Bab 2
Landasan Teori

2.1. Perencanaan dan Pengendalian Produksi

Perencanaan produksi merupakan suatu fungsi manajemen, dimana dalam perencanaan ditentukan usaha dan tindakan yang perlu diambil oleh pihak perusahaan, serta mempertimbangkan masalah yang akan timbul pada masa yang akan datang berdasarkan penyesuaian permintaan (*demand*) yang berasal dari peramalan dengan keseluruhan kemampuan yang ada.

Perencanaan dan pengendalian produksi adalah suatu proses perencanaan dan pengorganisasian mengenai pekerjaan, bahan baku, mesin dan peralatan serta modal yang diperlukan untuk memproduksi barang pada suatu periode tertentu sesuai dengan yang diramalkan dan kemampuan dari perusahaan.

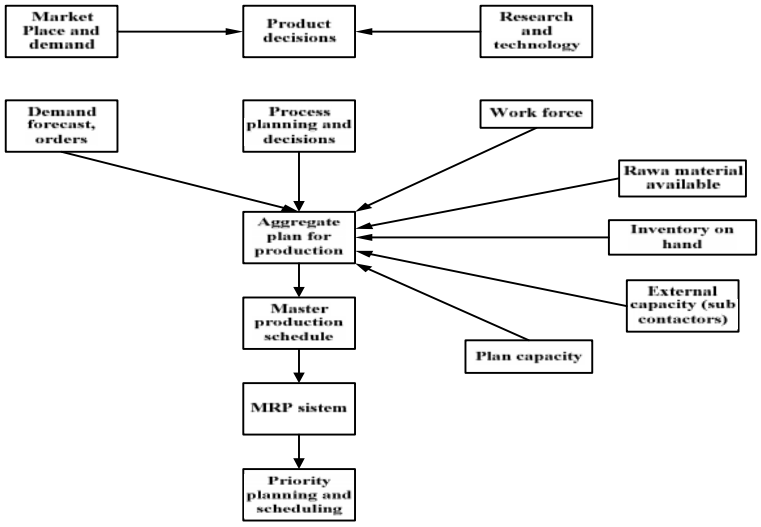
Proses perencanaan dan pengendalian produksi dapat membantu perusahaan dalam mengoptimalkan sumber daya yang dimilikinya untuk mencapai tujuan yang diinginkan. Proses perencanaan dan pengedalian produksi tersebut terdiri dari tahapan-tahapan yang telah tersusun secara sistematis dan saling terkait satu sama lain.

Pada dasarnya fungsi dasar yang harus dipenuhi oleh aktivitas perencanaan dan pengendalian produksi (Kusuma; 2004) adalah:

1. Meramalkan permintaan produk yang dinyatakan dalam jumlah produk sebagai fungsi dari waktu.
2. Menetapkan jumlah dan saat pemesanan bahan baku serta komponen secara ekonomis dan terpadu.
3. Menetapkan kesimbangan antara tingkat kebutuhan produksi, teknik pemenuhan pesanan serta memonitor tingkat persediaan produk jadi setiap saat, membandingkannya dengan rencana persediaan dan melakukan revisi atas rencana prodoksi pada saat yang ditentukan.

4. Membuat jadwal induk produksi, penugasan, pembebanan mesin dan tenaga kerja yang terperinci sesuai dengan ketersediaan kapasitas dan fluktuasi permintaan pada suatu periode.

Ruang lingkup sistem produksi secara luas digambarkan pada gambar 2.1. Berdasarkan gambar tersebut, manajer operasi tidak hanya menerima masukan dari departemen pemasaran yang berkaitan dengan peramalan permintaan, namun ia menangani pula data *financial*, pekerja, kapasitas dan bahan baku yang tersedia.



Gambar 2.1. Sistem Perencanaan Produksi

Tahapan-tahapan perencanaan dan pengendalian produksi antara lain peramalan, prencanaan kebutuhan material, perencanaan kebutuhan kapasitas dan penjadwalan.

2.2. Pengertian Penjadwalan

Penjadwalan diartikan sebagai rencana pengaturan kerja serta pengalokasian sumber, baik waktu maupun fasilitas untuk setiap operasi yang harus diselesaikan. Menurut Kenneth R. Baker penjadwalan adalah sebagai proses pengalokasian sumber-sumber untuk memilih sekumpulan tugas dalam jangka waktu tertentu. Fungsinya adalah sebagai alat untuk pengambilan keputusan yaitu untuk menetapkan suatu jadwal. Definisi lain mengatakan bahwa penjadwalan ialah

proses pengurutan pembuatan produk secara menyeluruh pada sejumlah mesin dalam jangka waktu tertentu.

Dari sekian banyak definisi penjadwalan yang telah ada pada saat ini, intinya adalah:

- Penjadwalan berfungsi sebagai alat pengambil keputusan.
- Penjadwalan merupakan teori yang berinsi prinsip-prinsip dasar, model, teknik dan kesimpulan logis dalam pengambilan keputusan.

Untuk menyelesaikan masalah penjadwalan yang dihadapi, dapat digunakan beberapa pendekatan. Pendekatan tersebut dibagi menjadi dua yaitu:

- Pendekatan yang lebih modern mencakup gabungan antara metode penelitian operasional, intelegensia tiruan, simulasi kejadian dan ide-ide yang diambil dari teori *control*.
- Pendekatan tradisional meliputi metode-metode penelitian operasional.

2.2.1. Tujuan Penjadwalan

Beberapa tujuan yang ingin dicapai dengan dilaksanakannya penjadwalan produksi adalah:

- Meningkatkan utilitas atau penggunaan sumber daya atau mengurangi waktu tunggunya, sehingga total waktu proses dapat berkurang dan produktivitas dapat meningkat.
- Mengurangi *makespan* yang juga berarti menurunkan *flow time* rata-rata dan *work in pro*cess rata-rata.
- Mengurangi persediaan barang setengah jadi atau mengurangi sejumlah pekerjaan yang menunggu dalam antrian ketika sumber daya yang ada masih mengerjakan tugas lain. Teori Baker mengatakan, “Jika aliran kerja suatu jadwal konstan, maka antrian yang mengurangi rata-rata waktu aliran akan mengurangi waktu persediaan”.
- Meminimasi biaya produksi.
- Mengurangi persediaan barang setengah jadi dengan jalan mengurangi jumlah pekerjaan yang menunggu antrian suatu mesin yang dalam keadaan sibuk. Hal

ini bertujuan untuk menghindari biaya *flow time*, yaitu biaya penyimpanan produksi setengah jadi.
- Memenuhi keinginan konsumen, naik dalam hal kualitas produk yang dihasilkan maupun dalam ketepatan waktu.
- Membantu dalam pengambilan keputusan sehingga penambahan biaya yang mahal dapat dihindarkan.

Pada saat merencanakan suatu jadwal produksi, yang harus dipertimbangkan adalah ketersediaan sumber daya yang dimiliki, baik berupa tenaga kerja, peralatan (*processor*) ataupun bahan baku. Karena sumber daya yang dimiliki dapat berubah-ubah (terutama *operator* dan bahan baku), maka penjadwalan merupakan proses yang dinamis.

