

Penerapan *Non Delay Schedule* Dengan Mempertimbangkan *Machine Availability Constraint* untuk Meminimasi *Makespan* pada PT Autotech Perkasa Mandiri, Cikarang, Jawa Barat

Anastasia Lidya Maukar¹, Ineu Widaningsih Sosodoro², Stefanus Setiady³

^{1,2,3} Faculty of Engineering, Industrial Engineering Department, President University
Jl. Ki Hajar Dewantara

Kota Jababeka, Cikarang, Bekasi - Indonesia 17550

Email: almaukar@president.ac.id, inesosodoro@yahoo.co.id, adysetiady86@gmail.com

ABSTRACT

One of the efforts to increase production efficiency is by implementing the right scheduling. PT Autotech Perkasa Mandiri is the job shop production system. Currently implements a heuristic method with a priority dispatch rule, namely the Earliest Due Date (EDD). The EDD rule has the highest priority for the job with the earliest due date to be processed on a machine. This study is to propose a more optimal scheduling by comparing the makespan of the current schedule with the proposed one. Previous research is the non delay schedule without considering machine availability constraints. In this study, machine availability constraints are taken into account, thus, they are more realistic when applied to the actual problem. The algorithm used is a non delay schedule with working time window (WT-NSA). Machine availability constraints that are taken into account are lunch-break hours (12: 00-13: 00) and shift changes (17: 00-08: 00). From the analysis, it is obtained that makespan from non delay schedule is 23 hours 6 minutes, while makespan from WT-NSA is 26 hours 6 minutes, and makespan from EDD schedule is 38 hours 54 minutes. Non-delay schedule has the smallest makespan, which is 23 hours 6 minutes, but WT-NSA is more realistic when applied to the real problem.

Keywords: Job Shop, Earliest Due Date, Non Delay Schedule, Machine Availability Constraint, Working Time Window, Makespan.

ABSTRAK

Dalam upaya meningkatkan efektifitas dan efisiensi produksi, ada beberapa cara yang dapat dilakukan, salah satunya adalah dengan menerapkan penjadwalan yang tepat. Dengan sistem produksi *job shop*, metode penjadwalan yang saat ini diterapkan PT Autotech Perkasa Mandiri adalah metode heuristik dengan *priority dispatch rule*, yaitu *Earliest Due Date* (EDD), dimana *job* yang memiliki *due date* yang paling awal merupakan *job* yang memiliki prioritas paling tinggi untuk diproses pada sebuah mesin. Tujuan dari penelitian ini adalah memberikan usulan penjadwalan yang lebih optimal dengan membandingkan *makespan* dari penjadwalan yang sekarang dengan penjadwalan yang diusulkan. Penelitian sebelumnya yaitu *non delay schedule* tanpa memperhitungkan *machine availability constraint*. Pada penelitian ini *machine availability constraint* diperhitungkan, agar lebih realistis jika diaplikasikan ke dalam permasalahan sebenarnya. Algoritma yang digunakan adalah *non delay schedule working time window* (WT-NSA). *Machine availability constraint* yang diperhitungkan adalah jam istirahat (12:00-13:00) dan pergantian *shift* (17:00-08:00). Dari analisis yang dilakukan diperoleh *makespan* dari *non delay schedule* adalah 23 jam 6 menit, sedangkan *makespan* dari WT-NSA adalah 26 jam 6 menit, dan *makespan* dari EDD *schedule* 38 jam 54 menit. *Non delay schedule* memiliki *makespan* paling kecil yaitu 23 jam 6 menit, namun WT-NSA lebih realistis apabila diaplikasikan ke dalam permasalahan sebenarnya.

Kata kunci: *Job Shop*, *Earliest Due Date*, *Non Delay Schedule*, *Machine Availability Constraint*, *Working Time Window*, *Makespan*.

1. Pendahuluan

PT Autotech Perkasa Mandiri merupakan perusahaan industri manufaktur yang memproduksi *precision part*, *jig & fixture*, *automation & dies*. Dimana barang-barang tersebut merupakan alat bantu pendukung proses produksi *customer*. Sebagai bagian dari rantai suplai industri manufaktur,

PT Autotech Perkasa Mandiri dituntut untuk menghasilkan barang dengan kualitas tinggi dengan pengiriman yang tepat waktu.

Sistem produksi di PT Autotech Perkasa Mandiri adalah *job shop*. Menurut Baker (1995) karakteristik *job shop* yaitu ada sejumlah m mesin dan sejumlah n *job*. Setiap *job* memiliki *routing* yang dapat berbeda satu sama lain, dan lebih kompleks dari *flow shop* (Champbell, Dudek, dan Smith, 1970). Selain itu, penjadwalan *job shop* adalah setiap *job* atau pesanan memiliki rute pengerjaan yang berbeda-beda, sesuai permintaan konsumen (*complex routing*) (Pinedo dan Chao, 1999). Karena kompleksnya aliran maka penjadwalan pun sangat kompleks. Aliran bersifat diskrit, dan *part* tidak bersifat multiguna (*part* yang mungkin menjadi *work-in process (WIP)* pada *job* yang satu tidak bisa digunakan pada *job* yang lain).

Bentuk sederhana dari model ini mengasumsikan bahwa setiap *job* hanya melewati satu jenis mesin sebanyak satu kali dalam rutenya pada proses tersebut. Namun ada juga model lainnya dimana setiap *job* diperbolehkan untuk melewati mesin sejenis lebih dari satu kali pada rutenya. Model ini disebut juga *job shop* dengan *recirculation* (pengulangan).

Penjadwalan produksi yang saat ini diterapkan adalah dengan metode *Earliest Due Date (EDD)*, yaitu dengan mengurutkan *job* berdasarkan *due date* dari *customer* dengan mengabaikan waktu kedatangan *job* dan total waktu proses setiap *job*. Artinya, *job* yang memiliki *due date* paling awal diantara *job* lain merupakan *job* yang memiliki prioritas tinggi untuk diproses pada sebuah mesin. Hal ini bertujuan agar meminimasi terjadinya keterlambatan pengiriman ke *customer* (Giffler dan Thompson, 1960). Data hasil observasi di lapangan mencatat bahwa *makespan* dari 11 *job* yang dijadwalkan pada minggu pertama bulan mei 2016 dengan metode *earliest due date* adalah sebesar 38 jam 54 menit.

Menurut Baker (1974), tujuan penjadwalan pada umumnya adalah meningkatkan produktifitas mesin, yaitu dengan mengurangi waktu mesin menganggur. Selain itu mengurangi persediaan barang setengah jadi dengan jalan mengurangi jumlah rata-rata pekerjaan yang menunggu dalam antrian suatu mesin karena mesin tersebut sibuk. Kemudian, mengurangi keterlambatan suatu pekerjaan, setiap pekerjaan mempunyai batas waktu (*due date*) penyelesaian, jika pekerjaan tersebut diselesaikan melewati batas waktu yang ditentukan maka pekerjaan tersebut dinyatakan terlambat. Dengan metode penjadwalan yang tepat maka keterlambatan ini dapat dikurangi, baik waktu maupun frekuensi.

