

PENJADWALAN BATCH PADA SISTEM PRODUKSI JOB SHOP SINGLE ITEM MULTI STAGE

Endang Widuri Asih¹

ABSTRACT

Peripatetic company in area manufacture will not get out of the that company ability to run its operation. Environmental problems optimization of manufacture usually hit the finite resource allocation of its amount in a few work area. With the certain criterion performance, the problems optimization is usually finished. One of form optimalitation in system manufacture is by scheduling existing production facility to finish some job. Two general criterion which often weared to evaluate the performance scheduling is makespan and flow time.

At this research is conducted to compare the minimum makespan is by using overlapping method and without using overlapping method. Overlapping method is division of a number of part to be produced by multi stage into some batch in such a way that the batch can be parallely processed.

Batch scheduling by using overlapping method represent the optimal solution by minimation makespan. Total the optimal batch as much 2. Makespan got is equal to 148745,08 minute, smaller than makespan got without using overlapping method that is equal to 152000 minute. Big of time thrift is equal to 2,14%.

Key word: Batch scheduling, Job Shop, makespan

INTISARI

Perusahaan yang bergerak dibidang *manufaktur* tidak akan lepas dari kemampuan perusahaan itu untuk menjalankan operasinya. Permasalahan optimasi lingkungan *manufaktur* biasanya mengenai pengalokasian sumber daya yang terbatas jumlahnya dalam beberapa bidang pekerjaan. Dengan kriteria performansi tertentu, permasalahan optimasi tersebut biasanya diselesaikan. Salah satu bentuk optimalisasi dalam sistem *manufaktur* adalah dengan menjadwalkan fasilitas-fasilitas produksi yang ada untuk menyelesaikan beberapa *job*. Dua kriteria umum yang sering dipakai untuk mengevaluasi performansi penjadwalan adalah *makespan* dan *flow time*.

Pada penelitian ini dilakukan untuk membandingkan *makespan* yang minimal adalah dengan cara menggunakan metode *overlapping* dan tanpa menggunakan metode *overlapping*. Metode *overlapping* adalah pembagian sejumlah *part* yang akan diproduksi secara *multi stage* ke dalam beberapa *batch* sedemikian sehingga *batch-batch* tersebut dapat diproses secara paralel.

Penjadwalan *batch* dengan menggunakan metode *overlapping* merupakan solusi yang optimal dengan meminimasi *makespan*. Jumlah *batch* optimal sebanyak 2. *makespan* yang didapat adalah sebesar 148745,08 menit, lebih kecil dari *makespan* yang didapat tanpa menggunakan metode *overlapping* yaitu sebesar 152000 menit. Besar penghematan waktu adalah sebesar 2,14%.

Kata Kunci: Penjadwalan Batch, Job Shop, Makespan

PENDAHULUAN

Sebuah perusahaan yang bergerak dibidang manufaktur tidak akan terlepas dari kemampuan perusahaan untuk menjalankan operasinya. Kemampuan perusahaan terdiri dari beberapa sumber daya yang harus disediakan untuk mendukung aktivitas produksinya. Untuk dapat mencapai

target produksi yang direncanakan, setiap industri harus memiliki strategi untuk mengatasi setiap permasalahan yang ada dari dalam ataupun luar sistemnya. Keadaan seperti ini dapat dicapai apabila ada suatu upaya optimalisasi dalam lingkungan sistem manufakturnya. Permasalahan optimasi lingkungan manufaktur biasanya

¹ Staf Pengajar Jurusan Teknik Industri, ISTA, Yogyakarta

mengenai pengalokasian sumber daya yang terbatas jumlahnya dalam beberapa bidang pekerjaan. Sumber daya tersebut bisa berupa sumber tenaga kerja, mesin, ataupun pekerja, yang dapat menjadi batasan bagi perusahaan untuk mencapai tujuannya yaitu tetap dapat bertahan dalam persaingan usaha dengan keuntungan maksimal. Dengan sebuah kriteria performansi tertentu, permasalahan optimasi tersebut biasanya diselesaikan dengan algoritma-algoritma atau dengan pendekatan model matematis tertentu.

Salah satu bentuk optimalisasi dalam sistem *manufaktur* adalah dengan menjadwalkan fasilitas-fasilitas produksi yang ada untuk menyelesaikan beberapa *job* atau yang dikenal dengan kasus penjadwalan atau *scheduling* yang merupakan kegiatan yang penting dalam perencanaan dan pengendalian produksi, karena jadwal yang baik akan memaksimalkan efektivitas pemanfaatan setiap sumber daya yang ada. Kasus *scheduling* biasanya diselesaikan dengan menggunakan algoritma.

Dua kriteria umum yang paling sering dipakai untuk mengevaluasi performansi penjadwalan adalah *makespan* dan *flow time*. Dengan demikian, disamping masalah penentuan urutan pemrosesan, masalah lainnya yang timbul adalah masalah penentuan urutan *batch* yang akan diproses.