Adapun tipe keputusan yang akan diperoleh dari pelaksanaan penjadwalan tersebut berupa:
- Pembebanan (*loading*)
- Pengurutan pekerjaan (*sequencing*)
- Penugasan (*dispatching*)
- Pengurutan operasi suatu *job* (*routing*)
- Penentuan waktu mulai dan selesai pekerjaan (*timing*)

Persoalan penjadwalan timbul apabila, beberapa pekerjaan (*job*) akan dikerjakan bersamaan, sedangkan sumber daya seperti mesin (peralatan) jumlahnya terbatas. Untuk mencapai hasil yang optimal dengan keterbatasan sumber daya yang dimiliki, maka diperlukan adanya penjadwalan sumber-sumber tersebut secara efisien. Terdapat berbagai model penjadwalan yang telah dikembangkan untuk mengatasi persoalan penjadwalan tersebut. Penjadwalan mempunyai beberapa elemen-elemen penting yang harus diperhatikan seperti *job*, operasi, mesin serta hubungan yang terjadi diantaranya:

a. *Job*

Job dapat didiartikan 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.

b. 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.

c. Mesin

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

2.2.2. Klasifikasi Permasalahan Penjadwalan

Permasalahan penjadwalan dapat diklasifikasikan berdasarkan faktor-faktor, yaitu:

a. Mesin
   - Mesin tunggal
   - 2 mesin
   - M mesin

b. Aliran proses
   - *Job shop*
   - *Flow shop*

c. Pola kedatangan
   - Statis
   - Dinamis

d. Elemen penjadwalan
   - Deterministik
   - Stokastik

1. Berdasarkan jumlah mesin

Dibedakan menjadi dua bagian yaitu:

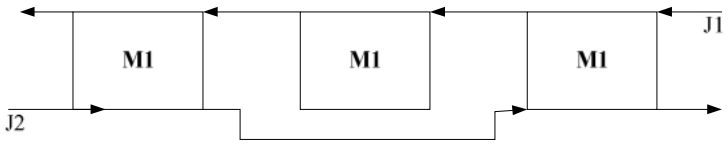
- Penjadwalan pada mesin tunggal
- Penjadwalan pada mesin ganda

2. Pola aliran proses

Pola aliran proses produksi dapat dibedakan menjadi, yaitu:

a. *Job shop*

Pada aliran *job shop*, masing-masing *job* memiliki urutan proses operasi yang unik. Setiap *job* bergerak dari satu mesin atau stasiun kerja menuju mesin/stasiun kerja lainnya dengan pola yang *random*. Lintasan prosesnya dapat dilihat pada gambar dibawah ini:



Gambar 2.2. Lintasan Proses *Job Shop*

Proses *job shop* mempunyai karakteristik dari pengurutan peralatan yang sama berdasarkan fungsi. Sebagaimanan aliran *job* dari stasiun kerja ke stasiun kerja lainya, atau dari suatu departemen ke departemen lainnya, maka karakteristik proses *job shop* adalah sebagai berikut:

- Proses penanganan material dan peralatan produksi multi-guna dapat diatur dan dimodifikasi untuk menangani produk yang berbeda.
- Produk-produk yang berbeda diproses dalam *lot-lot* atau *batch*.
- Pemrosesan *order-order* membutuhkan pengendalian dan perencanaan yang terperinci sehubungan dengan variasi pola-pola aliran dan pemisahan stasiun-stasiun kerja.

- Pengendalian membutuhkan informasi tentang *job* dan *shop* yang terperinci, meliputi urutan proses, prioritas *order*, waktu yang dibutuhkan oleh setiap *job* stasiun dari setiap *job n process*, kapasitas dari stasiun kerja dan kapasitas yang dibutuhkan dari stasiun kerja kritis pada suatu periode.
- Beban-beban stasiun kerja yang berbeda secara mencolok, masing-masing memiliki persentase utilitas kapasitas yang berbeda.
- Ketersediaan sumber-sumber meliputi: material, personal dan peralatan harus dikoordinasikan dengan perencanaan *order*.
- Sejumlah material *work in process* cenderung meningkat. Hal ini dalam aliran proses menyebabkan antrian-antrian dan *work in process* yang panjang.
- Menggunakan teknik-teknik penjadwalan tradisional, total waktu dari awal operasi pertama sampai operasi terakhir, relatif panjang dibandingkan dengan total waktu operasi.
- Para pekerja langsung biasanya memiliki *skill* (keahlian) yang lebih tinggi dan lebih terlatih dari pada pekerja untuk operasi *flow process*.

Dua permasalahan utama yang hendak diselesaikan dengan menggunakan penjadwalan:

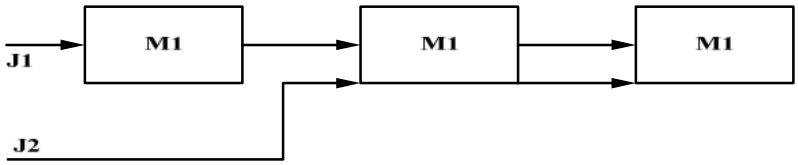
- Penentuan mesin yang akan digunakan (pengalokasian mesin) untuk menyelesaikan suatu proses produksi.
- Penjadwalan penetuan waktu pemakaian mesin tersebut (pengurutan).

b. *Flow shop* yang cenderung memiliki keamanan urutan operasi (*routing*) untuk semua *job*. *Flow shop* dibedakan menjadi:
   - *Pure flow shop*, yaitu *flow shop* yang memiliki jalur yang sama untuk semua tugas. Lintasan prosesnya dapat dilihat lebih jelas pada Gambar 2.3. dibawah ini.



Gambar 2.3. Lintasan Proses *Pure Flow Shop*

- *General flow shop*, yaitu *flow shop* yang memiliki pola aliran berbeda. Ini disebabkan adanya variasi dalam pekerjaan tugas, sehingga tugas yang datang tidak harus dikerjakan pada semua mesin. Jadi mungkin suatu proses dilewati. Lintasan prosesnya dapat dilihat pada Gambar 2.4. dibawah ini.



Gambar 2.4. Lintasan Proses *General Flow Shop*

3. Pola kedatangan pekerjaan (*job*)

Pola kedatangan pekerjaan dapat dibedakan atas:

a. Pola kedatangan statis, yaitu pola dimana pekerjaan datang secara bersamaan dan semua fasilitas tersedia saat kedatangan *job*.
b. Pola kedatangan dinamis, yaitu pola dimana pekerjaan datang secara acak atau kedatangan pekerjaan tidak menentu.

4. Sifat informasi

Dibagi menjadi dua bagian yaitu:

a. Informasi bersifat deterministik, yaitu suatu informasi yang didalamnya terdapat kepastian tentang pekerjaan dan mesin, misalnya mengenai waktu kedatangan pekerjaan dan waktu proses.
b. Informasi bersifat stokastik, yaitu model didalamnya terdapat kepastian mengenai pekerjaan dan mesin.

Informasi-informasi yang berhubungan dengan karakteristik *job*, yaitu saat kedatangan, batas waktu penyelesaian, perbedaan kepentingan diantara *job* yang dijadwalkan, banyak operasi dan waktu proses tiap operasi. Disamping itu terdapat pula informasi yang menyangkut karakteristik mesin seperti jumlah mesin, kapasitas, fleksibilitas serta efisiensi penggunaan yang berbeda untuk *job* yang berbeda.

2.2.3. *Input* Sistem Penjadwalan

Dalam melakukan aktivitas penjadwalan diperlukan *input* berupa kebutuhan kapasitas dari *order-orde*r yang akan dijadwalkan baik itu jenis serta jumlah sumber daya yang akan digunakan. Informasi ini dapat diperoleh dari, yaitu:

a. Lembar kerja operasi (OPC) yang berisi keterampilan dan peralatan yang dibutuhkan serta waktu standar pengerjaan.
b. *Bill of material* (BOM) yang berisi kebutuhan-kebutuhan akan komponen, sub komponen dan bahan pendukung.
c. Catatan terbaru mengenai status tenaga kerja, peralatan yang tersedia yang akan berpengaruh pada kualitas keputusan penjadwalan yang diambil.