Pada penelitian sebelumnya dengan judul “Usulan Penerapan *Non Delay Schedule* untuk Meminimasi *Makespan* di PT Autotech Perkasa Mandiri” (Stefanus Setiady, 2016), mencatat bahwa *makespan* dari 11 *job* yang sama, yang dijadwalkan pada minggu pertama bulan mei 2016, dengan metode *non delay schedule* adalah sebesar 23 jam 6 menit. Kemudian pada penelitian tersebut kendala ketersediaan mesin adalah tidak diperhitungkan, sehingga diasumsikan bahwa mesin selalu tersedia dan bekerja selama 8 jam per hari tanpa memperhitungkan jam istirahat dan pergantian *shift*. Menurut Baker (1974) *Non Delay Schedule (NSA)* adalah jadwal aktif dimana tidak ada mesin yang menganggur jika dapat memulai operasi tertentu, tujuannya adalah untuk meminimasi *makespan*.

Penjadwalan *job shop* dengan memperhitungkan kendala ketersediaan mesin adalah lebih kompleks dari penjadwalan *job shop* tanpa memperhitungkan kendala ketersediaan mesin. Algoritma penjadwalan *job shop* dengan memperhitungkan kendala ketersediaan mesin adalah lebih baik ketika diaplikasikan ke dalam permasalahan yang sebenarnya. Dalam sebuah hari kerja (*working day*) adalah termasuk jam kerja (*working period*) dan jam istirahat (*breaking period*). *Working period* adalah waktu dimana sebuah *job* dikerjakan pada sebuah mesin dan *breaking period* adalah waktu dimana mesin berhenti mengerjakan sebuah *job*. *Working time window* adalah algoritma yang digunakan untuk menghitung *finish time* baru apabila *breaking period* terjadi di antara waktu operasi (Ploydanai dan Mungwattana, 2010).

Pada penelitian kali ini kendala ketersediaan mesin diperhitungkan, yaitu adanya jam istirahat (*breaking period 1*) pada jam 12:00-13:00 (1 jam) dan pergantian *shift* (*breaking period 2*) pada jam 17:00-08:00 hari berikutnya. Dengan demikian diperlukan algoritma baru untuk menyelesaikan permasalahan penjadwalan *job shop* dengan memperhitungkan kendala ketersediaan mesin untuk meminimasi *makespan* yang dapat diaplikasikan ke dalam permasalahan yang sebenarnya. Penelitian kali ini akan membandingkan *makespan* dan utilisasi mesin antara penjadwalan EDD, penjadwalan *non delay (NSA)*, dan penjadwalan *non delay* dengan *working time window (WT-NSA)*.

2. Material dan Metode

Dalam melakukan analisis, beberapa data yang diperlukan telah dikumpulkan. Data yang digunakan pada penelitian ini adalah data yang diambil dari departemen PPIC dan Produksi PT Autotech Perkasa Mandiri, berupa data *job*/pesanan, dan fasilitas produksi yang digunakan. Dalam penelitian ini akan diusulkan suatu metode pejadwalan *job shop* non-deady dan *non delay* dengan *working time window* (WT-NSA). Adapun notasi model matematika *job shop scheduling* secara umum adalah sebagai berikut :

- $f_{i,j}$ = *finish time* dari *job j* pada mesin *i*
- $t_{i,j}$ = waktu mulai dari *job j* pada mesin *i*
- $p_{i,j}$ = waktu proses dari *job j* pada mesin *i*
- C_{max} = *makespan* (*finish time* dari *job* atau operasi paling akhir).

2.1 Algoritma Non Delay Schedule

Non delay schedule merupakan jadwal aktif, dimana tidak ada mesin mengangur apabila dapat memulai operasi tertentu. Langkah pertama dari *non delay schedule* adalah menghitung seluruh waktu proses dari seluruh operasi untuk setiap *job* yang dapat dikerjakan dalam tabel iterasi. Kemudian, operasi dengan waktu proses paling kecil dan *finish time* paling awal dipilih. Lalu, operasi yang dapat dikerjakan diperbarui dalam tabel iterasi dan langkah yang sama diteruskan sampai seluruh operasi terpilih (Ploydanai dan Mungwattana, 2010).

2.2 Algoritma Working Time Window

Algoritma *working time window* digunakan untuk menghitung *finish time* baru dengan memperhitungkan kendala ketersediaan mesin. Algoritma ini tepat digunakan untuk pabrik dengan banyak mesin, banyak *job*, dengan beberapa *breaking period*. Dalam satu hari kerja terdapat jam kerja (*working period*) dan jam istirahat (*breaking period*). Jam kerja (*working period*) adalah waktu dimana *job* dikerjakan pada sebuah mesin. Jam istirahat (*breaking period*) adalah waktu dimana mesin berhenti mengerjakan sebuah *job*. Jika *breaking period* tidak terjadi di antara waktu operasi, maka *finish time* = *start time* + *processing time*. Sebaliknya, jika *breaking period* terjadi di antara waktu operasi, maka *finish time* dari operasi tersebut harus dihitung kembali (Ploydanai dan Mungwattana, 2010). Perhitungan dalam algoritma *working time window* adalah sebagai berikut :

$f_{i,j} = t_{i,j} + p_{i,j}$, dengan aturan sebagai berikut :

- $b_1 = t_{i,j} \leq \mu_{a,b} \leq f_{i,j}$ (*true*) ; sebaliknya (*false*)
- $b_2 = t_{i,j} \leq \gamma_{a,b} \leq f_{i,j}$ (*true*) ; sebaliknya (*false*)
- $b_3 = \mu_{a,b} \leq t_{i,j} \leq \gamma_{a,b}$ (1) ; sebaliknya (0)
- $b_4 = \mu_{a,b} \leq f_{i,j} \leq \gamma_{a,b}$ (1) ; sebaliknya (0)

a = nomor hari kerja

B = nomor *breaking period*

$\mu_{a,b}$ = Waktu dari sebuah mesin berhenti

$\gamma_{a,b}$ = Waktu dari sebuah mesin beroperasi kembali

Selanjutnya untuk menentukan *finish time* yang baru dapat dilakukan dengan *if then else rule* berikut ini :

- If*, $\sim b_1 \wedge b_2 \wedge b_3 \wedge \sim b_4$, *then* : $t_{i,j} = \gamma_{a,b}$ and $f_{i,j} = f_{i,j} + (f_{i,j} - \mu_{a,b})$
- Else if*, $b_1 \wedge b_2 \wedge b_3 \wedge \sim b_4$, *then* : $t_{i,j} = \gamma_{a,b}$ and $f_{i,j} = f_{i,j} + (\gamma_{a,b} - \mu_{a,b})$
- Else if*, $b_1 \wedge b_2 \wedge \sim b_3 \wedge \sim b_4$, *then* : $t_{i,j} = \mu_{a,b}$ and $f_{i,j} = f_{i,j} + (\gamma_{a,b} - \mu_{a,b})$
- Else if*, $b_1 \wedge \sim b_2 \wedge \sim b_3 \wedge b_4$, *then* : $t_{i,j} = \mu_{a,b}$ and $f_{i,j} = f_{i,j} + (\gamma_{a,b} - \mu_{a,b})$

Algoritma *working time window* merupakan algoritma rekursif, yaitu digunakan secara berulang hingga *finish time* seluruh *job* selesai dihitung.

