses Operasi.....	32
4.1.4	Mesin yang di Gunakan dalam Proses Produksi	33
4.1.5	Waktu Setup Mesin.....	33
4.1.6	Waktu Proses Setiap Job di Setiap Mesin.....	34
4.2	Pengolahan Data.....	35
4.2.1	Mengidentifikasi mesin bottleneck dan Penentuan Time Buffer dengan Algoritma Zijm.....	35
4.2.2	Penjadwalan Mesin Bottleneck dengan Memperhatikan	

	Time Buffer Menggunakan Metode FCFS.....	45
4.2.3	Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer.....	47
4.2.4	Menentukan Aggregate Time Buffer.....	48
4.2.5	Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer.....	53
4.2.6	Penjadwalan Mesin Bottleneck dengan Memperhatikan Time Buffer Menggunakan Metode EDD.....	53
4.2.7	Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer.....	55
4.2.8	Menentukan Aggregate Time Buffer.....	56
4.2.9	Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer.....	61
4.2.10	Penjadwalan Mesin Bottleneck dengan Memperhatikan Time Buffer Menggunakan Metode SPT.....	61
4.2.11	Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer.....	63
4.2.12	Menentukan Aggregate Time Buffer.....	64
4.2.13	Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer.....	69
4.2.14	Penjadwalan Mesin Bottleneck dengan Memperhatikan Time Buffer Menggunakan Metode LPT.....	69
4.2.15	Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer.....	71
4.2.16	Menentukan Aggregate Time Buffer.....	72
4.2.17	Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer.....	76

4.2.18	Penjadwalan Mesin Bottleneck dengan Memperhatikan Time Buffer Menggunakan Metode CR.....	77
4.2.19	Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer.....	79
4.2.20	Menentukan Aggregate Time Buffer.....	80
4.2.21	Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer.....	84
4.2.22	Perbandingan Tardiness dan Lateness Setiap Job.....	85
BAB V PEMBAHASAN.....		88
BAB VI KESIMPULAN DAN SARAN.....		89
6.1	Kesimpulan.....	89
6.2	Saran.....	89
DAFTAR PUSTAKA		
LAMPIRAN		

DAFTAR TABEL

Tabel 4.1	Data Duedate Setiap Part.....	32
Tabel 4.2	Data waktu setup Mesin	33
Tabel 4.3	Data Waktu proses setiap Job di Setiap mesin	34
Tabel 4.4	Laju kedatangan <i>job</i> di setiap mesin	37
Tabel 4.5	Waktu proses sebuah <i>job</i> sudah termasuk waktu <i>set-up</i>	39
Tabel 4.6	Rata-rata beban kerja di setiap mesin	40
Tabel 4.7	Ekspektasi waktu menunggu setiap <i>job</i> di setiap mesin.....	41
Tabel 4.8	Ekspektasi rata-rata <i>leadtime</i> produksi (menit).....	43
Tabel 4.9	Ekspektasi rata-rata <i>leadtime</i> setiap <i>job</i>	44
Tabel 4.10	<i>ETC</i> dan <i>LTC</i> setiap <i>job</i> (menit).....	45
Tabel 4.11	Hasil perhitungan dengan metode FCFS.....	45
Tabel 4.12	Hasil perhitungan <i>ETC</i> dan <i>LTC</i>	47
Tabel 4.13	Duedate setiap job dengan memperhatikan time buffer.....	47
Tabel 4.14	Makespan setiap job dengan memperhatikan time buffer	47
Tabel 4.15	Hasil perhitungan latenees dan tardiness setiap job.....	48
Tabel 4.16	Hasil perhitungan total time buffer setiap job.....	48
Tabel 4.17	Total waktu proses setelah bottleneck dan total waktu proses....	49
Tabel 4.18	Hasil perhitungan Aggregate time buffer setiap job.....	50
Tabel 4.19	Waktu kedatangan setiap job pada mesin pertama.....	50
Tabel 4.20	Hasil perhitungan duedate.....	51
Tabel 4.21	Hasil perhitungan expexted duedate.....	51
Tabel 4.22	Data waktu kedatangan job pada mesin bottleneck.....	52
Tabel 4.23	Hasil sisa dari Aggregate Time Buffer.....	52
Tabel 4.24	Hasil sisa uji validasi.....	53
Tabel 4.25	Hasil perhitungan latenees dan tardiness setiap job.....	53

Tabel 4.26	Urutan pengerjaan job berdasarkan Earliness Duedate.....	53
Tabel 4.27	Hasil perhitungan dengan metode EDD.....	54
Tabel 4.28	Hasil perhitungan ETC dan LTC.....	55
Tabel 4.29	Duedate setiap job dengan memperhatikan time buffer.....	55
Tabel 4.30	Makespan setiap job dengan memperhatikan time buffer	56
Tabel 4.31	Hasil perhitungan latenees dan tardiness setiap job.....	56
Tabel 4.32	Hasil perhitungan total time buffer setiap job.....	57
Tabel 4.33	Total waktu proses setelah bottleneck dan total waktu proses....	57
Tabel 4.34	Hasil perhitungan Aggregate time buffer setiap job.....	58
Tabel 4.35	Waktu kedatangan setiap job pada mesin pertama.....	58
Tabel 4.36	Hasil perhitungan duedate.....	59
Tabel 4.37	Hasil perhitungan expexted duedate.....	59
Tabel 4.38	Data waktu kedatangan job pada mesin bottleneck.....	60
Tabel 4.39	Hasil sisa dari Aggregate Time Buffer.....	60
Tabel 4.40	Hasil sisa uji validasi.....	53
Tabel 4.41	Hasil perhitungan latenees dan tardiness setiap job.....	61
Tabel 4.42	Urutan pengerjaan job berdasarkan waktu terpendek.....	61
Tabel 4.43	Hasil perhitungan dengan metode SPT	62
Tabel 4.44	Hasil perhitungan ETC dan LTC.....	63
Tabel 4.45	Duedate setiap job dengan memperhatikan time buffer.....	63
Tabel 4.46	Makespan setiap job dengan memperhatikan time buffer	64
Tabel 4.47	Hasil perhitungan latenees dan tardiness setiap job.....	64
Tabel 4.48	Hasil perhitungan total time buffer setiap job.....	65
Tabel 4.49	Total waktu proses setelah bottleneck dan total waktu proses....	65
Tabel 4.50	Hasil perhitungan Aggregate time buffer setiap job.....	66
Tabel 4.51	Waktu kedatangan setiap job pada mesin pertama.....	66
Tabel 4.52	Hasil perhitungan duedate.....	67
Tabel 4.53	Hasil perhitungan expexted duedate.....	67

Tabel 4.54 Data waktu kedatangan job pada mesin bottleneck.....	67
Tabel 4.55 Hasil sisa dari Aggregate Time Buffer.....	68
Tabel 4.56 Hasil sisa uji validasi.....	68
Tabel 4.57 Hasil perhitungan latenees dan tardiness setiap job.....	69
Tabel 4.58 Urutan pengerjaan job berdasarkan waktu terpanjang.....	69
Tabel 4.59 Hasil perhitungan dengan metode LPT.....	70
Tabel 4.60 Hasil perhitungan ETC dan LTC.....	71
Tabel 4.61 Duedate setiap job dengan memperhatikan time buffer.....	71
Tabel 4.62 Makespan setiap job dengan memperhatikan time buffer	72
Tabel 4.63 Hasil perhitungan latenees dan tardiness setiap job.....	72
Tabel 4.64 Hasil perhitungan total time buffer setiap job.....	73
Tabel 4.65 Total waktu proses setelah bottleneck dan total waktu proses....	73
Tabel 4.66 Hasil perhitungan Aggregate time buffer setiap job.....	74
Tabel 4.67 Waktu kedatangan setiap job pada mesin pertama.....	74
Tabel 4.68 Hasil perhitungan duedate.....	74
Tabel 4.69 Hasil perhitungan expexted duedate.....	75
Tabel 4.70 Data waktu kedatangan job pada mesin bottleneck.....	75
Tabel 4.71 Hasil sisa dari Aggregate Time Buffer.....	76
Tabel 4.72 Hasil sisa uji validasi.....	76
Tabel 4.73 Hasil perhitungan latenees dan tardiness setiap job.....	77
Tabel 4.74 Urutan pengerjaan job berdasarkan waktu terpanjang.....	77
Tabel 4.75 Hasil perhitungan dengan metode LPT	77
Tabel 4.76 Hasil perhitungan ETC dan LTC.....	79
Tabel 4.77 Duedate setiap job dengan memperhatikan time buffer.....	79
Tabel 4.78 Makespan setiap job dengan memperhatikan time buffer	79
Tabel 4.79 Hasil perhitungan latenees dan tardiness setiap job.....	80
Tabel 4.80 Hasil perhitungan total time buffer setiap job.....	80
Tabel 4.81 Total waktu proses setelah bottleneck dan total waktu proses....	81

Tabel 4.82 Hasil perhitungan Aggregate time buffer setiap job.....	81
Tabel 4.83 Waktu kedatangan setiap job pada mesin pertama.....	82
Tabel 4.84 Hasil perhitungan due date.....	82
Tabel 4.85 Hasil perhitungan expected due date.....	83
Tabel 4.86 Data waktu kedatangan job pada mesin bottleneck.....	83
Tabel 4.87 Hasil sisa dari Aggregate Time Buffer.....	83
Tabel 4.88 Hasil sisa uji validasi.....	84
Tabel 4.89 Hasil perhitungan lateness dan tardiness setiap job.....	84
Tabel 4.90 Perbandingan Tardiness dan Lateness Masing-Masing Metode.	85
Tabel 4.91 Peningkatan Ratio Slack Masing-Masing Metode.....	84

DAFTAR GAMBAR

Gambar 2.1 ilustrasi Lingkungan flow Shop	9
Gambar 2.2 Teknik DBR	15
Gambar 2.3 DBR Kegiatan Mundur.....	15
Gambar 2.4 Model Aggregate Time Buffer.....	19
Gambar 3.1 Diagram Alir Kerangka Penelitian.....	23
Gambar 4.1 Struktur Produk	32

Abstraksi

Untuk dapat mengontrol *bottleneck* agar dapat melindungi laju produksi dari gangguan yang terjadi dalam sistem produksi dilakukan dengan menambah suatu penyangga *buffer*. Penambahan sejumlah *Time Buffer* dapat melindungi laju produksi, akan tetapi mempengaruhi *makespan* yang sangat berkaitan erat dengan performansi *duedate*. Pada penelitian ini agar diperoleh performansi *duedate* yang baik berdasarkan *tardiness* dan *lateness* maka dilakukan dengan pendekatan *TOC Aggregate*. Optimasi yang dilakukan melalui pendekatan *TOC Aggregate* dapat meningkatkan performansi *duedate* sebesar 95,522 %, Dengan demikian dapat diambil kesimpulan bahwa dengan menggunakan *TOC Aggregate* dapat meningkatkan performansi *duedate* setiap job berdasarkan *Aggregate time buffer* pada masing-masing job.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dewasa ini banyak filosofi pemanufakturan yang dapat diterapkan untuk mendukung sistem produksi yang tepat dan efisien. Salah satu filosofi pemanufakturan yang banyak diterapkan di dunia industri adalah *Theory of Constraints* (TOC). TOC yang murni berasal dari scheduling adalah *Optimizing Production Technology* (OPT). Teknik scheduling dari OPT lebih dikenal dengan nama *Drum Buffer Rope* (DBR).

TOC adalah cara untuk mengatasi *constraints*, pada awal kerjanya, Goldratt menyatakan bahwa mesin sebagai salah satu sumber yang membatasi aliran produksi, dimana ada ketidak seimbangan aliran produksi yang menyebabkan mesin *bottleneck*. Untuk mengidentifikasi mesin *bottleneck* perlu menganalisa beberapa komponen yang saling terkait yaitu sumber daya, kapasitas mesin dan pesanan. Pada perkembangan selanjutnya Goldratt cenderung menggunakan kata kendala (*constraints*) untuk menerangkan ide-ide yang sama pada situasi dimana hasil produksi (*throughput*) dibatasi oleh sesuatu hal. Sehingga *bottleneck* digunakan untuk mesin *bottleneck* sedangkan kendala (*constraint*) ditujukan untuk segala sesuatu yang membatasi hasil (*throughput*) termasuk kebijakan yang buruk ataupun situasi pasar. Mesin *bottleneck* merupakan salah satu contoh dari *constraints* pada rantai produksi.

Sejumlah peneliti telah mengimplementasikan prosedur dari konsep TOC dalam rantai produksi. Seperti yang dilakukan Goldrat (1990) dimana focus penelitian pada sumber daya *bottleneck*, kemudian menjaga sumber daya *bottleneck* agar tetap dapat mengoptimalkan kapasitasnya. Berdasarkan definisi tersebut ada beberapa sumber daya *Non-Bottleneck* yang memiliki kelebihan kapasitas. Kelebihan kapasitas tersebut menyebabkan *utilitas* mesin diatas 100% (Tu dan Li, 1998). Dalam TOC

kendala dikategorikan kedalam dua kondisi yaitu bottleneck permanent dan bottleneck sementara (Simons, Jr, *et.al*, 1996).

Adanya *bottleneck* pada rantai produksi dapat menurunkan performansi dari sistem produksi. Apabila *bottleneck* tersebut terjadi dilantai produksi, salah satunya adalah dapat menurunkan performansi *duedate*. Dimana *duedate* sangat berpengaruh kepada kepercayaan konsumen, perencanaan pemasaran dari pesaing, dan kapasitas produksi.

Berbagai metode sering digunakan untuk mengoptimalkan bottleneck, salah satunya adalah konsep *Drum Buffer Rope* (DBR) yang digunakan untuk menganalisa penyangga (*buffer*) yang berfungsi untuk menjaga agar mesin *bottleneck* dapat berproduksi 100%. Sedangkan teknik yang dipergunakan dalam masalah penggunaan *buffer* ini adalah *Aggregate Time Buffer* (ATB).

Dalam penelitian ini akan dikembangkan kombinasi urutan pengerjaan produk pada tipe lingkungan flowshop untuk mengoptimalkan performansi *duedate* dengan pendekatan *TOC Aggregate*. Dengan menggunakan konsep ATB diharapkan dapat meminimalkan rata-rata tardiness, lateness dan tardiness setiap job.

1.2 Rumusan Masalah

Dari uraian latar belakang penulisan tugas akhir ini maka dapat diidentifikasi serta diformulasikan permasalahan sebagai berikut :

1. Bagaimanakah jika time buffer yang digunakan dalam ATB diganti dengan Algoritma Zijm ?
2. Bagaimana pengaruhnya penggantian Algoritma Zijm terhadap due date yang dihasilkan ?

1.3 Batasan Masalah

Agar penelitian dapat mengarah sesuai dengan pokok permasalahan, maka ruang lingkup dalam penelitian ini diberi beberapa pembatasan sebagai berikut:

1. Penelitian dilakukan untuk melakukan penjadwalan dan pengontrolan produksi pada lingkungan manufaktur *Make To Order (MTO)*.
2. Penelitian diperoleh dari studi kasus yang didapat dari kasus penjadwalan mesin produksi tipe *flow shop*.
3. Sumberdaya seperti tenaga kerja tidak termasuk dalam penelitian ini.
4. Mekanisme kerja dari para operator dinilai sudah cukup baik sehingga penelitian ini hanya bertujuan untuk mencari urutan *job* yang baik tidak bertujuan untuk menyeimbangkan beban kerja untuk setiap mesin.
5. Waktu perpindahan material diabaikan karena letak setiap mesin berdekatan sehingga waktu perpindahan material diabaikan.
6. Tidak dilakukan *batch* pada *lot* produk.
7. Semua *job* yang akan dikerjakan dalam sebuah mesin tersedia di awal sehingga tidak ada keterlambatan kedatangan *job* serta tidak boleh ada interupsi lebih satu *job* dikerjakan pada suatu mesin.

1.4 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian tugas akhir ini adalah:

“Meningkatkan performansi *duedate* setiap *job* berdasarkan *Aggregate time buffer* masing-masing *job*.”



1.5 Manfaat Penelitian

Manfaat penelitian ini adalah:

1. Dapat menentukan due date yang optimal setiap job untuk meningkatkan performansi due date sehingga dapat menumbuhkan kepercayaan konsumen.
2. Dengan adanya buffer time bagi mesin bottleneck akan memperlancar aliran produksi sehingga ketangguhan sistem tetap terjaga.

1.6 Sistematika Penulisan

Untuk mempermudah penulisan tugas akhir ini maka dalam penyusunannya penulis memberikan sistematika penulisan berdasarkan bab demi bab yang berurutan, berdasarkan pokok-pokok permasalahan yang terbagi menjadi lima bab yaitu:

BAB II : LANDASAN TEORI

Bab ini merupakan penjelasan terperinci mengenai teori-teori yang digunakan sebagai landasan untuk pemecahan masalah.

BAB III : METODOLOGI PENELITIAN

Bab ini memberikan penjelasan tentang bahan atau materi penelitian, alat dan tata cara penelitian, variabel, data yang akan diteliti dan langkah-langkah analisis yang dipakai serta *flow chart* penelitian.

BAB IV : PENGUMPULAN DAN PENGOLAHAN DATA

Dalam bab ini menjelaskan mengenai pengumpulan data berdasarkan penelitian dan pengolahan data berdasarkan hasil perhitungan.

BAB V : PEMBAHASAN

Berisikan pembahasan yang diperoleh dari hasil pengolahan data yang dilakukan.

BAB VI : KESIMPULAN DAN SARAN

Bab ini merupakan bab terakhir yang berisikan kesimpulan yang diperoleh dari pemecahan masalah maupun dari hasil pengumpulan data serta diajukan beberapa saran untuk bahan peninjauan selanjutnya.

DAFTAR PUSTAKA**LAMPIRAN**

BAB II

LANDASAN TEORI

2.1 Pendahuluan

Dewasa ini banyak filosofi pemanufakturan yang dapat diterapkan untuk mendukung sistem produksi yang tepat dan efisien.. TOC adalah cara untuk mengatasi *constraints*, pada awal kerjanya, Goldratt menyatakan bahwa mesin sebagai salah satu sumber yang membatasi aliran produksi, dimana ada ketidak seimbangan aliran produksi yang menyebabkan mesin *bottleneck*. Untuk mengidentifikasi mesin *bottleneck* perlu menganalisa beberapa komponen yang saling terkait yaitu sumber daya, kapasitas mesin dan pesanan. Pada perkembangan selanjutnya Goldratt cenderung menggunakan kata kendala (*constraints*) untuk menerangkan ide-ide yang sama pada situasi dimana hasil produksi (*throughput*) dibatasi oleh sesuatu hal. Sehingga *bottleneck* digunakan untuk mesin *bottleneck* sedangkan kendala (*constraint*) ditujukan untuk segala sesuatu yang membatasi hasil (*throughput*) termasuk kebijakan yang buruk ataupun situasi pasar. Mesin *bottleneck* merupakan salah satu contoh dari *constraints* pada rantai produksi.

