

## BAB II TINJAUAN PUSTAKA

Tinjauan pustaka memiliki arti yaitu peninjauan kembali pustaka-pustaka yang terkait (*review of related literature*). Sesuai dengan arti tersebut, suatu tinjauan pustaka berfungsi sebagai peninjauan kembali (*review*) pustaka berkaitan dengan masalah yang diteliti yang dapat membantu peneliti dalam menganalisa permasalahan yang dihadapi.

### 2.1 Penelitian Terdahulu

Beberapa penelitian yang telah dilakukan dalam dengan penerapan metode perhitungan *integer linear programming* (ILP) yang digunakan sebagai referensi dalam penelitian ini adalah sebagai berikut :

1. Ekoanindiyo (2012) dalam Jurnal Dinamika Teknik volume 6 nomor 2 melakukan sebuah penelitian yang berjudul “Penjadwalan Produksi menggunakan Pendekatan *Theory of Constraints*”. Pada jurnal tersebut penulis menentukan penjadwalan suatu sistem produksi yang optimal dengan perbaikan melalui penerapan TOC pada suatu stasiun konstrain di sistem produksi. Hal ini akan mempermudah proses penjadwalan karena cukup hanya menjadwalkan stasiun konstrain dan stasiun lain akan menyesuaikan. Hasil yang diperoleh adalah dimana untuk menerapkan *Theory of Constraints* pada penjadwalan produksi dan penggunaan kapasitas untuk produksi merupakan langkah yang efektif dengan menggunakan sumber daya yang baik dalam tercapainya tujuan perusahaan yaitu meningkatkan keuntungan. Langkah-langkah yang dapat dilakukan antara lain menghitung kapasitas efektif di lintasan produksi, menentukan *constraints* atau kendala, dan menentukan produk mana yang akan di produksi terlebih dahulu di tiap stasiun kerja.
2. Rianto (2009) melakukan sebuah penelitian yang berjudul “Penerapan *Theory of Constraints* (TOC) dalam Upaya Peningkatan Kapasitas Produksi di PT. X”. Pada penelitian tersebut penulis bertujuan untuk mengeliminasi stasiun kerja *bottleneck* dengan menerakan lima prinsip perbaikan berkelanjutan *Theory of Constraints* (TOC) dalam mengoptimalkan perencanaan kapasitas. Dari penerapan TOC yang dilakukan untuk mengoptimalkan perencanaan produksi dengan menggunakan metode *linier programming* (LP) sehingga didapatkan hasil stasiun kerja *bottleneck* yang sebelumnya yaitu stasiun kerja 3 pada periode Oktober dapat dieliminasi dan pada stasiun kerja 4. Untuk jumlah produk maksimal diperoleh pada periode bulan

Juli 2007 dengan jumlah produksi untuk obat anti nyamuk bakar coil standar lokal 1.039.675.538 dC dengan *throughput* sebesar Rp 843.300.485,-

3. Nurwidiana dan Syakhroni (2010) melakukan sebuah penelitian yang berjudul “Upaya Maksimasi Keuntungan melalui Penentuan Volume Produksi pada Industri Konveksi di Pekalongan dengan menggunakan Metode *Theory of Constraints* (TOC)”. Penelitian dilakukan di BATIK MAS yang dimaksudkan untuk mengetahui apakah perusahaan telah menggunakan sumber daya yang dimiliki secara optimal dan menentukan berapa jumlah produk yang optimal pada masing-masing jenis produk yang memberikan keuntungan maksimal. Penyelesaian dengan perhitungan LP menekankan pada pengelolaan stasiun kendala (penghambat) yaitu *work center* penjahitan dan obras karena membutuhkan kapasitas yang lebih besar daripada kapasitas sumber daya yang tersedia sehingga didapatkan volume produksi optimal dan utilitas menunjukkan hasil angka kurang dari 100%, dimana hal ini menunjukkan tidak ada lagi *constraint resources* yang terjadi.

Untuk lebih jelasnya, perbandingan antara penelitian ini dengan penelitian terdahulu dapat ditunjukkan pada Tabel 2.1.