2.2.4. *Output* Sistem Penjadwalan

Untuk memastikan bahwa suatu aliran kerja yang lancer melalui tahapan produksi, maka sistem penjadwalan harus dibentuk aktivitas-aktivitas *output* sebagai berikut:

1. Pembebanan (*loading*)

Pembebanan melibatkan penyesuaian kebutuhan kapasitas untuk *order-order* yang diterima atau diperkirakan dengan kapasitas yang tesedia. Pembebanan dilakukan dengan menugaskan *order-order* pada fasilitas-fasilitas, *operator-operator* dan peralatan tertentu.

2. Pengurutan (*sequencing*)

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

3. Prioritas *job* (*dispatching*)

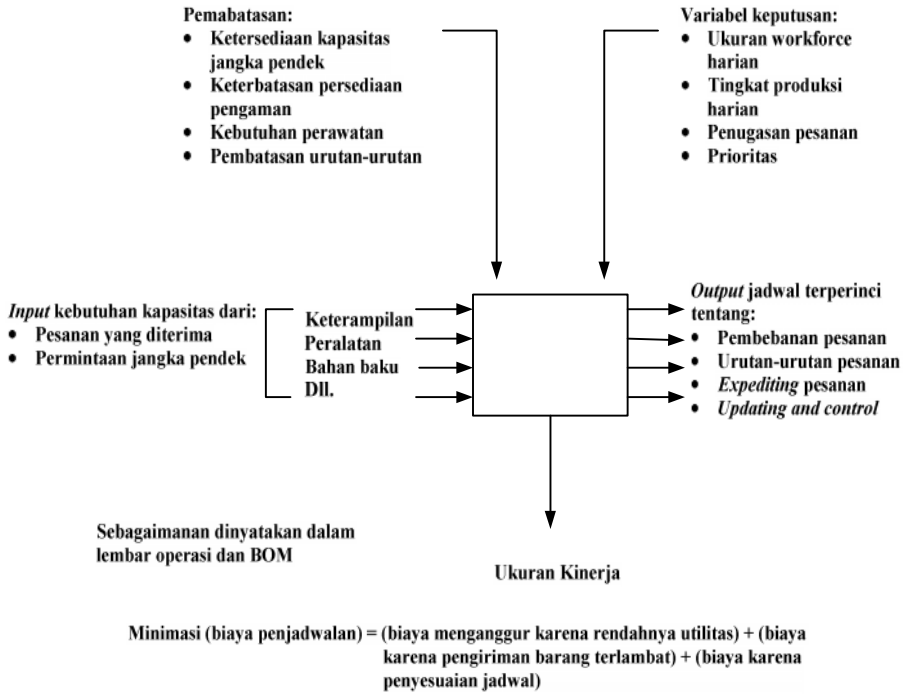
Prioritas *job* merupakan prioritas kerja tentang *job-job* mana yang akan diseleksi dan diprioritaskan untuk diproses.

4. Pengendalian kinerja penjadwalan, dilakukan dengan cara, yaitu:
   - Meninjau kembali status *order-order* pada saat melalui sistem tertentu.

- Mengatur kembali urutan-urutan, misalnya *expediting*, *order-order* yang jauh dibelakang atau mempunyai prioritas utama.

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

Elemen-elemen *input-output*, prioritas dan ukuran kinerja dari sistem penjadwalan akan tampak seperti dibawah ini:



Gambar 2.5. Elemen-elemen Sistem Penjadwalan (Ginting;2007)

2.2.5. Istilah –istilah dalam Penjadwalan

Dalam melakukan sebuah penjadwalan, terdapat beberapa istilah yang digunakan diantaranya adalah:

- *Processing time* (t_i)

  Waktu yang diperlukan untuk menyelesaikan satu operasi termasuk persiapan dan pengaturan proses.

- *Due date* (d_i)

  Batas waktu yang diperbolehkan untuk menyelesaikan suatu pekerjaan.

- *Completion time* (c_i)

  Rentang waktu mulai dari awal (t=0) sampai pekerjaan selesai dikerjakan.

- *Lateness* (L_i)

  Perbedaan antara *completion time* dengan *due date*, sehingga bisa positif (+) atau negatif (-).

  $L_i = c_i - d_i < 0$ ……(2.1)

  Keterangan: negatif yaitu saat penyelesaian memenuhi batas.

  $L_i = c_i - d_i > 0$ ……(2.2)

  Keterangan: positif yaitu saat penyelesaian memenuhi batas = *tardy job*.

- *Tardiness* (T_i)

  Keterlambatan penyelesaian suatu pekerjaan dari saat *due date*.

- *Slack time* (S_i)

  Waktu sisa yang tersedia bagi suatu pekerjaan (waktu proses – *due date*).

  $S_i = d_i – t_i$ ……...(2.3)

- *Flow time* (F_i)

  Waktu antara saat dimana pekerjaan 1 telah siap untuk dikerjakan sampai pekerjaan selesai.

- *Waiting time* (W_i)

  Waktu tungu pekerjaan 1 dari saat pekerjaan siap dikerjakan sampai saat operasi pendahulu selesai.

- *Makespan* (M_s)

  Jangka penyelesaian suatu penjadwalan (penjumlahan seluruh waktu proses).

  $M_s = C_{max}$ ……..(2.4)

- *Ready time* (R_i)

  Menunjukan saat pekerjaan ke-I dapat dikerjakan (siap dijadwalkan).

2.2.6. Tipe Lingkungan Penjadwalan

Lingkungan penjadwalan dalam suatu sistem produksi dapat dibedakan beberapa macam yang masing-masing mempunyai karateristik yang berbeda. Tipe-tipe lingkungan penjadwalan dalam sistem produksi, antara lain:

1. *Classic Job Shop*

   Karakteristik sistem produksi ini adalah produknya diskrit, alirannya kompleks, *job* unik dan *part-part* tidak *multi purpose* (kegunaan).

2. *Open Job Shop*

   Sistem produksi ini hanpir sama dengan *job shop*, tetapi perbedaannya pada *job* yang berulang dan *part* yang *multi purpose*. Selain pada sistem produksi ini *job-job* yang dikerjakan sering kali mempunyai alternatif *routing*.

3. *Batch Shop*

   Proses produksinya bisa diskrit atau kontinyu, alirannya kurang kompleks, banyak *job* berulang, *part multi purpose*, pengelompokkan dan penentuan ukuran *lot* menjadi suatu yang penting.

4. *Flow Shop*

   Proses produksinya bisa diskrit atau kontinyu, aliran *linear*, *job* mempunyai kemiripan yang tinggi, pengelompokkan dan penentuan ukuran *lot* menjadi suatu yang penting.

5. *Batch/Flow Shop*

   Mirip dengan *flow shop*, dengan perbedaan mempunyai proses *batch* yang kontinyu.

6. *Manufacturing Cell*

   Proses produksinya diskrit, mempunyai tipe *open job shop* atau *batch shop* yang terotomatis.

7. *Assembly Shop*

   Versi perakitan (*Assembly Version*) dari *open job shop* atau *batch shop*.

8. *Assembly Line*

   Volume produksinya tinggi dan variasinya rendah.

9. *Transfer Line*

   Sistem ini bercirikan volume produk sangat tinggi dan bervariasi rendah, fasilitas produksi yang *linear* dengan operasi yang terotomatis.