PT. Bakrie Tosanjaya, Bekasi Jawa Barat merupakan perusahaan yang bergerak dibidang peleburan logam dalam pembuatan *casting* otomotif (komponen-komponen kendaraan bermotor), dan bentuk aliran produksinya adalah bertipe *job shop*. Perusahaan memproduksi berdasarkan pesanan dari konsumen. Pesanan tersebut biasanya mempunyai periode pesanan (siklus per periode) yang tidak menentu lamanya dan penentuannya dilakukan dengan kesepakatan antara perusahaan dengan konsumen, artinya perusahaan menawarkan waktu yang paling mungkin bagi perusahaan untuk menyelesaikan pesanan dari konsumen berdasarkan perkiraan awal atau perhitungan kemampuan perusahaan. Perusahaan

masih dapat mengurangi periode pemesanan menjadi lebih kecil dari perkiraan awal berkaitan dengan tujuan perusahaan untuk memperoleh laba semaksimal mungkin.

Upaya agar pesanan konsumen dapat terpenuhi tepat waktu adalah dengan meminimasi *makespan*. Salah satu cara agar *makespan* dapat diminimalkan adalah dengan menentukan ukuran *batch* yang optimal dengan menggunakan metode *overlapping*. Metode *overlapping* adalah pembagian sejumlah *part* yang akan diproduksi secara *multi stage* ke dalam beberapa buah *batch* sedemikian sehingga *batch-batch* tersebut dapat diproses. Sedangkan *batch* itu sendiri dalam hal ini adalah ukuran kapasitas satu ledel (*tungku*) dalam satuan pieces.

PENJADWALAN

Penjadwalan merupakan upaya untuk mengatur kegiatan atau pekerjaan dengan tujuan untuk mencapai efisiensi penggunaan fasilitas, waktu, dan biaya. Hal ini diperlukan mengingat fasilitas produksi pada suatu industri jumlahnya terbatas, sementara itu pekerjaan yang akan dilakukan cukup banyak sehingga menuntut pengaturan jadwal pekerjaan yang baik dan oleh karena itu pihak manajemen harus mampu menjadwalkan pekerjaan sesuai dengan keterbatasan fasilitas produksi. Secara umum penjadwalan juga dapat didefinisikan sebagai proses pengalokasian sumber untuk memilih mengerjakan tugas dalam jangka waktu tertentu (*Baker, 1974*). Ada dua arti dari pengertian penjadwalan ini, yaitu yang pertama bahwa penjadwalan merupakan fungsi pengambilan keputusan yaitu dalam menentukan jadwal yang paling tepat, dan yang kedua bahwa penjadwalan merupakan sebuah teori yang berisi kumpulan prinsip, model, teknik, dan konklusi yang logis dalam proses pengambilan keputusan.

Kriteria Dan Asumsi Dalam Penjadwalan Produksi

Permasalahan penjadwalan pada intinya adalah menentukan urutan

produksi yang memberikan solusi terbaik dengan kriteria sebagai berikut:

- a. Memenuhi kendala teknologi yang ada (*technological constrains*).
- b. Memenuhi satu atau beberapa kriteria performansi yang telah didefinisikan terlebih dahulu sebelumnya.

Variabel ukuran performansi yang telah dikembangkan dalam lingkungan penjadwalan diantaranya adalah sebagai berikut (*Baker, 1974*):

- a. Completion time (C_i), yaitu waktu penyelesaian paling akhir suatu job i .
- b. Flow time (F_i) disebut juga dengan shop time atau manufakturing interval, yaitu waktu yang diperlukan suatu job i berada di shop floor.
- c. Waiting time (W_i), yaitu waktu menunggu antara waktu proses selesai diproses hingga mulai dioperasi berikutnya dari pengerjaan tiap operasi pada job i .
- d. Queue time (Q_i), yaitu waktu menunggu
- e. sebelum operasi dimulai.
- f. Idle time, yaitu waktu dimana operator atau sumber-sumber daya (misalnya : mesin) tidak menghasilkan produk.
- g. Lateness (L_i) = $C_i - d_i$, yaitu lamanya perbedaan antara completion time (C_i) dan due date (d_i) dari suatu job i .
- h. Tardiness (T_i) = $\max(0, L_i)$, yaitu lamanya keterlambatan waktu penyelesaian suatu job i .
- i. Slack (S_i) = $d_i - t_i$, yaitu perbedaan antara waktu yang tersisa dengan due date dan waktu proses job yang bersangkutan.

Dari variabel ukuran performansi diatas, dapat disimpulkan beberapa fungsi tujuan yang lazim digunakan, meliputi :

- a. Minimasi *flow time* aktual *job*
- b. Minimasi total waktu tunggu tinggal tertimbang (*total weighted flow time*)
- c. Minimasi keterlambatan tertimbang (*weighted lateness*)
- d. Minimasi keterlambatan maksimum
- e. Minimasi jumlah *job* yang terlambat

Beberapa asumsi yang sering digunakan dalam teori penjadwalan adalah :

a. Asumsi untuk karakteristik job :

1. Tiap *job* diselesaikan menurut proses yang sudah disusun.
2. Job merupakan suatu kesatuan yang mungkin terdiri dari beberapa unit.
3. Setiap job tidak ada proses menanti, jadi job diproses dari awal sampai akhir tanpa ada overlap antara proses sebelum dan sesudahnya.
4. Setiap job memiliki jumlah operasi yang sudah tertentu, dimana setiap operasi hanya dilakukan pada satu mesin.
5. Teknologi pembuatan sudah diketahui.

b. Asumsi untuk karakteristik mesin :

1. Kerja mesin dalam job adalah independent sehingga mesin dapat beroperasi pada kecepatan output maksimum.
2. Secara kontinue mesin siap untuk dibebani job selama periode penjadwalan.
3. Hanya satu job yang dapat dikerjakan tiap mesin pada saat yang sama.

c. Asumsi untuk karakteristik waktu proses

1. Waktu proses diambil pada saat *job* dikenakan proses pengerjaan.
2. Waktu *set-up*, dan waktu jika terjadi penghentian mesin termasuk dalam kategori waktu proses.