2.3 Algoritma *Non Delay Schedule* dengan *Working Time Window*

Algoritma *non delay* dengan *working time window* (WT-NSA) adalah algoritma yang dikembangkan dari *non delay schedule* dengan mempertimbangkan kendala ketersediaan mesin, yaitu dengan algoritma *working time window* untuk menentukan *finish time* dengan *machine availability constraint*. Algoritma ini bertujuan untuk meminimasi *makespan* dan dapat diaplikasikan ke permasalahan yang nyata. Langkah pertama, algoritma menghitung *finish time* dari seluruh operasi untuk setiap *job* jika operasi tersebut siap dikerjakan. Selanjutnya, algoritma memeriksa dan menghitung ulang *finish time* dengan menggunakan algoritma *working time window*. Kemudian, operasi dengan *start time* paling awal atau paling dekat dipilih. Jika beberapa *job* memiliki *start time* yang sama, maka algoritma memilih operasi dengan *finish time* paling dekat. Setelah itu, algoritma memperbarui operasi yang siap untuk dikerjakan pada mesin. Operasi yang terpilih tercatat dalam urutan untuk memperbarui *finish time* saat waktu operasi berubah. Kemudian, iterasi diteruskan sampai seluruh operasi untuk setiap *job* selesai (Ploydanai dan Mungwattana, 2010).

2.4 Langkah-langkah Penelitian

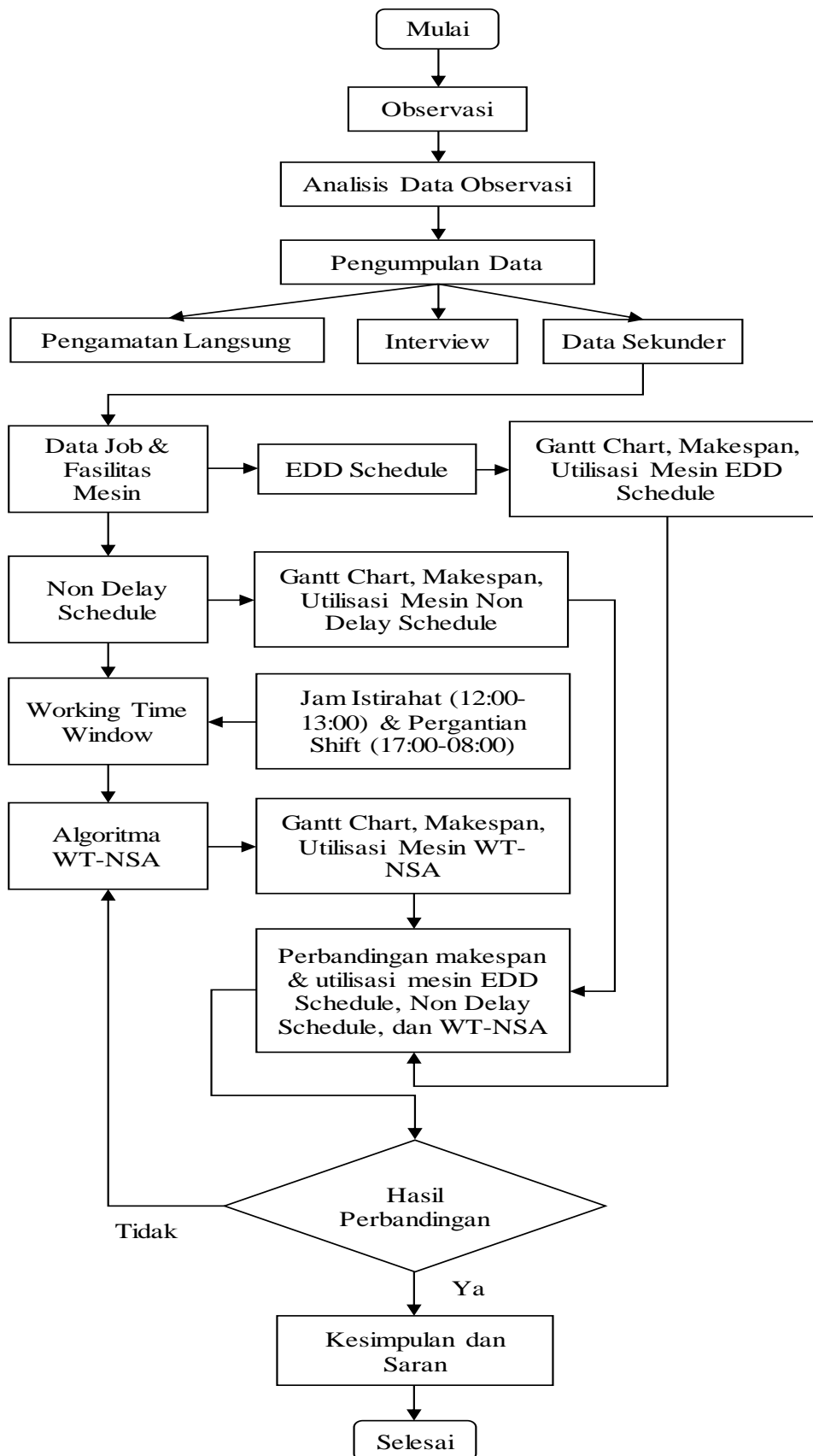
Langkah-langkah penyelesaian masalah dalam melakukan penelitian, dimana fokus dari penelitian ini adalah memperoleh usulan metode penjadwalan yang tepat untuk meminimasi *makespan*. Sebagai langkah awal, dilakukan pengambilan data yang berupa data sekunder yang diambil dari departemen terkait, yaitu PPIC. Berikut adalah data yang diambil Data *job* pada 2 Mei 2016, Fasilitas mesin yang digunakan dan *Gantt chart*, *makespan* dan utilisasi mesin dengan menggunakan penjadwalan saat ini (*EDD schedule*). Adapun utilisasi mesin adalah perbandingan antara waktu yang digunakan dalam menyelesaikan seluruh *job* dengan waktu yang tersedia (jam kerja).

Kemudian dengan data *job* dan fasilitas mesin yang sama dilakukan analisis dan pengolahan data dengan metode penjadwalan *non delay*. Pada tahap ini dilakukan pengolahan data sesuai dengan algoritma penjadwalan *non delay*. Kemudian hasil dari hasil penjadwalan *non delay*, disusunlah *Gantt chart* dan diperoleh *makespan* dan utilisasi mesin dari penjadwalan *non delay* untuk *job* yang masuk pada 2 Mei 2016.