10. *Flexible Transfer Line*

    Versi yang lebih modern dari sel dan lini transfer dimaksudkan untk memperoleh keuntungan dari tingginya produksi ke *item job shop*.

2.3. Aturan Prioritas

Aturan prioritas digunakan untuk memenuhi *job* mana yang akan dikerjakan terlebih dahulu mengklasifikasikan aturan-aturan prioritas ke dalam 2 tipe, yaitu:

1. Aturan Prioritas Lokal

Pada autran prioritas ini penugasan didasarkan pada informasi yang berkaitan dengan *job* yang berada pada antrian suatu mesin secara individual. Aturan yang termasuk pada tipe ini adalah:

a. *Shortest Processing Time* (SPT)

   Prioritas tertinggi diberikan pada *job* yang memiliki waktu proses terpendek. Aturan ini cenderung mengurangi *work in process*, *mean flow* serta *mean lateness*.

b. *Least Work Remaining* (LWKR)

   Prioritas tertinggi diberikan pada *job* yang memiliki sisa waktu yang terpendek.

c. *First Come First Served* (FCFS)

d. *Most Work Remaining* (MWKR)

   Prioritas tertinggi diberikan pada *job* yang memiliki waktu proses terbanyak.

e. *Most Operation Remaining* (MOPNR)

   Prioritas tertinggi diberikan pada *job* yang memiliki waktu proses terbanyak dan terpanjang.

2. Aturan Prioritas Global

Aturan prioritas global memanfaatkan informasi atau status dari mesin-mesin yang lainnya. Aturan yang tergolong tipe ini adalah:

a. *Anticipates Work In Next Queue* (AWINQ)

   Prioritas tertinggi diberikan kepada operasi yang berbeda pada stasiun dengan antrian terpendek.

b. *First Of First On* (FOFO)

   Prioritas tertinggi diberikan kepada operasi yang selesai paling awal.

Selain itu juga pengklasifikasian aturan ini berdasarkan informasi-informasi yang bersifat dinamis, diantaranya:

1. Aturan statis

Tipe ini memandang setiap *job* mempunyai prioritas yang sama. Prioritas yang termasuk ke dalam tipe ini adalah:

a. *First Arrival at The Shop Served* (FASFS)

   Prioritas tertinggi diberikan pada *job* yang tiba paling awal di shop.

b. *Total Work* (TWORK)

   Prioritas yang tertinggi diberikan pada *job* yang memilki total proses untuk seluruh operasi yang dilaksanakan paling sedikit.

c. *Earliest Due Date* (EDD)

   Prioritas yang tertinggi diberikan pada *job* yang memiliki due date paling cepat.

2. Aturan dinamis

Tipe ini memberikan prioritas yang berbeda untuk operasi-operasi yang berbeda pada suatu *job*. Aturan yang termasuk dalam tipe ini adalah:

a. *Operation Due Date* (OPNDD)

   Due date operasi suatu *job* dapat ditentukan dari perbandingan *interval* antara *due date job* dengan waktu kedatangan operasi. Disini prioritas tertinggi diberikan pada operasi *due date* tercepat.

b. *First Come First Server* (FCFS)

   Prioritas yang tertinggi diberikan pada operasi yang terlebih dahulu masuk ke dalam antrian suatu mesin.

c. *Slack Time per Operation* (ST/O)

   Prioritas yang tertinggi diberikan pada *job* yang memiliki harga *slack time per operation* paling kecil.

d. *Slack Time* (ST)

   Slack time diperoleh dengan cara mengurangi wakrtu proses dari *due date*. Prioritas tertinggi diberikan pada *job* yang memiliki *slack time* paling kecil.

Untuk menyelesaikan permasalahan *job shop*, banyak cara yang dapat digunakan diantaranya dengan metode matematis, *heuristic* dan simulasi. Kebanyakan ntuk menyelesaikan permasalahan ini digunakan metode *heuristic*, salah satunya adalah adanya *priority rules*. Biasanya *priority rules* ini dipakai baik untuk *operation schedulling* maupun *dispatching*. Ada beberapa aturan yang bisa digunakan dalam *priority rules* ini, yaitu:

1. *Random*

   Memilih *job* dalam antrian dengan kemungkinan yang sama pada setiap *job*.
2. *Critical Ratio* (CR)

   Prioritas tertinggi diberikan pada *job* yang memiliki *critical ratio* terkecil.
3. *Most Work Remaining* (MWKR)

   Prioritas tertinggi diberikan pada *job* yang memiliki waktu proses terbanyak.
4. *Shortest Processing Time* (SPT)

   Prioritas tertinggi diberikan pada *job* yang memiliki waktu proses terpendek. Aturan ini cenderung mengurangi *work in process*, *mean flow* serta *mean lateness*.
5. *Least Work Remaining* (LWKR)

   Aturan ini mempertimbangkan *successive operation* yaitu semua operasi yang tergantung dari operasi yang bersangkutan.
6. *Least Set-Up* (LSU)

   Memilih *job* yang memiliki waktu *set-up* yang terkecil, dengan demikian akan meminimasi *change over time* (perubahan waktu yang berlebihan).

2.4. Kriteria Optimalitas Dalam Penjadwalan

Pemilihan suatu sistem penjadwalan, pendekatan atau teknik yang digunakan tergantung pada tujuan jadwal dan kriteria optimalitas yang menjadi titik berat dan menjadi perhatian manajemen. Sehingga tujuan-tujuan dan kebijakan manajemen adalah dasar dari suatu penjadwalan. Berikut ini adalah beberapa perhitungan atau pengukuran dan istilah yang sering digunakan dalam menentukan kriteria optimalitas atau performansi dalam suatu system produksi *job shop*:

- *Process Time* (t_i)

  Waktu yang diperlukan untuk menyelesaikan suatu operasi j, termasuk didalamnya waktu untuk persiapan (*setup*).

- *Makespan* (M_s)

  Bisa diesbut juga *flow time* maksimum, yaitu jangka waktu penyelesaian suatu penjadwalan yang merupakan jumlah seluruh waktu proses.

  $M_s = \sum_{j=1}^{n} t_j$ ……...(2.5)

- *Ready Time* (r_j)

  Waktu yang dibutuhkan suatu *job* pada saat siap untuk dijadwalkan.

- *Due Date* (d_i)

  Batas waktu penyelesaian yang ditentukan untuk *job* i atau batas waktu akhir suatu *job* boleh diselesaikan.

  $d_i = t_i + W_i$ ……(2.6)

- *Waiting Time* (W_i)

  Waktu tunggu seluruh operasi dari suatu *job*.

  $W_i = \sum w_j$ ……..(2.7)

- *Flow Time* (F_i)

  Lamanya *job* I berada dilantai pabrik. *Flow time* dihitung sejak *job* dijadwalkan sampai *job* selesai dikerjakan.

  $F_i = t_i + r_j$ …….....(2.8)

  $F_i = C_i – r_j$ …….....(2.9)

- *Completion Time* (C_i)

  Waktu yang dibutuhkan sejak satu *job* mulai di *set-up* sampai selesai diproses.

  $C_i = F_i + r_j$ …….....(2.10)

- *Mean Flow Time*

  Rata-rata waktu yang dihabiskan oleh setiap *job* dilantai produksi.

  $$F_s = \frac{1}{n}\sum_{t=1}^{n} F_i \quad \text{.....(2.11)}$$

- *Lateness* (L_i)

  Besarnya simpangan waktu penyelesaian *job* i terhadap *due date* yang telah ditentukan untuk *job* tersebut.

