

Penjadwalan Produk *Painted* di PT. ABC Dengan Algoritma *Branch and Bound & Neighborhood Search* Untuk Meminimasi *Mean Flow Time*

Ary Kurniati¹, Lely Herlina², Bobby Kurniawan³

^{1, 2, 3}Jurusan Teknik Industri Universitas Sultan Ageng Tirtayasa

Arykurniati_091336@yahoo.com¹, lelyherlina@yahoo.com², b.kurniawan76@gmail.com³

ABSTRAK

PT. ABC adalah perusahaan manufaktur yang bergerak pada pelapisan metal (ZINCALUME®) dan pelapisan cat (COLORBOND®). Pembuatan kedua produk tersebut melewati plan produksi yang dinamakan Metal Coating Line 2 (MCL 2), produk pelapisan cat atau *painted* yang paling banyak diminta oleh konsumen khususnya di MCL 2. Sehingga dibutuhkan waktu pengerjaan produk yang cepat dengan cara menjadwalkan produk secara tepat. Latar belakang penelitian adalah PT. ABC belum dapat memenuhi semua order konsumen dikarenakan mesin MCL 2 masih bersifat baru sehingga menyebabkan *bottleneck* pada proses *welder* dan *surface* khususnya produk *painted* maka diperlukan penjadwalan yang tepat. Metode penelitian yang digunakan adalah algoritma *branch and bound* dan *neighborhood search*. Tujuan penelitian untuk meminimasi *mean flow time* pada MCL 2 dengan menggunakan algoritma *branch and bound* dan *neighborhood search*, yang akan dibandingkan dengan penjadwalan eksisting pada perusahaan. Kondisi penjadwalan eksisting pada perusahaan yaitu FCFS (First Come First Served) dimana order yang datang terlebih dahulu, akan diproses pada plan produksi. Pada penjadwalan ini, dibagi menjadi dua batch dengan dimensi yang berbeda-beda, dikarenakan pada perusahaannya, menetapkan sistem *sequencing* dengan ukuran yang lebih kecil, yang terlebih dahulu diproses. Sistem batch-nya terbagi menjadi dua dimensi yaitu batch pertama dengan dimensi 0,20 x 914mm dan batch kedua dengan dimensi 0,25 x 914mm. Hasil penelitian didapatkan, *mean flow time* pada batch 0,20 x 914mm dengan algoritma *branch and bound* dan *neighborhood search* adalah 18,42 jam dengan urutan job pada *branch and bound* adalah job 14- job 21- job 20- job 11- job 15 – job 8- job 3 –job 4- job 9- job 7, sedangkan dengan *neighborhood search*, urutan yang dihasilkan yaitu job 14- job 20- job 21- job 11- job 15 – job 8- job 3 –job 4- job 9- job 7 sedangkan nilai eksisting 43,23 jam. Sementara untuk batch 0,25 x 914mm, dengan menggunakan kedua algoritma, didapatkan nilai 147,13 jam dengan urutan yang berbeda pada masing-masing algoritmanya, dibandingkan nilai eksisting 182,76 jam. Pada algoritma *branch and bound*, urutan job nya adalah job 6- job 18- job 2- job 22- job 5- job 10- job 19- job 16- job 12- job 1- job 23- job 17- job 13. Sedangkan variasi jadwal dengan algoritma *neighborhood search* yaitu job 6- job 18- job 2- job 22- job 5- job 10- job 12- job 16- job 1- job 17- job 23- job 13. Penurunan *mean flow time* antara kondisi eksisting dengan menggunakan algoritma *branch and bound* dan *neighborhood search*, pada batch 0,20 x 914 mm dan 0,25 x 914 mm sebesar 57% dan 19%.

Kata Kunci: Algoritma *Branch And Bound*, Algoritma *Neighborhood Search*, *Mean Flow Time*, *SPT* (*Short Processing Time*), *Batch*.

PENDAHULUAN

Seiring dengan berkembangnya waktu disertai kompetensi pasar global yang semakin kuat, menimbulkan tantangan pada perusahaan manufaktur untuk beroperasi dengan biaya produksi yang rendah, *life cycle* pendek, oleh karena itu diperlukan penjadwalan. Penjadwalan menurut Kanneth R.Baker, yaitu proses pengalokasian sumber untuk memilih sekumpulan tugas dalam jangka waktu tertentu.