Selanjutnya, dengan data *job* dan fasilitas mesin yang sama dilakukan penjadwalan dengan algoritma WT-NSA, dimana jam istirahat (12:00-13:00) dan pergantian *shift* (17:00-08:00) diperhitungkan dalam penjadwalan *non delay* agar lebih realistic dengan menggunakan algoritma WT-NSA. Kemudian disusunlah *Gantt chart*, perhitungan *makespan* dan utilisasi dari algoritma WT-NSA ini.

Pada tahap selanjutnya dilakukan perbandingan antara *makespan* dan utilisasi mesin dari *EDD schedule*, *non delay schedule*, dan algoritma WT-NSA yang memperhitungkan *machine availability constraint*. Dimana hasil perbandingan yang dilakukan akan menjadi hasil dari penelitian yang dilakukan antara kondisi saat ini dengan metode yang diusulkan.

Tahap ini merupakan tahap dimana usulan yang dilakukan dalam penelitian ini dapat diterima atau ditolak. Apabila *makespan* dari penjadwalan *non delay* lebih tinggi dari *makespan* dengan penjadwalan sebelumnya, maka penelitian ini perlu ditinjau ulang (mengganti dengan metode lain), sebaliknya apabila *makespan* dari *non delay* lebih kecil, maka usulan dari penelitian ini dapat dilanjutkan untuk diperiksa dan menjadi data untuk pengambilan keputusan dalam penjadwalan selanjutnya. Kemudian dengan menambahkan *time window* pada penjadwalan *non delay* tentunya akan berpengaruh dengan *makespan* yang dihasilkan. Hasil yang diperoleh dapat lebih besar atau lebih kecil, namun lebih realistis. Langkah penelitian secara lengkap dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alir Langkah Penelitian

3. Hasil dan Perbandingan

Data *job* yang masuk pada minggu pertama Mei 2016 adalah 11 *job* (A-K) dengan jumlah komponen secara keseluruhan sebanyak 53 komponen (lihat Tabel 1). Kemudian, fasilitas mesin yang dimiliki adalah sebanyak 13 mesin dengan tipe dan fungsi yang berbeda-beda (lihat Tabel 2).

Tabel 1. Daftar *Job* PT Autotech Perkasa Mandiri

JOB	JOB NAME	Jumlah Part	Delivery Date
A	JIG STRENGHT FOOTREST 54P L/P, 14D L/R DAN 2PH R/L	9	17-May-16
B	INSPECTION JIG P/N BG7-F1288-30	5	24-May-16
C	INSPECTION JIG P/N BG7-F1288-40	4	24-May-16
D	INSPECTION JIG P/N BG7-F1288-50	5	24-May-16
E	INSPECTION JIG P/N BG7-F1288-60	2	24-May-16
F	JIG Y0818-E3-K03-3	10	25-May-16
G	JIG SPOT WELDING GA240-18380 OP 1	7	25-May-16
H	INSPECTION JIG P/N 50179-K56A-N001	2	9-May-16
I	INSPECTION JIG P/N 50176-K56A-N001	2	10-May-16
J	INSPECTION JIG P/N 50171-K56A-N002	2	16-May-16
K	INSPECTION JIG P/N 50174-K56A-N002	5	18-May-16

Tabel 2. Daftar Fasilitas Mesin PT Autotech Perkasa Mandiri

No	Nama Mesin	Spesifikasi	Brand	Jumlah	No. Mesin
1	Milling Manual	300x1200	Hitachi	4	ML 01-ML 04
2	Lathe	250x1000	Kern	1	LT 01
3	Surface Grinding	400x800	Okamoto	1	SG 01
4	Cylinder Grinding	120x400	Standart	1	CG 01
5	CNC Milling	800x400x400	Fanuc	2	CM 01-CM 02
6	CNC Lathe	350x600	Mazak	1	CL01
7	Wire Cut	320x260x180	Fanuc	1	WC 01
8	Tapping		Krisbow	1	TP 01
9	Drilling		Krisbow	1	DL 01

3.1 EDD Schedule

EDD *schedule* merupakan metode penjadwalan yang saat ini diterapkan oleh PT Autotech Perkasa Mandiri, dimana *job* yang memiliki *due date* yang paling awal di antara *job-job* lainnya dipilih sebagai *job* yang memiliki prioritas paling tinggi untuk diproses pada sebuah mesin. Berikut pada Tabel 3.3 dapat kita lihat bahwa *Job H* merupakan *job* dengan *due date* paling awal di antara *job* lainnya yaitu 9 Mei 2016, maka *Job H* diprioritaskan untuk diproses pada mesin. Selanjutnya, *Job I* dengan *due date* paling awal setelah *Job H* yaitu 10 Mei 2016 diprioritaskan untuk diproses pada mesin. Tabel 3 menampilkan perhitungan *completion time* seluruh *job*, yang disusun berdasarkan aturan *earliest due date* (EDD), dimana *completion time* terbesar merupakan *makespan* dari penjadwalan seluruh *job* tersebut. Kemudian dari Tabel 3 diperoleh *part 42* pada *Job G* memiliki *completion time* paling besar, yaitu 38 jam 54 menit atau 38,9 jam. Dengan demikian dapat disimpulkan bahwa PT Autotech Perkasa Mandiri memerlukan 38 jam 54 menit untuk menyelesaikan 11 *job* dengan 53 *part* (lihat Tabel 1) dengan menggunakan 13 mesin (lihat Tabel 2). Maka *makespan* dari penjadwalan seluruh *job* tersebut adalah 38 jam 54 menit. Selain itu, *completion date* dan *due date* untuk tiap-tiap *part/job* tidak ada yang mengalami keterlambatan, karena seluruh *part* selesai sebelum *due date* yang diberikan.

3.2 Algoritma Non-Delay Schedule

Non delay schedule merupakan jadwal aktif pada sistem produksi *job shop* dengan aturan *shortest processing time* (SPT), yaitu memprioritaskan *job* dengan waktu proses paling singkat dengan ketentuan tidak ada mesin yang menganggur jika dapat memulai operasi tertentu. Tujuan dari metode ini adalah meminimasi *makespan* dan meningkatkan utilisasi mesin. Pada bab sebelumnya telah diperoleh *makespan* dari EDD *schedule* yaitu 38 jam 54 menit, maka pada bab ini akan dilakukan analisis dengan algoritma *Non Delay Schedule* dengan tujuan untuk meminimasi *makespan*

dan meningkatkan utilisasi mesin. Selanjutnya, untuk mendapatkan operasi mana yang akan dijadwalkan terlebih dahulu, maka dengan menggunakan aturan *shortest processing time* (SPT), yaitu operasi dengan waktu proses paling kecil disusunlah iterasi seperti pada Tabel 4.

