

## BAB II TINJAUAN PUSTAKA

Pada subbab ini akan dijelaskan mengenai penelitian terdahulu yang akan dijadikan referensi dalam penelitian ini, penjelasan mengenai penjadwalan produksi, dan aturan *earliest due date*.

### 2.1 Penelitian Terdahulu

Penelitian yang telah dilakukan sebelumnya yang berhubungan dengan penjadwalan produksi diantaranya adalah:

1. Lazuardi (2015), melakukan penjadwalan pada perusahaan yang bertipe *generall* flowshop dengan adanya mesin paralel pada salah satu proses yang memiliki kapasitas berbeda di tiap mesinnya. Penelitian ini menggunakan algoritma berbasis *Earliest Due Date* (EDD) dengan aturan *Short Processing Time* (SPT), dengan tujuan untuk meminimasi total tardiness yang terjadi di perusahaan. Setelah dilakukan perhitungan total tardiness, akan dibuat jadwal produksi baru dan didapatkan hasil penurunan nilai total tardiness.
2. Ratnasari (2015), melakukan optimasi penjadwalan produksi di perusahaan plastik yang memproduksi *flexible packaging*. Perusahaan ini memiliki tipe produksi *hybrid flow shop* dengan aturan prioritas *First In First Out* (FIFO). Sistem yang dijalankan perusahaan masih belum baik karena terdapat beberapa pekerjaan yang tidak dikerjakan pada hari yang sama sehingga membebani produksi selanjutnya. Dengan menggunakan algoritma genetika dan *short processing time* untuk menyelesaikan masalah penjadwalan dengan tujuan untuk meminimasi nilai *makespan* dengan membandingkan hasil dari algoritma genetika dan hasil dari penjadwalan saat ini. Hasil yang didapatkan adalah penurunan nilai maksespan.

Penelitian yang dilakukan oleh Lazuardi (2015) dan Ratnasari (2015) akan menjadi rujukan pada penulisan penelitian ini dalam mengurutkan pekerjaan berdasarkan aturan EDD pada industri bordir yang memiliki tipe produksi *hybrid flow shop* dengan *priority dispatching rules* untuk meminimalkan *tardiness*.

Tabel 2.1  
Penelitian Terdahulu

Nomor	1	2	3
Nama Peneliti	Lazuardi (2015)	Ratnasari (2015)	Penelitian ini
Tipe Produksi	<i>General flowshop</i> dengan adanya mesin paralel	<i>Hybrid Flowshop</i>	<i>Hybrid Flowshop</i>
Metode	<i>Earliest due date</i>	Algoritma Genetika dan <i>Short Processing Time</i>	<i>Earliest due date</i> dengan <i>software</i> Matlab
Hasil	Didapatkan algoritma penjadwalan berbasis EDD dengan menyesuaikan keadaan dan batasan-batasan yang ada pada sistem produksi dengan mengurangi total tardiness sebanyak 28 hari	Nilai makespan dari hasil algoritma genetika mengalami penurunan sebesar 587,3 menit dibandingkan dengan menggunakan metode short processing time.	Hasil yang diharapkan dari penelitian ini adalah terjadinya penurunan nilai tardiness sehingga keterlambatan pesanan dapat berkurang dan bisa menerima pesanan sebanyak-banyaknya

## 2.2 Penjadwalan

Penjadwalan adalah aktivitas perencanaan untuk menentukan kapan dan di mana setiap operasi sebagai bagian dari pekerjaan secara keseluruhan harus dilakukan pada sumber daya yang terbatas. Menurut Baker (1974) penjadwalan adalah proses untuk melakukan tugas dengan menggunakan sumber-sumber yang tersedia pada waktu yang telah ditetapkan. Menurut Stevenson (1999) penjadwalan adalah membangun penentuan waktu penggunaan dari peralatan, fasilitas dan aktivitas manusia dalam suatu organisasi. Menurut Pinedo (2002) penjadwalan adalah proses pengambilan keputusan yang memegang peranan yang penting dalam manufaktur dan sistem produksi. Dengan kata lain, penjadwalan dapat diartikan suatu kegiatan perancangan berupa pengalokasian sumber daya baik mesin maupun tenaga kerja untuk menjalankan sekumpulan tugas sesuai prosesnya dalam jangka waktu tertentu.