Tabel 2.1 Perbandingan Penelitian ini dengan Penelitian Terdahulu

	Ekoanindiyo (2012)	Rianto (2009)	Nurwidiana dan Syakhroni (2010)	Penelitian ini (2014)
<b>Judul Penelitian</b>	Penjadwalan Produksi menggunakan Pendekatan <i>Theory of Constraints</i>	Penerapan <i>Theory of Constraints</i> (TOC) dalam Upaya Peningkatan Kapasitas Produksi di PT. X	Upaya Maksimasi Keuntungan melalui Penentuan Volume Produksi pada Industri Konveksi di Pekalongan dengan menggunakan Metode <i>Theory of Constraints</i> (TOC)	Optimasi Kapasitas Produksi Dalam Penyusunan Jadwal Induk Produksi Menggunakan <i>Integer Linear Programming</i> (ILP)
<b>Metode</b>	TOC	TOC, <i>linear programming</i> (LP)	TOC, <i>linear programming</i> (LP)	<i>Integer Linear programming</i> (ILP)
<b>Objek Penelitian</b>	-	Industri Manufaktur (PT. X)	Industri Manufaktur (Batik Mas)	Industri Manufaktur (CV. Pabrik Mesin Guntur)
<b>Aplikasi TOC</b>	Penjadwalan produksi	Peningkatan kapasitas produksi	Penentuan volume produksi dalam meningkatkan <i>profit</i>	Mengoptimalkan Kapasitas Produksi dalam penyusunan jadwal induk produksi
<b>Hasil Penelitian</b>	Penerapan TOC pada penjadwalan produksi dan kapasitas untuk produksi merupakan langkah yang efektif	Eliminasi stasiun kerja <i>bottleneck</i> dapat meningkatkan kapasitas produksi	Optimalisasi <i>resources</i> pada produksi produk yang memberikan <i>profit</i> maksimal	Menentukan jumlah produksi dengan mengelola dan meningkatkan <i>resources</i> yang ada guna meminimasi SOP dan mengoptimalkan perencanaan JIP

## 2.2 Jadwal Induk Produksi

Menurut Nasution dan Prasetyawan (2008) *Master Production Schedule* (MPS) atau yang lebih dikenal dengan jadwal induk produksi merupakan pernyataan akhir mengenai “berapa” banyak item-item akhir yang harus diproduksi dan “kapan” harus diproduksi. Sama seperti halnya yang dinyatakan oleh Fogarty (1992) Jadwal Induk Produksi merupakan suatu pernyataan tentang produk akhir dari suatu perusahaan industri manufaktur yang merencanakan dalam memproduksi output berkaitan dengan kuantitas dan periode waktu.

Sebagai suatu aktivitas proses, jadwal induk produksi (MPS) membutuhkan input utama sebagai berikut (Gaspersz, 1998) :

1. Data permintaan total, yang berkaitan dengan ramalan penjualan (*sales forecast*) dan pesanan-pesanan (*order*).
2. Status inventori, berkaitan dengan informasi tentang *on hand inventory*, *stock* yang dialokasikan untuk penggunaan tertentu, pesanan produksi dan pembelian yang dikeluarkan.
3. Perencanaan produksi menentukan tingkat produksi, inventori dan sumber daya lainnya.
4. Data perencanaan, berkaitan dengan aturan-aturan tentang *lot sizing*, *safety stock* dan waktu tunggu (*lead time*) dari masing-masing item.
5. Informasi dari *Rough Cut Capacity Planning* (RCCP), berupa kebutuhan kapasitas untuk mengimplementasikan MPS menjadi salah satu input bagi MPS. RCCP juga memberikan umpan balik kepada perencana atau penyusun jadwal induk produksi untuk mengambil tindakan perbaikan apabila ditemukan adanya ketidaksesuaian antara penjadwalan produksi induk dan kapasitas yang tersedia.