Tabel 3. Completion Time dan Makespan EDD Schedule

EDD SCHEDULE					EDD SCHEDULE					
Job	Part	Completion Time (jam)	Completion Date	Due Date	Job	Part	Completion Time (jam)	Completion Date	Due Date	
H	Part 43	2,7	02-Mei-16	09-Mei-16	C	Part 17	18,9	04-Mei-16	20-Mei-16	
	Part 44	2,9	02-Mei-16			Part 18	18,2	04-Mei-16		
I	Part 45	4,2	02-Mei-16	10-Mei-16	D	Part 19	19,9	04-Mei-16	20-Mei-16	
	Part 46	11,7	03-Mei-16			Part 20	20,4	04-Mei-16		
A	Part 1	11,9	03-Mei-16	13-Mei-16		Part 21	20,6	04-Mei-16		23-Mei-16
	Part 2	12,1	03-Mei-16			Part 22	20,9	04-Mei-16		
	Part 3	12,3	03-Mei-16		Part 23	21,2	04-Mei-16			
	Part 4	15,2	03-Mei-16		E	Part 24	22,9	04-Mei-16	20-Mei-16	
	Part 5	15,7	03-Mei-16			Part 25	22,7	04-Mei-16		
	Part 6	16,2	04-Mei-16		F	Part 26	26,2	05-Mei-16	23-Mei-16	
	Part 7	16,4	04-Mei-16			Part 27	27,2	05-Mei-16		
	Part 8	8,50	03-Mei-16			Part 28	28,7	05-Mei-16		
Part 9	7,4	02-Mei-16	Part 29	28,9		05-Mei-16				
J	Part 47	8,4	03-Mei-16	16-Mei-16	Part 30	30,2	05-Mei-16	23-Mei-16		
	Part 48	8,6	03-Mei-16		Part 31	31,9	05-Mei-16			
K	Part 49	16,6	04-Mei-16	18-Mei-16	Part 32	31,2	05-Mei-16			
	Part 50	16,8	04-Mei-16		Part 33	32,9	06-Mei-16			
	Part 51	13,7	03-Mei-16		Part 34	33,4	06-Mei-16			
	Part 52	14,7	03-Mei-16		Part 35	33,9	06-Mei-16			
	Part 53	15,7	03-Mei-16		G	Part 36	35,4		06-Mei-16	23-Mei-16
B	Part 10	17	04-Mei-16	Part 37		36,9	06-Mei-16			
	Part 11	17,2	04-Mei-16	Part 38		36,7	06-Mei-16			
	Part 12	17,4	04-Mei-16	Part 39		37,2	06-Mei-16			
	Part 13	17,6	04-Mei-16	Part 40		37,6	06-Mei-16			
C	Part 14	17,2	04-Mei-16	20-Mei-16	Part 41	37,7	06-Mei-16			
	Part 15	18,4	04-Mei-16		Part 42	38,9	06-Mei-16			
	Part 16	18,6	04-Mei-16							

Tabel 4 merupakan iterasi 1 dari *non delay schedule*, pada iterasi 1 *routing* operasi *j* 1 tiap-tiap *part* dimasukkan pada tabel, kemudian waktu proses tiap *part* dijumlahkan dengan waktu awal dimulainya operasi pada mesin tersebut ($r_j = C_j + t_j$), selanjutnya dipilihlah r_j yang terkecil. Pada Tabel 3.4 yang ditandai dengan warna merah adalah operasi yang terpilih. *Part* yang dapat memulai proses pada iterasi 1 yaitu *part* 9, 16, 17, 20, 21, dan 32. Kemudian pada iterasi berikutnya, operasi yang terpilih di iterasi 1 digantikan oleh operasi selanjutnya pada *part* tersebut. Selanjutnya proses yang sama pada iterasi 1 dilakukan, yaitu dengan memilih r_j terkecil. Kemudian proses diulang hingga operasi seluruh *part* terpilih.

Tabel 4. Iterasi 1 Non Delay Schedule

Stage	St	Cj	tj	rj	c*	m*	PSt
1	(1,1,1)	0	1	1	0,2	2	(9,1,2)
	(2,1,1)	0	1	1	0,5	ML01	(16,1,1)
	(3,1,1)	0	1	1	0,5	ML02	(17,1,1)
	(4,1,1)	0	1,5	1,5	0,5	ML03	(20,1,1)
	(5,1,1)	0	1,5	1,5	0,5	ML04	(21,1,1)
	(6,1,1)	0	1	1	1	6	(32,1,6)
	(7,1,2)	0	1	1			
	(8,1,2)	0	2	2			
	(9,1,2)	0	0,2	0,2			
	(10,1,1)	0	1	1			
	(11,1,1)	0	1	1			
	(12,1,1)	0	1	1			
	(13,1,1)	0	1	1			
	(14,1,2)	0	1	1			
	(15,1,1)	0	1	1			
	(16,1,1)	0	0,5	0,5			
	(17,1,1)	0	0,5	0,5			
	(18,1,2)	0	1	1			
	(19,1,1)	0	1	1			
	(20,1,1)	0	0,5	0,5			
	(21,1,1)	0	0,5	0,5			
	(22,1,1)	0	0,5	0,5			
	(23,1,2)	0	1	1			
	(24,1,1)	0	1	1			
	(25,1,1)	0	0,5	0,5			
	(26,1,1)	0	1	1			
	(27,1,2)	0	1	1			

Stage	St	Cj	tj	rj	c*	m*	PSt
1	(28,1,1)	0	1	1			
	(29,1,1)	0	1	1			
	(30,1,2)	0	1	1			
	(31,1,2)	0	1	1			
	(32,1,6)	0	1	1			
	(33,1,2)	0	1	1			
	(34,1,6)	0	1	1			
	(35,1,6)	0	1	1			
	(36,1,1)	0	1	1			
	(37,1,1)	0	2	2			
	(38,1,1)	0	1	1			
	(39,1,1)	0	1	1			
	(40,1,2)	0	1	1			
	(41,1,1)	0	1	1			
	(42,1,2)	0	1	1			
	(43,1,1)	0	1	1			
	(44,1,1)	0	1	1			
	(45,1,1)	0	1	1			
	(46,1,1)	0	0,5	0,5			
	(47,1,1)	0	0,5	0,5			
	(48,1,1)	0	0,5	0,5			
	(49,1,6)	0	1	1			
	(50,1,6)	0	1	1			
	(51,1,2)	0	1	1			
	(52,1,2)	0	1	1			
	(53,1,2)	0	1	1			

Selanjutnya, pada Tabel 5 menunjukkan *completion time* dan *makespan* untuk *Non Delay Schedule*. Berdasarkan Tabel 5, *completion time* paling besar adalah pada *part* 49 dengan waktu 23 jam 6 menit atau 23,1 jam, dengan demikian dapat disimpulkan bahwa *makespan* dari *Non Delay Schedule* untuk seluruh *job* adalah 23 jam 6 menit.