Sejumlah peneliti telah mengimplementasikan prosedur dari konsep TOC dalam rantai produksi. Seperti yang dilakukan Goldrat (1990) dimana focus penelitian pada sumber daya *bottleneck*, kemudian menjaga sumber daya *bottleneck* agar tetap dapat mengoptimalkan kapasitasnya. Berdasarkan definisi tersebut ada beberapa sumber daya *Non-Bottleneck* yang memiliki kelebihan kapasitas. Kelebihan kapasitas tersebut menyebabkan *utilitas* mesin diatas 100% (Tu dan Li, 1998). Dalam TOC kendala dikategorikan kedalam dua kondisi yaitu *bottleneck* permanent dan *bottleneck* sementara (Simons, Jr, *et. al*, 1996).

2.2 Konsep Penjadwalan / *Scheduling*.

2.2.1 Definisi Penjadwalan

Scheduling, secara umum didefinisikan sebagai kegiatan untuk mengalokasikan sumber daya yang dipunyai untuk mengerjakan sekumpulan pekerjaan (*job*). Dari definisi tersebut, dapat ditarik dua hal yang berkaitan dengan *scheduling*. Pertama, *scheduling* adalah sebagai fungsi pembuat keputusan. Dalam lingkungan manufaktur, dengan sumber daya yang terbatas, apabila terjadi permintaan dari konsumen, maka kegiatan *scheduling* sangat dibutuhkan untuk dapat memenuhi permintaan tersebut. Penjadwalan pengerjaan *job* yang kurang baik bisa mengakibatkan permintaan yang terjadi tidak bisa dipenuhi semuanya. Sehingga, dalam hal ini, *scheduling* merupakan fungsi pembuat keputusan untuk mengalokasikan sumber daya yang dipunyai dalam usaha untuk memenuhi permintaan. Kedua, *scheduling* merupakan dasar pemikiran optimasi. Untuk dapat melakukan kegiatan *scheduling* dengan baik, maka diperlukan adanya penguasaan prinsip, model, teknik, dan konklusi logis. Sehingga, dengan penguasaan hal tersebut, akan dapat mendasari kegiatan optimasi dalam semua bidang.

Penjadwalan (*scheduling*), didefinisikan oleh (Baker, 1974) sebagai proses pengalokasian sumber untuk memilih mengerjakan sekumpulan tugas dalam jangka waktu tertentu. Definisi umum ini dapat dijabarkan dalam dua arti yang berbeda. Yang pertama adalah bahwa penjadwalan merupakan sebuah fungsi pengambilan keputusan, yaitu dalam menentukan jadwal yang paling tepat. Arti ke - dua adalah bahwa penjadwalan merupakan sebuah teori yang berisi kumpulan prinsip, model, teknik dan konklusi logis dalam proses pengambilan keputusan.

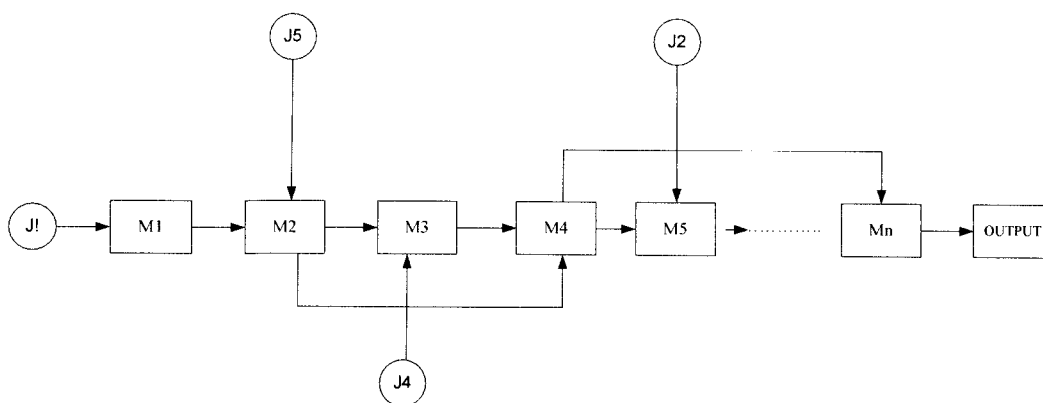
(Morton, 1993) mendefinisikan penjadwalan sebagai pengambilan keputusan tentang penyesuaian aktivitas dan sumber daya dalam rangka menyelesaikan sekumpulan pekerjaan agar tepat pada waktunya dan mempunyai kualitas seperti yang diharapkan. Keputusan yang dibuat dalam penjadwalan meliputi pengurutan pekerjaan (*sequencing*), waktu mulai dan selesai pekerjaan (*timing*), dan urutan operasi untuk suatu pekerjaan (*routing*). Masalah penjadwalan selalu berkaitan dengan pengurutan produksi (*sequencing*), yang dengan demikian didefinisikan sebagai penentuan urutan kedatangan dari bermacam-macam pekerjaan yang harus diselesaikan dalam jangka waktu tertentu.

(Andrew, 1991) mendefinisikan penjadwalan sebagai pengambilan keputusan tentang penyesuaian aktivitas dan sumber daya dalam rangka menyelesaikan sekumpulan pekerjaan agar tepat pada waktunya dan mempunyai kualitas seperti yang diharapkan. Keputusan yang dibuat dalam penjadwalan meliputi pengurutan pekerjaan (*sequencing*), waktu mulai dan selesai pekerjaan (*timing*), dan urutan operasi untuk suatu pekerjaan (*routing*).

2.2.2 FLOWSHOP SCHEDULING

Dalam beberapa situasi, terkadang *part-part* penyusun produk membutuhkan beberapa operasi dengan mesin yang berbeda. Sehingga dapat dikatakan bahwa setiap *part (job)* memiliki m pekerjaan yang membutuhkan sejumlah m mesin. Apabila sebuah produk membutuhkan n *part (job)*, yang urutan proses setiap *part*-nya sama, maka dapat dikatakan bahwa sistem tersebut mempunyai n *job* dan m mesin.

Dalam sistem diatas setiap *part* membutuhkan urutan proses yang sama sehingga mesin-mesin akan disusun secara seri berdasarkan urutan proses setiap *part*. Sebagai berikut ilustrasinya:



Gambar 2.1. Ilustrasi Lingkungan *Flow Shop*

2.2.3 Tujuan Penjadwalan

Secara umum tujuan penjadwalan adalah (Baker, 1974) :

1. Meningkatkan produktifitas mesin, yaitu dengan mengurangi waktu mesin menganggur.
2. Mengurangi persediaan barang setengah jadi dengan jalan mengurangi jumlah rata-rata pekerjaan yang menunggu antrian suatu mesin karena mesin tersebut sibuk.
3. Meminimasi biaya produksi.
4. Mengurangi keterlambatan karena telah melampaui batas waktu, dengan cara :
 - a. Mengurangi maksimum keterlambatan
 - b. Mengurangi jumlah pekerjaan yang terlambat.

2.3 Theory Of Constraints (TOC)

2.3.1 Definisi

Constraints adalah segala sesuatu di dalam organisasi yang menjadi pembatas untuk menuju pada tujuan organisasi. Ada dua tipe *constraints* menurut (Umble,1996) adalah *bottleneck* dan sumber kapasitas (*capacity constraints resource*). Sumber daya berkendala kapasitas (CCR) adalah sumber daya yang jika tidak dijadwalkan sebagaimana mestinya akan menghambat aliran produk yang menyimpang dari perencanaan aliran semula. Sumber daya berkendala kapasitas (CCR) merupakan sebuah kegiatan untuk mengalokasikan sumber daya yang dimiliki untuk mengerjakan sekumpulan tugas-tugas atau *job-job* yang terfokus pada sumber daya kritis. Sumber daya berkendala kapasitas pada penjadwalan biasanya berupa *bottleneck*

Teori *constraints* relatif baru dalam perkembangannya yang berkaitan dengan pengambilan keputusan yang mengandung batasan-batasan. *Theory Of Constraints* (TOC) pertama kali diperkenalkan oleh seorang ilmuwan berkebangsaan Israel, Dr. Eliyahu M. Goldratt pada akhir tahun 1980. Pada awalnya dikenal dengan *Optimized Production Timeable* dan selanjutnya dikenal dengan *Optimized Production Technology* (OPT).

Optimized Production Technology (OPT) atau disebut juga dengan *Synchronous Manufacturing* (Umble dan Srikanth, 1996) adalah sebuah filosofi manajemen yang dibangun berlandaskan kepada sejumlah asumsi yang terbatas dan dirancang untuk memberikan sebuah proses peningkatan yang berkesinambungan (*continuous improvement*). Bentuk asumsi yang mendasar OPT adalah sebuah sistem output yang ditentukan oleh pembatas (*constraints*).

Bentuk asumsi yang mendasar ini dikaitkan dengan *throughput*, *inventory*, dan *operating expense*. Definisi ini dirancang untuk mendukung tujuan utama (*goal*) sebuah perusahaan/industri. Tujuan utama dari setiap perusahaan adalah mencetak uang sebanyak-banyaknya dan meningkatkan kinerja baik saat ini maupun saat yang akan datang (Goldratt dan Cox, 1984). *Throughput*, *inventory* dan *operating expense*

adalah ukuran yang digunakan untuk mengukur kinerja (*performance*) dari tujuan utama perusahaan.

Troughput didefinisikan sebagai kecepatan dari perputaran sejumlah uang oleh sebuah perusahaan melalui penjualan pada periode tertentu. Secara khusus *troughput* didefinisikan sebagai barang yang terjual. Produk akhir tidak termasuk kategori *troughput* tetapi *inventori*.

Inventory didefinisikan sebagai uang yang diinvestasikan dalam bentuk bahan baku yang sudah dibeli dan termasuk seluruh sumber yang digunakan untuk proses produksi seperti fasilitas dan alat-alat.

Operating expense didefinisikan sebagai sejumlah uang yang dikeluarkan oleh perusahaan untuk merubah *inventori* menjadi *troughput* pada periode tertentu (Umble dan Srikanth, 1990). *Operating expense* (pengeluaran biaya operasi) yaitu termasuk biaya produksi seperti pekerja langsung dan tidak langsung, biaya simpan bahan baku, biaya depresiasi dan biaya administrasi. Dalam hal ini tidak ada pemisahan antara biaya pekerja langsung dengan tidak langsung. Kenaikan *troughput* secara berkesinambungan akan menurunkan biaya *inventori* dan biaya operasi yang akan memberikan kontribusi terhadap tujuan memperoleh uang (Chase dan Aquilano, 1989).

Dalam sistem aturan permodalan dan peningkatan kinerja sistem dalam mencapai tujuannya, OPT mempunyai filosofi yang memfokuskan pada aturan pembatas atau disebut pula dengan *constraint*. Oleh karena itu, teori ini disebut dengan *Theory of Constrain*.

2.3.2 Bottleneck dan Non Bottleneck

Sumber *bottleneck* merupakan sumber dimana kapasitasnya sama atau lebih kecil daripada permintaan yang dibebankan pada sumber tersebut. Ketika permintaan

melebihi kapasitas pada sebuah *bottleneck* maka pihak manajemen harus mencari bagaimana mengatasi hal tersebut.

Sumber *non bottleneck* adalah suatu sumber dimana kapasitasnya lebih besar daripada permintaan yang dibebankan pada sumber tersebut (kelebihan kapasitas). Dalam mengoperasikan sebuah sumber *non bottleneck* kapasitasnya tidak akan menaikkan *throughput* pada suatu pabrik. Peningkatan *throughput* dengan pendekatan TOC dapat dilakukan dengan lima langkah utama dalam TOC.

2.3.3 Lima Langkah Utama Dalam TOC

Pembatas adalah elemen yang membatasi sistem dari pencapaian tujuan utama perusahaan. Jenis pembatas yang biasanya ditemui pada lingkungan industri manufaktur adalah pemasaran, bahan baku, kapasitas, logistik, pengurusan, perilaku. Prosedur dari peningkatan kinerja ini terdiri dari langkah-langkah sebagai berikut (Heizer dan Render, 2004) :

1. Identifikasi constraints sistem
Bagaimana dari sistem yang memiliki hubungan terlemah? Masalah fisik atau kebijakan?
2. Putuskan bagaimana menghilangkan *constraints* tersebut.
Tentukan bagaimana menghilangkan *constraints* yang telah ditemukan dengan mempertimbangkan perubahan dengan biaya terendah.
3. Subordinasi segala sesuatu yang lain.
Setelah *constraints* ditemukan lalu putuskan apa yang akan dilakukan terhadap *constraints* tersebut. Setelah itu harus dievaluasi apakah *constraints* tersebut masih menjadi *constraints* pada performansi sistem atau tidak. Jika tidak maka langsung menuju ke langkah 5 tetapi jika sistem masih memiliki *constraints* teruskan ke langkah 4.

4. Evaluasi *constraints*.

Jika langkah ini dilakukan maka langkah ke-2 dan ke-3 tidak berhasil menangani *constraints*. Maka harus ada perubahan besar dalam sistem seperti reorganisasi, perbaikan modal atau modal substansi sistem.

5. Waspada terhadap *constraints* lain yang mungkin timbul.

Berdasarkan aturan produksi kontrol didasarkan kendala, Goldratt membuat 10 aturan global mengenai optimasi. Aturan Goldratt adalah sebagai berikut (Sipper, 1998) :

1. Seimbangkan aliran produksi, bukan kapasitas produksi.
2. Penggunaan Stasiun kerja *non bottleneck* tidak ditentukan oleh potensi stasiun kerja tersebut tetapi oleh stasiun kerja bottleneck atau sumber kritis lainnya.
3. Penggunaan sumber daya *non bottleneck* mengikuti sumber daya *bottleneck*.
4. Mesin *non bottleneck* tidak perlu melakukan aktivitas yang tidak menambah kegunaan yang akan menyebabkan inventori
5. Satu jam kehilangan pada *bottleneck* merupakan satu jam kehilangan sistem keseluruhan.
6. Satu jam penghematan pada *non bottleneck* merupakan suatu penghematan semu.
7. *Bottleneck* mempengaruhi *throughput* dan inventori.
8. *Batch transfer* tidak selalu sama jumlahnya dengan *batch process*.
9. *Batch process* sebaiknya tidak tetap (variabel).
10. Penjadwalan (kapasitas dan prioritas) dilakukan dengan memperhatikan semua kendala (*constraint*) yang ada secara simultan.

Dalam TOC terdapat sebuah metode yang dapat membantu pengimplementasian dari lima langkah tersebut yaitu metode *Drum Buffer Rope*.

2.3.4 Drum Buffer Rope (DBR)

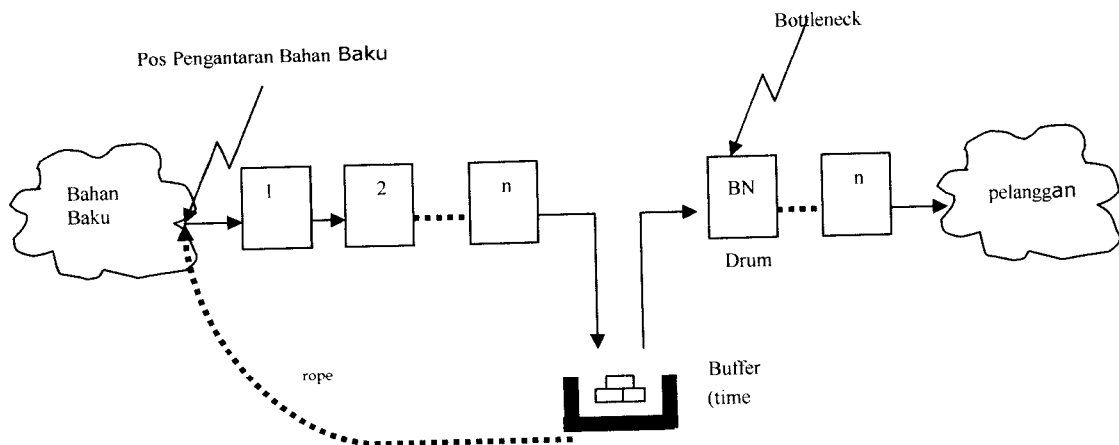
Drum Buffer Rope adalah suatu teknik kontrol produksi untuk menunjukkan pengeksploasi, subordinasi langkah-langkah TOC. Jika suatu sistem memiliki *bottleneck* maka *bottleneck* tersebut akan menjadi titik kontrol. *Bottleneck* menjadi titik kontrol karena untuk menjamin pengoperasian sistem cukup untuk memproduksi.

Drum merupakan stasiun dengan kapasitas terendah yang menjadi *constrains* dalam sistem produksi. Stasiun ini akan menentukan aliran produksi keseluruhan sistem maka stasiun ini perlu mendapat perlindungan dari gangguan yang selalu terjadi dalam sistem produksi. Perlindungan ini dilakukan dengan memberikan *buffer* yang ditempatkan sebelum *bottleneck*. Tujuan dari *buffer* ini adalah untuk menjaga *bottleneck* dari fluktuasi stasiun-stasiun kerja sebelumnya. Selain itu juga berfungsi agar aliran produksi tidak terganggu oleh gangguan yang terjadi dalam sistem produksi. Terdapat dua macam *buffer* (penyangga), (Umble dan Srikanth, 1996) yaitu:

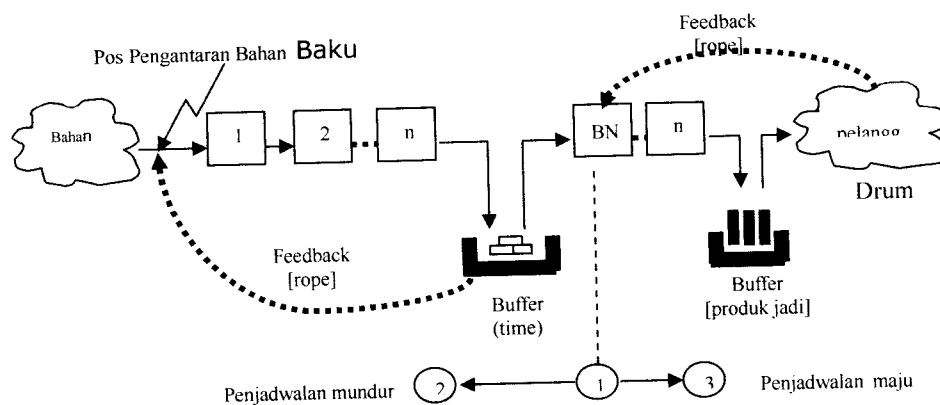
1. *Time Buffer*, yaitu waktu yang dijadikan penyangga dengan tujuan untuk melindungi laju produksi (*throughput*) sistem dari gangguan yang selalu terjadi dalam sistem produksi.
- 2 *Stock Buffer*, yaitu produk akhir dijadikan penyangga dengan tujuan untuk memperbaiki kemampuan menanggapi sistem produksi terhadap permintaan.