Penjadwalan mempunyai beberapa elemen-elemen penting yang harus diperhatikan seperti *job*, operasi, mesin serta hubungan yang terjadi diantaranya:

### 1. *Job*

*Job* dapat diartikan sebagai suatu pekerjaan yang harus diselesaikan untuk mendapatkan suatu produk. *Job* biasanya terdiri dari beberapa operasi yang harus dikerjakan (minimal 1 operasi). Manajemen melalui perencanaan yang telah dibuat atau berdasarkan pesanan dari pelanggan memberikan *job* kepada bagian *shop floor*

untuk dikerjakan. Informasi yang dipunyai oleh suatu *job* dilakukan didalamnya, saat harus diselesaikan dan saat *job* mulai dikerjakan.

## 2. Operasi

Operasi adalah himpunan bagian dari *job*, untuk menyelesaikan suatu *job*, operasi-operasi dalam *job* diurutkan dalam suatu urutan pengerjaan tertentu. Urutan tersebut ditentukan pada saat perencanaan proses. Suatu operasi baru dapat dikerjakan apabila operasi atau proses yang mendahuluinya sudah dikerjakan terlebih dahulu. Setiap operasi mempunyai waktu proses, waktu proses ( $t_{ij}$ ) adalah waktu pengerjaan yang diperlukan untuk melakukan operasi tersebut. Waktu proses operasi untuk suatu *job* biasanya telah diketahui sebelumnya dan mempunyai nilai tertentu.

## 3. Mesin

Mesin adalah sumber daya yang diperlukan untuk mengerjakan proses penyelesaian suatu *job*. Setiap mesin hanya dapat memproses satu tugas pada saat tertentu.

### 2.2.1 Tujuan penjadwalan

Tujuan penjadwalan menurut Baker (1974) adalah sebagai berikut.

1. Meningkatkan produktifitas mesin, yaitu dengan mengurangi waktu mesin menganggur.
2. Mengurangi persediaan barang setengah jadi dengan jalan mengurangi jumlah rata-rata pekerjaan yang menunggu dalam antrian suatu mesin karena mesin tersebut sibuk.
3. 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 maka keterlambatan ini dapat dikurangi, baik waktu maupun frekuensi.

### 2.2.2 Definisi dan Notasi dalam Penjadwalan

Sebelum membahas teori yang berkenaan dengan penjadwalan yang akan dikerjakan pada mesin-mesin yang ada dalam sistem produksi, terlebih dahulu diberikan pengertian dari beberapa definisi yang digunakan dalam penjadwalan (Ginting, 2007).

#### 1. *Processing Time* ( $t_i$ )

Adalah waktu yang dibutuhkan untuk mengerjakan suatu pekerjaan. Dalam waktu proses ini sudah termasuk waktu yang dibutuhkan untuk persiapan dan pengaturan (*set-up*) selama proses berlangsung.

2. *Due Date* ( $d_i$ )

Adalah batas waktu dimana operasi terakhir dari suatu pekerjaan harus selesai. Batas waktu yang ditentukan untuk tugas yang telah lewat, yang akan dinyatakan dengan terlambat. Diasumsikan bahwa akan diberikan denda bila terlambat. *Due date* dinyatakan dengan  $d_i$ .

3. *Slack Time* ( $SL_i$ )

Adalah waktu tersisa yang muncul akibat dari waktu prosesnya lebih kecil dari *due date* nya.

$$SL_i = d_i - t_i \quad (2-1)$$

Sumber: Ginting (2007)

4. *Flow Time* ( $F_i$ )

*Flow time*, rentang waktu antara satu titik dimana tugas tersedia untuk diproses dengan suatu titik ketika tugas tersebut selesai. Jadi, *flow time* sama dengan *processing time* dijumlahkan dengan waktu ketika tugas menunggu sebelum diproses. *Flow time* dinyatakan dengan  $F_i$ .