PT. ABC adalah perusahaan manufaktur yang bergerak di bidang pelapisan metal

(ZINCALUME®) dan pelapisan cat (COLORBOND®). Pada pembuatan produknya dilakukan pada *plant* MCL 2 (*Metal Coating Line 2*), dimana MCL 2 ini termasuk mesin yang masih bersifat baru dan prosesnya yang belum ter-*sequence* sesuai dengan dimensi yang paling tipis yang terlebih dahulu diproses sehingga mengakibatkan proses pewarnaan dan pelapisan kurang khususnya *painted* tidak terselesaikan tepat waktu sesuai yang direncanakan dan adanya *bottleneck* pada proses *welder* dan *surface*, yang mengakibatkan aliran waktu penyelesaian cukup

lama dan WIP (*Work In Process*) sehingga tidak dapat memenuhi *order customer* secara tepat waktu sehingga diperlukan penjadwalan yang tepat untuk MCL 2. Tipe produksi pada PT. ABC yaitu *make to order*, dibuat berdasarkan pesanan, dengan aliran produksinya *flow shop*. Metode penjadwalan eksisting yang digunakan yaitu FCFS (*First Come First Serve*) yaitu *order* yang datang terlebih dahulu, yang akan diproses terlebih dahulu. Berdasarkan permasalahan yang terjadi, maka peneliti melakukan penjadwalan ulang dengan menggunakan algoritma yaitu algoritma *Branch & Bound* dan *Neighborhood Search*.

Algoritma *Branch & Bound* adalah suatu prosedur paling umum untuk mencari solusi optimal seperti masalah penjadwalan. Di dalam algoritma *Branch and Bound*, terdapat tiga buah bagian utama, yaitu : ekspresi batas bawah (*Lower Bound (LB)*), strategi pencarian dan percabangan (*branching*). Di dalam prosedur ini, suatu masalah dipecah menjadi beberapa sub masalah yang merepresentasikan pembagian kerja secara parsial. Simpul-simpul terus bercabang lebih jauh sampai diperoleh solusi lengkap (Sutanto,2004).

Metode *Neighborhood Search* merupakan teknik penjadwalan dengan menentukan urutan awal suatu pengerjaan *job* lalu mengevaluasi terkait ukuran kinerja yang ditentukan yang ditandai dengan notasi T' . Jika urutan *job* belum maksimal maka dapat diperoleh melalui perubahan urutan *job*, dimana *job* ditukarkan dengan *job* di sebelahnya yang ditandai dengan notasi T'' . Apabila ukuran kinerja yang diinginkan maksimal maka proses pertukaran *job* dapat dihentikan yang ditandai dengan $T'=T''$ (Triyanto,2012).

Tujuan penelitian ini adalah membuat mekanisme penjadwalan baru pada produk *printed* di MCL 2 dengan menggunakan algoritma *Branch & Bound* dan *Neighborhood Search* untuk meminimasi *mean flow time*.

METODE PENELITIAN

Tahapan penelitian yang akan dilakukan untuk membuat variasi jadwal yang baru dengan menggunakan algoritma *Branch and Bound* untuk meminimasi *mean flow time*, sebagai berikut :

Data yang diperlukan dalam penelitian ini adalah data primer dan data sekunder. Data primer seperti wawancara yang dilakukan penulis yaitu permasalahan yang dialami oleh PT.ABC. Kemudian untuk data sekunder yang digunakan adalah data waktu proses untuk setiap

job nya pada periode maret 2013 sebanyak 23 *job*.

Tahap pertama yang dilakukan adalah perhitungan penjadwalan eksisting pada perusahaan. Dimulai dari menentukan metode yang digunakan oleh perusahaan. Kemudian dibagi menjadi dua *batch* yaitu *batch* pertama dengan dimensi 0,20 x 914mm dan *batch* kedua dengan dimensi 0,25 x 914 mm, dilakukan perhitungan untuk mencari *mean flow time* pada masing-masing *batch* nya.

Tahap kedua adalah melakukan perhitungan jadwal inisial sebagai penentu kriteria yang dicari seperti meminimasi *mean flow time*, yang bertujuan untuk mengetahui variasi jadwal yang memiliki *mean flow time* minimum dan sebagai usulan jadwal inisial untuk perhitungan algoritma *Branch and Bound* dan *Neighborhood Search* pada masing-masing *batch* nya.

Tahap ketiga adalah perhitungan penjadwalan dengan algoritma *Branch and Bound* dimana jadwal inisial yang digunakan pada masing-masing *batch*. Kemudian dilakukan perhitungan dengan algoritma *Branch and Bound* yang pertama dengan melakukan percabangan di tiap simpul nya kemudian dari percabangan tersebut, dipilih node dengan total waktu proses yang terkecil yang dijadikan sebagai *lower bound* untuk melakukan percabangan sampai semua *job* sudah terjadwalkan dan memiliki nilai *fathomed* yaitu $lower\ bound < \text{atau} = upper\ bound$. *Upper bound* didapatkan dari hasil perhitungan dengan teknik heuristik.

Tahap keempat adalah perhitungan penjadwalan dengan algoritma *neighborhood search* dimana jadwal insial yang dihitung pada tahap kedua dengan urutannya sebagai dasar perhitungan untuk algoritma untuk mencari nilai *mean flow time* yang minimum. Pada tahap ini, perhitungan dilakukan dengan cara menukarkan sepasang *job* sampai proses pertukaran berhenti yang ditandai dengan hasil $T'=T''$. Jika $T' \neq T''$ maka dilanjutkan ke *stage* berikutnya, dengan hasil variasi jadwal dari T'' yang akan dijadikan sebagai jadwal inisial untuk melakukan pertukaran *job* sampai hasil yang didapatkan $T'' = T'''$ (Triyanto, 2012).