  $L_i = C_i - d_i$ .....(2.12)

  $L_i < 0$, saat penyelesaian memenuhi batas akhir.

  $L_i > 0$, saat penyelesaian melewati batas akhir.

- *Mean Lateness*

  $$L_s = \frac{1}{n}\sum_{t-1}^{n} L_i \quad \text{.....(2.13)}$$

- *Tardiness* (Ti)

  Keterlambatan penyelesaian suatu pekerjaan hingga saat *due date*. Ditunjukan oleh *lateness* yang berharga positif. Jika lateness berharga negative maka besarnya *tardiness* adalah nol.

  $T_i = \max(L_i, 0)$ .....(2.14)

- *Mean Tardiness*

  Rata-rata keterlambatan seluruh *job* yang dijadwalkan.

  $$T_s = \frac{1}{n}\sum_{t=1}^{n} T_i \quad \text{.....(2.15)}$$

- *Earliness* (E_i)

  Penyelesaian suatu *job* lebih awal dari *due date* atau *lateness* yang bernilai negatif.

  $E_i = \min(L_i, 0)$ .....(2.16)

- *Slack Time* (S_i)

  Waktu sisa yang tersedia bagi suatu *job*.

  $S_i = d_i - t_i$ .....(2.17)

- Utilitas Mesin (U)

  Merupakan bagian dari kapasitas mesin yang dibebani untuk menjalankan proses-proses yang dibutuhkan terhadap waktu yang tersedia.

  $$U = \frac{\sum_{n=i}^{n} t_i}{C_{max}} \quad \text{.....(2.18)}$$

- *Number of Tardy Job*

  Menunjukan kuantitas *job* yang terlambat.

  $N_t = \sum_{i=1}^{n} d_t$ .....(2.19)

  Dimana:

  $d_i = 1$ jika $T_i \geq 0$ .....(2.20)

  $d_i = 0$ jika $T_i \leq 0$ .....(2.21)

- *Critical Ratio* (CR)

  Prioritas tertinggi diberikan kepada *job* yang memiliki CR terkecil.

  $C = \frac{a_{j(t)}}{P_j}$ .....(2.22)

  Dimana:

  $a_{j(t)} = d_j - t$ .....(2.23)

  Keterangan:

  $a_{j(t)}$ = *allowance*

  d_j = *due date*

  P_j = waktu yang dibutuhkan untuk menyelesaikan operasi j, sehingga:

  $P_j = a_{j(t)} - S_j$ .....(2.24)

  S_j = *slack time*

Sedangkan kriteria optimalisasi penjadwalan yang sering digunakan pada saat ini adalah:

1. Berkaitan dengan waktu
   - Minimasi *mean flow time*
   - Minimasi *max flow time*
   - Minimasi *mean tardiness*
   - Minimasi *mean lateness*
   - Minimasi *max tardiness*

2. Berkaitan dengan ongkos

Kriteria ini lebih menekankan pada ongkos yang timbul dari penjadwalan yang dilakukan seperti ongkos penalty akibat keterlambatan, ongkos *flow time* dan ongkos *inventory*. Adapun tujuan akhir yang ingin dicapai adalah mendapatkan

ongkos yang minimal. Beberapa kriteria yang berkaitan secara langsung terhadap ongkos ini adalah:

- Minimasi jumlah produk yang mengalami keterlambatan.
- Maksimasi utilitas mesin.

3. Kriteria gabungan

Merupakan gabungan atau kombinasi dari beberapa optimalitas. Sedangkan mengelompokkan beberapa besaran yang digunakan untuk mengukur keberhasilan suatu penjadwalan adalah sebagai berikut:

a. Berdasarkan waktu penyelesaian
   - Minimasi maksimal *makespan*
   - Minimasi rata-rata *makespan*
   - Minimasi waktu penyelesaian maksimal

b. Berdasarkan batas penyelesaian
   - Minimasi maksimal keterlambatan
   - Minimasi rata-rata waktu keterlambatan
   - Minimasi jumlah *job* yang terlambat
   - Minimasi rata-rata jumlah *job* yang terlambat

c. Berdasarkan penggunaan sumber
   - Rata-rata jumlah *job* yang menunggu untuk diproses
   - Rata-rata waktu mesin menunggu

2.5. Klasifikasi Penjadwalan

Penjadwalan terbagi menjadi 3 kategori yaitu:

1. *Single Machine Schedulling*
   - *Shortest Processing* Time (SPT)
   - *Earliest Due Date* (EDD)
   - *Minimazing Number of Job Tardy*
2. *Flow Shop Schedulling*
   - *Johnson Algorithm*
3. *Job Shop Schedulling*
   - *Priority Dispatching Rules*

2.6. *Job Shop Schedulling*

Penjadwalan *job shop* adalah pengurutan pekerjaan untuk lintasan produk yang tidak beraturan (tata letak pabrik berdasarkan proses). Penjadwalan pada proses produksi tipe *job shop* lebih sulit dibandingkan dengan penjadwalan *flow shop*. Hal ini disebabkan oleh 3 alasan, yaitu:

1. *Job shop* menangani variasi produk yang sangat banyak, dengan pola aliranyang berbeda-beda melalui *work center*.
2. Peralatan pada *job shop* digunakan secara bersama-sama oleh bermacam-macam *order* dalam prosesnya, sedangkan peralatan flow shop digunakan khususnya hanya untuk satu jenis produk.
3. *Job* yang berbeda mungkin ditentukan oleh prioritas yang berbeda pula. Hal ini mengakibatkan *order* tertentu yang dipilih harus diproses seketika pada saat *order* tertentu yang dipilih harus diproses seketika padaa saat *order* tersebut ditugaskan pada suatu *work center*. Sedangkan pada flow shop tidak terjadi permasalahan seperti diatas karena keseragaman output yang diproduksi untuk persediaan. Prioritas *order* pada *flow shop* dipengaruhi terutama pada pengirimannya dibandingkan tanggal pemrosesan.

Ukuran keberhasilan dari suatu pelaksanaan aktivitas penjadwalan khususnya penjadwalan *job shop* adalah meminimasi kriteria-kriteria keberhasilan sebagai berikut:

a. Rata-rata waktu alir (*mean flow time*) sksn mengurangi persediaan barang setengah jadi.
b. Rata-rata keterlambatan (*mean tardiness*).
c. Jumlah *job* yang terlambat, akan meminimasi dari maksimum ukuran keterlambatan.
d. Jumlah mesin yang menganggur.
e. Jumlah persediaan.

Masalah yang terjadi pada penjadwalan dapat diselesaikan dengan cara sebagai berikut:

1. *Sequencing*, dapat diselesaikan dengan metode:
   a. *Priority rule*
   b. *Queue*
2. *Timing*, awal dan akhir tiap *job* dihitung berdasarkan pada urutan, *routing* dan waktu proses.

Metode-metode yang dapat digunakan dalam penyelesaian masalah penjadwalan, yaitu:

a. Heuristik
b. Matematis
c. Simulasi

2.6.1. Ruang Jawab Persoalan Penjadwalan *Job Shop*

Persoalan *job shop* mempunyai ciri khas yaitu aliran penjadwalan pekerjaan dalam shop tidak searah. Persoalan ini biasanya membutuhkan matriks waktu proses yang menyatakan waktu pemrosesan tiap operasi dari suatu *job* dan matriks *routing* yang menunjukan urutan mesin untuk mengerjakan beberapa operasi dari suatu *job*. Penyelesaian persoalan n *job* m mesin pada lintasan produksi *job shop* $(n!)^m$ jadwal, suatu jadwal dikatakan layak jika memenuhi kriteria:

1. Tidak ada *overlap* diantara waktu serta proses operasi.
2. Hubungan ketergantungan antar operasi setiap *job* tidak dilanggar.