Berdasarkan kedua definisi *buffer* diatas, maka tipe *buffer* yang paling sesuai untuk menjadi *buffer* di stasiun *constrains* adalah *time buffers*, karena tujuan dari *time buffers* adalah melindungi *throughput* dari gangguan internal yang muncul. *Inventori* yang terjadi di stasiun konstrain tampak seperti *stock bufferr*s untuk melindungi stasiun *constrains*, sesungguhnya inventori tersebut muncul karena setiap order diberikan *time buffers* di stasiun *constrains* sehingga order tiba sebelum jadwalnya. *Rope* adalah penghubung laju produksi antara stasiun *constrains* dan stasiun *non-constrains*. Adanya *rope* mengurangi jumlah persediaan di setiap stasiun kerja dan menjaganya pada tingkat tertentu. Hal tersebut dikarenakan stasiun kerja *non-*

constrains berproduksi sesuai dengan kebutuhan stasiun *constrains*. Dengan kata lain rope adalah alat komunikasi antara *bottleneck* dan bagian pengantaran bahan baku. Bahan baku yang diantar ini hanya sejumlah persediaan pada *buffer* sehingga pada *buffer* tersedia persediaan yang dibutuhkan oleh *bottleneck*. Teknik DBR ini dapat diilustrasikan seperti pada gambar 2.2 dan 2.3



Gambar 2.2. Teknik DBR



Prosedur Penjadwalan :

- 1) Penjadwalan CCR
- 2) Penjadwalan mundur untuk pengantaran bahan baku
- 3) Penjadwalan maju untuk produk jadi

Gambar 2.3. DBR kegiatan mundur

2.4 Algoritma Zijm

Algoritma zijm ditemukan oleh seorang profesor dari sebuah universitas di belanda yang bernama WHK Zijm. Algoritma zijm memberikan logika pendeteksian mesin *bottleneck* yang terjadi pada suatu sistem, berdasarkan parameter – paraeter yang telah ditentukan, setelah mendeteksi terjadi *bottleneck* zijm juga memberikan salah satu parameter yang dibutuhkan untuk meningkatkan kinerja mesin *bottleneck* yaitu menghasilkan *time buffers*. Tujuan dari *buffer* ini adalah untuk menjaga *bottleneck* dari fluktuasi stasiun-stasiun kerja sebelumnya. Selain itu juga berfungsi agar aliran produksi tidak terganggu oleh gangguan yang terjadi dalam sistem produksi. (Umble dan Srikanth, 1996)

Sedangkan logika dasar dari algoritma zijm adalah mencari mesin *bottleneck* yaitu dengan cara mendeteksi mesin yang mempunyai beban kerja terbanyak, dan ekspektasi waktu menunggu setiap *job* di mesin paling besar. Selanjutnya zijm setelah diketahui mesin yang *bottleneck* zijm akan menghasilkan *time buffers* yang selanjutnya digunakan untuk menjaga *fluktuasi* setasiun kerja sebelumnya. Model matematis yang digunakan adalah sebagai berikut (Ridwan, 2003) :

1. Hitung laju kedatangan setiap *job*.

$$\lambda_{jk}^{(h)} = \frac{D^{(h)}}{Q^{(h)}} \times \delta_{jk}^{(h)} \quad \dots\dots\dots(2.1)$$

dengan :

$$\lambda_{jk}^{(h)} = \text{Laju kedatangan } job \text{ h operasi ke k di mesin j}$$

$$D^{(h)} = \text{Laju permintaan / } job \text{ h}$$

$$Q^{(h)} = \text{Ukuran lot produksi } job \text{ h} = 1$$

$$D^{(h)} = \frac{1}{MLT} = \frac{1}{(d^{(h)} - r^{(h)})m} \dots\dots\dots(2.2)$$

$\delta_{jk}^{(h)} \begin{cases} 1, \text{ jika } job \text{ h, proses k, dikerjakan pada mesin j} \\ 0, \text{ untuk yang lainnya} \end{cases}$

MLT = Manufacturing *Lead Time* (waktu pengerjaan)

m = Jumlah mesin

2. Hitung waktu proses setiap *job*.

$$P_{jk}^{(h)} = Z_{jk}^{(h)} + (Q_{jk}^{(h)} \times a_{jk}^{(h)}) \dots\dots\dots(2.3)$$

dengan :

$Z_{jk}^{(h)}$ = Waktu *set-up job* h , proses ke k di mesin j

$a_{jk}^{(h)}$ = waktu proses *job* h , proses ke k di mesin j

3. Hitung rata – rata beban kerja setiap mesin

$$P_j = \sum_{h,k} \lambda_{jk}^{(h)} \times P_{jk}^{(h)} \dots\dots\dots(2.4)$$

4. Hitung ekspektasi rata – rata waktu menunggu setiap *job* di mesin

$$E(j) = \frac{\sum_{h,k} \lambda_{jk}^{(h)} \times (P_{jk}^{(h)})^2}{2(1 - P_j)} \dots\dots\dots(2.5)$$

5. Hitung ekspektasi rata – rata lead time produksi operasi k *job* h di mesin j

$$E(T_{jk}^{(h)}) = E(j) + P_{jk}^{(h)} \dots\dots\dots(2.6)$$

6. Hitung ekspektasi rata – rata lead time *job* h

$$E(T^{(h)}) = \sum_{j,k} \delta_{jk}^{(h)} \times E(T_{jk}^{(h)}) \dots\dots\dots(2.7)$$

$$ETC = R + \sum_{i=1}^m (ti - wi) \dots\dots\dots(2.8)$$

$$LTC = d - \sum_{i=1}^m (ti + wi) + tb \dots\dots\dots(2.9)$$

Dengan :

m = jumlah mesin sebelum mesin *bottleneck*

r = *release time*

d = *duedate*

t_i = waktu proses

w_i = waktu tunggu

M = jumlah mesin dari *bottleneck* sampai akhir

t_b = waktu pengerjaan di mesin *bottleneck* sebelum ditambah waktu tunggu

ETC = Early Time Complete

LTC = Last Time Complete

2.5 Aggregate Time Buffer

Konsep *Aggregate Time Buffer* (ATB) murni berasal dari TOC (Goldrat,1999). Menurut Tu dan Li (1998) Time Buffer adalah jumlah dari waktu proses ditambah waktu setup ditambah sejumlah waktu untuk memastikan bahwa produk tetap siap untuk diproduksi ketika berada di mesin bottleneck. Menurut Leach (1999) time buffer harus diaggregatkan untuk menghindari ketidakpastian dari aktivitas produksi yang berkaitan dengan performansi due date.

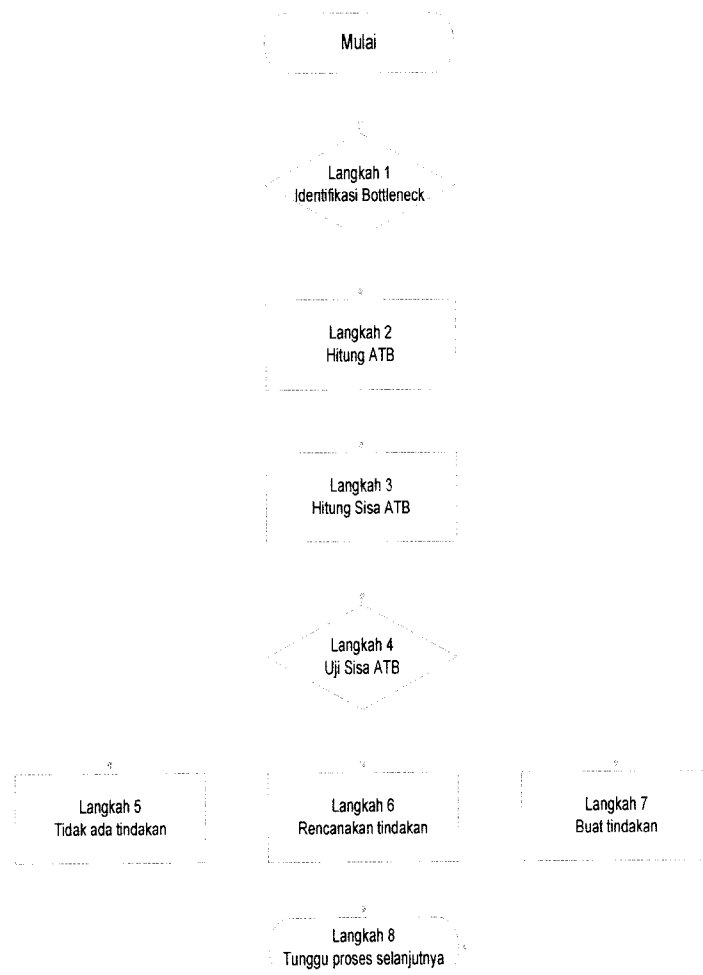
2.5.1 Aggregate Time Buffer Control

Dalam pemakaian konsep ATB hanya stasiun kerja yang bottleneck yang perlu untuk diperhatikan. Dalam pemakaian konsep ATB sama seperti pengontrolan inventory. Time buffer akan digunakan dalam waktu yang cepat, tetapi dilain waktu tidak akan digunakan sama sekali. Oleh karena itu perlu direncanakan kapan time buffer diperlukan dan berapa banyak time buffer diperlukan. Aturan pengontrolan Time Buffer adalah sebagai berikut (Chang, 2002)

1. Jika baffle yang terjadi dilantai produksi hanya $1/3$ dari total jumlah buffer yang ada, maka bottleneck tidak akan diberi tindakan apapun.

2. Jika baffle yang terjadi dilantai produksi $2/3$ buffer dari total jumlah buffer yang ada, maka akan dibuat perencanaan untuk mengatasi bottleneck.
3. Jika baffle yang terjadi dilantai produksi $3/3$ buffer dari total jumlah buffer yang ada, maka akan langsung diberi tindakan untuk mengatasi bottleneck.

Selama proses dilantai produksi total ATB harus tetap dimonitor. Jika buffer negatif atau lebih kecil dari $1/3$ total ATB maka bottleneck tidak akan diberi tindakan apapun. Oleh karena itu ATB perlu dirubah dikontrol disesuaikan dengan kondisi lantai produksi agat ATB tetap update. Algoritma untuk pengontrolan penganalisaan ATB dideskripsikan dalam gambar di bawah ini.



Gambar 2.4 Model Aggregate Time Buffer

Langkah 1 : Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2. Jika tidak ada bottleneck masuk ke langkah 8.

Langkah 2 : Mengitung total ATB dari waktu aktivitas di rantai produksi. Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \dots\dots\dots(2.10)$$

Dimana :

i = Job ke i

j = Stasiun kerja/mesin pengerjaan job

k = Stasiun bottleneck

e = Stasiun terakhir untuk pengerjaan job

P_{ij} = Waktu proses job ke i pada stasiun j

AB_i = Aggregate Time Buffer job ke i

AB_{ik} = Aggregate Time Buffer job ke i pada mesin bottleneck ke k

Langkah 3.1 : Menghitung due date. Formula yang digunakan adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} x d_{ik} \right) \dots\dots\dots(2.11)$$

Dimana :

d_i = Due date job

a_i = Waktu kedatangan job pada mesin pertama

d_{ik} = Waktu proses di mesin bottleneck

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan berdasarkan due date. Formula yang digunakan adalah sebagai berikut:

$$et_{ik} = a_i + (d_i - a_i) \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \dots\dots\dots(2.12)$$

Dimana:

et_{ik} = Waktu penyelesaian yang diharapkan untuk job i pada stasiun bottleneck k .

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik}) \dots\dots\dots(2.13)$$

Dimana :

t_{ik} = Waktu mulai job ke i pada stasiun bottleneck k

SB_{ik} = Aggregate time buffer ketika job ke i sampai ke stasiun bottleneck k

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

a. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$,(2.14)

maka langsung masuk ke langkah 5

b. Jika $\frac{1}{3} AB_{ik} \leq AB_{ik} - SB_{ik} \leq \frac{2}{3} AB_{ik}$,(2.15)

maka langsung masuk ke langkah 6

c. Jika $AB_{ik} - SB_{ik} \geq \frac{2}{3} AB_{ik}$,(2.16)

maka langsung ke langkah 7

Langkah 5 : Mengevaluasi penjadwalan berdasarkan performansi duedate untuk prioritas dispatcing.

Langkah 6 : Memonitor penjadwalan pada rantai produksi secara intensif.

$AB_{ik} - SB_{ik} \geq \frac{2}{3} AB_{ik}$, kemudian langsung ke langkah 7

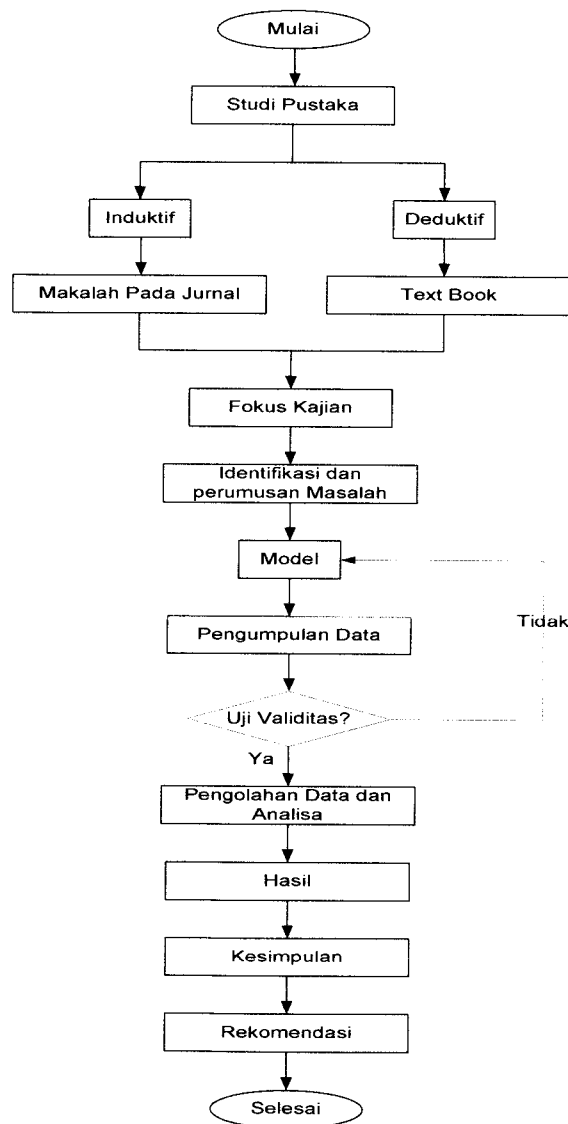
Langkah 7 : Berdasarkan penggunaan time buffer, maka penjadwalan akan diselesaikan tepat waktu.

Langkah 8 : Menunggu untuk proses selanjutnya

BAB III

METODOLOGI PENELITIAN

Langkah- langkah penelitian perlu disusun secara baik untuk mempermudah penyusunan laporan penelitian. Adapun langkah – langkah penelitian dapat dipresentasikan seperti gambar 3.1



Gambar 3.1 Diagram Alir Kerangka Penelitian



3.1 Studi Pustaka

Ada dua macam studi pustaka yang dilakukan yaitu studi pustaka induktif dan deduktif. Kajian induktif adalah kajian pustaka yang bermakna untuk menjaga keaslian penelitian, dan bermanfaat bagi peneliti untuk menjadi kekinian topik penelitian. Kajian ini diperoleh dari jurnal, prosiding, seminar, majalah dan lain sebagainya. Pada kajian induktif, dapat diketahui perkembangan penelitian, batas-batas dan kekurangan penelitian terdahulu. Disamping itu dapat diketahui perkembangan metode – metode mutakhir yang pernah dilakukan peneliti lain. Kajian deduktif membangun konseptual yang mana fenomena – fenomena atau parameter – parameter yang relevan disistematika, diklasifikasikan dan dihubungkan – hubungkan sehingga bersifat umum. Kajian deduktif merupakan landasan teori yang dipakai sebagai acuan untuk memecahkan masalah penelitian.

3.2 Penentuan Obyek Penelitian

Penelitian ini dilakukan untuk melanjutkan penelitian sebelumnya (Aditya Gatot, 2006) di Laboratorium Sistem Produksi Jurusan Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia.

3.3 Analisis Model

Model yang akan dianalisa adalah model yang diperoleh berdasarkan kajian literatur induktif, yaitu model matematis yang pernah diajukan oleh *Sheng-Hung Chang*

(2002). Langkah analisis model adalah melakukan identifikasi parameter diperlukan untuk penelitian.

Model matematis yang digunakan adalah sebagai berikut :

Model yang akan dianalisa yaitu suatu model matematis yang berawal dari suatu penjadwalan produksi yang memperhatikan beberapa kriteria yaitu *Time Buffers*, *makespan* dan *duedate* tiap *part*.

Model matematis dapat dilihat pada persamaan 1 :

$$S = \max\{c_i; j_i\} - \left(a_i + \left(\sum_{j=1}^{j=e} E(j) + p_{ij} \right) + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} x d_{ik} \right) \right) \dots \text{Persamaan 1}$$

Dimana :

S = Slack

C_i = Completion Time

i = Job ke i

j_i = Stasiun kerja/mesin pengerjaan job

k = Mesin bottleneck

e = Stasiun terakhir untuk pengerjaan job

P_{ij} = Waktu proses job ke i pada stasiun j

a_i = Waktu mulai di stasiun ke i

$E(j)$ = Rata-rata waktu menunggu

d_{ik} = Waktu proses job ke i di mesin bottleneck

3.4 Identifikasi dan Perumusan Masalah

Proses ini dilakukan untuk merumuskan masalah yang maknanya merumuskan butir – butir yang lebih atau sudah jelas dan sistematis atas permasalahan yang diungkapkan

di latar belakang masalah. Identifikasi ini diperlukan supaya rumusan masalah , latar belakang masalah dan judul penelitian saling berkaitan.