Tahap kelima adalah perbandingan *mean flow time* pada kondisi eksisting dengan penjadwalan menggunakan kedua metode tersebut yaitu algoritma *Branch & Bound* dan *Neighborhood Search* pada masing-masing *batch* nya.

HASIL DAN PEMBAHASAN

Pengambilan data pada penelitian ini bersifat sekunder dan primer, data sekunder yaitu belum dapat memenuhi semua *order* konsumen secara

sisipan, *job* sisipan yang dilakukan seperti *breezing*, yaitu adanya proses produk *bare* pada aliran *flow shop* selama pembersihan drum untuk produk *painted*. Kemudian variasi jadwal SPT digunakan sebagai jadwal inisial untuk perhitungan algoritma *branch and bound* dan *neighborhood search*.

Langkah-langkah penjadwalan menggunakan algoritma *branch and bound*, antara lain :

1. Mencari jadwal inisial

Sebelumnya jadwal inisial telah dihitung, dan didapatkan *mean flow time* nya 18,42 jam dengan urutan *job 14- job 20- job 21- job 11- job 15 - job 8- job 3 -job 4- job 9- job 7* pada *batch 0,20 x 914mm* dan pada *batch 0,25 x 914mm* senilai 148,45 dengan urutan *job* nya yaitu *job 6- job 20- job 21- job 11- job 15 - job 8- job 3 -job 4- job 9- job 7*.

2. Inisialisasi Parameter

Tempatkan P(0) pada list yang aktif. Nilai yang terkait pada node ini adalah $V_0 = 0$ dan $p(\Phi) = \sum_{j=1}^n P_j$. Dikarenakan nilai pada algoritma *branch and bound*, menggunakan aturan *integer programming* yaitu 1,0. Yang artinya untuk mesin dalam posisi diam, diberi kode 0 dan ketika mesin mengerjakan suatu *job* diberi tanda 1.

P(0) = Waktu proses pada *job* pada mesin dalam keadaan diam, belum berproses.

$V_0 = value$ pada P(0) dalam keadaan mesin diam

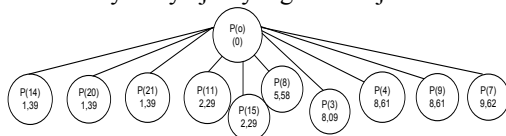


Gambar 1. Node Pertama untuk *batch 0,20 x 914 mm*

3. Membuat percabangan $X_1=1$ untuk masing-masing *job* i. Ketika mesin mengerjakan suatu *job* diberi tanda 1, P(s), dari list yang aktif dengan node baru untuk masing-masing *job* yang belum terjadwalkan. Kemudian notasikan K sebagai nomor *job* dalam urutan parsial s. Jika $k=n$, stop: yang berarti urutan penjadwalan sudah optimal. Jika belum optimal, dilanjutkan dengan langkah 3 dan langkah 4.

K= level

n= banyak nya *job* yang akan dijadwalkan.



Gambar 2. Percabangan Pertama untuk *batch 0,20 x 914 mm*

Contoh perhitungan :

$$P(s') = P(14) = \text{Job ke } 14 = 1,39,$$

$$V_{js} = V_s + p(s') ; V_{14s} = 0 + 1,39 = 1,39$$

Kemudian, dilanjutkan ke langkah selanjutnya, dikarenakan level k tidak sama dengan n, atau level 1 tidak sama dengan 10 *job*, maka belum dikatakan optimal.

4. Menentukan *lower bound*

Dilanjutkan dengan memilih *lower bound* (batas bawah). Node yang akan dipilih, yang mempunyai nilai V_s nya lebih kecil diantara yang node-node yang lain.

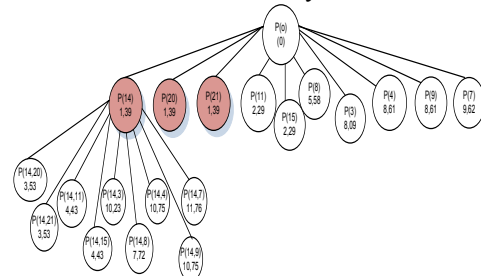


Gambar 3. *Lower Bound* Pada Percabangan Pertama untuk *batch 0,20 x 914 mm*

Dan didapatkan node yang dipilih sebagai *lower bound* (batas bawah) yaitu node dengan P (14), P(20) dan P(21) dengan masing-masing *job* ke 14, 20 dan 21. Yang kemudian dilanjutkan dengan pembuatan node baru yang dicabangkan pada node yang telah dipilih.