Jadwal yang layak tersebut dapat diklasifikasikan sebagai berikut:

1. Jadwal *feasible*

Suatu jadwal dikatakan *feasible* jika seluruh operasi semua *job* telah ditugaskan dan ketentuan *routing* operasi telah terpenuhi (dengan kata lain tidak ada *overlap* antar operasi).

2. Jadwal semi aktif

Jadwal semi aktif diperoleh jika tidak ada satu operasi pun yang dapat dikerjakan lebih awal tanpa mengubah susunan operasi lainnya. Penjadwalan ini diusulkan oleh Giffler dan Thompson.

Adapun langkah-langkahnya yaitu sebagai berikut:

P_{st} = Suatu jadwal parsial yang mengandung sejumlah t operasi yang telah dijadwalkan.

St = kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

t = *Stage* (tahap)

σ_j = Saat paling awal dimana operasi σ_k pada St siap dijadwalkan atau dikerjakan

ϕ_j = Saat paling awal operasi $j \in s_t$ dapat diselesaikan dimana $\phi_j = \sigma_j + t_{ij}$

t_k = Waktu pemrosesan operasi dari *job* ke i pada operasi ke j. Saat paling awal, yaitu suatu operasi dapat dimulai (ϕ_j*) pada mesin m ditentukan oleh penyelesaian dari operasi pendahulunya (j-i) dan penyelesaian operasi terakhir pada mesin m, sehingga berlaku ϕ_j * = max (ϕ_{j-1},t_m).

Algoritma jadwal semi aktif adalah sebagai berikut:

Step 1 : t = 0, P_{st} = 0 (yaitu jadwal parsial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu. Dengan kata lain, merupakan sesuatu yang awal pada *job* tersebut.

Step 2 : Tentukan $\phi^* = \min(\phi_j)$ dimana ϕ_j adalah saat paling awal operasi j dapat diselesaikan ($\phi_j = \sigma_j + t_{ij}$). Tentukan m*, yaitu mesin di mana ϕ^* dapat direalisasi.

Step 3 : Untuk setiap operasi dalam P_{st} yang memerlukan mesin m* dan memiliki $\sigma_j < \phi^*$ untuk suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar ke dalam P_{st} sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.

Step 4 : Membuat suatu jadwal parsial baru P_{t+1} dan memperbaiki kumpulan data dengan cara:

- Menghilangkan operasi j dar S_t
- Membuat S_{t+1} dengan cara menambah pengikut langsung operasi k yang telah dihilangkan
- Menambah satu pada t.

Step 5 : Jika ada beberapa opersi yang belum terjadwalkan (t<nm) , kembali ke langkah 2 sampai seluruh pekerjaan terjadwalkan. JIka telah terjadwalkan semua, berhenti.

3. Jadwal Aktif

Jadwal aktif adalah kumpulan jadwal *feasible* dimana tidak satupun operasi dapat dipindahkan lebih awal tanpa menunda operasi lain dengan aturan prioritas SPT (*Short Processing Time*) yaitu prioritas diberikan kepada pekerjaan yang memiliki waktu paling singkat dijadwalkan pada urutan pertama, kemudian diikuti *job* yang terbesar.

Adapun langkah-langkahnya yaitu sebagai berikut:

P_{st} = Suatu jadwal parsial yang mengandung sejumlah t operasi yang telah dijadwalkan.

St = kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

t = *Stage* (tahap)

R_j = Saat paling awal dimana operasi $j \in s_t$ siap dijadwalkan atau dikerjakan

C_j = Saat paling awal operasi $j \in s_t$ dapat diselesaikan dimana $J = R_j + t_{ij}$

t_{ij} = Waktu pemrosesan operasi dari *job* ke i pada operasi ke j. Saat paling awal, yaitu suatu operasi dapat dimulai (R_j*) pada mesin m ditentukan oleh penyelesaian dari operasi pendahulunya (j-i) dan penyelesaian operasi terakhir pada mesin m, sehingga berlaku $R_j^* = \max (R_{j-1}, t_m)$.

Algoritma jadwal aktif adalah sebagai berikut:

Step 1 : $t = 0$, $P_{st} = 0$ (yaitu jadwal parsial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

Step 2 : Tentukan $r^* = \min (r_j)$ diaman r_j adalah saat paling awal operasi j dapat diselesaikan ($r_j = c_j + t_{ij}$). Tentukan m*, yaitu mesin di mana r* dapat direalisasi.

Step 3 : Untuk setiap operasi dalam P_{st} yang memerlukan mesin m* dan memiliki $c_j < r^*$ untuk suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar ke dalam P_{st} sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.

Step 4 : Membuat suatu jadwal parsial baru P_{t+1} dan memperbaiki kumpulan data dengan cara:

- Menghilangkan operasi j dar S_t
- Membuat S_{t+1} dengan cara menambah pengikut langsung operasi k yang telah dihilangkan
- Menambah satu pada t.

Step 5 : Kembali ke langkah 2 sampai seluruh pekerjaan terjadwalkan.

4. Jadwal *Non Delay*

Jadwal *non delay* adalah kumpulan jadwal *feasible* dimana tidak satupun mesin dibiarkan menganggur jika pada saat yang sama terdapat operasi yang memerlukan mesin tersebut.

Adapun langkah-langkah dari metode ini adalah:

Notasi-notasi yang dihunakan dalam teknik ini adalah sebagai berikut:

P_{st} = Suatu jadwal parsial yang mengandung sejumlah t operasi yang telah dijadwalkan.

t = *Stage* (tahap)

R_j = Saat paling awal dimana operasi $j \in s_t$ siap dijadwalkan atau dikerjakan

C_j = Saat paling awal operasi $j \in s_t$ dapat diselesaikan dimana $J = R_j + t_{ij}$

t_{ij} = waktu proses pekerjaan i pada operasi j.

Algoritma jadwal *non delay* ini adalah sebagai berikut:

Step 1 : $t = 0$, $P_{st} = 0$ (yaitu jadwal parial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

Step 2 : Tentukan $c^* = \min (c_j)$ diaman c_j adalah saat paling awal operasi j dapat mulai dikerjakan. Tentukan m*, yaitu mesin di mana c* dapat direalisasi.

Step 3 : Untuk setiap operasi dalam P_{st} yang memerlukan mesin m* dan memiliki $c_j = c^*$ untuk suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar ke dalam P_{st} sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.

Step 4 : Membuat suatu jadwal parsial baru P_{t+1} dan memperbaiki kumpulan data dengan cara:

- Menghilangkan operasi j dar S_t
- Membuat S_{t+1} dengan cara menambah pengikut langsung operasi k yang telah dihilangkan
- Menambah satu pada t.

Step 5 : Kembali ke langkah 2 sampai seluruh pekerjaan terjadwalkan.

2.6.2. Asumsi Dasar Penjadwalan

Dalam menyelesaikan suatu masalah penjadwalan biasanya diberlakukannya beberapa asumsi yang menyangkut karakteristik tugas, operasi mesin yang digunakan dan waktu proses. Hal ini dimaksudkan untuk menyederhanakan masalah penjadwalan itu sendiri. Asumsi-asumsi dasar tersebut, antara lain:

a. Karakteristik *job*

- *Job* terdiri dari urutan operasi yang telah ditentukan.
- Suatu operasi hanya bisa dikerjakan pada satu tipe mesin dari setiap tipe mesin dalam *shop*.
- Waktu proses diketahui dengan pasti seperti halnya *due date*.
- Urutan waktu *set-up* bersifat independen dan waktu transportasi antar mesin dapat diabaikan.
- Operasi yang sedang dikerjakan pada mesin tidak dapat di interupsi.
- Operasi tidak dapat dimulai sampai operasi pendahulunya diselesaikan.