Asumsi lain yang sangat berpengaruh terhadap penjadwalan adalah selalu tersedianya kebutuhan-kebutuhan yang diperlukan dalam proses operasi, seperti tersedianya peralatan dan bahan pendukung dalam jumlah yang cukup.

Batching

Ada dua alternatif penjadwalan yang memperpendek siklus produksi yaitu dengan metode *splitting* dan *overlapping* (*fogarty, 1991*). *Splitting* yaitu teknik untuk mengurangi total waktu penyelesaian dengan membagi *lot*

menjadi dua atau lebih sesuai dengan jumlah mesin identik yang ada. Misalnya ada dua mesin D yang identik, maka *lot* pekerjaan D dibagi menjadi dua dan waktu proses masing-masing mesin menjadi setengah. Metode berikutnya adalah *overlapping*, yaitu teknik untuk mengurangi total waktu penyelesaian dengan membagi setengah *lot* (yang akan diproses) ke dalam dua atau lebih yang lebih kecil, sehingga suatu kelompok *lot* yang telah selesai diproses dalam suatu stage (proses) tertentu bisa diteruskan ke stage berikutnya tanpa harus menunggu seluruh *lot* selesai diproses pada stage berikut.

Metode Overlapping

Kecepatan suatu sistem produksi dalam menyelesaikan pesanan biasanya dengan ukuran performansi *makespan* atau *lead time*. *Makespan* adalah waktu yang dibutuhkan untuk menyelesaikan sejumlah pekerjaan dari mulai proses pertama untuk pekerjaan pertama sampai selesai proses terakhir dari pekerjaan terakhir. Sementara itu *lead time* adalah waktu yang dibutuhkan untuk menyelesaikan sebuah pekerjaan. *Total lead time* adalah hasil penjumlahan dari *lead time* masing-masing pekerjaan. *Lead time* meliputi waktu proses, waktu antrian, waktu transportasi, dan waktu inspeksi.

Dalam usaha mengurangi total *lead time* pemrosesan sejumlah produk, Fogarty (1991) mengusulkan suatu metode yang disebut dengan metode *overlapping*. Dalam metode yang ditampilkan oleh Fogarty (1991) diasumsikan semua pekerjaan yang sudah berada pada rantai produksi sejak waktu nol dan dikirim secara bersamaan pada saat *part* terakhir diproses. Dengan demikian seluruh *part* mempunyai *lead time* yang sama, yaitu sebesar *lead time* maksimum (yaitu *lead time* yang selesai terakhir). Conway (1967) menyatakan bahwa *lead time* maksimum akan sama dengan *makespan*. Teknik yang digunakan dalam metode *overlapping* adalah membagi sejumlah *part* yang akan diproses ke dalam dua *batch* atau lebih sehingga *total lead time* dapat dikurangi.

Fogarty (1991) menyatakan beberapa hal dasar berikut mengenai metode *overlapping*:

1. Sejumlah *part* yang akan diproses dibagi menjadi paling sedikit 2 *batch*.
2. Segera setelah *batch* pertama selesai diproses pada stage 1, *batch* pertama tersebut dipindahkan ke stage ke 2 untuk segera diproses lebih lanjut.
3. Pada waktu stage 1 sedang memproses *batch* kedua, stage 2 memproses *batch* secara *overlapping*.
4. Ketika stage 1 selesai memproses *batch* kedua, *batch* kedua ini segera pada *job shop* dua stage, jika stage 2 mempunyai operasi yang lebih cepat dari stage 1, *batch* yang pertama harus cukup besar untuk mengisi *idle time* pada stage 2. perhitungan minimum *batch* pertama ini adalah seperti pada gambar 1. Keterangan dari gambar 1 adalah