5. Membuat node selanjutnya

Pada node baru ini, P(S) yang akan dihitung, akan menggabungkan *job* yang belum terjadwalkan di akhir urutan parsial *job* dan atribut lainnya yang digunakan untuk membuat node baru dan memasukkan mereka sesuai urutan listnya.



Gambar 4. Percabangan Kedua untuk *batch 0,20 x 914 mm*

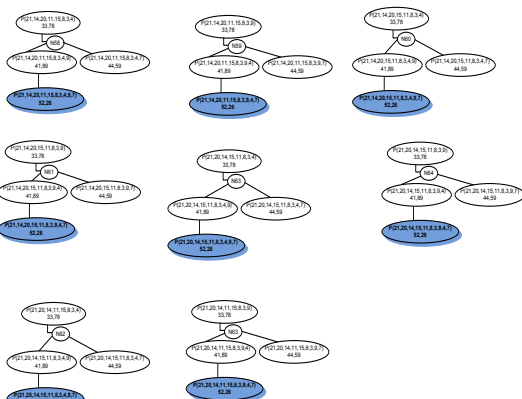
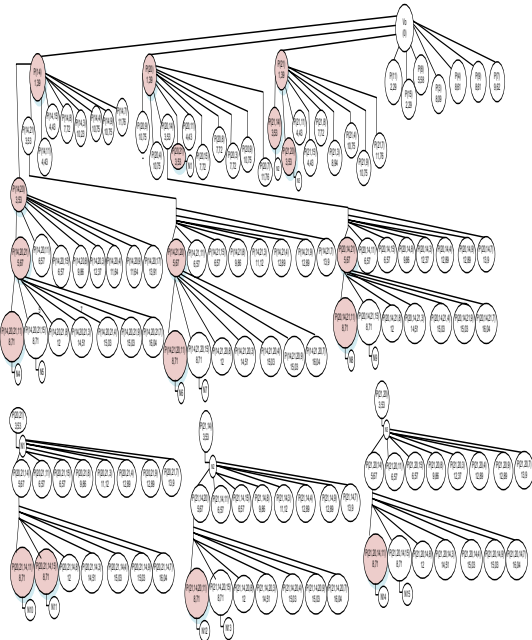
Contoh perhitungan :

$$P(s') = P(14,20) = \text{Job ke } 14 = 1,39, \text{ job ke } 20 = 1,39$$

$V_{js} = V_s + p(s') ; V_{14,20s} = 0 + 1,39 + 0,75 + 1,39 = 3,53$; dalam hal ini, disertai *job* sisipan dengan waktu 0,75 menit dikarenakan *job* 14 dan *job* 20 pekerjaan dengan warna yang berbeda sehingga diperlukan *job* sisipan dan dihasilkan variasi jadwal dengan total *makespan* yang terkecil, yang kemudian akan dicari nilai *mean flow time* yang minimum yang akan dipilih.

Berikut dibawah ini model hasil perhitungan dengan menggunakan *branch and bound* pada masing-masing *batch* :

a. *Branch and Bound Solution* pada *batch* 1 dengan dimensi panjang dan tebalnya yaitu 0,20 x 914 mm :



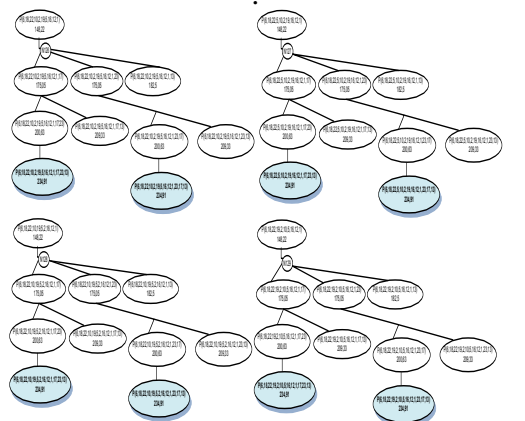
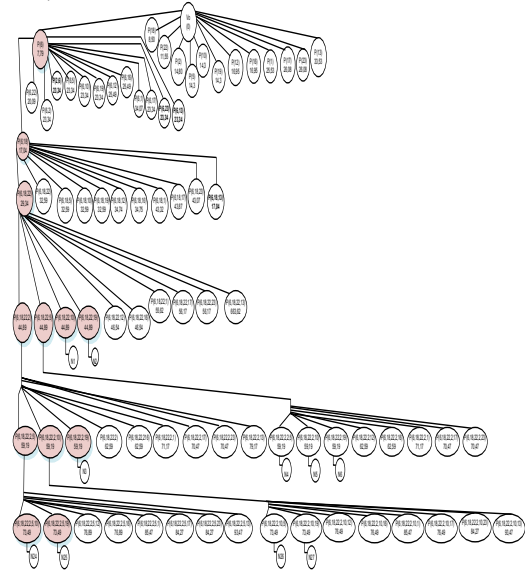
Gambar 5. Struktur *Tree* untuk *batch* kedua 0,20 x 914 mm