## 2.3 Kapasitas

Kapasitas merupakan tingkat dimana *system manufacturing* (tenaga kerja, mesin, pusat kerja, departemen, pabrik) berproduksi. Dengan kata lain, kapasitas merupakan tingkat *output* yang dapat dicapai dengan spesifikasi produk, *product mix*, tenaga kerja, dan peralatan yang ada sekarang. Dalam CRP, kapasitas berkaitan dengan tingkat *output* kerja dalam setiap pusat kerja. (Nurwidiana dan Syakhroni, 2010)

Gaspersz (1998) mengemukakan beberapa definisi yang akan banyak dipergunakan dalam pembahasan yang berkaitan dengan perencanaan kapasitas sebagai berikut:

1. Pusat Kerja (*Work Center*)

Merupakan suatu fasilitas produksi spesifik yang terdiri dari satu atau lebih orang dan atau mesin dengan kemampuan yang sama atau identik, yang dapat dipertimbangkan sebagai satu unit untuk tujuan perencanaan kebutuhan kapasitas (CRP) dan penjadwalan terperinci (*detailed scheduling*).

2. Pesanan Manufaktur (*Manufacturing Orders*)

Merupakan suatu dokumen atau identitas jadwal yang memberikan kewenangan untuk membuat *part* tertentu atau produk dalam jumlah tertentu. Pesanan *manufacturing* dapat berupa salah satu; *open orders*, *already in process*, atau *planned orders*, sebagaimana dijadwalkan melalui proses MRP.

3. Routing

Merupakan sekumpulan informasi yang merinci metode pembuatan item tertentu, termasuk operasi yang dilakukan, sekuensi operasi, berbagai pusat kerja yang terlibat, serta standar waktu setup (*setup time*) dan waktu pelaksanaan kerja (*run time*).

4. Beban (*load*)

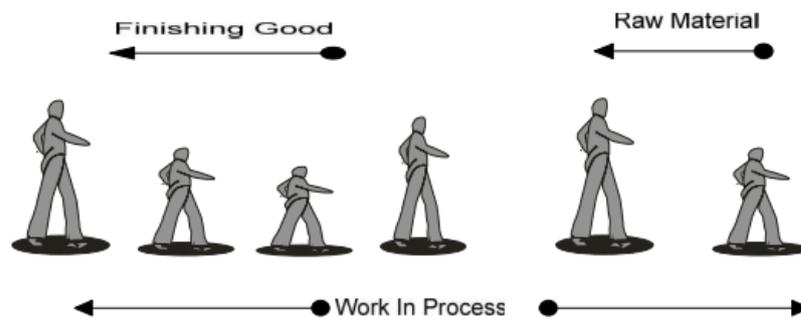
Adalah banyaknya kerja yang dijadwalkan untuk dilakukan oleh fasilitas *manufacturing* dalam periode waktu yang ditetapkan. Beban (*load*) biasa dinyatakan dalam ukuran jam kerja atau unit produksi. *Load* merupakan volume kerja yang dikerjakan. Sebagaimana yang digunakan dalam CRP, beban (*load*) menggambarkan waktu setup (*setup time*) dan waktu pelaksanaan (*runtime*) yang dibutuhkan dari suatu pusat kerja.

### 2.3.1 Hubungan Kapasitas dengan Beban (*Capacity-Load Relationship*)

Dalam aplikasi CRP menunjukkan perbandingan antara beban yang ditetapkan pada pusat-pusat kerja berdasarkan pesanan kerja yang ada dan kapasitas dari setiap pusat kerja selama periode waktu tertentu. Penelitian Nurwidiana dan Syakhroni (2010) menyatakan bahwa adanya tindakan perencanaan kembali (*replanning*) dapat dilakukan untuk menghilangkan situasi guna mencapai suatu keseimbangan antara beban dan kapasitas.

Pada buku *The Goal*, Goldratt mengungkapkan suatu sinkronisasi manufaktur dalam perumpaannya pada pejalan kaki yang dapat disamakan untuk aliran proses. Sepanjang gerakan, beberapa pejalan kaki lebih lambat dibanding yang lain (Ekoanindiyo, 2012). Tujuan kelompok dapat terhambat dengan adanya orang paling

lambat, seperti *bottleneck* (orang yang paling lambat) menjadi sumber daya yang menghalangi *throughput* yang ditunjukkan pada Gambar 2.1.