$$Q = Q_1 + Q_2 \dots \dots \dots (1)$$

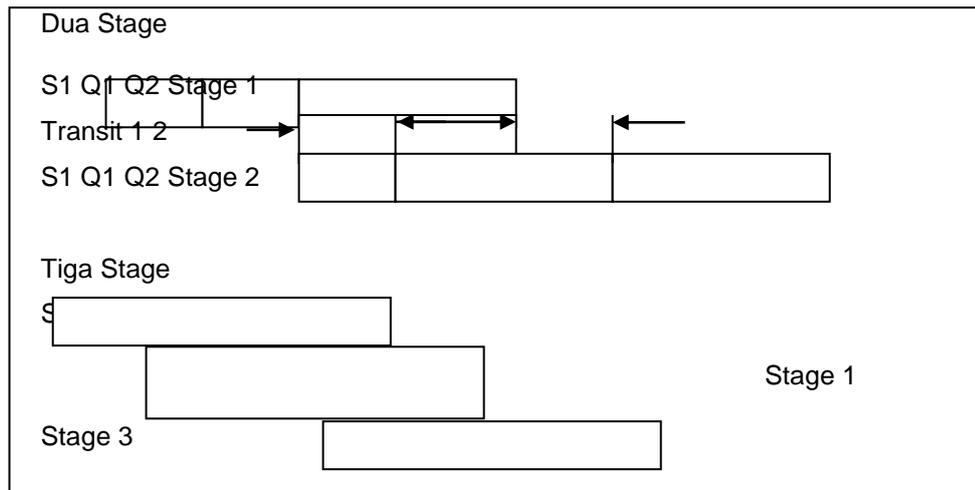
$$Q_1 p_2 + T_{12} + S_2 \geq Q_2 P_1 + T_{12} \dots \dots (2)$$

- Q = Jumlah part yang harus dikerjakan
- Q_1 = Ukuran minimum dari batch pertama
- Q_2 = Ukuran minimum dari batch kedua
- S_2 = Waktu set-up pada stage kedua
- P_1 = Waktu proses per unit pada stage pertama
- P_2 = Waktu proses per unit pada stage kedua
- T_{12} = Waktu transit antara stage 1 dan 2

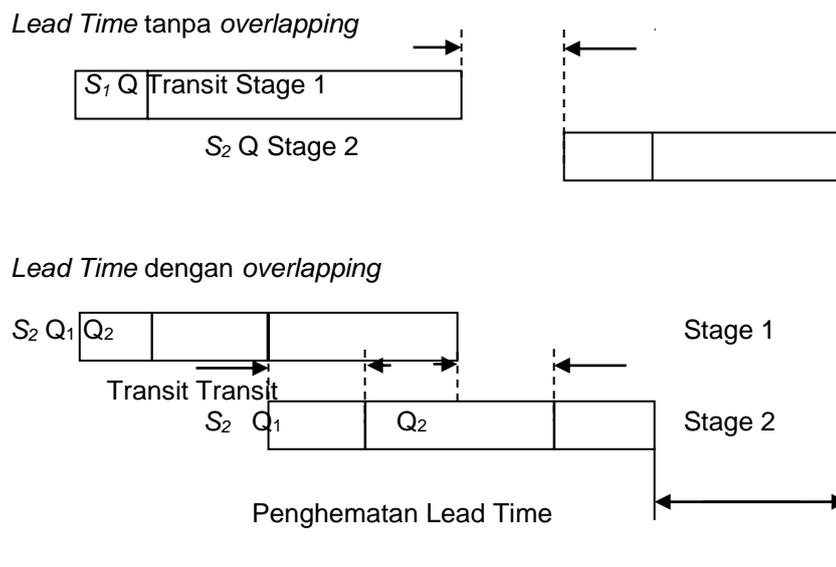
Keuntungan menggunakan metode *overlapping* adalah berkurangnya total *manufacturing lead time* (gambar 2.4), Kerugiannya adalah muncul ongkos tambahan yang berasal dari bertambahnya perencanaan dan kontrol yang diperlukan untuk menambah jumlah *batch* dan mengawasi perpindahan material. Sedangkan keuntungan bila tidak menggunakan metode

overlapping adalah proses batchnya lebih sederhana (apa adanya) dan tidak ada ongkos tambahan, kerugiannya akan diperoleh total *manufacturing lead time* (gambar 2) yang akan bertambah karena pada ukuran dan jumlah *batch* dari suatu jenis produk yang akan yang

akan diproduksi tidak dibagi-bagi lagi menjadi beberapa batch.



Gambar 1. Skema Overlapping Operasi (Sumber : Fogarty, 1991)



Gambar 2. Perbandingan *Lead Time* dengan dan tanpa menggunakan *Overlapping* (Sumber : Fogarty, 1991)

Identifikasi Stage Kritis

Pada kasus dimana terdapat 3 stage kritis sebagai contoh yaitu di stage 1, stage 2, dan stage 3, untuk mendapatkan *makespan* minimum stage 2 tidak boleh mengalami *idle time* (karena stage 2 merupakan stage yang waktu prosesnya paling lambat/*bottleneck*). Pada penelitian ini ditemukan untuk mencapai nilai *makespan* minimum, maka (N-1) persamaan dibagi ke dalam dua kelompok. Kelompok pertama adalah persamaan yang menggunakan stage 1 dan stage 2 (misalnya $= t_2 Q_N + S_2 = Q_{N-1} + S_1$).

Kelompok kedua yang menggunakan stage 2 dan stage 3 (misalnya : $t_2 Q_1 + S_2 = Q_2 t_3 + S_3$). Dari Persamaan-persamaan tersebut hubungan Q_1 dapat ditentukan.