$$F_i = C_i - r_i \quad (2-2)$$

$$F_i = t_i + W_i \quad (2-3)$$

Sumber: Ginting (2007)

5. *Completion Time* ( $C_i$ )

Adalah waktu yang dibutuhkan untuk menyelesaikan pekerjaan mulai dari saat tersedianya pekerjaan ( $t=0$ ) sampai pada pekerjaan tersebut selesai dikerjakan.

$$C_i = F_i + r_i \quad (2-4)$$

Sumber: Ginting (2007)

6. *Lateness* ( $L_i$ )

Adalah selisih antara *completion time* ( $C_i$ ) dengan *due date* nya ( $d_i$ ). Suatu pekerjaan memiliki *lateness* yang bernilai positif apabila pekerjaan tersebut diselesaikan setelah *due date*. Pekerjaan tersebut akan memiliki keterlambatan yang negatif. Sebaliknya jika pekerjaan diselesaikan setelah batas waktunya, pekerjaan tersebut memiliki keterlambatan yang positif.

$$L_i = C_i - d_i, \quad (2-5)$$

$L_i \leq 0$ , saat penyelesaian memenuhi batas waktu

$L_i > 0$ , saat penyelesaian melebihi batas waktu

Sumber: Ginting (2007)

### 7. *Tardiness* ( $T_i$ )

Adalah ukuran waktu terlambat yang bernilai positif jika suatu pekerjaan dapat diselesaikan lebih cepat dari *due date*, pekerjaan tersebut akan memiliki keterlambatan yang negatif. Sebaliknya jika pekerjaan diselesaikan setelah batas waktunya, pekerjaan tersebut memiliki keterlambatan positif. *Tardiness* dinyatakan dengan  $T_i$ , dimana  $T_i$  adalah maksimum dari  $\{0, L_i\}$ .

### 8. *Maskepan* (M)

Adalah total waktu penyelesaian pekerjaan-pekerjaan mulai dari urutan pertama yang dikerjakan pada mesin atau *work center* pertama sampai kepada urutan pekerjaan terakhir pada mesin atau *work center* terakhir.

$$M = \sum_{i=1}^{n+1} S_{[i-1],[i]} + \sum_{i=1}^n t_i \quad (2-6)$$

Sumber: Ginting (2007)

### 9. *Heuristic*

Prosedur penyelesaian suatu masalah atau aturan ibu jari (*rule of thumb*) yang ditunjukkan untuk memproduksi hasil yang baik tetapi tidak menjamin hasil yang optimal.

## 2.3 *Input dan Output Penjadwalan*

### 2.3.1 *Input Penjadwalan*

Pekerjaan-pekerjaan yang merupakan alokasi kapasitas untuk setiap pesanan, penugasan prioritas *job*, dan pengendalian jadwal produksi membutuhkan informasi terperinci, dimana informasi-informasi tersebut akan menyatakan input dari sistem penjadwalan. kebutuhan-kebutuhan kapasitas dari setiap pesanan yang dijadwalkan dalam hal jumlah dan macam sumber daya yang digunakan. Untuk produk-produk tertentu, informasi ini bisa diperoleh dari lembar kerja operasi dan *bill of material* (BOM). Kualitas dari keputusan-keputusan penjadwalan sangat dipengaruhi oleh ketetapan estimasi input-input tersebut. Oleh karena itu, pemeliharaan catatan terbaru tentang status tenaga kerja dan peralatan yang tersedia, dan perubahan kebutuhan kapasitas yang diakibatkan perubahan desain produk/ proses menjadi sangat penting.

### 2.3.2 *Output Penjadwalan*

Untuk memastikan bahwa suatu aliran kerja yang lancar akan melalui tahapan produksi, maka sistem penjadwalan harus membentuk aktivitas-aktivitas *output* sebagai berikut (Ginting R, 2007).

1. Pembebanan (*loading*)

Pembebanan melibatkan penyesuaian kebutuhan kapasitas untuk order- order yang diterima/diperkirakan dengan kapasitas yang tersedia. Pembebanan dilakukan dengan menugaskan order-order fasilitas-fasilitas, operator-operator, dan peralatan tertentu.