3.5 Pengumpulan Data

Pengumpulan data dilakukan dalam dua cara .

1. Wawancara bebas tidak didokumentasikan secara terstruktur
2. Studi lapangan, yang dilakukan pengamatan secara langsung dan pencatatan data produksi seperti data mesin, data waktu , pekerja, data biaya – biaya yang diperlukan.

Kedua pengambilan data tersebut diatas termasuk dalam kriteria pengambilan data primer dan sekunder.

3.6 Pengolahan Data dan Analisis Hasil

1. Identifikasi Bottleneck

Data-data yang sudah terkumpul kemudian diolah. Pertama identifikasi bottleneck dengan cara menjumlahkan waktu proses setiap job pada setiap mesin. Jumlah waktu permesinan yang paling lama akan menjadi stasiun bottleneck.

2. Menghitung buffer time dengan menggunakan Algoritma Zijm. Langkah-langkahnya sebagai berikut :

1. Hitung laju kedatangan setiap *job*.

$$\lambda_{jk}^{(h)} = \frac{D^{(h)}}{Q^{(h)}} \times \delta_{jk}^{(h)} \quad \dots\dots\dots(2.1)$$

dengan :

$\lambda_{jk}^{(h)}$ = Laju kedatangan *job* h operasi ke k di mesin j

$D^{(h)}$ = Laju permintaan / job h

$Q^{(h)}$ = Ukuran lot produksi job h = 1

$\delta_{jk}^{(h)} \begin{cases} 1, & \text{jika job h, proses k, dikerjakan pada mesin j} \\ 0, & \text{untuk yang lainnya} \end{cases}$

$$D^{(h)} = \frac{1}{MLT} = \frac{1}{(d^{(h)} - r^{(h)})m} \dots\dots\dots(2.2)$$

MLT = Manufacturing *Lead Time* (waktu pengerjaan)

m = Jumlah mesin

2. Hitung waktu proses setiap job.

$$P_{jk}^{(h)} = Z_{jk}^{(h)} + (Q_{jk}^{(h)} \times a_{jk}^{(h)}) \dots\dots\dots(2.3)$$

dengan :

$Z_{jk}^{(h)}$ = Waktu *set-up* job h , proses ke k di mesin j

$a_{jk}^{(h)}$ = waktu proses job h , proses ke k di mesin j

3. Hitung rata – rata beban kerja setiap mesin

$$P_j = \sum_{h,k} \lambda_{jk}^{(h)} \times P_{jk}^{(h)} \dots\dots\dots(2.4)$$

4. Hitung ekspektasi rata – rata waktu menunggu setiap job di mesin

$$E(j) = \frac{\sum_{h,k} \lambda_{jk}^{(h)} \times (P_{jk}^{(h)})^2}{2(1 - P_j)} \dots\dots\dots(2.5)$$

5. Hitung ekspektasi rata – rata lead time produksi operasi k job h di mesin j

$$E(T_{jk}^{(h)}) = E(j) + P_{jk}^{(h)} \dots\dots\dots(2.6)$$

6. Hitung ekspektasi rata – rata lead time job h

$$E(T^{(h)}) = \sum_{j,k} \delta_{jk}^{(h)} \times E(T_{jk}^{(h)}) \dots\dots\dots(2.7)$$

$$ETC = R + \sum_{i=1}^m (ti - wi) \dots\dots\dots(2.8)$$

$$LTC = d - \sum_{i=1}^m (ti + wi) + tb \dots\dots\dots(2.9)$$

Dengan :

m = jumlah mesin sebelum mesin *bottleneck*

r = *release time*

d = *duedate*

t_i = waktu proses

w_i = waktu tunggu

M = jumlah mesin dari *bottleneck* sampai akhir

t_b = waktu pengerjaan di mesin *bottleneck* sebelum ditambah waktu tunggu

ETC = Early Time Complete

LTC = Last Time Complete

3. Menghitung Aggregate Time Buffer. Model yang digunakan adalah sebagai berikut.

Langkah 1 : Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2. Jika tidak ada bottleneck masuk ke langkah 8.

Langkah 2 : Mengitung total ATB dari waktu aktivitas di rantai produksi. Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \dots\dots\dots(2.10)$$

Dimana :

i = Job ke i

j = Stasiun kerja/mesin pengerjaan job

k = Stasiun bottleneck

e = Stasiun terakhir untuk pengerjaan job

P_{ij} = Waktu proses job ke i pada stasiun j

AB_i = Aggregate Time Buffer job ke i

AB_{ik} = Aggregate Time Buffer job ke i pada mesin bottleneck ke k

Langkah 3.1 : Menghitung duedate. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d'_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \right) x d_{ik} \dots\dots\dots(2.11)$$

Dimana :

d'_i = Duedate job

a_i = Waktu kedatangan job pada mesin pertama

d_{ik} = Waktu proses di mesin bottleneck

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan berdasarkan duedate. Formula yang digunakan adalah sebagai berikut:

$$et_{ik} = a_i + (d'_i - a_i) x \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \dots\dots\dots(2.12)$$

Dimana:

et_{ik} = Waktu penyelesaian yang diharapkan untuk job i pada stasiun bottleneck k .

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik}) \dots\dots\dots(2.13)$$

Dimana :

t_{ik} = Waktu mulai job ke i pada stasiun bottleneck k

SB_{ik} = Aggregate time buffer ketika job ke i sampai ke stasiun bottleneck k

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

a. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$,(2.14)

maka langsung masuk ke langkah 5

b. Jika $\frac{1}{3}AB_{ik} \leq AB_{ik} - SB_{ik} \leq \frac{2}{3}AB_{ik}, \dots \dots \dots (2.15)$

maka langsung masuk ke langkah 6

c. Jika $AB_{ik} - SB_{ik} \geq \frac{2}{3}AB_{ik}, \dots \dots \dots (2.16)$

maka langsung ke langkah 7

Langkah 5 : Mengevaluasi penjadwalan berdasarkan performansi duedate untuk prioritas dispatcing.

Langkah 6 : Memonitor penjadwalan pada lantai produksi secara intensif.

$AB_{ik} - SB_{ik} \geq \frac{2}{3}AB_{ik}$, kemudian langsung ke langkah 7

Langkah 7 : Berdasarkan penggunaan time buffer, maka penjadwalan akan diselesaikan tepat waktu.

Langkah 8 : Menunggu untuk proses selanjutnya

3.7 Hasil Penelitian

Hasil penelitian yang diperoleh dari pengolahan dan analisis data kemudian didiskusikan untuk mengetahui kemungkinan kekurangan atau kelebihan dari hasil penelitian sehingga dapat dibuat suatu rekomendasi terhadap hasil penelitian ini.

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

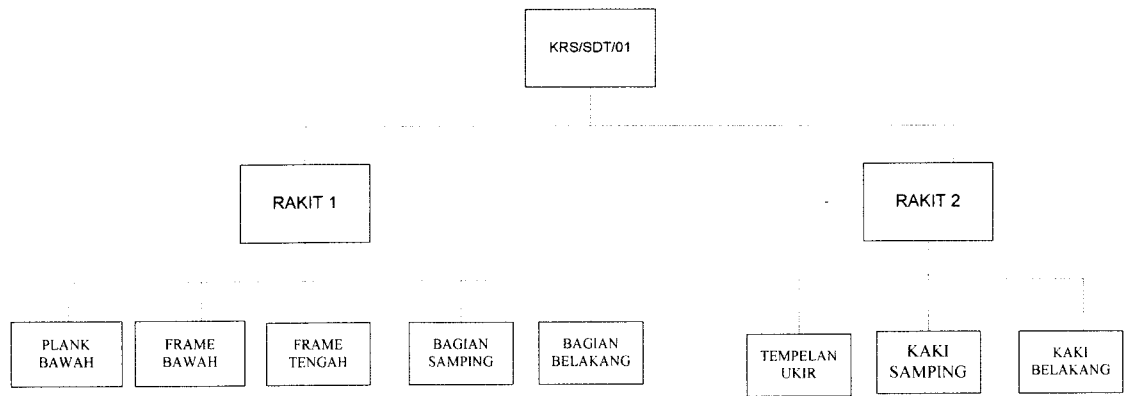
4.1 Pengumpulan Data

Data diperoleh dari literatur peneliti sebelumnya yang penelitiannya dilakukan di Cv. Anugrah Furniture pada departemen produksi. Obyek yang diteliti adalah kursi sudut ukir. Penelitian berfokus pada kursi sudut ukir dikarenakan kursi sudut ukir mengalami penjadwalan yang tidak baik dan permintaan akan produk ini terbesar dibandingkan dengan produk yang lain. Adapun data yang diperlukan :

- a. Struktur produk / *Bill Of Material* produk
- b. *Duedate* setiap part
- c. Peta proses operasi (terlampir)
- d. Mesin yang digunakan dalam proses produksi
- e. Waktu *setup* mesin
- f. Waktu proses setiap *job*

4.1.1 Struktur Produk

Struktur produk dapat dilihat dibawah ini :



Gambar 4.1 Struktur Produk

4.1.2 *Duedate* Setiap Part

Data *duedate* setiap part pada saat pengambilan data adalah sebagai berikut :Tabel 4.1 Data *Duedate* Setiap Part

<i>Job</i>	<i>Duedate</i>
J1	39.07
J2	40.37
J3	33.33
J4	25.68
J5	20.29
J6	24.75
J7	33.14
J8	28.75

4.1.3 Peta Proses Operasi

Peta proses operasi dapat dilihat dilampiran

4.1.4 Mesin Yang di Gunakan Dalam Proses Produksi

Mesin yang digunakan untuk produksi pada departemen produksi perusahaan anugrah dapat dilihat seperti dibawah ini :

- M1 : Mesin Bandsaw
- M2 : Mesin Ukir
- M3 :Mesin gergaji
- M4 : Mesin Spindel
- M5 : Mesin planer
- M6 : Mesin Bor
- M7 : Mesin Bubut
- M8 : Mesin Router
- M9 : Mesin Sending

4.1.5 Waktu *Setup* Mesin

Waktu *setup* mesin dapat dilihat pada tabel dibawah ini :

Tabel 4.2 Data waktu *set-up* mesin

Mesin	Waktu Setup (menit)
1	2
2	2
3	1.5
4	2.5
5	1
6	2
7	2.5
8	1.5
9	1.5

4.1.6 Waktu Proses Setiap Job di Setiap Mesin

Waktu proses setiap *job* dapat dilihat seperti tabel dibawah ini :

Tabel 4.3 Data waktu proses setiap *job* pada setiap mesin tanpa waktu *set-up*.

No	Jenis Part	No order	Waktu proses tanpa waktu <i>set-up</i> (menit)									Total
			M1	M2	M3	M4	M5	M6	M7	M8	M9	
1	Plank Bawah	J1	1	1.5	0.5	1.45	1.7	1	1.56	0.5	0.17	9.38
2	Frame Bawah	J2	1.2	1	0.5	1.45	1.2	1.2	1.58	0.5	0.33	8.96
3	Frame Tengah	J3	1.2	2	0.75	1..5	1.58	0.75	1.2	0.75	0.33	8.56
4	Bagian Samping	J4	0.5	0.45	1.08	1.08	1	0.75	0.8	1.5	0.28	7.44
5	Bagian Belakang	J5	0.75	0.45	1.08	1.08	1	0.1	0.8	1.75	0.28	7.29
6	Tempelan Ukir	J6	1	1	1.5	1.02	0.8	0.7	1	0.5	0.13	7.65
7	Kaki Samping	J7	0.25	1.23	1.85	1.75	0.8	0.7	0.75	0.75	0.25	8.33
8	Kaki Belakang	J8	0.25	1.23	1.7	1.75	0.8	0.7	1.25	0.75	0.25	8.68

4.2 Pengolahan Data

Untuk mencapai tujuan penelitian ini, dari data yang telah dikumpulkan selanjutnya dilakukan pengolahan data. Ketepatan pengolahan data tergantung ketepatan perhitungannya. Jika hal tersebut di penuhi diharapkan keakuratan hasil yang diperoleh akan sesuai dengan tujuan yang diharapkan.

4.2.1 Mengidentifikasi Mesin *Bottleneck* dan Penentuan *Time Buffer* dengan *Algoritma Zijm*

Adapun langkah-langkah pengolahan data yang dilakukan adalah:

Langkah 1 : Menghitung laju kedatangan *job*.

laju kedatangan *job* dihitung dengan menggunakan persamaan (2.1) dan (2.2) Data saat siap dan *due date* diambil dari tabel 4.1. Pada kasus ini tidak dilakukan *batch* pada *lot* produksi maka $Q^{(h)} = 1$. Contoh perhitungan laju kedatangan *job* adalah sebagai berikut:

Job 1

$$D^1 = \frac{1}{(39.07 - 0).9} = 0.002843898 \text{ menit}$$

$$\lambda_{11}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit.}$$

$$\lambda_{22}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{33}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{44}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{55}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{66}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{77}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{88}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

$$\lambda_{99}^1 = \frac{0.002843898}{1} \times 1 = 0.002843898 \text{ menit}$$

Hasil lengkap perhitungan laju kedatangan *job* dapat dilihat pada tabel 4.5:

Langkah 2 : Menghitung waktu proses sebuah *job*

Penentuan waktu proses sebuah *job* menggunakan persamaan (2.3). Data waktu proses diambil dari tabel 4.3 dan data waktu *set-up* diambil dari tabel 4.2 Contoh perhitungannya adalah sebagai berikut:

Job 1

$$P_{11}^1 = 2 + (1 \times 1) = 3 \text{ Menit}$$

$$P_{22}^1 = 2 + (1 \times 1.5) = 3.5 \text{ Menit}$$

$$P_{33}^1 = 1.5 + (1 \times 0.5) = 2 \text{ Menit}$$

$$P_{44}^1 = 2.5 + (1 \times 1.45) = 3.95 \text{ Menit}$$

$$P_{55}^1 = 1 + (1 \times 1.7) = 2.7 \text{ Menit}$$

$$P_{66}^1 = 2 + (1 \times 1) = 3 \text{ Menit}$$

$$P_{77}^1 = 2.5 + (1 \times 1.56) = 4.06 \text{ Menit}$$

$$P_{88}^1 = 1.5 + (1 \times 0.5) = 2 \text{ Menit}$$

$$P_{99}^1 = 1.5 + (1 \times 0.17) = 1.67 \text{ Menit}$$

Hasil lengkap perhitungan waktu proses sebuah *job* dapat dilihat pada tabel 4.6

Tabel 4.5 Waktu proses sebuah *job* sudah termasuk waktu *set-up*.

Job	Waktu Proses (menit)								
	M1	M2	M3	M4	M5	M6	M7	M8	M9
J1	3	3.5	2	3.95	2.7	3	4.06	2	1.67
J2	3.2	3	2	3.95	2.2	3.2	4.08	2	1.83
J3	3.2	4	2.25	4	2.58	2.75	3.7	2.25	1.83
J4	2.5	2.45	2.58	3.58	2	2.75	3.3	3	1.78
J5	2.75	2.45	2.58	3.58	2	2.1	3.3	3.25	1.78
J6	3	3	3	3.52	1.8	2.7	3.5	2	1.63
J7	2.25	3.23	3.35	4.25	1.8	2.7	3.25	2.25	1.75
J8	2.25	3.23	3.2	4.25	1.8	2.7	3.75	2.25	1.75

Langkah 3 : Menghitung rata-rata beban kerja di setiap mesin.

Perhitungan rata-rata beban kerja ini menggunakan persamaan (2.4)

$$\begin{aligned}
 P_l &= (0.002843898 \times 3) + (0.002752319 \times 3.2) + (0.003333667 \times 3.2) + \\
 &\quad (0.004326757 \times 2.5) + (0.005476151 \times 2.75) + (0.014828 \times 3) + \\
 &\quad (0.003352779 \times 2.25) + (0.003864734 \times 2.25) \\
 &= 0.083590576 \text{ Menit}
 \end{aligned}$$

Hasil lengkap perhitungan rata-rata beban kerja di setiap mesin dapat dilihat pada table 4.7 :

Tabel 4.6 Rata-rata beban kerja di setiap mesin

Mesin	Rata-rata beban kerja(Menit)
M1	0.083590576
M2	0.092342975
M3	0.081051662
M4	0.117011038
M5	0.063012636
M6	0.081513697
M7	0.108561821
M8	0.074689028
M9	0.05328411

Langkah 4 : Menghitung ekspektasi waktu menunggu setiap *job* di setiap mesin

Perhitungan ini menggunakan persamaan (2.5). Contoh perhitungannya sebagai berikut:

$$E(1) = \frac{(0.0255950 \ 86) + (0.02818374 \ 5) + (0.03413674 \ 7) + (0.02704222 \ 9) + (0.04141339 \ 5) + (0.04040404 \) + (0.01697344 \ 6) + (0.01956521 \ 7)}{2(1 - 0.027800955 \)}$$

$$= 0.106905531 \text{ Menit}$$

Hasil perhitungan ekspektasi waktu menunggu setiap mesin dapat dilihat pada tabel 4.8 :

Tabel 4.7 Ekspektasi waktu menunggu setiap *job* di setiap mesin.