3.3 Algoritma with Working Time Window

Working time window adalah algoritma untuk menghitung *finish time* dengan memperhitungkan kendala ketersediaan mesin (*machine availability constraint*). Pada bab sebelumnya telah diperoleh *makespan* dari *Non Delay Schedule*, yaitu sebesar 23 jam 6 menit namun tanpa memperhitungkan kendala ketersediaan mesin (*machine availability constraint*). Maka, agar lebih realistis untuk diterapkan kendala ketersediaan mesin perlu diperhitungkan. Selanjutnya, diasumsikan bahwa PT Autotech Perkasa Mandiri memiliki jam kerja dari jam 08:00-12:00 dan 13:00-17:00 (8 jam kerja) dengan jam istirahat (*breaking period* 1) dari jam 12:00-13:00 (1 jam) juga jam pergantian *shift* (*breaking period* 2) dari jam 17:00-08:00. Jam istirahat (*breaking period* 1) dan jam pergantian *shift* (*breaking period* 2) ini merupakan *working time window* yang akan diperhitungkan.

Tabel 3.5 Completion Time dan Makespan Non Delay Schedule

NON DELAY SCHEDULE					NON DELAY SCHEDULE					
Job	Part	Completion Time (jam)	Completion Date	Due Date	Job	Part	Completion Time (jam)	Completion Date	Due Date	
A	Part 1	5,4	02-Mei-16	13-Mei-16	F	Part 28	17,2	04-Mei-16	23-Mei-16	
	Part 2	8,9	03-Mei-16			Part 29	16,4	04-Mei-16		
	Part 3	2,7	02-Mei-16			Part 30	21,9	04-Mei-16		
	Part 4	21,1	04-Mei-16			Part 31	11,1	02-Mei-16		
	Part 5	22,3	04-Mei-16			Part 32	7,5	02-Mei-16		
	Part 6	6,7	02-Mei-16			Part 33	9,4	03-Mei-16		
	Part 7	2,4	02-Mei-16			Part 34	3,4	02-Mei-16		
	Part 8	16,7	04-Mei-16			Part 35	4,4	02-Mei-16		
	Part 9	0,4	02-Mei-16							
B	Part 10	9,9	03-Mei-16	20-Mei-16	G	Part 36	6,2	02-Mei-16	23-Mei-16	
	Part 11	4,2	02-Mei-16			Part 37	22,1	04-Mei-16		
	Part 12	4,6	02-Mei-16			Part 38	7,7	02-Mei-16		
	Part 13	5,2	02-Mei-16			Part 39	8,20	03-Mei-16		
	Part 14	3,20	02-Mei-16			Part 40	10,1	03-Mei-16		
C	Part 15	11,6	03-Mei-16	20-Mei-16		Part 41	6,5	02-Mei-16		09-Mei-16
	Part 16	1,2	02-Mei-16			Part 42	10,9	03-Mei-16		
	Part 17	1,4	02-Mei-16		H	Part 43	6,7	02-Mei-16		
D	Part 18	4,7	02-Mei-16	20-Mei-16	I	Part 44	7,20	02-Mei-16	10-Mei-16	
	Part 19	12,6	03-Mei-16			Part 45	8,70	03-Mei-16		
	Part 20	1,7	02-Mei-16		J	Part 46	3,2	02-Mei-16	16-Mei-16	
	Part 21	1,9	02-Mei-16			Part 47	2	02-Mei-16		
	Part 22	2,2	02-Mei-16			Part 48	2,5	02-Mei-16		
E	Part 23	5,70	02-Mei-16	20-Mei-16	K	Part 49	23,1	04-Mei-16	18-Mei-16	
	Part 24	13,6	03-Mei-16			Part 50	9,6	03-Mei-16		
F	Part 25	3	02-Mei-16	23-Mei-16		Part 51	19,7	04-Mei-16		18-Mei-16
	Part 26	18,8	04-Mei-16			Part 52	20,7	04-Mei-16		
	Part 27	15,4	03-Mei-16			Part 53	21,7	04-Mei-16		

3.4 Algoritma Non-Delay Schedule dengan Working Time Window

Pada bab sebelumnya telah diasumsikan bahwa PT Autotech Perkasa Mandiri memiliki jam kerja dari jam 08:00-12:00 dan 13:00-17:00 (8 jam kerja) dengan jam istirahat (*breaking period 1*) dari jam 12:00-13:00 (1 jam) juga jam pergantian *shift* (*breaking period 2*) dari jam 17:00-08:00, maka pada bab ini akan dilakukan analisis *Non Delay Schedule* dengan *Working Time Window*. Tujuan dari metode ini adalah meminimasi *makespan* dengan tetap memperhitungkan kendala ketersediaan mesin, sehingga dapat diaplikasikan ke dalam permasalahan yang sebenarnya. Tabel 6 merupakan tabel iterasi 5 *Non-Delay Schedule* yang terjadi *breaking period* dari operasi yang terpilih.

Tabel 6. Iterasi 5 Non Delay Schedule

Stage	St	Cj	tj	rj	c*	m*	PSt	
5	(1,2,3)	1	1	2	1	ML01	(13,1,1)	
	(2,2,3)	1	1	2	1	ML02	(15,1,1)	
	(3,3,8)	1,5	0,2	1,7	1	ML03	(19,1,1)	
	(4,1,1)	0	1,5	1,5	1	2	(23,1,2)	
	(5,1,1)	0	1,5	1,5	1	ML04	(24,1,1)	
	(6,2,3)	1	0,5	1,5	1	3	(25,2,3)	
	(7,3,8)	1,5	0,2	1,7	1	6	(50,1,6)	
	(8,1,2)	0	2	2	1,2	8	(20,3,8)	
	(10,2,3)	1	1	2	1,2	CM02	(48,3,5)	
	(11,2,5)	1	1	2	1,5	CM01	(46,2,5)	
	(12,2,5)	1	1	2	1,5	4	(18,2,4)	
	(13,1,1)	0	1	1	2	7	(14,3,7)	
	(14,3,7)	1,5	0,5	2				
	(15,1,1)	0	1	1				
	(18,2,4)	1	0,5	1,5				
	(19,1,1)	0	1	1				
	(20,3,8)	1	0,2	1,2				
	(21,3,8)	1	0,2	1,2				
	(22,3,8)	1	0,2	1,2				
	(23,1,2)	0	1	1				
	(24,1,1)	0	1	1				
	(25,2,3)	0,5	0,5	1				
	(26,1,1)	0	1	1				
	(27,1,2)	0	1	1				
	(28,1,1)	0	1	1				
	5	(29,1,1)	0	1	1			
		(30,1,2)	0	1	1			
		(31,1,2)	0	1	1			
(32,2,1)		1	0,5	1,5				
(33,1,2)		0	1	1				
(34,2,4)		1	0,5	1,5				
(35,2,4)		1	0,5	1,5				
(36,1,1)		0	1	1				
(37,1,1)		0	2	2				
(38,1,1)		0	1	1				
(39,1,1)		0	1	1				
(40,1,2)		0	1	1				
(41,1,1)		0	1	1				
(42,1,2)		0	1	1				
(43,1,1)		0	1	1				
(44,1,1)		0	1	1				
(45,1,1)		0	1	1				
(46,2,5)		0,5	1	1,5				
(48,3,5)		0,7	0,5	1,2				
(49,2,1)		1	1	2				
(50,1,6)		0	1	1				
(51,1,2)		0	1	1				
(52,1,2)		0	1	1				
(53,1,2)		0	1	1				