2. Pengurutan (*sequencing*)

Pengurutan merupakan penugasan tentang order-order mana yang diprioritaskan untuk diproses dahulu bila suatu fasilitas harus memproses banyak *job*.

3. Prioritas *Job* (*dispatching*)

*Dispatching* merupakan prioritas kerja tentang job-job mana yang diseleksi dan diprioritaskan untuk diproses.

4. *Updating Schedules*

Pelaksanaan jadwal biasanya selalu ada masalah baru yang berbeda dari saat pembuatan jadwal, maka jadwal harus segera di-update bila ada permasalahan baru yang memang perlu diakomodasi.

5. Pengendalian Kinerja Penjadwalan

Pengendalian kinerja penjadwalan dilakukan dengan:

- a. Meninjau kembali status setiap pesanan pada saat melalui sistem tertentu.
- b. Mengatur kembali urutan-urutan, misalnya *expediting order-order* yang jauh dibelakang atau mempunyai prioritas utama.

6. *Up-dating* Jadwal

*Up-dating* jadwal dilakukan sebagai refleksi kondisi operasi yang terjadi dengan merevisi prioritas-prioritas.

## 2.4 Penjadwalan Produksi

Penjadwalan produksi secara umum didefinisikan sebagai penetapan waktu dari penggunaan peralatan, fasilitas, dan aktivitas manusia dalam sebuah organisasi (Adam dan Elbert, 1992). Penjadwalan produksi mencakup tahapan *loading*, *sequencing*, dan *detailed scheduling*. Pada tahap *loading*, setiap *job* ditentukan prosesnya, kemudian beban (*load*) setiap mesin ditentukan melalui pekerjaan yang harus diproses, dan ditentukan urutan pengerjaan *job* yang dikenal dengan sebutan *sequencing*. Dari urutan tersebut diatur waktu mulai dan selesainya pekerjaan melalui penjadwalan secara mendetail.

## 2.5 Penjadwalan *Hybrid Flow Shop*

Dalam bukunya Ginting (2009), menurut Baker (1974) penjadwalan *flowshop* merupakan suatu pergerakan unit-unit yang terus-menerus melalui suatu rangkaian stasiun-stasiun kerja yang disusun berdasarkan produk. *Hybrid Flow Shop* adalah generalisasi dari permasalahan *flowshop* klasik dengan adanya beberapa mesin paralel setiap stage atau tahap dari suatu proses. (Oguz, Janiak, & Lichtenstein, 2001).

Menurut Ruiz & Rodriguez (2010) *Hybrid Flow Shop* merupakan lingkungan manufaktur secara umum dimana sejumlah  $n$  job akan diproses pada  $m$  stage/tahap secara berurutan. Ada beberapa karakteristik yang sama untuk *Hybrid Flow Shop* yakni:

1. Jumlah tahapan proses  $m$  minimal dua buah -Setiap tahapan memiliki  $M^{(k)} \geq 1$  mesin dalam bentuk paralel dan paling tidak di salah satu stage  $M^{(k)} > 1$ .
2. Semua job diproses mengikuti alur produksi yang sama seperti stage 1, stage 2, sampai dengan stage  $m$ . Sebuah job mungkin saja melewati beberapa stage dimana job ini tetap diproses paling tidak di salah satu stage tersebut.

## 2.6 Aturan *Earliest Due Date*

Aturan lainnya yang perlu diketahui ialah aturan *Earliest Due Date* (EDD). Aturan ini menyebutkan bahwa pengurutan pekerjaan berdasarkan batas waktu (*Due Date*) tercepat. Pekerjaan dengan saat jatuh tempo paling awal harus dijadwalkan terlebih dahulu daripada pekerjaan dengan saat jatuh tempo belakangan. Aturan ini bertujuan untuk meminimasi kelambatan maksimum (*Maksimum Lateness*) atau meminimasi ukuran kelambatan maksimum (*Maksimum Tardiness*) suatu pekerjaan. Buruknya, aturan ini akan menyebabkan jumlah pekerjaan yang terlambat akan menjadi besar serta akan menambah ukuran kelambatan rata-rata (*Mean Tardiness*) (Bedworth, 1982).