Persamaan yang digunakan pada kasus ini adalah persamaan kelompok kedua, yaitu kelompok yang menggunakan stage 2 dan stage 3, dengan cara sebagai berikut :

$$Q_n(t_3) + S_3 = Q_{N-1}(t_2) + S_2 \dots\dots\dots(3)$$

Dari persamaan (3) didapatkan rumus umum untuk hubungan Q_1 dan Q_2 adalah sebagai berikut :

$$Q = Q_1 \left(\frac{t_2}{t_3} \right)^{i-1} + (S_{21} - S_3) \sum_{k=0}^{i-2} \frac{(t_2)^k}{(t_3)^{k+1}} \dots(4)$$

Karena $\sum_{i=1}^N Q_1 = n$, maka didapatkan persamaan sebagai berikut :

$$Q_2 = \frac{n - (S_2 - S_3) \sum_{i=2}^N \sum_{k=0}^{i-2} \frac{(t_2)^k}{(t_3)^{k+1}}}{1 + \sum_{i=2}^N \left[\frac{t_2}{t_3} \right]^{i-1}} \dots\dots\dots(5)$$

$$Q_1 = \frac{Q_{N-1} t_2 + (S_2 - S_3)}{t_3} \dots\dots\dots(6)$$

Perumusan Jumlah Batch Optimal

Jumlah *batch* yang ideal adalah jumlah *batch* yang menghasilkan nilai *makespan* yang lebih kecil. Untuk menentukan jumlah *batch* (N) yang ideal dapat ditentukan dengan mencoba-coba

harga N (mulai dari N = 2,3,4, dan seterusnya) sampai ditemukan suatu titik maksimum *makespan*.

Algoritma Perhitungan Makespan

Langkah awal dari penyelesaian dengan menggunakan metode *overlapping* adalah menghitung *makespan* yang terjadi pada sistem produksi *job shop*. *Makespan* akan dihitung secara bertahap. Tahap pertama adalah menghitung saat selesai untuk *batch* pertama pada setiap stage. Tahap selanjutnya adalah menghitung saat selesai *batch* kedua pada setiap stage. Saat selesai *batch* kedua pada stage kesatu adalah saat selesai *batch* pertama pada stage kesatu ditambahkan dengan waktu *setup* dan waktu proses *batch* kedua pada stage kesatu. Sedangkan saat selesai *batch* kedua pada stage kedua dapat dihitung dari saat mulainya *batch* kedua diproses pada stage dua ditambah waktu proses *batch* kedua pada stage dua. Waktu mulainya *batch* kedua diproses pada stage dua adalah maksimum antara dua nilai : saat selesai *batch* pertama pada stage kedua di tambah waktu *setup* untuk *batch* kedua pada stage dua atau saat selesai *batch* pada stage satu. Demikian sehingga didapatkan *makespan*, yaitu selang waktu antara *batch* pertama mulai diproses pada stage satu sampai saat selesai *batch* terakhir pada stage terakhir. Algoritma yang diusulkan ini menghitung secara *forward*, yaitu mulai dari *batch* pertama yang diproses pada stage satu sampai *batch* terakhir pada stage dua. Untuk lebih jelasnya perhitungannya adalah sebagai berikut :

Langkah 1:

Hitung $C_{r,N} = Q_N \cdot t_r$
untuk semua r (r = 1,2,...,m)

Langkah2:

Hitung $C_{1,N-1} = C_{1,N} + S_1 + Q_{N-1} \cdot t_1$

Langkah 3:

Hitung $C_{2,N-1}$ dengan
 $C_{2,N-1} = \max (C_{1,N-1}; C_{2,N} + S_2) + Q_{n-1} \cdot t_2$

Ulangi langkah 3: Untuk semua $C_{2,N-1}$ dengan :

$$C_{r,i} = \max (C_r - t_i; C_{r,i+1} + S_r) + Q_i \cdot t_r$$

Langkah 4:

Ulangi langkah 2 dan langkah 3 untuk $l = N-2, N-3, \dots, 1$ dengan $C_{l,1} + S_1 + Q_1.t_1$

Langkah 5:

Tentukan $makespan = C_{m-1}$

Perumusan Ukuran Dari Setiap Batch

Ukuran suatu *batch* akan memberikan efek yang signifikan terhadap ukuran performansi terhadap waktu, seperti *flow time* dan *makespan*. Sebagai contoh gambar 3, memperlihatkan perbedaan *makespan* yang terjadi akibat perubahan ukuran *batch* (Fogarty, 1991). Situasi yang ditampilkan adalah *stage 2* mempunyai waktu proses yang lebih lambat 1,5 kali dari waktu proses (dalam satuan waktu, *sw*) satu *part* pada *stage 1*, dan waktu *setup* kedua *stage* adalah sama. Pada kondisi (a) *batch* Q_1 diproses selama 3 *sw* pada *stage 1*, untuk kemudian diproses selama 4,5 *sw* pada *stage 2* dan *batch* Q_2 diproses selama 2,5 *sw* pada *stage 1* dan 3,75 *sw* pada *stage 2*. keadaan ini menyebabkan *batch* Q_2 telah diproses pada *stage 1*, pada saat *batch* Q_2 masih diproses distage 2 sehingga *batch* Q_2 harus menunggu (mengalami *Queue time*) sampai *batch* Q_1 selesai diproses oleh *stage 2*. pada kondisi (b), yang terjadi adalah sebaliknya dari kondisi (a), yaitu *stage 2* telah selesai memproses *batch* Q_1 dan disetup untuk memproses *batch* Q_2 tetapi *batch* Q_2 belum selesai diproses distage 1 sehingga *stage 2* mengalami *idle time*. Kondisi (c) menggambarkan keadaan yang ideal, yaitu tidak terjadi *idle time* dari mesin dan tidak terjadi *queue time* dari *batch*. Dari gambar 3 dapat terlihat bahwa ukuran *batch* akan mempengaruhi terjadi atau tidaknya *Idle time* dan *queue time* pada aliran produksi yang pada akhirnya mempengaruhi *makespan* dan hal ini menyebabkan *makespan* menjadi semakin besar. *Makespan* minimum terjadi pada saat tidak terjadi *idle time* maupun *queue time*. Karena itu pengembangan metode ini akan ditunjukkan untuk menemukan cara dalam menentukan jumlah dan ukuran *batch* yang dapat meminimasi *idle time* dan *queue time* yang terjadi