Tabel 6. Variasi Jadwal *Batch* 0,20x914mm Untuk Keseluruhan Jadwal

No	Jadwal	Nilai Mean Flow time (jam)
1	S1	18,42
2	S2	18,42
3	S3	18,43
4	S4	19,10
5	S5	18,42
6	S6	18,42
7	S7	18,42
8	S8	18,62
9	S9	18,42
10	S10	18,42
11	S11	18,42
12	S12	18,42
13	S13	18,42
14	S14	18,42
15	S15	18,42
16	S16	18,42
17	S17	18,42
18	S18	18,42
19	S19	18,42

Dan didapatkan nilai *mean flow time* terkecil diantara 10 *sequencing* adalah *sequence* ke 1 dengan nilai 18,42 dengan *sequencing* yang dihasilkan 14-20-21-11-15-8-3-4-9-7.

b. *Branch and Bound Solution* pada *batch* 1 dengan dimensi panjang dan tebalnya yaitu 0,25 x 914 mm :



Gambar 6. Struktur *Tree* untuk *batch* kedua 0,25 x 914 mm

Tabel 7. Variasi Jadwal *Batch* 0,20x914mm Untuk Keseluruhan Jadwal

No	Jadwal	Nilai Mean Flow time (jam)	No	Jadwal	Nilai Mean Flow time (jam)	No	Jadwal	Nilai Mean Flow time (jam)	No	Jadwal	Nilai Mean Flow time (jam)
1	S1	148,45	19	S19	148,45	37	S37	148,45	55	S55	148,45
2	S2	149,16	20	S20	148,45	38	S38	148,45	56	S56	148,45
3	S3	148,45	21	S21	148,45	39	S39	148,45	57	S57	148,45
4	S4	148,45	22	S22	148,45	40	S40	148,45	58	S58	148,45
5	S5	148,45	23	S23	148,45	41	S41	148,45	59	S59	148,45
6	S6	148,45	24	S24	148,45	42	S42	148,45	60	S60	148,45
7	S7	148,45	25	S25	148,45	43	S43	148,45	61	S61	148,45
8	S8	148,45	26	S26	148,45	44	S44	148,45	62	S62	148,45
9	S9	148,45	27	S27	148,45	45	S45	148,45	63	S63	148,45
10	S10	148,45	28	S28	148,45	46	S46	148,45	64	S64	148,45
11	S11	151,03	29	S29	148,45	47	S47	148,45	65	S65	148,45
12	S12	147,13	30	S30	148,45	48	S48	148,45	66	S66	148,45
13	S13	148,45	31	S31	148,45	49	S49	148,45	67	S67	148,45
14	S14	148,45	32	S32	148,45	50	S50	148,45	68	S68	148,45
15	S15	148,45	33	S33	148,45	51	S51	148,45	69	S69	148,45
16	S16	148,45	34	S34	148,45	52	S52	148,45	70	S70	148,45
17	S17	148,45	35	S35	148,45	53	S53	148,45	71	S71	148,45
18	S18	148,45	36	S36	148,45	54	S54	148,45			

Didapatkan nilai *mean flow time* terkecil diantara 71 *sequencing* adalah *sequence* ke 12 dengan nilai 147,13 jam dengan *sequencing* yang dihasilkan 6-18-2-22-5-10-19-16-12-1-23-17-13. Kemudian melakukan perhitungan dengan *neighborhood search*, Langkah-langkah yang dilakukan, antara lain :

1. Penentuan posisi *stage* penetapan nilai kriteria awal ($s=1$).

Dengan jadwal inisial SPT untuk melakukan perhitungan. Pada *stage* (s) = 1, $T' = 18,42$ jam, $S' = 14-20-21-11-15-8-3-4-9-7$.

2. Proses Pertukaran *Job*

Seluruh *job* akan dipertukarkan dengan ketentuan ($n-1$) yang dapat diilustrasikan sebagai berikut:

$$N = n - 1$$

Dimana : N = posisi *job* yang akan dipertukarkan

$$n = \text{posisi job awal}$$

untuk *job* 10 ($n = 1$)

Urutan Jadwal Awal (S') = 14-20-21-11-15-8-3-4-9-7

Pertukaran *job* ke 1 (*job* 10) dengan $n = 1$

$N = n - 1$, $1 - 1 = 0$. *Job* ke 1 ditukar dengan *job* ke 0 (tidak terjadi pertukaran).