Berikut merupakan perhitungan *finish time* untuk operasi yang terpilih dari iterasi 5 Non Delay Schedule pada Tabel 6 :

$$\begin{aligned}
 f_{1,13} &= 10:30+01:00 = 11:30 & f_{1,24} &= 11:00+01:00 = 12:00 & f_{5,48} &= 10:00+00:30 = 10:30 \\
 f_{1,15} &= 11:00+01:00 = 12:00 & f_{3,25} &= 09:42+00:30 = 10:12 & f_{5,46} &= 10:00+01:00 = 11:00 \\
 f_{1,19} &= 11:00+01:00 = 12:00 & f_{6,50} &= 13:00+01:00 = 14:00 & f_{4,18} &= 11:12+00:30 = 11:42 \\
 f_{2,23} &= 11:12+01:00 = 12:12 & f_{8,20} &= 09:30+00:12 = 09:42 & f_{7,14} &= 10:42+00:30 = 11:12
 \end{aligned}$$

Dari hasil perhitungan di atas menunjukkan bahwa operasi yang terpilih dari iterasi 5 pada tabel 3.6 ada terjadi *breaking period*, yaitu pada operasi $f_{2,23}$ (23,1,2). Berikut adalah perhitungan *finish time* baru untuk operasi $f_{2,23}$ (23,1,2) :

$$\begin{aligned}
 b_1 &= t_{2,23} \leq \mu_{1,1} \leq f_{2,23} & b_1 \wedge \neg b_2 \wedge \neg b_3 \wedge b_4, & \text{ maka : } t_{i,j} = \mu_{\alpha,B} \text{ dan } f_{i,j} = f_{i,j} + (\gamma_{\alpha,B} - \mu_{\alpha,B}) \\
 b_1 &= 11:12 \leq 12:00 \leq 12:12 \text{ (true)} & t_{2,23} &= 13:00 \text{ dan } f_{2,23} = 12:12 + (13:00 - 12:00) \\
 b_2 &= t_{2,23} \leq \gamma_{1,1} \leq f_{2,23} & t_{2,23} &= 12:00 \text{ dan } f_{2,23} = 13:12 \\
 b_2 &= 11:12 \leq 13:00 \leq 12:12 \text{ (false)} & & \\
 b_3 &= \mu_{1,1} \leq t_{2,23} \leq \gamma_{1,1} & & \\
 b_3 &= 12:00 \leq 11:12 \leq 13:00 \text{ (0)} & & \\
 b_4 &= \mu_{1,1} \leq f_{2,23} \leq \gamma_{1,1} & & \\
 b_4 &= 12:00 \leq 12:12 \leq 13:00 \text{ (1)} & &
 \end{aligned}$$

Dari perhitungan di atas diperoleh *finish time* baru untuk operasi (23,1,2) yaitu 13:12. Selanjutnya, perhitungan *finish time* dilakukan hingga seluruh operasi terjadwalkan, dengan pedoman, apabila tidak terjadi *breaking period* maka $f_{i,j} = t_{i,j} + p_{i,j}$, lalu apabila terjadi *breaking period* maka perhitungan *finish time* sama seperti dengan operasi (23,1,2) di atas. Berikut pada Tabel 7 menunjukkan *completion time* dan *makespan* untuk *Non-Delay Schedule* dengan *Working Time Window*.

Tabel 7. *Completion Time* dan *Makespan Non-Delay Schedule* dengan *Working Time Window*

WT-NSA					WT-NSA					
Job	Part	Completion Time (jam)	Completion Date	Due Date	Job	Part	Completion Time (jam)	Completion Date	Due Date	
A	Part 1	6,4	02-Mei-16	13-Mei-16	F	Part 28	19,2	04-Mei-16	23-Mei-16	
	Part 2	9,9	03-Mei-16			Part 29	18,4	04-Mei-16		
	Part 3	2,7	02-Mei-16			Part 30	24,9	04-Mei-16		
	Part 4	24,1	04-Mei-16			Part 31	12,1	03-Mei-16		
	Part 5	25,3	04-Mei-16			Part 32	8,5	02-Mei-16		
	Part 6	7,7	02-Mei-16			Part 33	10,4	03-Mei-16		
	Part 7	2,4	02-Mei-16			Part 34	3,4	02-Mei-16		
	Part 8	18,7	04-Mei-16			Part 35	5,4	02-Mei-16		
	Part 9	0,4	02-Mei-16							
B	Part 10	10,9	03-Mei-16	20-Mei-16	G	Part 36	7,2	02-Mei-16	23-Mei-16	
	Part 11	5,2	02-Mei-16			Part 37	25,1	04-Mei-16		
	Part 12	5,6	02-Mei-16			Part 38	8,7	02-Mei-16		
	Part 13	6,2	02-Mei-16			Part 39	9,20	03-Mei-16		
	Part 14	3,20	02-Mei-16			Part 40	11,1	03-Mei-16		
C	Part 15	12,6	03-Mei-16	20-Mei-16	H	Part 41	7,5	02-Mei-16	09-Mei-16	
	Part 16	1,2	02-Mei-16			Part 42	11,9	03-Mei-16		
	Part 17	1,4	02-Mei-16			Part 43	7,7	02-Mei-16		
D	Part 18	5,7	02-Mei-16	20-Mei-16	I	Part 44	8,20	02-Mei-16	10-Mei-16	
	Part 19	14,6	03-Mei-16			Part 45	9,70	03-Mei-16		
	Part 20	1,7	02-Mei-16		J	Part 46	3,2	02-Mei-16	16-Mei-16	
	Part 21	1,9	02-Mei-16			Part 47	2	02-Mei-16		
	Part 22	2,2	02-Mei-16			Part 48	2,5	02-Mei-16		
E	Part 23	6,70	02-Mei-16	20-Mei-16	K	Part 49	26,1	04-Mei-16	18-Mei-16	
	Part 24	15,6	03-Mei-16			Part 50	10,6	03-Mei-16		
F	Part 25	3	02-Mei-16	23-Mei-16		Part 51	21,7	04-Mei-16		18-Mei-16
	Part 26	20,7	04-Mei-16			Part 52	23,7	04-Mei-16		
	Part 27	17,4	04-Mei-16			Part 53	24,7	04-Mei-16		

Dari Tabel 7 diperoleh bahwa *completion time* paling besar yaitu pada *part 49* dengan waktu 26 jam 6 menit atau 26,1 jam, dengan demikian dapat disimpulkan bahwa *makespan* dari *Non Delay Schedule* dengan *Working Time Window* untuk seluruh *job* adalah 26 jam 6 menit. *Gantt chart* dari penjadwalan *non-delay WT-NSA* dapat dilihat pada lampiran.