agar fungsi tujuan untuk meminimasi *makespan* dapat tercapai.

Pengumpulan Data

1 Jumlah Mesin Pada Plant II

Pada proses produksi di plant II, menggunakan beberapa jumlah mesin yang sangat terkait hubungannya satu dengan lainnya pada masing-masing stasiun kerja, yaitu terdiri atas :

- Stasiun kerja 1 : Melting
- Stasiun kerja 2 : Mesin sanplant
- Stasiun kerja 3 : Mesin moulding
- Stasiun kerja 4 : Pouring
- Stasiun kerja 5 : Trimming
- Stasiun kerja 6 : Shot blast

2 Waktu Proses Setiap Mesin

Waktu proses setiap mesin produksi dalam mengolah suatu jenis produk sangat diperlukan, karena dapat menentukan kapan produk tersebut harus selesai dibuat dan dikirimkan tepat waktu. Waktu proses setiap mesin produksi di plant II adalah sebagaimana tabel 1.

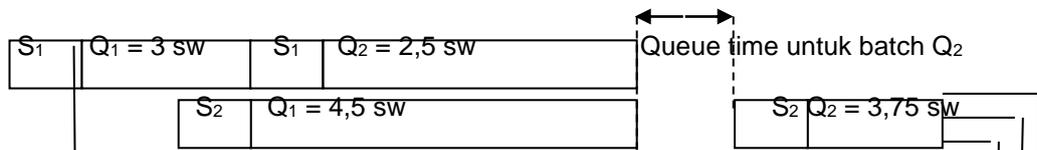
Pengolahan Data

Algoritma Perhitungan Makespan

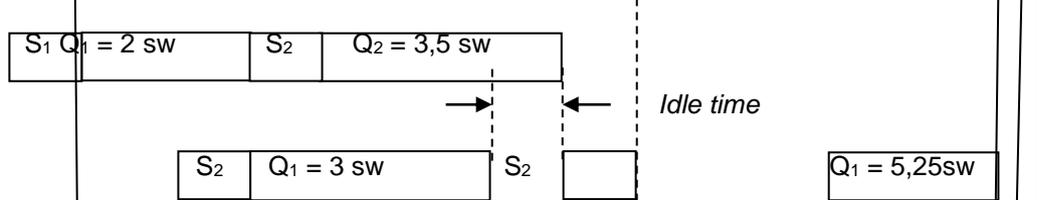
Langkah awal yang akan dilakukan dalam penelitian ini adalah menghitung *makespan* yang terjadi pada sistem produksi *job shop*. Notasi dan defenisi yang digunakan dalam perumusan dapat dijelaskan sebagai berikut :

- N : Jumlah *batch*
- m : Jumlah *stage*
- n : Jumlah *part* yang harus diproduksi
- g : Jumlah jenis *item* yang harus diproduksi
- Q_i : Ukuran *batch* i
- $T_{i,h}$: Waktu yang dibutuhkan untuk menyelesaikan satu *part* dari *item* h pada *stage* r
- $S_{r,i}$: Waktu *setup* *batch* i pada *stage* r
- $C_{r,i}$: Saat selesainya *batch* i pada *stage* r ($i=1,2,\dots,N$; $r=1,2,\dots,m$)
- $P_{i,r}$: Waktu yang diperlukan oleh *stage* r untuk mengerjakan *batch* i

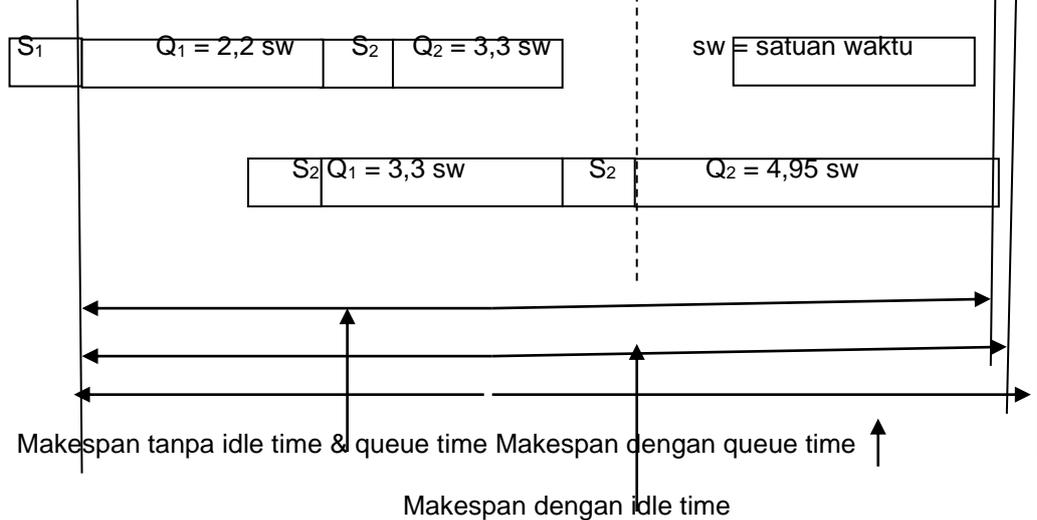
a) Terjadi *batch* menunggu (*queue time*)