Variasi jadwal dihasilkan (S1) pada *stage* 1 = (14-20-21-11-15-8-3-4-9-7). Proses penukaran akan terus berlanjut hingga *job* urutan ke 10. Selam pergantian warna diperlukan *set up* selama 0,5 jam dan *job* sisipan selam 45 menit. Dan berikut dibawah ini tabel perhitungan S1 :

Tabel 8. Variasi Jadwal S1 Pada batch 0,20 x 914mm (Stage 1)

No	No Job	Item Description	Total Runtime	Start Time	Finish Time	Completion Time
1	14	0.20 x 914mm Gemilang AZ100 G300 Steel Merah Merapi DB NIR	1,39	0	1,39	1,39
2	J.S	Job Sisipan	0,75	1,39	2,14	2,14
3	20	0.20 x 914mm Gemilang AZ100 G300 Steel Hijau Bromo DB NIR	1,39	2,14	3,54	3,54
4	J.S	Job Sisipan	0,75	3,54	4,29	4,29
5	21	0.20 x 914mm Gemilang AZ100 G300 Steel Biru Bromo DB NIR	1,39	4,29	5,68	5,68
6	J.S	Job Sisipan	0,75	5,68	6,43	6,43
7	11	0.20 x 914mm Gemilang AZ100 G300 Steel Merah Carita DB NIR	2,29	6,43	8,72	8,72
8	15	0.20 x 914mm Gemilang AZ100 G300 Steel Merah Carita DB NIR	1,79	8,72	10,50	10,50
9	8	0.20 x 914mm Gemilang SS AZ100 G300 SI Merah Carita DB NIR	5,08	10,50	15,58	15,58
10	J.S	Job Sisipan	0,75	15,58	16,33	16,33
11	3	0.20 x 914mm Gemilang SS AZ100 G300 Steel Biru Bromo DB NIR	8,09	16,33	24,42	24,42
12	J.S	Job Sisipan	0,75	24,42	25,17	25,17
13	4	0.20 x 914mm Gemilang SS AZ100 G300 SI Hijau Bromo DB NIR	8,61	25,17	33,78	33,78
14	9	0.20 x 914mm Gemilang SS AZ100 G300 SI Hijau Bromo DB NIR	8,11	33,78	41,90	41,90
15	J.S	Job Sisipan	0,75	41,90	42,65	42,65
16	7	0.20 x 914mm Gemilang SS AZ100 G300 SI Merah Merapi DB NIR	9,62	42,65	52,27	52,27
Total Flow Time						294,78
Mean Flow Time						18,42

Start Time = 0

Finish Time = Start time + processing time = 0 + 1,39 = 1,39

Completion Time = lamanya waktu pengerjaan 1 *job* dalam sistem = finish time = 1,39

Total Flow Time = \sum Completion Time = 294,78

Mean Flow Time = $\frac{\sum \text{Completion Time}}{n \text{ job}} = \frac{294,78}{16} =$

18,42

Dan dibawah ini keseleruhan *job* yang dipertukarkan :

Tabel 9. Variasi Jadwal Keseluruhan Pada Batch 0,20 x 914mm (Stage 1)

No	Jadwal	Nilai Mean Flow time (jam)
1	S1	18,42
2	S2	18,42
3	S3	18,42
4	S4	18,66
5	S5	18,42
6	S6	18,63
7	S7	18,89
8	S8	19,81
9	S9	18,42
10	S10	24,53

Nilai *mean flow time* yang terkecil didapatkan $T'' = 18,42$ pada S1. Dikarenakan nilai $T'' = T''$ yaitu hasil *mean flow time* jadwal inisial sama dengan hasil *mean flow time* dengan menggunakan variasi jadwal dari perhitungan metode *neighborhood search*, maka proses berhenti untuk *batch* 0,20 x 914 mm. Kemudian dilakukan perhitungan *neighborhood search* pada *batch* 0,25 x 914 dengan langkah-langkah yang sama dan variasi jadwal SPT yang telah dihitung sebelumnya. Dan berikut, tabel hasil perhitungan keseluruhan urutan *job* untuk *batch* 0,25x914mm pada *stage* 1:

Tabel 10. Variasi Jadwal Batch 0,25x914mm Untuk Keseluruhan (T'')

Jadwal	Nilai Mean Flow time
S1	148,45
S2	148,52
S3	148,75
S4	147,13
S5	148,45
S6	148,45
S7	148,45
S8	149,24
S9	148,45
S10	150,22
S11	152,47
S12	148,45
S13	153,13

Dan didapatkan nilai *mean flow time* yang terkecil pada *batch* 0,25 x 914 mm pada *stage* 2 senilai 147,13 pada variasi *job* S4 yaitu 6-18-2-22-5-10-19-12-16-1-17-23-13, dikarenakan nilai $T' \neq T''$ yaitu hasil *mean flow time* jadwal inisial belum sama dengan hasil *mean flow time* dengan menggunakan variasi jadwal dari perhitungan metode *neighborhood search*, maka dilanjutkan ke *stage* selanjutnya yaitu *stage* 2 dengan menggunakan variasi jadwal pada hasil *stage* 1 dengan langkah-langkah yang sama. Kemudian dihasilkan nilai kriteria terbaik untuk *batch* 0,25 x 914mm pada *stage* 2 yang terlihat pada dibawah ini :

Tabel 12. Variasi Jadwal Batch 0,25x914mm Untuk Keseluruhan (T^{''})

Jadwal	Nilai Mean Flow time
S1	147,13
S2	147,19
S3	147,19
S4	148,45
S5	148,52
S6	147,13
S7	147,13
S8	147,93
S9	147,13
S10	148,93
S11	161,95
S12	150,39
S13	155,22

Dan didapatkan variasi jadwal dengan nilai *mean flow time* terkecil adalah variasi jadwal S1 dengan nilai 147,13 jam dengan *sequencing* 6-18-2-22-5-10-19-12-16-1-17-23-13 dan dikarenakan nilai $T'' = T'''$ maka proses berhenti sampai disini.