- Setiap mesin hanya dapat memproses satu operasi pada satu waktu.
- Setiap *part* (bagian) hanya dapat diproses di suatu mesin pada satu mesin.

b. Karakteristik operasi
   - Setiap operasi merupakan satu kesatuan, walaupun mungkin terdiri dari beberapa unit.
   - Setiap operasi yang telah dimulai pengerjaannya pada suatu mesin harus diselesaikan.
   - Setiap operasi tidak boleh diproses lebih dari satu mesin pada waktu yang sama.
   - Setiap operasi dikerjakan menurut yang telah disusun dan tidak didasarkan pada urutan lainnya.
   - Setiap operasi boleh diproses satu kali pada mesin yang sama.
   - Setiap operasi dapat diproses pada beberapa jenis mesin yang mampu melaksanakan operasi tersebut.
   - Setiap *job* hanya mempunyai satu *routing* dalam memproses operasional.

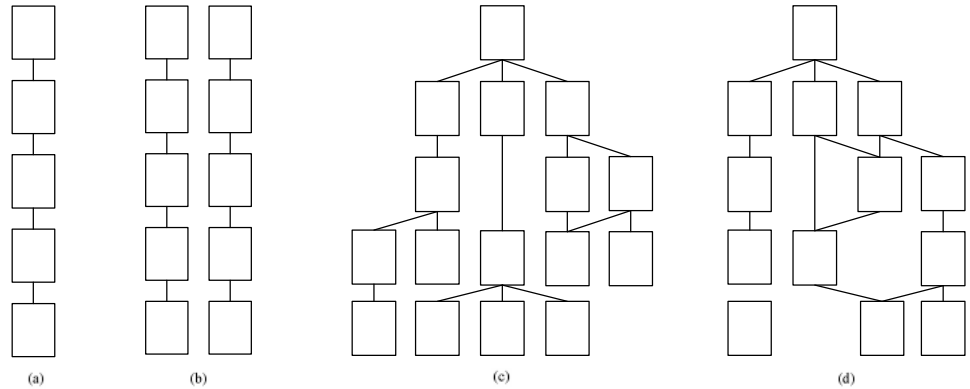
c. Karakteristik mesin
   - Setiap mesin hanya memproses satu tugas pada suatu saat tertentu.
   - Setiap mesin secara kontinyu siap untuk dibebani tugas selama proses.
   - Penjadwalan apabila tidak mengalami interupsi akibat kerusakan atau perawatan.
   - Setiap mesin operasi sesuai dengan informasi waktu dan distribusi yang diketahui secara tepat.

d. Karakteristik waktu proses
   - Waktu proses telah dikerahui baik rata-rata maupun distribusinya.
   - Waktu proses independen terhadap jadwal, artinya urutan *set-up time*.
   - Bersifat independen dan *move time* antara mesin dapat diabaikan.
   - Setiap waktu proses secara implisit sudah mencakup waktu pemindahan benda kerja, *set-up* dan penghentian mesin.

2.6.3. Penjadwalan Produk Berstruktur *Multi Level*

Berdasarkan strukturnya, setiap produk dapat dibedakan atas produk yang berstruktur satu *level* dan produk yang berstruktur multi *level*. Produk-produk yang berstruktur satu *level* biasanya disebut juga sebagai komponen tunggal yaitu produk-produk yang hanya membubuhkan serangkaian operasi pemesinan dengan urutan proses (*routing*) tertentu.

Produk-produk ynag mempunyai struktur multi *level* terdiri dari sejumlah *part* (komponen) yang harus diproses melalui serangkaian operasi pemesian dan peralatan hingga menjadi produk akhir. Seperti yang digambarkan oleh Billington pada tahun 1983 mengenai struktur produk pada gambar 2.6. dibawah ini.



Gambar 2.6. Empat Macam Struktur Produk

(a) seri: satur produk pada satu *level*; (b) parallel: multi produk satu *level* dengan proses seri; (c) *assembly*: satu produk multi *level*, tidak terjadi pengamanan komponen; (d) general: satu produk multi *level*, terjadi kesamaan komponen.

Meskipun tidak dinyatakan eksplinsit, persoalan penjadwalan pada umumnya berkaitan dengan penjadwalan produk-produk yang berstruktur satu *level*, pada hal sistem manufaktur yang sebenarnya produk-produk yang berstruktur multi *level* lebih banyak dijumpai. Pada struktur multi *level*, hubungan ketergantungan (*precedence relationship*) diantara proses pemesinan dan proses perakitan dapat dinyatakan sebagai suatu diagrap G. setiap tahapan proses dinyatakan dengan proses, dimana penomoran *level* produk dengan *level* 0 untuk produk akhir, kemudian nomor *level* bertambah satu pada setiap langkah mundur hingga mencapai komponen dasar.

2.6.4. Matriks *Routing* dan Matriks Waktu Untuk Persoalan *Job Shop*

Dalam penjadwalan *job shop* diperlukan input berupa jmlah *job*, jumlah operasi dalam tiap *job* dan urutan operasi beserta mesin yang memprosesnya (*routing*). Hal ini ditampilkan dalam bentuk matriks waktu proses yang menyatakan urutan mesin yang memproses tiap-tiap urutan operasi.

Suatu penjadwalan digambarkan dengan susunan balok-balok, dimana setiap balok merupakan *triplet* dari *job*-operasi-mesin. Panjang balok menyatakan waktu proses pekerjaan yang bersangkutan. Notasi *triplet* yang digunakan tiap balok (I,j,k) dimana I menunjukan nomor pekerjaan, j menyatakan urutan pekerjaan dan k mewakili mesin yang diperlukan. Masalah penjadwalan ini di deskripsikan secara grafis dengan menggunakan skala *gantt chart*. Contoh matriks waktu proses dan matriks *routing* mesin dapat dilihat pada tabel di bawah ini.

Tabel 2.1. Matriks Waktu Proses

Job	**Operasi**		
	1	**2**	**3**
1	3	2	4
2	4	4	1
3	2	2	3
4	3	3	3

Tabel 2.2. Matriks *Routing*

Job	**Operasi**		
	1	**2**	**3**
1	2	3	1
2	1	3	2
3	2	4	3
4	3	1	2

Pada matriks waktu proses, operasi 1 dari *job* 1 mewali;I waktu proses 3 satuan waktu pada dan matriks *routing*, operasi 1 dan *job* 1 dikerjakan dimesin 1, untuk menulis kondisi tersebut sering kali digunakan notasi Q_{ijk} untuk mempresentasikan satu operasi j dari *job* I diproses dimesin k dan t_{ijk} untk memepresentasikan waktu proses operasi j dai *job* i diproses di mesin k.

Setelah *input* dari masing-masing telah didefinisikan, proses selanjutnya adalah penugasan operasi *job* dari tiap mesin. Penugasan ini mempunyai aturan yang bermacam-macam dan penggunaannya biasanya ditentukan oleh kebijakan manajemen berdasarkan sistem produksinya serta kriteria optimalitas yang diinginkan.