b) Terjadi mesin menunggu (*idle time*)



c) Tidak terjadi *idle time* dan *queue time*



Gambar 3. Pengaruh *idle time* dan *queue time* terhadap *makespan*
(Sumber : Fogarty, 1991)

Tabel.1 Waktu Proses (menit)

Operasi	Mesin (menit)		Total (menit)
	1	2	
1	90	-	90
2	2	1	3
3	59	1	60
4	2	-	2
5	1	-	1
6	3	1	4

Tabel 2 *Routing* Pekerjaan

JOB	OPERASI KE -					
	1	2	3	4	5	6
1	SK ₁	SK ₄	-	-	-	-
2	SK ₂	SK ₃	-	-	-	-
3	SK ₂	SK ₃	SK ₁	SK ₄	SK ₅	SK ₆

Setelah mengetahui notasi dan definisi, dilakukan perhitungan *makespan* untuk job. Langkah-langkah yang diambil untuk menghitungnya adalah :

a. Perhitungan Manual *Makespan* Dengan Metode *Overlapping*

Jumlah part yang harus diproduksi

(n) = 1000 pcs

Waktu proses mesin 1 (t_1) = 60

Waktu proses mesin 2 (t_2) = 90

Waktu proses mesin 3 (t_3) = 2

Waktu *setup* mesin 1 (S_1) = 0

Waktu *setup* mesin 2 (S_2) = 0

Waktu *setup* mesin 3 (S_3) = 0

Langkah pertama adalah dengan menentukan jumlah *batch* (N) yang optimal.

- Untuk $N = 2$, dengan menggunakan rumus 5 yaitu:

$$Q_2 = \frac{1000 - (0) \sum_{i=2}^2 \sum_{k=0}^{i-1} \left(\frac{90}{2}\right)^k}{1 + \sum_{i=2}^2 \left(\frac{90}{2}\right)^{i-1}}$$

$$Q_2 = \frac{1000}{1 + \left(\frac{90}{2}\right)^1} = \frac{1000}{1 + 45} = \frac{1000}{46} = 21,7 \text{ pcs}$$

Dan dengan rumus 6 didapatkan:

$$Q_1 = \frac{21,74 * 90 + 0}{2} = 978,3 \text{ pcs}$$

Setelah menentukan jumlah batch yang ideal, maka akan dicari *makespan*nya dengan metode, dengan langkah-langkah sebagai berikut :

Langkah 1 :

Hitung $C_{r,N} = Q_{N,r} \cdot t_r$ untuk semua r ($r = 1, 2, \dots, m$)

$$C_{1,1} = 978,3 * 60 = 58698 \text{ menit}$$

$$C_{1,2} = 978,3 * 90 = 88047 \text{ menit}$$

$$C_{1,3} = 978,3 * 2 = 1956,6 \text{ menit}$$

$$C_{2,1} = 21,74 * 60 = 1304,4 \text{ menit}$$

$$C_{2,2} = 21,74 * 90 = 1956,6 \text{ menit}$$

$$C_{2,3} = 21,74 * 2 = 43,48 \text{ menit}$$

Langkah 2 :

Hitung $C_{1,N-1} = C_{1,N} + S_1 + Q_{N-1} \cdot t_1$

$$F_{1,1} = 0 + 58698 = 58698 \text{ menit}$$

$$F_{1,2} = 58698 + 88047 = 146745 \text{ menit}$$

$$F_{1,3} = 146745 + 1956,6 = 148701,6 \text{ menit}$$

Langkah 3 :

Hitung $C_{2,N-1}$ dengan

$$C_{2,N-1} = \max(C_{1,N-1}; C_{2,N} + S_2) + Q_{N-1} \cdot t_2$$

$$F_{2,1} = 58698 + 1304,4 = 60002,4 \text{ menit}$$

$$F_{2,2} = \max(60002,4; 146745) + 1956,6 = 148701,6 \text{ menit}$$

$$F_{2,3} = \max(148701,6; 148701,6) + 43,48 = 148745,08 \text{ menit}$$

Langkah 4 :

Tentukan $makespan = C_{m-1}(4)$

Jadi *makespan* untuk $N = 2$ adalah sebesar 148745,08 menit.

b. Perhitungan Manual *Makespan* Tanpa Metode *Overlapping*

Tabel. 3 Hasil Perhitungan *Makespan* Tanpa Metode *Overlapping*

Mesin	Waktu Operasi (menit)	Total Produk	Jumlah Waktu operasi (menit)
1	60	1000	60000
2	90		90000
3	2		2000
Total <i>makespan</i> (menit)			152000