Dari hasil perhitungan yang dilakukan dengan beberapa metode yang berbeda-beda maka didapatkan *makespan* terbaik dari masing-masing perhitungan metode tersebut. Berikut dibawah ini, hasil perhitungan yang didapatkan pada masing-masing *batch* :

a. *Batch* Pertama 0,20 X 914 mm

Tabel 13. Perbandingan Nilai Kriteria (Mean Flow Time)

Nilai Kriteria	Algoritma Neighborhood Search	Algoritma Branch and Bound	Existing (Perusahaan)
Mean flow time	18,42 jam	18,42 jam	43,23 jam
Urutan Job	(14-20-21-11-15-8-3-4-9-7)	(14-21-20-11-15-8-3-4-9-7)	(3-4-7-8-9-11-14-15-20-21)

b. *Batch* Kedua 0,25 X 914 mm

Tabel 14. Perbandingan Nilai Kriteria (Mean Flow Time)

Nilai Kriteria	Algoritma Neighborhood Search	Algoritma Branch and Bound	Existing (Perusahaan)
Mean flow time	147,13 jam	147,13 jam	182,76 jam
Urutan Job	(6-18-2-22-5-10-19-12-16-1-17-23-13)	(6-18-2-22-5-10-19-16-12-1-23-17-13)	(1-2-3-6-10-12-13-16-17-18-19-22-23)

Dari tabel 13 dan 14, didapatkan dari *batch* 1 dan *batch* 2, dengan menggunakan dua metode penjadwalan ini yaitu metode *neighborhood search* dan algoritma *branch and bound*, hasil *mean flow time* nya sama yaitu untuk *batch* 1 dengan dimensi 0,20 x 914 mm yaitu 18,42 jam dengan urutan *job* yaitu 14-20-21-11-15-8-3-4-9-7 dan 14-21-20-11-15-8-3-4-9-7. Pada *batch* 2 dengan dimensi 0,25 x 914 mm yaitu 147,13 dengan urutan *job* yaitu 6-18-2-22-5-10-19-12-16-1-17-23-13 yang menggunakan metode *neighborhood search* dan untuk algoritma *branch and bound*, urutan yang didapatkan berbeda dengan metode sebelumnya, 6-18-2-22-5-10-19-16-12-1-23-17-13.

ANALISA

Pada hasil perhitungan yang telah dilakukan, pada *batch* 0,20 x 914 mm, hasilnya terlihat sama dengan menggunakan 3 metode yaitu SPT,

neighborhood search dan algoritma *branch and bound* yaitu 18,42 jam dengan urutan *job* yang berbeda yaitu 14-20-21-11-15-8-3-4-9-7 dan 14-21-20-11-15-8-3-4-9-7. Dapat dilihat sebenarnya, bahwa perhitungan menggunakan jadwal inisial saja sudah cukup. Tetapi kedua metode ini selain metode SPT dapat dilakukan untuk mencari nilai kriteria yang terbaik dengan melakukan pertukaran *job* untuk setiap variasi *job* nya dan melakukan enumerasi dengan batasan yang sudah ditentukan yaitu *upper bound* dan *lower bound*. Terdapat perbedaan urutan *job* ke 2 dan ke 3 yaitu 20 dan 21.

Pada penelitian ini, hasil perhitungannya mempunyai nilai yang sama dengan menggunakan ketiga metode ini dikarenakan beberapa urutan *job* yang berbeda mempunyai jenis produk yang sama, waktu prosesnya yang relatif sama, tonase yang diproses sama, sehingga prosesnya bersifat *continue* dan diperlukan *job* sisipan, yang hanya diperlukan ketika pergantian jenis produk dan warna yang berbeda. Waktu yang diperlukan untuk *job* sisipan ini adalah 45 menit atau 0,75 jam.

Kemudian untuk nilai hasil perhitungan pada *batch* kedua dengan dimensi 0,25 x 914 mm, nilai terbaik didapatkan dari perhitungan menggunakan metode *neighborhood search* dengan urutan *job* yang berbeda pula dengan penggunaan jadwal inisial dan perhitungan algoritma *branch bound* dengan hasil *mean flow time* yang didapatkan 147,13 jam dengan urutan jadwal yang berbeda dengan menggunakan dua algoritma ini. Urutan jadwal pada algoritma *branch bound* yaitu (6-18-2-22-5-10-19-16-12-1-23-17-13). Kemudian variasi jadwal dengan algoritma *neighborhood search* yaitu (6-18-2-22-5-10-19-12-16-1-17-23-13), yang berbeda pada urutan ke 8-9 dan 11-12.