Proses pengerjaan *job* pada *Earliest Due Date*, dilakukan dengan mengerjakan *job* dengan *due date* yang paling awal (kecil) dijadwalkan pada urutan yang pertama. Adapun langkah penjadwalannya adalah sebagai berikut.

1. Urutkan pekerjaan berdasarkan *Earliest Due Date* atau batas waktu terawal/pendek.
2. Terapkan hasil *Earliest Due Date* pada masing-masing prosesor secara berurutan.

## 2.7 Teknik *Priority Dispatching*

Menurut Kusuma (1999) *dispatching* adalah salah satu jenis metode pejadwalan dimana waktu siap dari setiap mesin ditentukan sedemikian rupa sehingga berurutan naik.

Keputusan pemilihan produk yang akan di proses dapat dilakukan pada saat mesin siap menerima produk (mesin kosong).

Pada teknik *priority dispatching* ditentukan aturan prioritas untuk memilih satu operasi diantara operasi-operasi yang mengalami konflik pada mesin  $m^*$  pada setiap tahap. Aturan prioritas ini harus dapat menjamin agar tidak terjadi dua operasi atau lebih yang mempunyai skala prioritas yang sama. Untuk itu biasanya dipakai dua buah aturan prioritas yaitu aturan prioritas yang pertama dan aturan prioritas kedua.

### 2.7.1 Pemilihan Aturan Prioritas

Penjadwalan dengan pendekatan *heuristic* menggunakan aturan pengurutan atau *priority dispatching*, pada *priority dispatching* ditentukan aturan prioritasnya untuk memilih suatu operasi diantara operasi-operasi yang mengalami gangguan. Penentuan prioritas bergantung pada tujuan yang ingin dicapai. Beberapa aturan prioritas menurut Fogarty (1991), adalah:

1. *Random* (R)

Pemilihan operasi secara random artinya setiap pekerjaan yang akan dikerjakan diurutkan secara *random* (tiap *job* mempunyai kemungkinan yang sama untuk dipilih).

2. *Most Working Remaining* (MWR)

Prioritas tertinggi diberikan pada operasi dengan sisa waktu proses terlama. Aturan prioritas ini cocok digunakan untuk menghasilkan jadwal dengan *makespan* terpendek.

3. *Least Working Remaining* (LWR)

Prioritas tertinggi diberikan pada operasi dengan sisa waktu proses terpendek. Aturan ini biasanya meminimasi *flow time* rata-rata.

4. *Most Operation Remaining* (MOR)

Prioritas tertinggi diberikan pada operasi dengan *successor* terbanyak.

5. *Fewest Operation* (FO)

Prioritas tertinggi diberikan pada produk dengan sisa operasi paling sedikit. Aturan ini biasanya mengurangi WIP, waktu penyelesaian produk, dan rata-rata waktu keterlambatan.

6. *Shortest Processing Time* (SPT)

Prioritas tertinggi diberikan pada operasi dengan waktu proses terpendek, aturan ini biasanya meminimasi WIP, rata-rata keterlambatan (*mean lateness*) dan waktu penyelesaian rata-rata (*mean flow time*) produk.

7. *First Come First Serve* (FCFS)

Prioritas tertinggi diberikan pada operasi yang masuk lebih dahulu. Aturan ini untuk tipe organisasi dimana konsumen mementingkan waktu pelayanan.

8. *Earliest Due Date* (EDD)

Prioritas tertinggi diberikan pada produk dengan *due date* (batas waktu penyelesaian) terpendek. Aturan ini berjalan dengan baik bila waktu proses *job-job* relatif sama.