PEMBAHASAN

Job atau *Stage* Kritis

Tujuan dari penelitian yang dihadapi ini adalah mengurutkan pekerjaan *part* pada *stage* kritis untuk meminimasi *makespan*. Sehingga konsep yang dipakai untuk menyelesaikan permasalahan yang dihadapi adalah konsep minimasi. Seperti yang telah dibahas dalam bab sebelumnya, identifikasi *stage* kritis diambil berdasarkan *stage bottleneck*, yaitu mesin yang mempunyai waktu proses paling lama. Pada *job* ini *stage bottleneck* terdapat pada *stage* 1 dengan waktu proses 90 menit. Dalam penelitian ini menggunakan 3 *stage* kritis yaitu *stage* sebelum *bottleneck* (*stage* 3), *stage bottleneck* itu sendiri (*stage* 1), dan *stage* sesudah *bottleneck* (*stage* 4).

Idle Time Batch

a. *Idle Time Batch* pada mesin 1 :

Pada mesin 1, batch 2 mengalami *idle time* sebesar = $(88047 - 60000)$ menit = 28047 menit.

b. *Idle Time Batch* pada mesin 4 :

Pada mesin 4, batch 2 mengalami *idle time* sebesar = $(91956,6 - 90000)$ menit = 1956,6 menit.

Variabel Yang Berpengaruh Pada Hasil Penjadwalan

a. Waktu proses

Waktu proses pada suatu *stage* akan mempengaruhi nilai *makespan* minimum yang didapat, jumlah *batch* untuk mencapai *makespan* minimum tersebut dan status *stage* kritis. Dengan semakin lamanya waktu yang diperlukan untuk memproses sebuah *part* (membesarnya nilai *t*) pada suatu *stage*, maka nilai *makespan* minimum yang didapatkan akan semakin membesar, jumlah *batch* yang diperlukan untuk mencapai nilai minimum juga semakin banyak.

Hal ini terjadi karena dengan bertambah lamanya suatu *stage* memproses suatu *part* berarti kemungkinan *stage* untuk menjadi *bottleneck* semakin besar, sehingga untuk meminimumkan *idle time* yang terjadi diperlukan pembagian *batch* yang lebih banyak. Hal lain adalah pada kondisi tersebut kemungkinan *stage* tersebut menjadi *stage bottleneck* semakin besar.

b. Perbandingan *Makespan* dengan dan Tanpa Metode *Overlapping*.

Pada job ini tanpa menggunakan metode *Overlapping* nilai *makespan* yang didapatkan sebesar 152000 menit, sedangkan apabila dihitung dengan menggunakan metode *Overlapping* didapatkan *makespan* yang lebih kecil, dari keadaan awal yaitu sebesar 148745,08 menit. besar penghematannya sebesar :

$$\begin{aligned} &= \frac{152000 - 148745,08}{152000} \times 100\% \\ &= \frac{3254,92}{152000} \times 100\% \\ &= 2,14\% \end{aligned}$$

KESIMPULAN

Penarikan kesimpulan ini bertujuan untuk menjawab permasalahan yang telah dirumuskan pada awal tahap penelitian ini. Adapun kesimpulannya adalah sebagai berikut :

- Dari hasil perhitungan dan pengolahan data dapat menghasilkan ukuran *batch* dan jumlah *batch* yang menghasilkan nilai

makespan minimum. Ukuran *batch* yang didapat adalah sebagai berikut:

1. *Batch* 1 = 978,3 pcs

2. *Batch* 2 = 21,7 pcs

- Dengan menggunakan metode *Overlapping* berarti memecah sejumlah part ke dalam *batch* yang lebih banyak (sampai batas tertentu), sehingga dapat menghasilkan nilai *makespan* yang lebih kecil dari keadaan semula.
- Penjadwalan dengan metode *overlapping* untuk jumlah batch 2 dihasilkan *makespan*nya adalah 148745,08 menit. Dan tanpa metode *overlapping* adalah 152000
- Penjadwalan batch *job shop* dengan metode *overlapping* akan menghemat waktu *makespan* sebesar 2,14 % jika dibandingkan penjadwalan tanpa metode *overlapping*

DAFTAR PUSTAKA

- Baker, K.R., 1974, *Introduction To Sequencing And Scheduling*, John Wiley & Sons, Inc., New York.
- Bedworth, D.D. and James E. Bailey, 1987, *Integrated Production Control System : Management, Analysis, design, 2E*, John Wiley & Sons, Canada.
- Blackstone Hoffman, Fogarty, 1991, *Production & Inventory Management : Second Edition*, South. Western Publishing Co, Cincinnati
- Buffa E.S, 1996, *Management Operasi Dan Produksi Modern*, Binarupa Aksara, Jakarta.
- Conway, R. W., Maxwell, W.L., and Miller, L. W., 1967, *Theory Of Scheduling* Reading, Massachusetts, Adison Wesley, Publishing Company.
- Gasperz, Vincent, 1998, *Production Planning and Inventory Control*. Pt. Gramedia Pustaka Utama, Jakarta.