Gambar 2.1 Gambaran Pejalan Kaki sebagai Aliran Proses  
Sumber : Penelitian Ekoandiyo (2012)

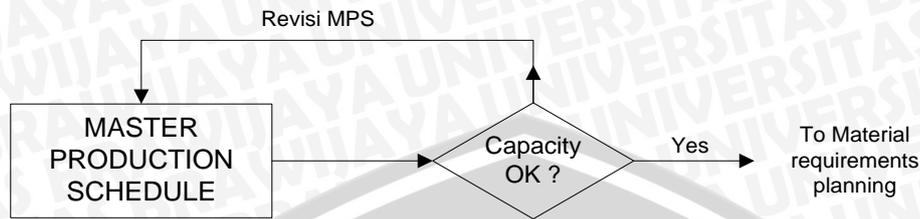
Tujuan dari perencanaan kapasitas adalah berusaha mengatur pesanan kerja yang datang dan kapasitas dari pusat kerja secara bersama-sama dalam mencapai aliran yang seimbang (Nurwidiana dan Syakhroni, 2010). Beban yang bertambah membuat banyaknya antrian sehingga waktu tunggu pusat kerja lebih panjang. Sebaliknya bila beban dikurangi maka waktu tunggu akan lebih pendek. Sehingga dapat disimpulkan bahwa penanganan hubungan antara kapasitas dan beban didasarkan pada kemampuan sistem perencanaan dan pelaksanaan yang menyesuaikan dari tingkat kedatangan pesanan kapasitas selama jam kerja dalam interval waktu tertentu.

### 2.3.2 Rought Cut Capacity Planning (RCCP)

Menurut Nasution dan Prasetyawan (2008) ketidaklayakan dari suatu jadwal induk produksi atau *master production schedule* (MPS) diperbaiki dengan suatu pendekatan yang disebut RCCP, dimana RCCP akan mengkonversi MPS dengan kebutuhan-kebutuhan kapasitas untuk sumberdaya-sumberdaya dan kemudian menentukan apakah MPS tersebut layak terhadap keterbatasan-keterbatasan kapasitas yang ada. Jika MPS tidak sesuai maka MPS harus direvisi, sehingga MPS tersebut tetap sesuai dengan keterbatasan kapasitas yang ada.

*Rought Cut Capacity Planning* (RCCP) merupakan urutan kedua (level 2 dalam Hierarki Perencanaan dan Pengendalian Kapasitas) yang berperan dalam pengujian MPS. RCCP melakukan validasi terhadap MPS guna menetapkan sumber-sumber daya yang spesifik tertentu, khususnya yang diperkirakan akan menjadi *potential bottleneck*.

RCCP merupakan proses iterasi yang dilakukan bersama-sama dengan MPS dimana dalam penerapannya seringkali beberapa iterasi dibutuhkan sebelum MPS yang realistis dibuat seperti ditunjukkan pada Gambar 2.2.



Gambar 2.2 Hubungan MPS dengan Perencanaan Kapasitas  
Sumber : Nasution dan Prasetyawan (2008)

Pada dasarnya RCCP didefinisikan sebagai proses konversi dari rencana produksi dan atau MPS ke dalam kebutuhan kapasitas yang berkaitan dengan sumber-sumber daya kritis seperti; tenaga kerja, mesin dan peralatan, kapasitas gudang, kapabilitas pemasok material dan *parts*, dan sumber daya keuangan (Gaspersz, 1998).

Menurut Gaspersz (1998) terdapat empat langkah yang diperlukan untuk melaksanakan RCCP, yaitu :

1. Memperoleh informasi tentang rencana produksi dari MPS.
2. Memperoleh informasi tentang struktur produk dan waktu tunggu (*lead time*).
3. Menentukan *bill of resources*.
4. Menghitung kebutuhan sumber daya spesifik dan membuat laporan RCCP.

### 2.3.3 Metode Pengukuran Kapasitas

Menurut Gaspersz (1998) terdapat tiga metode dalam pengukuran kapasitas, antara lain sebagai berikut :

1. *Theoretical Capacity (Maximum Capacity, Design Capacity)*

Merupakan kapasitas maksimum yang mungkin dari sistem manufaktur yang didasarkan pada asumsi mengenai adanya kondisi ideal seperti; tiga *shift* per hari, tujuh hari per minggu, tidak ada *downtime* mesin, dan lain-lain. dengan demikian *Theoretical Capacity* diukur berdasarkan pada jam kerja yang tersedia untuk melakukan pekerjaan, tanpa suatu kesempatan untuk berhenti atau istirahat, *downtime* mesin, atau alasan lainnya.