Dan berbeda jauh sekali dengan hasil eksisting dan nilai *mean flow time* pada jadwal inisial dengan menggunakan metode SPT (*Short Processing Time*). Hal ini dikarenakan *job* yang ditukarkan sangat berpengaruh besar seperti pertukaran *job* dengan warna, waktu proses yang berbeda pula, adanya *job* sisipan yang diperlukan dan proses enumerasi yang terjadi pada perhitungan *branch and bound* sehingga *sequencing* nya menjadi lebih banyak. Dan penurunan yang terjadi antara kondisi eksisting dengan menggunakan dua algoritma ini yaitu **57 % pada batch 0,20 x 914mm dan 19 % pada batch 0,25 x 914 mm.**

KESIMPULAN

Penjadwalan produk painted di MCL 2 dengan menggunakan algoritma *branch and bound* dan

neighborhood search, didapatkan nilai *mean flow time* nya yaitu sebesar 18,42 jam untuk urutan *job* dengan dimensi 0,20 x 914mm dibandingkan kondisi eksisting yaitu 43,23 jam. Kemudian untuk dimensi 0,25 x 914 mm, didapatkan nilai *mean flow time* nya sebesar 147,13 jam dibandingkan kondisi eksisting dengan nilai 182,76 jam. Dari dua nilai tersebut didapatkan bisa meminimasi *mean flow time*. Untuk algoritma *branch and bound* dan *neighborhood search*. Pada produk dengan dimensi 0,20 x 914 mm, nilai *mean flow time* 18,42 dengan urutan *job* (14-20-21-11-15-8-3-4-9-7) dan (14-21-20-11-15-8-3-4-9-7). Kemudian untuk dimensi produk 0,25 x 914 mm, nilai *mean flow time* 147,13 jam dengan urutan *job* yang berbeda yaitu (6-18-2-22-5-10-19-16-12-1-23-17-13) dan (6-18-2-22-5-10-19-12-16-1-17-23-13), dibandingkan dengan jadwal *existing* yang memiliki nilai sebesar 43,23 jam untuk dimensi 0,20 x 914 mm dan 182,76 jam untuk dimensi produk 0,25 x 914mm. Sehingga menunjukkan penurunan dari kondisi eksisting yaitu 57 % pada *batch* 0,20 x 914mm dan 19 % pada *batch* 0,25 x 914.

DAFTAR PUSTAKA

- Baker, K. R. 2008. *Element of Sequencing dan Scheduling*. John Wiley dan Sons Inc. New York.
- Berlianty, I., et. al. 2010. *Teknik- Teknik Optimasi Heuristik*. Graha Ilmu. Yogyakarta.
- Brucker, P. 2007. *Scheduling Algorithms Fifth Edition*. Springer-Verlag Berlin Heidelberg. Germany.
- Hakim, P. 2008. *Perencanaan & Pengendalian Produksi*. Guna Widya. Surabaya.
- Hasbullah, M.K. 2009. *Usulan Penjadwalan Produksi Sepatu Nike Pada PT. Pratama Abadi Industri Dengan Menggunakan Metode Shortest Processing Time*. Jurnal Teknik Industri : Universitas Mercu Buana Jakarta.
- Palit, H.C., et. al., *Penjadwalan Produksi Flexible Flow Shop Dengan Sequence-Dependent Setup Times Menggunakan Metode Relaksasi Lagrangian* (Studi Kasus Pada PT. Cahaya Angkasa Abadi).
<http://puslit.petra.ac.id/journals/pdf.php?PublishedID=IND03050205>. Diakses pada tanggal 13 Januari 2012.
- Pinedo, M., 2008. *Scheduling Theory, Algorithms, and System Thirt Edition*. Springer. New York.
- Sutanto, G. 2000. *Algoritma Branch And Bound Dan Algoritma Genetika Untuk Penjadwalan Flowshop Dengan Fungsi Tujuan Ganda*. Skripsi Jurusan Teknik Industri Fakultas Teknik Universitas Kristen Petra : Surabaya.
- Sutanto, J., et. al. 2004. *Algoritma Brach and Bound untuk Masalah Penjadwalan pada Mesin Paralel*. Jurnal Teknik Informatika. Laboratorium Ilmu dan Rekayasa Komputasi. Departemen Teknik Informatika : ITB
- Suyanto. 2010. *Algoritma Optimasi Deterministik atau Probabilistik*. Graha Ilmu. Yogyakarta.
- Trijayanto, B.E. 2012. *Perbandingan Metode Penjadwalan Pada Pola Aliran Flow Shop Mesin Tunggal Dengan Jadwal Eksisting, Teknik Sisipan, Dan Neighborhood Search*. Skripsi Jurusan Teknik Industri Fakultas Teknik Universitas Sultan Ageng Tirtayasa : Cilegon.