9. *Critical Ratio* (CR)

Urutkan *job* berdasarkan CR terkecil (mengurangi *lateness*)

$$CR = \frac{\text{Due date} - \text{Now}}{\text{Remaining lead time}} \quad (2-7)$$

Sumber: Fogarty (1991)

10. *Slack Time* (ST)

Merupakan variasi dari EDD. Aturan ini akan melibatkan selisih waktu proses dengan batas waktu yang sudah ditetapkan.

$$ST = \text{Remaining time} - \text{Setup} - \text{Run time} \quad (2-8)$$

Sumber: Fogarty (1991)

11. *Slack Per Remaining Operation* (S/OPN)

Merupakan variasi dari ST, urutkan *job* berdasarkan S/OPN terkecil (aturan ini mengurangi *lateness*).

$$S/OPN = \frac{(\text{Due date} - \text{Present date}) - \text{Remaining Processing time}}{\text{Remaining number of operations}} \quad (2-9)$$

Sumber: Fogarty (1991)

12. *Least Setup* (LSU)

Urutkan *job* berdasarkan waktu setup terkecil (aturan ini mengurangi *makespan*).

## 2.8 Gantt Chart

Menurut Stevenson (1999), *Gantt Chart* adalah chart yang digunakan sebagai gambaran bagan untuk memuat dan menjadwalkan sesuatu yang dapat dijadwalkan. *Gantt Chart* dapat digunakan dengan cara yang berbedabeda. Tujuan dari *Gantt Chart* adalah mengorganisir dan memperjelas penggunaan sumber daya secara nyata atau sesuai dengan yang diharapkan dalam kurun waktu tertentu. dalam beberapa kasus, skala waktu diwakili oleh sumbu horisontal (sumbu *x*) dan sumber daya yang akan dijadwalkan ditulis pada sumbu vertikal (sumbu *y*). Pemakaian sumber daya itu dapat dilihat didalam bagan tersebut. Ada dua jenis *Gantt Chart* yaitu:

1. *Load chart*

Sebuah *Gantt Chart* yang menunjukkan pembebanan dan waktu mengganggu sekelompok mesin atau departemen. *Chart* ini juga menunjukkan kapan pekerjaan tersebut mulai untuk dijadwalkan dan harus selesai, dan dimana seharusnya menempatkan waktu mengganggu tersebut.

## 2. *Schedule Chart*

Sebuah *Gantt Chart* yang menunjukkan pesanan yang dipesan atau pekerjaan yang sedang dalam tahap pengerjaan dan apakah order tersebut dapat terjadwalkan. *Schedule chart* biasanya sering digunakan oleh manager untuk merekam segala jenis pekerjaan yang sedang dalam proses pengerjaan. *Chart* ini mengindikasikan pekerjaan mana yang terjadwalkan dan mana yang seharusnya berada diawal.

## 2.9 MATLAB

MATLAB merupakan sebuah singkatan dari *Matrix Laboratory*, yang pertama kali dikenalkan oleh *University of New Mexico* dan *University of Stanford* pada tahun 1970. *Software* ini pertama kali memang digunakan untuk keperluan analisis numerik, aljabar linier dan teori tentang matriks. Saat ini, kemampuan dan fitur yang dimiliki oleh Matlab sudah jauh lebih lengkap dengan ditambahkannya *toolbox* yang sangat luar biasa. Beberapa manfaat yang didapatkan dari MATLAB antara lain:

1. Perhitungan matematika
2. Komputasi numerik
3. Simulasi dan pemodelan
4. Visualisasi dan analisis data
5. Pembuatan grafik untuk keperluan sains dan teknik
6. Pengembangan aplikasi, misalnya dengan memanfaatkan GUI.

MATLAB dapat dipadang sebagai sebuah kalkulator dengan fitur yang lengkap. Kita pernah menggunakan kalkulator dengan dengan fasilitas minimal, misalnya hanya terdapat fasilitas penambahan, pengurangan perkalian dan pembagian. Kalkulator yang lebih lengkap lagi adalah kalkulator *scientific* dimana fasilitas yang diberikan tidak hanya yang disebutkan di atas, melainkan sudah ada fungsi-fungsi trigonometri, bilangan kompleks, akar kuadrat dan logaritma. MATLAB mirip dengan kalkulator tersebut, tetapi dengan fitur-fitur yang lengkap diantaranya dapat digunakan untuk memrogram, aplikasi berbasis GUI dan lengkap dengan *toolbox* yang dapat dimanfaatkan untuk memecahkan masalah sains dan teknik.