Mesin	Ekspektasi waktu menunggu (menit)
M1	0.106905531
M2	0.130472469
M3	0.102057998
M4	0.199680303
M5	0.062490167
M6	0.101586219
M7	0.173759753
M8	0.088154483
M9	0.044213335

Langkah 5 : Menghitung ekspektasi rata-rata leadtime produksi

Perhitungan dalam langkah 5 ini menggunakan persamaan (2.6). Contoh perhitungannya adalah sebagai berikut:

$$E(T_{11}^{(1)}) = 0.106905531 + 3 = 3.106905531 \text{ Menit}$$

$$E(T_{22}^{(1)}) = 0.130472469 + 3.5 = 3.630472469 \text{ Menit}$$

$$E(T_{33}^{(1)}) = 0.102057998 + 2 = 2.102057998 \text{ Menit}$$

$$E(T_{44}^{(1)}) = 0.199680303 + 3.95 = 4.149680303 \text{ Menit}$$

$$E(T_{55}^{(1)}) = 0.062490167 + 2.7 = 2.762490167 \text{ Menit}$$

$$E(T_{66}^{(1)}) = 0.101586219 + 3 = 3.101586219 \text{ Menit}$$

$$E(T_{77}^{(1)}) = 0.173759753 + 4.06 = 4.233759753 \text{ Menit}$$

$$E(T_{88}^{(1)}) 0.088154483+2 = 2.088154483 \text{ Menit}$$

$$E(T_{99}^{(1)}) 0.044213335+1.67 = 1.714213335 \text{ Menit}$$

Hasil perhitungan secara lengkap dapat dilihat pada tabel 4.9

Tabel 4.8 Ekspektasi rata-rata *leadtime* produksi (menit)

Job	M1	M2	M3	M4	M5	M6	M7	M8	M9
J1	3.106905531	3.630472469	2.102057998	4.149680303	2.762490167	3.101586219	4.233759753	2.088154483	1.714213335
J2	3.306905531	3.130472469	2.102057998	4.149680303	2.262490167	3.301586219	4.253759753	2.088154483	1.874213335
J3	3.306905531	4.130472469	2.352057998	4.199680303	2.642490167	2.851586219	3.873759753	2.338154483	1.874213335
J4	2.606905531	2.580472469	2.682057998	3.779680303	2.062490167	2.851586219	3.473759753	3.088154483	1.824213335
J5	2.856905531	2.580472469	2.682057998	3.779680303	2.062490167	2.201586219	3.473759753	3.338154483	1.824213335
J6	3.106905531	3.130472469	3.102057998	3.719680303	1.862490167	2.801586219	3.673759753	2.088154483	1.674213335
J7	2.356905531	3.360472469	3.452057998	4.449680303	1.862490167	2.801586219	3.423759753	2.338154483	1.794213335
J8	2.356905531	3.360472469	3.302057998	4.449680303	1.862490167	2.801586219	3.923759753	2.338154483	1.794213335

Langkah 6 : Menghitung ekspektasi rata-rata leadtime setiap *job*.

Perhitungan ini menggunakan persamaan (2.7). Contoh perhitungannya sebagai berikut:

Job 1

$$E(T^{(1)}) = 3.106905531 + 3.630472469 + 2.102057998 + 4.149680303 + 2.762490167 + \\ 3.101586219 + 4.233759753 + 2.088154483 + 1.714213335 \\ = 26.88932026 \text{ Menit}$$

Hasil perhitungan ekspektasi rata-rata *leadtime* setiap *job* secara lengkap dapat dilihat pada tabel 4.10.

Tabel 4.9 Ekspektasi rata-rata *leadtime* setiap *job*.

JOB	Rata-rata leadtime (menit)
J1	26.88932026
J2	26.46932026
J3	27.56932026
J4	24.94932026
J5	24.79932026
J6	25.15932026
J7	25.83932026
J8	26.18932026

Setelah diketahui ekspektasi rata-rata *leadtime* setiap *job* selanjutnya menghitung *ETC* dan *LTC* setiap *order*. *ETC* dan *LTC* merupakan *time buffer* bagi *job* yang akan masuk mesin 4 (mesin *bottleneck*).

$$ETC_1 = 0 + 11.910884 = 11.910884$$

$$LTC_1 = 39.07 - (12.9891163) + 3.95 = 30.030884$$

Hasil lengkapnya dapat dilihat pada tabel 4.11 dan 4.12 :

Tabel 4.10 *ETC* dan *LTC* setiap *job* (menit)

Order	<i>ETC</i>	<i>LTC</i>
J1	11.910884	30.030884
J2	11.610884	31.630884
J3	12.910884	23.340884
J4	10.570884	17.610884
J5	10.820884	11.970884
J6	11.980884	15.210884
J7	12.540884	23.770884
J8	12.390884	19.530884

4.2.2 Penjadwalan Mesin *Bottleneck* dengan Memperhatikan *Time Buffer* Menggunakan Pendekatan Metode FCFS

Tabel 4.11 Hasil perhitungan dengan metode FCFS

No Order	M1			M2		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 1	0	3	3	3	3.5	6.5
Job 2	3	3.2	6.2	6.5	3	9.5
Job 3	6.2	3.2	9.4	9.5	4	13.5
Job 4	9.4	2.5	11.9	13.5	2.45	15.95
Job 5	11.9	2.75	14.65	15.95	2.45	18.4
Job 6	14.65	3	17.65	18.4	3	21.4
Job 7	17.65	2.25	19.9	21.4	3.23	24.63
Job 8	19.9	2.25	22.15	24.63	3.23	27.86

Tabel 4.11 Hasil perhitungan dengan metode FCFS

No Order	M3			M4		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 1	6.5	2	8.5	8.5	3.95	12.45
Job 2	9.5	2	11.5	12.45	3.95	16.4
Job 3	13.5	2.25	15.75	16.4	4	20.4

Tabel 4.11 Hasil perhitungan dengan metode FCFS (lanjutan)

Job 4	15.95	2.58	18.53	20.4	3.58	23.98
Job 5	18.53	2.58	21.11	23.98	3.58	27.56
Job 6	21.4	3	24.4	27.56	3.52	31.08
Job 7	24.63	3.35	27.98	31.08	4.25	35.33
Job 8	27.98	3.2	31.18	35.33	4.25	39.58

Tabel 4.11 Hasil perhitungan dengan metode FCFS

No Order	M5			M6		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 1	12.45	2.7	15.15	15.15	3	18.15
Job 2	16.4	2.2	18.6	18.6	3.2	21.8
Job 3	20.4	2.58	22.98	22.98	2.75	25.73
Job 4	23.98	2	25.98	25.98	2.75	28.73
Job 5	27.56	2	29.56	29.56	2.1	31.66
Job 6	31.08	1.8	32.88	32.88	2.7	35.58
Job 7	35.33	1.8	37.13	37.13	2.7	39.83
Job 8	39.58	1.8	41.38	41.38	2.7	44.08

Tabel 4.11 Hasil perhitungan dengan metode FCFS

No Order	M7			M8		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 1	18.15	4.06	22.21	22.21	2	24.21
Job 2	21.8	4.08	25.88	25.88	2	27.88
Job 3	25.73	3.7	29.43	29.43	2.25	31.68
Job 4	28.73	3.3	32.03	32.03	3	35.03
Job 5	31.66	3.3	34.96	35.03	3.25	38.28
Job 6	35.58	3.5	39.08	39.08	2	41.08
Job 7	39.83	3.25	43.08	43.08	2.25	45.33
Job 8	44.08	3.75	47.83	47.83	2.25	50.08

Tabel 4.11 Hasil perhitungan dengan metode FCFS

No Order	M9		
	Mulai	Durasi	Selesai
Job 1	24.21	1.67	25.88
Job 2	27.88	1.83	29.71
Job 3	31.68	1.83	33.51
Job 4	35.03	1.78	36.81
Job 5	38.28	1.78	40.06
Job 6	41.08	1.63	42.71
Job 7	45.33	1.75	47.08
Job 8	50.08	1.75	51.83

Tabel 4.12 Hasil perhitungan ETC dan LTC

Job	ETC	LTC
Job 1	11.9108837	30.0308837
Job 2	11.6108837	31.6308837
Job 3	12.9108837	23.3408837
Job 4	10.5708837	17.6108837
Job 5	10.8208837	11.9708837
Job 6	11.9808837	15.2108837
Job 7	12.5408837	23.7708837
Job 8	12.3908837	19.5308837

4.2.3 Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer

Tabel 4.13. Duedate setiap job dengan memperhatikan time buffer

Job	Duedate (d)
Job 1	39.07
Job 2	40.37
Job 3	33.33
Job 4	25.68
Job 5	20.29
Job 6	24.75
Job 7	33.14
Job 8	28.75

Tabel 4.14. Makespan setiap job dengan memperhatikan time buffer

Job	Makespan ($C_{i,9}$)
Job 1	25.88
Job 2	29.71
Job 3	33.51
Job 4	36.81
Job 5	40.06
Job 6	42.71
Job 7	47.08
Job 8	51.83

$$\begin{aligned}
 \text{Lateness} &= C_{i,9} - d \\
 &= 25.88 - 39.07 \\
 &= -13.19
 \end{aligned}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.15.

Job	Slack	Keterangan
Job 1	-13.19	Lateness
Job 2	-10.66	Lateness
Job 3	0.18	Tardiness
Job 4	11.13	Tardiness
Job 5	19.77	Tardiness
Job 6	17.96	Tardiness
Job 7	13.94	Tardiness
Job 8	23.08	Tardiness

4.2.4 Menentukan Aggregate Time Buffer

Model yang digunakan adalah sebagai berikut.

Langkah 1: Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2.

Langkah 2.1: Mengitung total time buffer dari waktu aktivitas di lantai produksi.

Contoh perhitungannya Total Time Buffer *Job* 1 pada mesin 4

$$AB_i = (C_{ik} - ETC) + (LTC - C_{ik})$$

$$AB_1 = (C_{14} - ETC) + (LTC - C_{14})$$

$$AB_1 = (12.45 - 11.9108837) + (30.0308837 - 12.45)$$

$$AB_1 = 18.12$$

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.16.

Job	Total Time Buffer		
	ETC	LTC	Total
Job 1	0.539116	17.58088	18.12
Job 2	4.789116	15.23088	20.02
Job 3	7.489116	2.940884	10.43
Job 4	13.40912	6.369116	19.77823
Job 5	16.73912	15.58912	32.32823

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.16. (lanjutan)

Job 6	19.09912	15.86912	34.96823
Job 7	22.78912	11.55912	34.34823
Job 8	27.18912	20.04912	47.23823
Total			217.2312

Langkah 2.2 : Mengitung aggregate time buffer dari waktu aktivitas di lantai produksi.

Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i x \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Tabel 4.17. Total waktu proses setelah bottleneck dan total waktu proses

Job	Mesin Bottleneck	Total Leadtime Setelah Bottleneck $\sum_{j=k}^{j=e} P_{ij}$	Total Leadtime $\sum_{j=1}^{j=e} P_{ij}$
Job 1	M4	18.04988426	26.88932026
Job 2	M4	17.92988426	26.46932026
Job 3	M4	17.77988426	27.56932026
Job 4	M4	17.07988426	24.94932026
Job 5	M4	16.67988426	24.79932026
Job 6	M4	15.81988426	25.15932026
Job 7	M4	16.66988426	25.83932026
Job 8	M4	17.16988426	26.18932026

Contoh perhitungannya Aggregate Time Buffer *Job* 1 pada mesin 4

$$AB_{14} = AB_1 x \frac{\sum_{j=4}^{j=9} P_{1j}}{\sum_{j=1}^{j=9} P_{1j}}$$

$$AB_{14} = 18.12 x \frac{18.04988426}{26.88932026} = 12.16333844$$

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.18.

Job	Aggregate Time Buffer
Job 1	12.16333844
Job 2	13.56122029
Job 3	6.726469535
Job 4	13.53984478
Job 5	21.74378864
Job 6	21.98761282
Job 7	22.15929275
Job 8	30.96968453
Total	142.8512518

Langkah 3.1 : Menghitung due date. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} x t_{ik} \right)$$

Tabel 4.19. Waktu kedatangan setiap job pada mesin pertama

Job	M1
	a_1
Job 1	0
Job 2	3
Job 3	6.2
Job 4	9.4
Job 5	11.9
Job 6	14.65
Job 7	17.65
Job 8	19.9

Rata-rata Lead Time setiap job dapat dilihat pada tabel 4.10

Contoh perhitungan due date :

$$d_i = 0 + 26.88932026 + \left(\frac{18.04988426}{26.88932026} x 3.95 \right)$$

Hasil perhitungan due date secara lengkap dapat dilihat pada tabel 4.20.

Job	Due date
Job 1	29.54082063
Job 2	32.1449856
Job 3	36.34898259
Job 4	36.80012794
Job 5	39.10720826
Job 6	42.0226548
Job 7	46.23114966
Job 8	48.87564718

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan (Expected Finishing Time) berdasarkan due date . Formula yang digunakan untuk menghitung adalah sebagai berikut :

$$et_{ik} = a_i + (d_i' - a_i) x \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Contoh perhitungannya :

$$et_{14} = 0 + (29.54082063 - 0) x \frac{18.04988426}{26.88932026} \\ = 19.82974609$$

Hasil perhitungan expected due date secara lengkap dapat dilihat pada tabel 4.21.

Job	Expected Finishing Time
Job 1	19.82974609
Job 2	22.74233616
Job 3	25.6435487
Job 4	28.1576659
Job 5	30.19941628
Job 6	31.86160295
Job 7	36.08873802
Job 8	38.89661784

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik})$$

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.22.

Job	t_{ik}
Job 1	8.5
Job 2	11.5
Job 3	15.75
Job 4	18.53
Job 5	21.11
Job 6	24.4
Job 7	27.98
Job 8	31.18

Contoh perhitungannya :

$$SB_{14} = 12.16333844 - (8.5 - 19.82974609) \\ = 23.49308$$

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.23.

Remaining Aggregate Time Buffer (SB)	
Job	
Job 1	23.49308
Job 2	24.80356
Job 3	16.62002
Job 4	23.16751
Job 5	30.8332
Job 6	29.44922
Job 7	30.26803
Job 8	38.6863
Total	217.3209

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

a. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$, maka langsung masuk ke langkah 5

Contoh perhitungannya :

$$AB_{14} - SB_{14} \leq \frac{1}{3} AB_{14}$$

$$12.16333844 - 23.49308 \leq \frac{1}{3} \times 12.16333844 \\ -11.32974609 \leq 4.054446147$$

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.24.

	AB-SB	Keterangan	1/3 AB
1	-11.32974609	<=	4.054446147
2	-11.24233616	<=	4.520406762
3	-9.893548698	<=	2.242156512
4	-9.627665899	<=	4.513281594
5	-9.089416275	<=	7.247929547
6	-7.461602951	<=	7.329204272
7	-8.108738021	<=	7.386430916
8	-7.716617839	<=	10.32322818

4.2.5 Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer

$$\begin{aligned}
 \text{Lateness} &= C_{i9} - et_{i4} \\
 &= 25.88 - 29.54082063 \\
 &= -3.66082
 \end{aligned}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.25

Job	Selisih	Keterangan
Job 5	-3.66082	Lateness
Job 6	-2.43499	Lateness
Job 4	-2.83898	Lateness
Job 8	0.009872	Tardiness
Job 7	0.952792	Tardiness
Job 3	0.687345	Tardiness
Job 1	0.84885	Tardiness
Job 2	2.954353	Tardiness

4.2.6 Penjadwalan Mesin *Bottleneck* Dengan Memperhatikan *Time Buffer* Menggunakan Metode EDD

Tabel 4.26. Urutan pengerjaan job berdasarkan Earliness Duedate

Urutan Pengerjaan Job	Duedate
Job 5	20.29

Tabel 4.26. Urutan pengerjaan job berdasarkan Earliness Duedate (lanjutan)

Job 6	24.75
Job 4	25.68
Job 8	28.75
Job 7	33.14
Job 3	33.33
Job 1	39.07
Job 2	40.37

Tabel 4.27 Hasil perhitungan dengan metode EDD

Job	M1			M2		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	0	2.75	2.75	2.75	2.45	5.2
Job 6	2.75	3	5.75	5.75	3	8.75
Job 4	5.75	2.5	8.25	8.75	2.45	11.2
Job 8	8.25	2.25	10.5	11.2	3.23	14.43
Job 7	10.5	2.25	12.75	14.43	3.23	17.66
Job 3	12.75	3.2	15.95	17.66	4	21.66
Job 1	15.95	3	18.95	21.66	3.5	25.16
Job 2	18.95	3.2	22.15	25.16	3	28.16

Tabel 4.27 Hasil perhitungan dengan metode EDD

Job	M3			M4		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	5.2	2.58	7.78	7.78	3.58	11.36
Job 6	8.75	3	11.75	11.75	3.52	15.27
Job 4	11.75	2.58	14.33	15.27	3.58	18.85
Job 8	14.43	3.2	17.63	18.85	4.25	23.1
Job 7	17.66	3.35	21.01	23.1	4.25	27.35
Job 3	21.66	2.25	23.91	27.35	4	31.35
Job 1	25.16	2	27.16	31.35	3.95	35.3
Job 2	28.16	2	30.16	35.3	3.95	39.25

Tabel 4.27 Hasil perhitungan dengan metode EDD

Job	M5			M6		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	11.36	2	13.36	13.36	2.1	15.46
Job 6	15.27	1.8	17.07	17.07	2.7	19.77
Job 4	18.85	2	20.85	20.85	2.75	23.6
Job 8	23.1	1.8	24.9	24.9	2.7	27.6
Job 7	27.35	1.8	29.15	29.15	2.7	31.85
Job 3	31.35	2.58	33.93	33.93	2.75	36.68
Job 1	35.3	2.7	38	38	3	41
Job 2	39.25	2.2	41.45	41.45	3.2	44.65

Tabel 4.27 Hasil perhitungan dengan metode EDD

Job	M7			M8		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	15.46	3.3	18.76	18.76	3.25	22.01
Job 6	19.77	3.5	23.27	23.27	2	25.27
Job 4	23.6	3.3	26.9	26.9	3	29.9
Job 8	27.6	3.75	31.35	31.35	2.25	33.6
Job 7	31.85	3.25	35.1	35.1	2.25	37.35
Job 3	36.68	3.7	40.38	40.38	2.25	42.63
Job 1	41	4.06	45.06	45.06	2	47.06
Job 2	44.65	4.08	48.73	48.73	2	50.73

Tabel 4.27 Hasil perhitungan dengan metode EDD

Job	M9		
	Mulai	Durasi	Selesai
Job 5	22.01	1.78	23.79
Job 6	25.27	1.63	26.9
Job 4	29.9	1.78	31.68
Job 8	33.6	1.75	35.35
Job 7	37.35	1.75	39.1
Job 3	42.63	1.83	44.46
Job 1	47.06	1.67	48.73
Job 2	50.73	1.83	52.56

Tabel 4.28 Hasil perhitungan ETC dan LTC

Job	ETC	LTC
Job 5	10.8208837	11.9708837
Job 6	11.9808837	15.2108837
Job 4	10.5708837	17.6108837
Job 8	12.3908837	19.5308837
Job 7	12.5408837	23.7708837
Job 3	12.9108837	23.3408837
Job 1	11.9108837	30.0308837
Job 2	11.6108837	31.6308837

4.2.7 Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer

Tabel 4.29. Tabel due date setiap job dengan memperhatikan time buffer

Job	Due date (d)
Job 5	20.29
Job 6	24.75
Job 4	25.68

Tabel 4.29. Tabel due date setiap job dengan memperhatikan time buffer (lanjutan)

Job 8	28.75
Job 7	33.14
Job 3	33.33
Job 1	39.07
Job 2	40.37

Tabel 4.30. Tabel makespan setiap job dengan memperhatikan time buffer

Job	Makespan (C_{i9})
Job 5	23.79
Job 6	26.9
Job 4	31.68
Job 8	35.35
Job 7	39.1
Job 3	44.46
Job 1	48.73
Job 2	52.56

$$\text{Lateness} = C_{i9} - d$$

$$= 23.79 - 20.29$$

$$= 3.5$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada

tabel 4.31.