2.6.5. Teknik Penyelesaian Masalah *Job Shop*

Dalam penyelesaian masalah *job shop* dapat menggunakan beberapa teknik pendekatan, daitara teknik tersebut adalah:

1. Teknik pendekatan optimal

Teknik pendekatan optimal merupakan pendekatan yang memberikan solusi terbaik terhadap suatu permasalahan ditinjau dari kriteria tertentu. Pendekatan optimal memiliki 2 metode, yaitu metode program integer dan metode *branch and bound*. Namun kedua metode ini memiliki tingkat kesukaran yang tinggi dan belum tentu menghasilkan jadwal yang benar-benar optimal.

2. Teknik pendekatan heuristik

Teknik pendekatan heuristik digunakan dalam masalah penjadwalan untuk jumlah operasi lebih kompleks dengan waktu penyelesaian yang relatif lebih cepat. Walaupun pendekatan heuristik tidak menghasilkan jadwal yang optimal, namun pendekatan heuristik dapat menghasilkan jadwal yang baik dan mendekati optimal. Teknik pendekatan heuristik terbagi atas:

- *Priority dispatching rules*

  Priority dispatching rules adalah satu jenis metode peramalan dimana waktu siap (*ready time*) dari setiap mesin ditentukan sedemikian rupa sehingga berurutan naik. Keputusan pemilihan produk yang akan diproses dapat dilakukan pada saat mesin siap menerima produk (mesin menganggur). Pada teknik *dispatching* digunakan aturan prioritas untuk memilih salah satu operasi diantara operasi-operasi yang mengalami konflik pada mesin m* pada setiap tahap.
- *Sampling procedures*
- *Probabilistic dispatching procedures*

2.7. Metode Penjadwalan

Proses perancangan algoritma penjadwalan dengan memperhatikan aktivitas perawatan pencegah merupakan pengembangan dari rancangan algoritma penjadwalan heuristik yang telah dikenal antara lain:

1. Meode *Forward*

Metode ini menjadwalkan proses kerja dalam setiap sumber daya mulai sejak awal produksi (saat t=0) sampai dapat diselesaikannya suatu produk direncanakan. Tujuan dari metode ini adalah menjadwalkan produksi apabila titik waktu mulainya telah ditentukan sebelumnya dan tidak diberikan batas waktu harus diselesaikannya keseluruhan suatu produk. Kelebihan metode ini adalah bahwa penjadwalan dapat disusun secara *short processing time* (SPT) sehingga didapat suatu penjadwalan produksi dengan *flow time* yang minimum. Kelemahan dari metode ini adalah karena titik awal ditentukan terlebih dahulu, maka ada resiko keterlambatan sehingga dapat menimbulkan kerugian berupa *penalty cost* dan kepercayaan konsumen, selain itu jika selesai sebelum *due date* maka ada ongkos simpan.

2. Metode *Backward*

Metode ini menjadwalkan produksi mulai dari batas akhir diselesaikannya keseluruhan produk (*due date*) kemudian ters mundur kebelakang sampai didapatkan waktu mulainya produksi. Kelebihan metode ini adalah hasil penjadwalan dengan metode *backward* tidak akan terlambat, karena dijadwalkan mundur atau dengan kata lain *due date* selalu dapat dipenuhi. Kelemahan metode ini tidak bisa diterapkan pada penjadwalan n *job* m mesin jika ada beberapa *job* yang mempunyai *due date* yang sama dan operasi terakhir pada mesin yang sama. Kelemahan lain dalam penerapan metode ini bahwa penjadwalan tidak dapat mendeteksi adanya sumber daya yang menganggur sehingga utilitas sumber daya yang ada tidak dapat maksimum. Selain itu ada kemungkinan menghasilkan penjadwalan yang tidak fleksibel yaitu wajtu mulai proses pertama kali sudah terlewat dari waktu sekarang (waktu saat menjadwalkan).

3. Metode Kompromi

Metode ini merupakan gabungan antara metode *forward* dan metode *backward*, sehingga dari penggabungan metode tersebut maka dapat mereduuksi kelemahan metode masing-masing. Tujuan dari metode ini adalah membentuk suatu penjadwalan dengan *flow time* yang minimum, dapat memenuhi *due date* serta memiliki tingkat utilitas sumber daya yang maksimum. Kelebihan dari metode ini adalah adanya beberapa kriteria optimalisasi yang dapat tercakup langsung dengan metode tersebut, yaitu minimasi *flow time*, pemenuhan *due dat*e, maksimasi utilitas sumber daya. Kelemahanya adalah bahwa metode ini lebih rumit perhitungan serta masih belum mampu mendeteksi kriteria optimalisasi minimasi ongkos produksi.

4. Metode *Forced*

Metode ini merupakan metode penjadwalan dimana satu atau beberapa kegiatan yang dipaksakan untuk dilaksanakan pada suatu waktu tertentu sehingga tidak dapat lagi ditukar atau diganti dengan operasi lainnya. Kegiatan yang dipaksakan tersebut misalnya memerlukan mesin produksi khusus yang harus disewa dengan harga yang sangat mahal tiap sama waktunya, sehingga semua operasi yang membutuhkan mesin tersebut dipaksakan untuk dikerjakan pada waktu mesin disewa. Kelebihan metode ini adalah kemampuannya untuk mengatasi beberapa permasalahan yang harus dikerjakan pada satuan waktu tertentu. Kelemahannya adalah bahwa dengan memaksakan beberapa bagian didalam penjadwalan maka diperoleh suatu penjadwalan produksi yang optimal.

2.8. *Gantt Chart*

Gantt chart atau yang disebut juga *bar chart* yang dikembangkan oleh Henry Gantt sekitar tahun 1900. *Chart* ini terdiri dari koordinat dan aksis, dimana satu aksis mempresentasikan waktu yang telah dilalui dan aksis lainnya mempresentasikan pekerjaan atau aktivitas yang dilakukan. Pekerjaan ini dinyatakan dalam bentuk batangan.

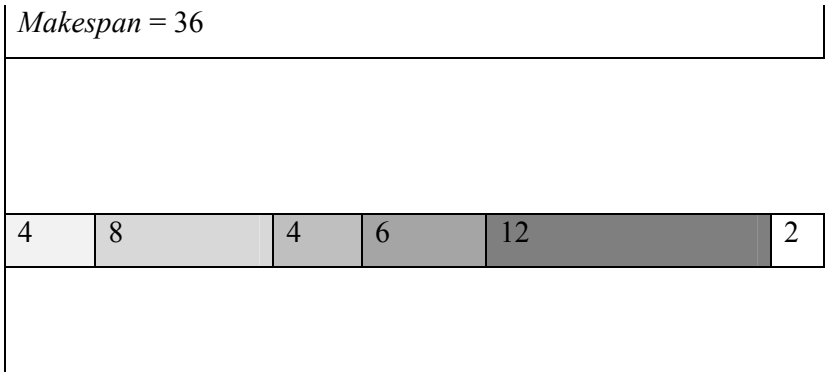
Gantt chart memiliki keuntungan dalan kelemahan pula, yang diantaranya:

a. Keuntungan *gantt chart* adalah:

- Dalam situasi keterbatasan sumber, penggunaan *Gantt chart* memeungkinkan evaluasi lebih awal mengenai penggunaan sumber daya yang telah direncanakan.
- Kemajuan pekerjaan mudah diperiksa pada setiap waktu karena sudah tergambar dengan jelas.
- Semua pekerjaan diperlihatkan secara grafis dalam suatu peta yang mudah dipahami.

b. Kelemahan *gantt chart* adalah:

- Saling ketergantungan antar aktivitas
- Perkembangan proyek
- Ketidakpastian



Gambar 2.7. *Gantt Chart*