3.5 Perbandingan

Tujuan dari penelitian ini adalah mendapatkan usulan penjadwalan *job shop* yang optimal yang dapat diaplikasikan ke dalam permasalahan yang sebenarnya. Hasil yang dapat diukur dari analisis penjadwalan yang dilakukan adalah *makespan* dan utilisasi mesin. Untuk itu dari hasil analisis yang dilakukan dengan algoritma *Non-Delay Schedule* dan hasil analisis dengan algoritma WT-NSA akan

dibandingkan dengan kondisi aktual perusahaan. Selanjutnya, apabila hasil perbandingan dengan metode yang diusulkan lebih baik daripada kondisi aktual perusahaan, maka usulan dapat diterima untuk menjadi masukan pada penjadwalan selanjutnya. Algoritma WT-NSA merupakan algoritma yang dikembangkan dari *Non Delay Schedule*, yaitu dengan memperhitungkan *machine availability constraint* (kendala ketersediaan mesin) agar lebih realistik.

Pada sub bab 3.2 sebelumnya telah dilakukan analisis dengan metode penjadwalan *non delay*, yang kemudian diperoleh kesimpulan, bahwa untuk menyelesaikan 53 *part* dari 11 *job* memerlukan *makespan* 23 jam 6 menit atau 23,1 jam. Selanjutnya pada bab 3.4 juga telah dilakukan analisis dengan algoritma WT-NSA, yang juga diperoleh kesimpulan, bahwa untuk menyelesaikan *job* yang sama dengan memperhitungkan jam istirahat (12:00-13:00) dan pergantian *shift* (17:00-08:00) memerlukan *makespan* 26 jam 6 menit. Kemudian hasil dari analisis dengan penjadwalan *non delay* dan WT-NSA dibandingkan dengan kondisi aktual perusahaan, yaitu membandingkan *makespan*. Hal ini dilakukan untuk mengetahui apakah usulan dari penerapan metode penjadwalan *non delay* dengan *working time window* (WT-NSA) dapat diterima untuk penjadwalan selanjutnya.

Setelah dilakukan analisis dan perhitungan, ternyata *makespan* dari penjadwalan *non delay* dengan *working time window* (WT-NSA) lebih kecil dari *makespan* dari kondisi aktual perusahaan. Hal tersebut didukung dan dapat dilihat pada bab 3.2 dan 3.4 dimana *makespan* dari penjadwalan *non delay* adalah 23,1 jam, dan *makespan* dari algoritma WT-NSA adalah 26,1 jam, sedangkan pada bab 3.1 kita peroleh *makespan* dari kondisi aktual perusahaan adalah 38,9 jam. Maka, algoritma *non delay schedule* dengan *working time window* (WT-NSA) dapat diterima untuk penjadwalan berikutnya karena dapat meminimasi *makespan* dan lebih realistik apabila diaplikasikan dalam permasalahan yang sebenarnya, karena memperhitungkan *machine availability constraint*, yaitu jam istirahat (12:00-13:00) dan pergantian *shift* (17:00-08:00). Berikut pada Tabel 8 menyajikan perbandingan *makespan* dari analisis yang telah dilakukan.

Tabel 8. Perbandingan *Makespan*

No.	Penjadwalan	Makespan
1	EDD	38 Jam 54 Menit
2	NSA	23 Jam 6 Menit
3	WT-NSA	26 Jam 6 Menit

Berdasarkan Tabel 8, *makespan* dari penjadwalan EDD adalah 38 jam 54 menit, sedangkan *makespan* dari penjadwalan *non delay* adalah 23 jam 6 menit, dan *makespan* dari penjadwalan *non delay* dengan *working time window* adalah 26 jam 6 menit. *Makespan* dari penjadwalan *non delay* merupakan hasil paling kecil, namun demikian tidak realistik karena tidak memperhitungkan kendala ketersediaan mesin. Sedangkan *makespan* dari penjadwalan *non delay* dengan *working time window* lebih kecil dari penjadwalan EDD dan realistik karena memperhitungkan kendala ketersediaan mesin. Selain itu, akan dilakukan perbandingan antara utilisasi mesin. Tabel 9 menyajikan perbandingan utilisasi mesin dari ketiga metode.

Tabel 9. Perbandingan Utilisasi Mesin

Machine	Utilisasi			Machine	Utilisasi		
	EDD	NSA	WT-NSA		EDD	NSA	WT-NSA
ML01	20%	37%	33%	CM01	29%	52%	25%
ML02	20%	35%	31%	CM02	29%	28%	29%
ML03	23%	33%	29%	CL01	10%	18%	16%
ML04	16%	33%	29%	WC01	7%	13%	12%
LT01	40%	68%	61%	TP01	11%	23%	20%
SG01	51%	61%	58%	DL01	10%	18%	16%
CG01	25%	44%	39%				

Dari Tabel 9 diperoleh bahwa utilisasi mesin dengan *non delay schedule* maupun *non delay schedule* dengan *working time window* lebih tinggi dibandingkan dengan *EDD schedule*.

4. Kesimpulan

Berdasarkan analisis dan pengolahan data yang telah dilakukan, maka diperoleh kesimpulan, yaitu algoritma *non delay schedule* dengan *working time window* dapat meminimasi *makespan* 15 jam 48 menit dan

meningkatkan rata-rata utilisasi mesin sebesar 13%. Selain itu, algoritma *non delay schedule* dengan *working time window* lebih realistis untuk diterapkan karena memperhitungkan kendala ketersediaan mesin, yaitu jam istirahat (12:00 - 13:00) dan pergantian *shift* (17:00 - 08:00).

Daftar Pustaka

1. Baker, K. R. (1974). Introduction to Sequencing and Scheduling, John Wiley, NY.
2. Baker, K. R. (1995). Element of Sequencing and Scheduling, Kenneth R. Baker, Tuck School of Bussiness Administration, Dartmouth College, Hanover, NH.
3. Campbell, H. G., Dudek, R. A., and Smith, M. L. (1970). A Heuristic Algorithm for the n Job m Machine Sequencing Problem, Management Science, Vol.16, pp. B630-B637.
4. Giffler, B., and Thompson, G. L. (1960). Algorithms for Solving Production Scheduling Problems, Operation Research, Vol. 8, pp. 487-503.
5. Pinedo, M., dan Chao, X.,(1992). A Parametric Adjustment Method for Dispatching, Technical Report, Department of Industrial Engineering adn Operation Research, Columbia University, New York, NY.
6. Pinedo, M., and Chao, X. (1999). Operations Scheduling with Applications in Manufacturing and Services, Irwin / McGraw-Hill, Burr Ridge, Illinois.
7. Ploydanai, K., and Mungwattana, A. (2010). Algorithm for Solving Job Shop Scheduling Problem Based on Machine Availability Constraint, Department of Industrial Engineering, Faculty of Engineering, Kasetsart University, Bangkok, Thailand.
8. Setiady, S. (2016). Usulan Penerapan Non Delay Schedule untuk Meminimasi Makespan di PT Autotech Perkasa Mandiri, Program Studi Teknik Industri, Fakultas Teknik, President University, Cikarang, Indonesia.

LAMPIRAN: GANTT CHART PENJADWALAN NON-DELAY WT-NSA