Job	Slack	Keterangan
Job 5	3.5	Tardiness
Job 6	2.15	Tardiness
Job 4	6	Tardiness
Job 8	6.6	Tardiness
Job 7	5.96	Tardiness
Job 3	11.13	Tardiness
Job 1	9.66	Tardiness
Job 2	12.19	Tardiness
Total	57.19	

4.2.8 Menentukan Aggregate Time Buffer

Model yang digunakan adalah sebagai berikut.

Langkah 1: Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2.

Langkah 2.1 : Mengitung total time buffer dari waktu aktivitas di rantai produksi.

Contoh perhitungannya Total Time Buffer *Job* 1 pada mesin 4

$$AB_i = (C_{ik} - ETC) + (LTC - C_{ik})$$

$$AB_1 = (C_{14} - ETC) + (LTC - C_{14})$$

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.32.

Job	Total Time Buffer		
	ETC	LTC	Total
Job 5	0.539116	0.610884	1.15
Job 6	3.289116	0.059116	3.348233
Job 4	8.279116	1.239116	9.518233
Job 8	10.70912	3.569116	14.27823
Job 7	14.80912	3.579116	18.38823
Job 3	18.43912	8.009116	26.44823
Job 1	23.38912	5.269116	28.65823
Job 2	27.63912	7.619116	35.25823
Total			137.0476

Langkah 2.2 : Mengitung aggregate time buffer dari waktu aktivitas di rantai produksi.

Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

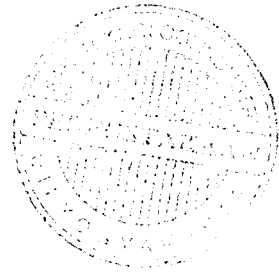
$$AB_{ik} = AB_i \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Tabel 4.33. Tabel total waktu proses setelah bottleneck dan total waktu proses

Job	Mesin Bottleneck	LT Setelah Bottleneck $\sum_{j=k}^{j=e} P_{ij}$	LT $\sum_{j=1}^{j=e} P_{ij}$
Job 5	M4	16.67988426	24.79932026
Job 6	M4	15.81988426	25.15932026
Job 4	M4	17.07988426	24.94932026
Job 8	M4	17.16988426	26.18932026

Tabel 4.33. Tabel total waktu proses setelah bottleneck dan total waktu proses (lanjutan)

Job 7	M4	16.66988426	25.83932026
Job 3	M4	17.77988426	27.56932026
Job 1	M4	18.04988426	26.88932026
Job 2	M4	17.92988426	26.46932026



$$AB_{14} = AB_1 \times \frac{\sum_{j=4}^{j=9} P_{ij}}{\sum_{j=1}^{j=9} P_{ij}}$$

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.34.

Job	Aggregate Time Buffer
Job 5	0.773483575
Job 6	2.105329226
Job 4	6.51602166
Job 8	9.360899741
Job 7	11.86291691
Job 3	17.05687736
Job 1	19.23729483
Job 2	23.88334961
Total	90.79617291

Langkah 3.1 : Menghitung due date. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} \right) xt_{ik}$$

Tabel 4.35. Tabel waktu kedatangan setiap job pada mesin pertama

Job	M1
	a_1
Job 5	0
Job 6	2.75
Job 4	5.75
Job 8	8.25
Job 7	10.5

Tabel 4.35. Tabel waktu kedatangan setiap job pada mesin pertama (lanjutan)

Job 3	12.75
Job 1	15.95
Job 2	18.95

Hasil perhitungan duedate secara lengkap dapat dilihat pada tabel 4.36.

Job	Duedate
Job 5	27.2072083
Job 6	30.1226548
Job 4	33.1501279
Job 8	37.2256472
Job 7	39.0811497
Job 3	42.8989826
Job 1	45.4908206
Job 2	48.0949856

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan (Expected Finishing Time) berdasarkan duedate . Formula yang digunakan untuk menghitung adalah sebagai berikut :

$$et_{ik} = a_i + (d_i' - a_i) \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Hasil perhitungan expected duedate secara lengkap dapat dilihat pada tabel 4.37.

Job	Expected Finishing Time
Job 5	18.2994163
Job 6	19.961603
Job 4	24.5076659
Job 8	27.2466178
Job 7	28.938738
Job 3	32.1935487
Job 1	35.7797461
Job 2	38.6923362

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik})$$

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.38.

t_{ik}
7.78
11.75
14.33
17.63
21.01
23.91
27.16
30.16

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.39.

Remaining Aggregate Time Buffer (SB)	
Job	
Job 5	11.2929
Job 6	10.31693
Job 4	16.69369
Job 8	18.97752
Job 7	19.79165
Job 3	25.34043
Job 1	27.85704
Job 2	32.41569
Total	162.6858

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

b. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$, maka langsung masuk ke langkah 5

Contoh perhitungannya :

$$AB_{14} - SB_{14} \leq \frac{1}{3} AB_{14}$$

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.40.

	AB-SB	Keterangan	1/3 AB
Job 5	-10.519416	<=	0.257827858
Job 6	-8.211603	<=	0.701776409
Job 4	-10.177666	<=	2.17200722

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.40. (lanjutan)

Job 8	-9.6166178	<=	3.120299914
Job 7	-7.928738	<=	3.954305637
Job 3	-8.2835487	<=	5.685625787
Job 1	-8.6197461	<=	6.41243161
Job 2	-8.5323362	<=	7.961116538

4.2.9 Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer

$$\text{Lateness} = C_{i9} - et_{i4}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.41

Job	Slack	Keterangan
Job 5	-3.41721	Lateness
Job 6	-3.22265	Lateness
Job 4	-1.47013	Lateness
Job 8	-1.87565	Lateness
Job 7	0.01885	Tardiness
Job 3	1.561017	Tardiness
Job 1	3.239179	Tardiness
Job 2	4.465014	Tardiness

4.2.10 Penjadwalan Mesin *Bottleneck* dengan Memperhatikan *Time Buffer* Menggunakan Metode SPT

Tabel 4.42. Urutan pengerjaan job berdasarkan waktu proses terpendek

Urutan Pengerjaan Job	Total Waktu Proses
Job 5	23.79
Job 4	23.94
Job 6	24.15
Job 7	24.83
Job 8	25.18

Tabel 4.42. Urutan pengerjaan job berdasarkan waktu proses terpendek (lanjutan)

Job 2	25.46
Job 1	25.88
Job 3	26.56

Tabel 4.43 Hasil perhitungan dengan metode SPT

No Order	M1			M2		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	0	2.75	2.75	2.75	2.45	5.2
Job 4	2.75	2.5	5.25	5.25	2.45	7.7
Job 6	5.25	3	8.25	8.25	3	11.25
Job 7	8.25	2.25	10.5	11.25	3.23	14.48
Job 8	10.5	2.25	12.75	14.48	3.23	17.71
Job 2	12.75	3.2	15.95	17.71	3	20.71
Job 1	15.95	3	18.95	20.71	3.5	24.21
Job 3	18.95	3.2	22.15	24.21	4	28.21

Tabel 4.43 Hasil perhitungan dengan metode SPT

No Order	M3			M4		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	5.2	2.58	7.78	7.78	3.58	11.36
Job 4	7.78	2.58	10.36	11.36	3.58	14.94
Job 6	11.25	3	14.25	14.94	3.52	18.46
Job 7	14.48	3.35	17.83	18.46	4.25	22.71
Job 8	17.83	3.2	21.03	22.71	4.25	26.96
Job 2	21.03	2	23.03	26.96	3.95	30.91
Job 1	24.21	2	26.21	30.91	3.95	34.86
Job 3	28.21	2.25	30.46	34.86	4	38.86

Tabel 4.43 Hasil perhitungan dengan metode SPT

No Order	M5			M6		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	11.36	2	13.36	13.36	2.1	15.46
Job 4	14.94	2	16.94	16.94	2.75	19.69
Job 6	18.46	1.8	20.26	20.26	2.7	22.96
Job 7	22.71	1.8	24.51	24.51	2.7	27.21
Job 8	26.96	1.8	28.76	28.76	2.7	31.46
Job 2	30.91	2.2	33.11	33.11	3.2	36.31
Job 1	34.86	2.7	37.56	37.56	3	40.56
Job 3	38.86	2.58	41.44	41.44	2.75	44.19

Tabel 4.43 Hasil perhitungan dengan metode SPT

No Order	M7			M8		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	15.46	3.3	18.76	18.76	3.25	22.01

Tabel 4.43 Hasil perhitungan dengan metode SPT (lanjutan)

Job 4	19.69	3.3	22.99	22.99	3	25.99
Job 6	22.96	3.5	26.46	26.46	2	28.46
Job 7	27.21	3.25	30.46	30.46	2.25	32.71
Job 8	31.46	3.75	35.21	35.21	2.25	37.46
Job 2	36.31	4.08	40.39	40.39	2	42.39
Job 1	40.56	4.06	44.62	44.62	2	46.62
Job 3	44.19	3.7	47.89	47.89	2.25	50.14

Tabel 4.43 Hasil perhitungan dengan metode SPT

Job	M9		
	Mulai	Durasi	Selesai
Job 5	22.01	1.78	23.79
Job 4	25.99	1.78	27.77
Job 6	28.46	1.63	30.09
Job 7	32.71	1.75	34.46
Job 8	37.46	1.75	39.21
Job 2	42.39	1.83	44.22
Job 1	46.62	1.67	48.29
Job 3	50.14	1.83	51.97

Tabel 4.44 Hasil perhitungan ETC dan LTC

Job	ETC	LTC
Job 5	10.8208837	11.9708837
Job 4	10.5708837	17.6108837
Job 6	11.9808837	15.2108837
Job 7	12.5408837	23.7708837
Job 8	12.3908837	19.5308837
Job 2	11.6108837	31.6308837
Job 1	11.9108837	30.0308837
Job 3	12.9108837	23.3408837

4.2.11 Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer

Tabel 4.45. Tabel due date setiap job dengan memperhatikan time buffer

Job	Due date (d)
Job 5	20.29
Job 4	25.68
Job 6	24.75
Job 7	33.14
Job 8	28.75

Tabel 4.45. Tabel due date setiap job dengan memperhatikan time buffer (lanjutan)

Job 2	40.37
Job 1	39.07
Job 3	33.33

Tabel 4.46. Tabel makespan setiap job dengan memperhatikan time buffer

Job	Makespan (C_{i9})
Job 5	23.79
Job 4	27.77
Job 6	30.09
Job 7	34.46
Job 8	39.21
Job 2	44.22
Job 1	48.29
Job 3	51.97

$$\text{Lateness} = C_{i9} - d$$

$$= 23.79 - 20.29$$

$$= 3.5$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada

tabel 4.47.

Job	Slack	Keterangan
Job 5	3.5	Tardiness
Job 4	2.09	Tardiness
Job 6	5.34	Tardiness
Job 7	1.32	Tardiness
Job 8	10.46	Tardiness
Job 2	3.85	Tardiness
Job 1	9.22	Tardiness
Job 3	18.64	Tardiness
Total	54.42	

4.2.12 Menentukan Aggregate Time Buffer

Model yang digunakan adalah sebagai berikut.

Langkah 1: Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2.

Langkah 2.1 : Mengitung total time buffer dari waktu aktivitas di rantai produksi.

Contoh perhitungannya Total Time Buffer *Job* 1 pada mesin 4

$$AB_i = (C_{ik} - ETC) + (LTC - C_{ik})$$

$$AB_1 = (C_{14} - ETC) + (LTC - C_{14})$$

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel

4.48.

Job	Total Time Buffer		
	ETC	LTC	Total
Job 5	0.539116	0.610884	1.15
Job 4	4.369116	2.670884	7.04
Job 6	6.479116	3.249116	9.728233
Job 7	10.16912	1.060884	11.23
Job 8	14.56912	7.429116	21.99823
Job 2	19.29912	0.720884	20.02
Job 1	22.94912	4.829116	27.77823
Job 3	25.94912	15.51912	41.46823
Total			140.4129

Langkah 2.2 : Mengitung aggregate time buffer dari waktu aktivitas di lantai produksi.

Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Tabel 4.49. Tabel total waktu proses setelah bottleneck dan total waktu proses

Job	Mesin Bottleneck	LT Setelah Bottleneck $\sum_{j=k}^{j=e} P_{ij}$	LT $\sum_{j=1}^{j=e} P_{ij}$
Job 5	M4	16.67988426	24.79932026
Job 4	M4	17.07988426	24.94932026
Job 6	M4	15.81988426	25.15932026
Job 7	M4	16.66988426	25.83932026
Job 8	M4	17.16988426	26.18932026

Tabel 4.49. Tabel total waktu proses setelah bottleneck dan total waktu proses (lanjutan)

Job 2	M4	17.92988426	26.46932026
Job 1	M4	18.04988426	26.88932026
Job 3	M4	17.77988426	27.56932026

$$AB_{14} = AB_1 x \frac{\sum_{j=4}^{j=9} P_{ij}}{\sum_{j=1}^{j=9} P_{ij}}$$

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.50.

Job	Aggregate Time Buffer
Job 5	0.773483575
Job 4	4.819465378
Job 6	6.116998084
Job 7	7.244880994
Job 8	14.42218064
Job 2	13.56122029
Job 1	18.64658082
Job 3	26.74350942
Total	92.32831921

Langkah 3.1 : Menghitung due date. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} xt_{ik} \right)$$

Tabel 4.51. Tabel waktu kedatangan setiap job pada mesin pertama

Job	M1
	a_i
Job 5	0
Job 4	2.75
Job 6	5.25
Job 7	8.25
Job 8	10.5
Job 2	12.75
Job 1	15.95
Job 3	18.95

Hasil perhitungan due date secara lengkap dapat dilihat pada tabel 4.52.

Job	Due date
Job 5	27.2072083
Job 4	30.1501279
Job 6	32.6226548
Job 7	36.8311497
Job 8	39.4756472
Job 2	41.8949856
Job 1	45.4908206
Job 3	49.0989826

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan (Expected Finishing Time) berdasarkan due date . Formula yang digunakan untuk menghitung adalah sebagai berikut :

$$et_{ik} = a_i + (d_i' - a_i) \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Hasil perhitungan expected due date secara lengkap dapat dilihat pada tabel 4.53.

Job	Expected Finishing Time
Job 5	18.29941628
Job 4	21.5076659
Job 6	22.46160295
Job 7	26.68873802
Job 8	29.49661784
Job 2	32.49233616
Job 1	35.77974609
Job 3	38.3935487

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik})$$

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.54.

t_{ik}
7.78
10.36

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.54.

(lanjutan)

14.25
17.83
21.03
23.03
26.21
30.46

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.55.

Remaining Aggregate Time Buffer (SB)	
Job	
Job 5	11.2929
Job 4	15.9671
Job 6	14.3286
Job 7	16.1036
Job 8	22.8888
Job 2	23.0236
Job 1	28.2163
Job 3	34.6771
Total	166.498

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

c. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$, maka langsung masuk ke langkah 5

Contoh perhitungannya :

$$AB_{14} - SB_{14} \leq \frac{1}{3} AB_{14}$$

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.56.

	AB-SB	Keterangan	1/3 AB
Job 5	-10.51941628	<=	0.257827858
Job 4	-11.1476659	<=	1.606488459
Job 6	-8.211602951	<=	2.038999361
Job 7	-8.858738021	<=	2.414960331
Job 8	-8.466617839	<=	4.807393548
Job 2	-9.462336162	<=	4.520406762

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.56. (lanjutan)

Job 1	-9.569746095	<=	6.21552694
Job 3	-7.933548698	<=	8.914503141

4.2.13 Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer

$$\text{Lateness} = C_{i,9} - et_{i,4}$$

Hasil perhitungan latenees dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.57

Job	Slack	Keterangan
Job 5	-3.4172	Lateness
Job 4	-2.3801	Lateness
Job 6	-2.5327	Lateness
Job 7	-2.3711	Lateness
Job 8	-0.2656	Lateness
Job 2	2.32501	Tardiness
Job 1	2.79918	Tardiness
Job 3	2.87102	Tardiness

4.2.14 Penjadwalan Mesin *Bottleneck* dengan Memperhatikan *Time Buffer* Menggunakan Pendekatan Metode LPT

Tabel 4.58. Urutan pengerjaan job berdasarkan waktu proses terpanjang

Urutan Pengerjaan Job	Total Waktu Proses
Job 3	26.56
Job 1	25.88
Job 2	25.46
Job 8	25.18
Job 7	24.83
Job 6	24.15
Job 4	23.94
Job 5	23.79

Tabel 4.59 Hasil perhitungan dengan metode LPT

No Order	M1			M2		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 3	0	3.2	3.2	3.2	4	7.2
Job 1	3.2	3	6.2	7.2	3.5	10.7
Job 2	6.2	3.2	9.4	10.7	3	13.7
Job 8	9.4	2.25	11.65	13.7	3.23	16.93
Job 7	11.65	2.25	13.9	16.93	3.23	20.16
Job 6	13.9	3	16.9	20.16	3	23.16
Job 4	16.9	2.5	19.4	23.16	2.45	25.61
Job 5	19.4	2.75	22.15	25.61	2.45	28.06

Tabel 4.59 Hasil perhitungan dengan metode LPT

No Order	M3			M4		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 3	7.2	2.25	9.45	9.45	4	13.45
Job 1	10.7	2	12.7	13.45	3.95	17.4
Job 2	13.7	2	15.7	17.4	3.95	21.35
Job 8	16.93	3.2	20.13	21.35	4.25	25.6
Job 7	20.16	3.35	23.51	25.6	4.25	29.85
Job 6	23.51	3	26.51	29.85	3.52	33.37
Job 4	26.51	2.58	29.09	33.37	3.58	36.95
Job 5	29.09	2.58	31.67	36.95	3.58	40.53

Tabel 4.59 Hasil perhitungan dengan metode LPT

No Order	M5			M6		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 3	13.45	2.58	16.03	16.03	2.75	18.78
Job 1	17.4	2.7	20.1	20.1	3	23.1
Job 2	21.35	2.2	23.55	23.55	3.2	26.75
Job 8	25.6	1.8	27.4	27.4	2.7	30.1
Job 7	29.85	1.8	31.65	31.65	2.7	34.35
Job 6	33.37	1.8	35.17	35.17	2.7	37.87
Job 4	36.95	2	38.95	38.95	2.75	41.7
Job 5	40.53	2	42.53	42.53	2.1	44.63

Tabel 4.59 Hasil perhitungan dengan metode LPT

No Order	M7			M8		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 3	18.78	3.7	22.48	22.48	2.25	24.73
Job 1	23.1	4.06	27.16	27.16	2	29.16
Job 2	26.75	4.08	30.83	30.83	2	32.83
Job 8	30.1	3.75	33.85	33.85	2.25	36.1
Job 7	34.35	3.25	37.6	37.6	2.25	39.85
Job 6	37.87	3.5	41.37	41.37	2	43.37

Tabel 4.59 Hasil perhitungan dengan metode LPT (lanjutan)

Job 4	41.7	3.3	45	45	3	48
Job 5	44.63	3.3	47.93	48	3.25	51.25

Tabel 4.59 Hasil perhitungan dengan metode LPT

No Order	M9		
	Mulai	Durasi	Selesai
Job 3	24.73	1.83	26.56
Job 1	29.16	1.67	30.83
Job 2	32.83	1.83	34.66
Job 8	36.1	1.75	37.85
Job 7	39.85	1.75	41.6
Job 6	43.37	1.63	45
Job 4	48	1.78	49.78
Job 5	51.25	1.78	53.03

Tabel 4.60 Hasil perhitungan ETC dan LTC

Job	ETC	LTC
Job 3	12.9108837	23.3408837
Job 1	11.9108837	30.0308837
Job 2	11.6108837	31.6308837
Job 8	12.3908837	19.5308837
Job 7	12.5408837	23.7708837
Job 6	11.9808837	15.2108837
Job 4	10.5708837	17.6108837
Job 5	10.8208837	11.9708837

4.2.15 Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer

Tabel 4.61. Tabel due date setiap job dengan memperhatikan time buffer

Job	Due date (d)
Job 3	33.33
Job 1	39.07
Job 2	40.37
Job 8	28.75
Job 7	33.14
Job 6	24.75
Job 4	25.68
Job 5	20.29

Tabel 4.62. Tabel makespan setiap job dengan memperhatikan time buffer

Job	Makespan (C_{19})
Job 3	26.56
Job 1	30.83
Job 2	34.66
Job 8	37.85
Job 7	41.6
Job 6	45
Job 4	49.78
Job 5	53.03



$$\begin{aligned} \text{Lateness} &= C_{19} - d \\ &= 26.56 - 33.33 \\ &= -6,7 \end{aligned}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.63.

Job	Slack	Keterangan
Job 3	-6.77	Latenees
Job 1	-8.24	Latenees
Job 2	-5.71	Latenees
Job 8	9.1	Tardiness
Job 7	8.46	Tardiness
Job 6	20.25	Tardiness
Job 4	24.1	Tardiness
Job 5	32.74	Tardiness
Total	73.93	

4.2.16 Menentukan Aggregate Time Buffer

Model yang digunakan adalah sebagai berikut.

Langkah 1: Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2.

Langkah 2.1 : Mengitung total time buffer dari waktu aktivitas di lantai produksi.

Contoh perhitungannya Total Time Buffer *Job* 1 pada mesin 4

$$AB_i = (C_{ik} - ETC) + (LTC - C_{ik})$$

$$AB_1 = (C_{14} - ETC) + (LTC - C_{14})$$

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.64.

Job	Total Time Buffer		
	ETC	LTC	Total
Job 3	0.53912	9.89088	10.43
Job 1	5.48912	12.6309	18.12
Job 2	9.73912	10.2809	20.02
Job 8	13.2091	6.06912	19.2782
Job 7	17.3091	6.07912	23.3882
Job 6	21.3891	18.1591	39.5482
Job 4	26.3791	19.3391	45.7182
Job 5	29.7091	28.5591	58.2682
Total			234.771

Langkah 2.2 : Mengitung aggregate time buffer dari waktu aktivitas di lantai produksi.

Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i x \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Tabel 4.65. Tabel total waktu proses setelah bottleneck dan total waktu proses

Job	Mesin Bottlenect	LT Setelah Bottleneck $\sum_{j=k}^{j=e} P_{ij}$	LT $\sum_{j=1}^{j=e} P_{ij}$
1	M4	17.77988426	27.56932026
2	M4	18.04988426	26.88932026
3	M4	17.92988426	26.46932026
4	M4	17.16988426	26.18932026
5	M4	16.66988426	25.83932026
6	M4	15.81988426	25.15932026
7	M4	17.07988426	24.94932026
8	M4	16.67988426	24.79932026

$$AB_{14} = AB_1 x \frac{\sum_{j=4}^{j=9} P_{ij}}{\sum_{j=1}^{j=9} P_{ij}}$$

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.66.

Job	Aggregate Time Buffer
Job 3	6.726469535
Job 1	12.16333844
Job 2	13.56122029
Job 8	12.63893141
Job 7	15.08859856
Job 6	24.86746287
Job 4	31.2979317
Job 5	39.19088772
Total	155.5348405

Langkah 3.1 : Menghitung due date. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} xt_{ik} \right)$$

Tabel 4.67. Tabel waktu kedatangan setiap job pada mesin pertama

Job	M1
	a_1
Job 5	0
Job 4	3.2
Job 6	6.2
Job 7	9.4
Job 8	11.65
Job 2	13.9
Job 1	16.9
Job 3	19.4

Hasil perhitungan due date secara lengkap dapat dilihat pada tabel 4.68.

Job	Due date
Job 3	30.14898259
Job 1	32.74082063
Job 2	35.3449856

Hasil perhitungan due date secara lengkap dapat dilihat pada tabel 4.68. (lanjutan)

Job 8	38.37564718
Job 7	40.23114966
Job 6	41.2726548
Job 4	44.30012794
Job 5	46.60720826

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan (Expected Finishing Time) berdasarkan due date . Formula yang digunakan untuk menghitung adalah sebagai berikut :

$$et_{ik} = a_i + (d_i' - a_i) \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Hasil perhitungan expected due date secara lengkap dapat dilihat pada tabel 4.69.

Job	Expected Finishing Time
Job 3	19.4435487
Job 1	23.02974609
Job 2	25.94233616
Job 8	28.39661784
Job 7	30.08873802
Job 6	31.11160295
Job 4	35.6576659
Job 5	37.69941628

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik})$$

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.70.

t_{ik}
9.45
12.7
15.7
20.13
23.51
26.51
29.09
31.67

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.71.

Remaining Aggregate Time Buffer (SB)	
Job	
Job 3	16.72
Job 1	22.4931
Job 2	23.8036
Job 8	20.9055
Job 7	21.6673
Job 6	29.4691
Job 4	37.8656
Job 5	45.2203
Total	218.145

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

d. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$, maka langsung masuk ke langkah 5

Contoh perhitungannya :

$$AB_{14} - SB_{14} \leq \frac{1}{3} AB_{14}$$

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.72.

	AB-SB	Keterangan	1/3 AB
Job 3	-9.993548698	<=	2.242156512
Job 1	-10.32974609	<=	4.054446147
Job 2	-10.24233616	<=	4.520406762
Job 8	-8.266617839	<=	4.212977138
Job 7	-6.578738021	<=	5.029532854
Job 6	-4.601602951	<=	8.289154291
Job 4	-6.567665899	<=	10.4326439
Job 5	-6.029416275	<=	13.06362924

4.2.17 Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer

$$\text{Lateness} = C_{i9} - et_{i4}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.73

Job	Slack	Keterangan
Job 3	-3.589	Lateness
Job 1	-1.9108	Lateness
Job 2	-0.685	Lateness
Job 8	-0.5256	Lateness
Job 7	1.36885	Tardiness
Job 6	3.72735	Tardiness
Job 4	5.47987	Tardiness
Job 5	6.42279	Tardiness

4.2.18 Penjadwalan Mesin *Bottleneck* dengan Memperhatikan *Time Buffer* Menggunakan Metode CR

Tabel 4.74. Urutan pengerjaan job berdasarkan Critical Ratio

Urutan Pengerjaan Job	Critical Ratio
Job 5	0.85288
Job 6	1.02484
Job 4	1.07268
Job 8	1.14178
Job 3	1.25489
Job 7	1.33468
Job 1	1.50966
Job 2	1.58562

Tabel 4.75 Hasil perhitungan dengan metode CR

No Order	M1			M2		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	0	2.75	2.75	2.75	2.45	5.2
Job 6	2.75	3	5.75	5.75	3	8.75
Job 4	5.75	2.5	8.25	8.75	2.45	11.2
Job 8	8.25	2.25	10.5	11.2	3.23	14.43
Job 3	10.5	3.2	13.7	14.43	4	18.43
Job 7	13.7	2.25	15.95	18.43	3.23	21.66
Job 1	15.95	3	18.95	21.66	3.5	25.16
Job 2	18.95	3.2	22.15	25.16	3	28.16

Tabel 4.75 Hasil perhitungan dengan metode CR

No Order	M3			M4		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	5.2	2.58	7.78	7.78	3.58	11.36
Job 6	8.75	3	11.75	11.75	3.52	15.27
Job 4	11.75	2.58	14.33	15.27	3.58	18.85
Job 8	14.43	3.2	17.63	18.85	4.25	23.1
Job 3	18.43	2.25	20.68	23.1	4	27.1
Job 7	21.66	3.35	25.01	27.1	4.25	31.35
Job 1	25.16	2	27.16	31.35	3.95	35.3
Job 2	28.16	2	30.16	35.3	3.95	39.25

Tabel 4.75 Hasil perhitungan dengan metode CR

No Order	M5			M6		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	11.36	2	13.36	13.36	2.1	15.46
Job 6	15.27	1.8	17.07	17.07	2.7	19.77
Job 4	18.85	2	20.85	20.85	2.75	23.6
Job 8	23.1	1.8	24.9	24.9	2.7	27.6
Job 3	27.1	2.58	29.68	29.68	2.75	32.43
Job 7	31.35	1.8	33.15	33.15	2.7	35.85
Job 1	35.3	2.7	38	38	3	41
Job 2	39.25	2.2	41.45	41.45	3.2	44.65

Tabel 4.75 Hasil perhitungan dengan metode CR

No Order	M7			M8		
	Mulai	Durasi	Selesai	Mulai	Durasi	Selesai
Job 5	15.46	3.3	18.76	18.76	3.25	22.01
Job 6	19.77	3.5	23.27	23.27	2	25.27
Job 4	23.6	3.3	26.9	26.9	3	29.9
Job 8	27.6	3.75	31.35	31.35	2.25	33.6
Job 3	32.43	3.7	36.13	36.13	2.25	38.38
Job 7	35.85	3.25	39.1	39.1	2.25	41.35
Job 1	41	4.06	45.06	45.06	2	47.06
Job 2	44.65	4.08	48.73	48.73	2	50.73

Tabel 4.75 Hasil perhitungan dengan metode CR

No Order	M9		
	Mulai	Durasi	Selesai
Job 5	22.01	1.78	23.79
Job 6	25.27	1.63	26.9
Job 4	29.9	1.78	31.68
Job 8	33.6	1.75	35.35
Job 3	38.38	1.83	40.21
Job 7	41.35	1.75	43.1

Tabel 4.75 Hasil perhitungan dengan metode CR (lanjutan)

Job 1	47.06	1.67	48.73
Job 2	50.73	1.83	52.56

Tabel 4.76 Hasil perhitungan ETC dan LTC

Job	ETC	LTC
Job 5	10.8208837	11.9708837
Job 6	11.9808837	15.2108837
Job 4	10.5708837	17.6108837
Job 8	12.3908837	19.5308837
Job 3	12.9108837	23.3408837
Job 7	12.5408837	23.7708837
Job 1	11.9108837	30.0308837
Job 2	11.6108837	31.6308837

4.2.19 Penentuan Tardiness dan Lateness Dengan Memperhatikan Time Buffer

Tabel 4.77. Tabel due date setiap job dengan memperhatikan time buffer

Job	Due date (d)
Job 5	20.29
Job 6	24.75
Job 4	25.68
Job 8	28.75
Job 3	33.33
Job 7	33.14
Job 1	39.07
Job 2	40.37

Tabel 4.78. Tabel makespan setiap job dengan memperhatikan time buffer

Job	Makespan (C_{i9})
Job 5	23.79
Job 6	26.9
Job 4	31.68
Job 8	35.35
Job 3	40.21
Job 7	43.1
Job 1	48.73
Job 2	52.56

$$\begin{aligned} \text{Lateness} &= C_{j_9} - d \\ &= 25.88 - 39.07 \\ &= -13,19 \end{aligned}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.79.

Job	Slack	Keterangan
Job 5	3.5	Tardiness
Job 6	2.15	Tardiness
Job 4	6	Tardiness
Job 8	6.6	Tardiness
Job 3	6.88	Tardiness
Job 7	9.96	Tardiness
Job 1	9.66	Tardiness
Job 2	12.19	Tardiness
Total	56.94	

4.2.20 Menentukan Aggregate Time Buffer

Model yang digunakan adalah sebagai berikut.

Langkah 1: Mengidentifikasi bottleneck, jika ada masuk ke langkah ke 2.

Langkah 2.1: Mengitung total time buffer dari waktu aktivitas di lantai produksi.

Contoh perhitungannya Total Time Buffer *Job* 1 pada mesin 4

$$AB_i = (C_{ik} - ETC) + (LTC - C_{ik})$$

$$AB_1 = (C_{14} - ETC) + (LTC - C_{14})$$

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel

4.80.

Job	Total Time Buffer		
	ETC	LTC	Total
Job 5	0.53912	0.61088	1.15
Job 6	3.28912	0.05912	3.34823
Job 4	8.27912	1.23912	9.51823
Job 8	10.7091	3.56912	14.2782
Job 3	14.1891	3.75912	17.9482
Job 7	18.8091	7.57912	26.3882

Hasil perhitungan total time buffer setiap *job* secara lengkap dapat dilihat pada tabel

4.80. (lanjutan)

Job 1	23.3891	5.26912	28.6582
Job 2	27.6391	7.61912	35.2582
Total			136.548

Langkah 2.2 : Mengitung aggregate time buffer dari waktu aktivitas di lantai produksi.

Formula yang digunakan untuk menghitung ATB adalah sebagai berikut:

$$AB_{ik} = AB_i x \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Tabel 4.81. Tabel total waktu proses setelah bottleneck dan total waktu proses

Job	Mesin Bottlenect	LT Setelah Bottleneck $\sum_{j=k}^{j=e} P_{ij}$	LT $\sum_{j=1}^{j=e} P_{ij}$
1	M4	16.67988426	24.79932026
2	M4	15.81988426	25.15932026
3	M4	17.07988426	24.94932026
4	M4	17.16988426	26.18932026
5	M4	17.77988426	27.56932026
6	M4	16.66988426	25.83932026
7	M4	18.04988426	26.88932026
8	M4	17.92988426	26.46932026

$$AB_{14} = AB_1 x \frac{\sum_{j=4}^{j=9} P_{ij}}{\sum_{j=1}^{j=9} P_{ij}}$$

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.82.

Job	Aggregate Time Buffer
Job 5	0.773483575

Hasil perhitungan Aggregate time buffer setiap *job* secara lengkap dapat dilihat pada tabel 4.82. (lanjutan)

Job 6	2.105329226
Job 4	6.51602166
Job 8	9.360899741
Job 3	11.5750949
Job 7	17.02400756
Job 1	19.23729483
Job 2	23.88334961
Total	90.4754811

Langkah 3.1 : Menghitung due date. Formula yang digunakan untuk menghitung due date adalah sebagai berikut:

$$d_i = a_i + LT + \left(\frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}} xt_{ik} \right)$$

Tabel 4.83. Tabel waktu kedatangan setiap job pada mesin pertama

Job	M1
	a_1
Job 5	0
Job 6	3
Job 4	6.2
Job 8	9.4
Job 3	11.9
Job 7	14.9
Job 1	17.65
Job 2	19.9

Hasil perhitungan due date secara lengkap dapat dilihat pada tabel 4.84.

Job	Due date
Job 5	27.20720826
Job 6	30.1226548
Job 4	33.15012794
Job 8	37.22564718
Job 3	40.64898259
Job 7	42.28114966
Job 1	45.49082063
Job 2	48.0949856

Langkah 3.2 : Menghitung waktu selesai pengerjaan produk yang diharapkan (Expected Finishing Time) berdasarkan due date . Formula yang digunakan untuk menghitung adalah sebagai berikut :

$$et_{ik} = a_i + (d'_i - a_i) \times \frac{\sum_{j=k}^{j=e} P_{ij}}{\sum_{j=1}^{j=e} P_{ij}}$$

Hasil perhitungan expected due date secara lengkap dapat dilihat pada tabel 4.85.

Job	Expected Finishing Time
Job 5	18.29941628
Job 6	19.96160295
Job 4	24.5076659
Job 8	27.24661784
Job 3	29.9435487
Job 7	32.13873802
Job 1	35.77974609
Job 2	38.69233616

Langkah 3.3 : Menghitung sisa dari Aggregate Time Buffer. Formula yang digunakan untuk menghitung sisa ATB adalah sebagai berikut :

$$SB_{ik} = AB_{ik} - (t_{ik} - et_{ik})$$

Data waktu kedatangan job pada mesin bottleneck dapat dilihat pada tabel 4.86.

t_{ik}
7.78
11.75
14.33
17.63
20.68
25.01
27.16
30.16

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.87.

Remaining Aggregate Time Buffer (SB)	
Job	
Job 5	11.2929

Hasil sisa dari Aggregate Time Buffer secara lengkap dapat dilihat pada tabel 4.87.

(lanjutan)

Job 6	10.3169
Job 4	16.6937
Job 8	18.9775
Job 3	20.8386
Job 7	24.1527
Job 1	27.857
Job 2	32.4157
Total	162.545

Langkah 4 : Menguji validasi Aggregate Time Buffer berdasarkan perhitungan pada langkah 3.3. Formula yang digunakan adalah sebagai berikut :

e. Jika $AB_{ik} - SB_{ik} \leq \frac{1}{3} AB_{ik}$, maka langsung masuk ke langkah 5

Contoh perhitungannya :

$$AB_{14} - SB_{14} \leq \frac{1}{3} AB_{14}$$

Hasil sisa uji validasi secara lengkap dapat dilihat pada tabel 4.88.

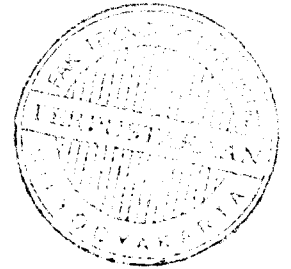
	AB-SB	Keterangan	1/3 AB
Job 5	-10.51941628	<=	0.257827858
Job 6	-8.211602951	<=	0.701776409
Job 4	-10.1776659	<=	2.17200722
Job 8	-9.616617839	<=	3.120299914
Job 3	-9.263548698	<=	3.858364967
Job 7	-7.128738021	<=	5.674669185
Job 1	-8.619746095	<=	6.41243161
Job 2	-8.532336162	<=	7.961116538

4.2.21 Penentuan Tardiness dan Lateness Dengan Memperhatikan Aggregate Time Buffer

$$\text{Lateness} = C_{i9} - et_{i4}$$

Hasil perhitungan lateness dan tardiness setiap *job* secara lengkap dapat dilihat pada tabel 4.89

Job	Slack	Keterangan
Job 5	-3.4172	Lateness
Job 6	-3.2227	Lateness
Job 4	-1.4701	Lateness
Job 8	-1.8756	Lateness
Job 3	-0.439	Lateness
Job 7	0.81885	Tardiness
Job 1	3.23918	Tardiness
Job 2	4.46501	Tardiness



4.2.22 Perbandingan Tardiness dan Lateness Setiap Job

Tabel 4.90 Perbandingan Tardiness dan Lateness Masing-Masing Metode.

Job	FCFS				EDD			
	Sebelum ATB		Sesudah ATB		Sebelum ATB		Sesudah ATB	
	Tardiness	Lateness	Tardiness	Lateness	Tardiness	Lateness	Tardiness	Lateness
Job 1		-13.19		-3.6608	9.66		3.23918	
Job 2		-10.66		-2.435	12.19		4.46501	
Job 3	0.18			-2.839	11.13		1.56102	
Job 4	11.13		0.00987		6			-1.4701
Job 5	19.77		0.95279		3.5			-3.4172
Job 6	17.96		0.68735		2.15			-3.2227
Job 7	13.94		0.84885		5.96		0.01885	
Job 8	23.08		2.95435		6.6			-1.8756
Total	86.06	-23.85	5.45321	-8.9348	57.19	0	9.28406	-9.9856

Tabel 4.90 Perbandingan Tardiness dan Lateness Masing-Masing Metode.

Job	SPT				LPT			
	Sebelum ATB		Sesudah ATB		Sebelum ATB		Sesudah ATB	
	Tardiness	Lateness	Tardiness	Lateness	Tardiness	Lateness	Tardiness	Lateness
Job 1	9.22		2.79918			-8.24		-1.9108
Job 2	3.85		2.32501			-5.71		-0.685
Job 3	18.64		2.87102			-6.77		-3.589
Job 4	2.09			-2.3801	24.1		5.47987	
Job 5	3.5			-3.4172	32.74		6.42279	
Job 6	5.34			-2.5327	20.25		3.72735	
Job 7	1.32			-2.3711	8.46		1.36885	
Job 8	10.46			-0.2656	9.1			-0.5256
Total	54.42	0	7.99521	-10.967	94.65	-20.72	16.9989	-6.7104

Tabel 4.90 Perbandingan Tardines dan Lateness Masing-Masing Metode.

Job	CR			
	Sebelum ATB		Sesudah ATB	
	Tardiness	Lateness	Tardiness	Lateness
Job 1	9.66		3.23918	
Job 2	12.19		4.46501	
Job 3	6.88			-0.439
Job 4	6			-1.4701
Job 5	3.5			-3.4172
Job 6	2.15			-3.2227
Job 7	9.96		0.81885	
Job 8	6.6			-1.8756
Total	56.94	0	8.52304	-10.425

Tabel 4.91 Peningkatan Ratio Slack Masing-Masing Metode.

Metode	Slack		Ratio
	Sebelum ATB	Sesudah ATB	
FCFS	109.91	14.38800099	95.522 %
EDD	57.19	19.26969968	37.9203 %
SPT	54.42	18.96199901	35.458 %
LPT	115.37	23.70929535	91.6607 %
CR	56.94	18.94766487	37.9923 %

BAB V

PEMBAHASAN

Penelitian ini dilakukan dengan tujuan untuk mengontrol *bottleneck* agar dapat melindungi laju produksi dari gangguan yang terjadi dalam sistem produksi dengan menambah suatu penyangga (buffer). Penambahan sejumlah *Time Buffer* dapat melindungi laju produksi dan akan mempengaruhi *makespan* yang sangat berkaitan erat dengan performansi *duedate*. Dari kondisi tersebut maka diperlukan suatu tindakan bagaimana mengestimasi *duedate*, agar diperoleh performansi *duedate* yang baik berdasarkan tardiness dan lateness dengan pendekatan *TOC Aggregate*.

Dari hasil pengolahan data pada bab IV, berdasarkan algoritma zijm, pada rantai produksi mengalami *bottleneck* pada mesin ke empat, ini dibuktikan dengan waktu proses sebesar 31.08 menit, selain waktu proses di mesin ke empat juga mengalami beban kerja terbesar yaitu 0.117011038 menit dan ekspektasi menunggu setiap job terbesar yaitu 0.199680303 menit. Selanjutnya setelah mendeteksi mesin *bottleneck* dengan menggunakan Algoritma Zijm kemudian menentukan *time buffer*, dengan cara menganalisa slack antara ETC, LTC dengan *completion time* di mesin empat. Pada penelitian ini perhitungan *makespan* dilakukan dengan tujuan mencari urutan job yang optimal, dengan metode FCFS yang hasilnya dapat dilihat pada tabel 4.11, EDD yang hasilnya dapat dilihat pada tabel 4.27, SPT yang hasilnya dapat dilihat pada tabel 4.43, LPT yang hasilnya dapat dilihat pada tabel 4.59 dan CR yang hasilnya dapat dilihat pada tabel 4.75. Penambahan sejumlah *Time Buffer* akan mempengaruhi *makespan* yang sangat berkaitan erat dengan performansi *duedate*. Performansi *duedate* dapat diketahui dengan menganalisa tardiness dan latenessnya, yang hasilnya dapat dilihat pada tabel 4.15, 4.31, 4.47, 4.63, 4.79.

Peningkatan performansi *duedate* melalui *TOC Aggregate* untuk menghindari ketidakpastian dari aktivitas produksi agar diperoleh performansi *duedate* yang lebih baik berdasarkan *tardiness* dan *lateness*. Algoritma yang digunakan pada *TOC Aggregate* adalah *Aggregate Time Buffer (ATB)*. Dalam pemakaian konsep *TOC Aggregate* hanya stasiun kerja yang *bottleneck* yang perlu diperhatikan. Performansi *duedate* yang lebih baik dapat diketahui dengan menghitung *tardiness* dan *lateness*nya, yang dapat dilihat pada tabel 4.25, 4.41, 4.57, 4.73, 4.89. Dari hasil analisa dengan menggunakan ATB dapat meningkatkan performansi *duedate* setiap job berdasarkan *Aggregate time buffer* pada masing-masing job yang hasilnya dilihat pada tabel 4.90.

Algoritma ATB dapat meningkatkan performansi *duedate* secara signifikan. Untuk metode FCFS terjadi peningkatan sebesar 95.522 %, metode LPT terjadi peningkatan sebesar 91.6607 %, metode CR terjadi peningkatan sebesar 37.9923 %, metode EDD terjadi peningkatan sebesar 37.9203 %, dan peningkatan SPT sebesar 35.458 %.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Setelah dilakukan pembahasan terhadap hasil penelitian, maka dapat diambil kesimpulan sebagai berikut :

1. Setelah menggunakan ATB terjadi peningkatan performansi untuk metode FCFS terjadi peningkatan sebesar 95.522 %, metode LPT terjadi peningkatan sebesar 91.6607 %, metode CR terjadi peningkatan sebesar 37.9923 %, metode EDD terjadi peningkatan sebesar 37.9203 %, dan peningkatan SPT sebesar 35.458 %.
2. Berdasarkan peningkatan performansi *duedate* metode yang tepat adalah metode FCFS dengan urutan job adalah job 1, job 2, job 3, job 4, job 5, job 6, job 7, dan job 8. Sedangkan estimasi *duedate* untuk job 1 sebesar 29.54082063 menit, job 2 sebesar 32.1449856 menit, job 3 sebesar 36.34898259 menit, job 4 sebesar 36.80012794 menit, job 5 sebesar 39.10720826 menit, job 6 sebesar 42.0226548 menit, job 7 sebesar 46.23114966 menit, dan job 8 sebesar 48.87564718 menit.

6.2 Saran

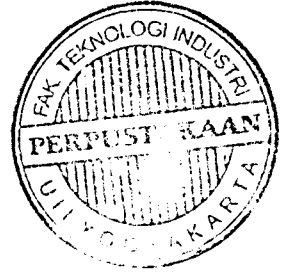
Perlu dilakukan penelitian lebih lanjut untuk mengembangkan penelitian ini dengan metode-metode lain sebagai pembanding. Misalkan dengan *Genetic Algorithm*, *Simulated Annealing*, *Tabu Search* atau *Imun Algorithm*.

DAFTAR PUSTAKA

- Baker, Kenneth R (1974). *Introduction to Sequencing and Scheduling*. John Wiley & Son
- Gatot, Aditya (2006). *Penjadwalaan Mesin Bottleneck dengan Pendekatan Algoritma Genetik*, Jurusan Teknik Industri, Fakultas Teknologi Industri, Universitas Islam Indonesia.
- Goldratt, E. M., dan Cox Jeff (1984). *The Goal*. Croton on Hudson, NY: North RiverPress Inc.
- Kusiak Andrew (1991). *Intelligent Design and Manufacturing*, New York : John Willey & sons,.
- Leach, L. P., (1999). *Critical Chain Project Management Improves Project Performance*, Project Management Journal, June, 39-51.
- Morton E Thomas, Pentico, david W (1993). *Heuristic Scheduling System. With Applications to Production System and Project Management*. John Willey & sons, Inc
- Heizer Jay, dan Render Barry (2004). *Operations Management, New Jersey* : Pearson Education Inc.
- Ridwan AP, Muhammad, (2003). *Proceeding Seminar Nasional Teknoin*, FTI UII 2003.
- Sheng-Hung Chang, (2001). *A Study of Due Date Control By using TOC's Aggregates Buffer Approach*. Department of Industrial Engineering and Management Minghsin Institute of Technology
- Simon, J.V., et al. (1999). *Simulation Versus Sequential Scheduling of Multiple Resources Which Constrain System Throughput*. Int. J. of Prod. Res., pp.21-23, Vol.37, No.1
- Tersine, Richard J, (1994). *Principles Of Inventory and Materials Requirement Management*, Edisi IV, New Jersey : Prentice – Hall International Inc.

Umble, M. Michael, M.L Srikanth, (1998). *Synchronous Manufacturing : Principles for World Class Excellence*, Cincinnati : South Western Publishing Co.

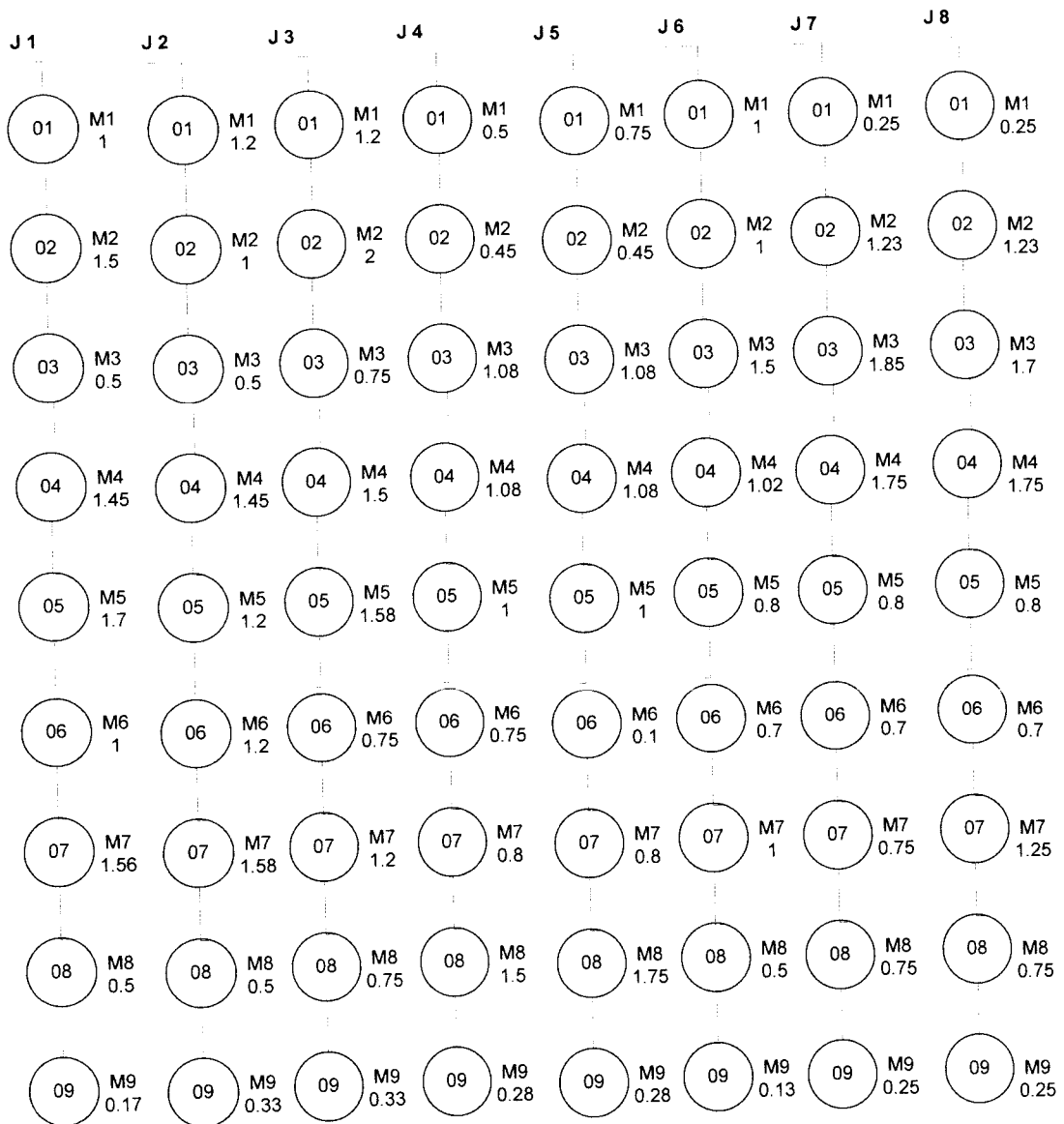
Y,-M. Tu, and R.-K, Li, (1998). *Contraint Time Buffer Determination Model*, International Journal Production Research, 36 (4), 1091-1103.



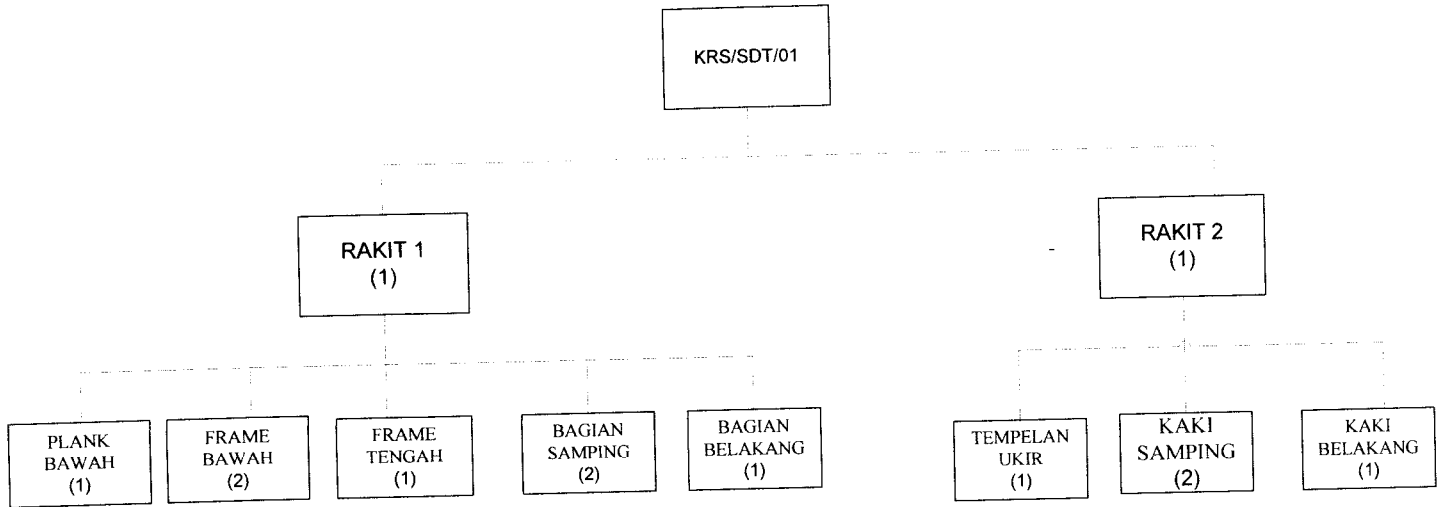
Lampiran

Peta Proses Operasi

Jenis Produk : 01/krs ukr/sdt.
No peta : 01
Dipetakan oleh : Aditya Gatot P
Tanggal Di Petakan : 10 Maret 2006



Bill of Material Produk KRS/SDT/01



Bill Of Material Produk KRS/SDT/01

No.	Nama Part	Kode Part	Per Parent (unit)
1	Plank Bawah	J1	1
2	Frame Bawah	J2	2
3	Frame Tengah	J3	1
4	Bagian Samping	J4	2
5	Bagian Belakang	J5	1
6	Tempelan Ukir	J6	1
7	Kaki Samping	J7	2
8	Kaki Belakang	J8	1