2. *Demonstrated Capacity (Actual Capacity, Effective Capacity)*

Merupakan tingkat output yang dapat diharapkan berdasarkan pada pengalaman, yang mengukur produksi secara aktual dari pusat kerja di waktu lalu,

yang biasanya diukur menggunakan angka rata-rata berdasarkan beban kerja normal.

### 3. *Rated Capacity (Calculated Capacity, Nominal Capacity)*

Pengukuran didasarkan pada penyesuaian kapasitas teoritis dengan faktor produktivitas yang telah ditentukan oleh *Demonstrated Capacity*. Dihitung melalui penggandaan waktu kerja yang tersedia dengan faktor utilisasi dan efisiensi.

Utilisasi adalah pecahan yang menggambarkan persentase *clock time* yang tersedia dalam pusat kerja yang secara aktual digunakan untuk produksi berdasarkan pengalaman masa lalu. Menurut Gaspersz (1998) formula untuk menghitung utilisasi dapat dilihat pada persamaan (2-1).

$$\text{Utilisasi} = \frac{\text{Jam aktual yang digunakan untuk produksi}}{\text{Jam yang tersedia menurut jadwal}} \quad (2-1)$$

Efisiensi adalah faktor yang mengukur performansi aktual dari pusat kerja relatif terhadap standar yang ditetapkan. Faktor efisiensi dapat lebih besar dari 1,0. Menurut Gaspersz (1998) formula untuk menghitung efisiensi dapat dilihat pada persamaan (2-2).

$$\text{Efisiensi} = \frac{\text{Jam standar yang diperoleh atau diproduksi}}{\text{jam aktual yang digunakan untuk produksi}} \quad (2-2)$$

Dengan demikian formulasi perhitungan *rated (calculated) capacity* dapat dilihat sesuai persamaan (2-3).

$$\begin{aligned} \text{Calculated capacity per periode} \\ &= \text{banyaknya orang atau mesin} \times \text{jam per shift} \times \text{shifts per hari} \\ &\quad \times \text{hari kerja per periode} \times \text{utilisasi} \times \text{efisiensi} \\ &= \text{waktu yg tersedia per periode waktu} \times \text{utilisasi} \times \text{efisiensi} \end{aligned} \quad (2-3)$$

#### 2.3.4 *Capacity Requirement Resources (CCR)*

Menurut Fogarty (1991) kendala dapat didefinisikan sebagai segala sesuatu yang menghambat suatu sistem untuk mencapai kinerja yang lebih tinggi. Terdapat dua tipe pokok dari kendala, yaitu kendala fisik dan kendala non fisik. Kendala fisik adalah

kendala yang berhubungan dengan kapasitas mesin, sedangkan kendala non fisik adalah kendala yang berupa permintaan terhadap produk dan prosedur kerja. Sumber daya berkendala kapasitas atau *Capacity Constraints Resources* (CCR) merupakan sumber daya yang tidak dijadwalkan dengan baik maka akan menghambat aliran produksi dan menyimpang dari perencanaan yang sudah ditetapkan.

## 2.4 Pengukuran Waktu Kerja

Menurut Wingjosoebroto (2003) pengukuran kerja merupakan suatu aktivitas yang dilakukan untuk mendapatkan waktu baku penyelesaian pekerjaan. Pengukuran kerja adalah metode penetapan keseimbangan antara kegiatan manusia yang dikontribusikan dengan unit output yang dihasilkan (Wingjosoebroto, 2003).

Terdapat dua teknik pengukuran kerja yang dapat dilakukan yaitu pengukuran kerja secara langsung dan tidak langsung. Pengukuran kerja langsung adalah pengukuran kerja yang dilakukan secara langsung ditempat dimana pekerjaan atau kerja yang diukur tersebut dilakukan. Sedangkan pengukuran kerja tidak langsung adalah cara perhitungan waktu kerja dimana peneliti tidak harus berada ditempat kerja yang diukur.

Dalam penelitian ini, pengukuran waktu kerja yang dilakukan adalah pengukuran kerja secara langsung dimana menggunakan metode jam henti atau *stopwatch time study*.

### 2.4.1 Pengukuran dengan Metode Jam Henti (*Stopwatch Time Study*)

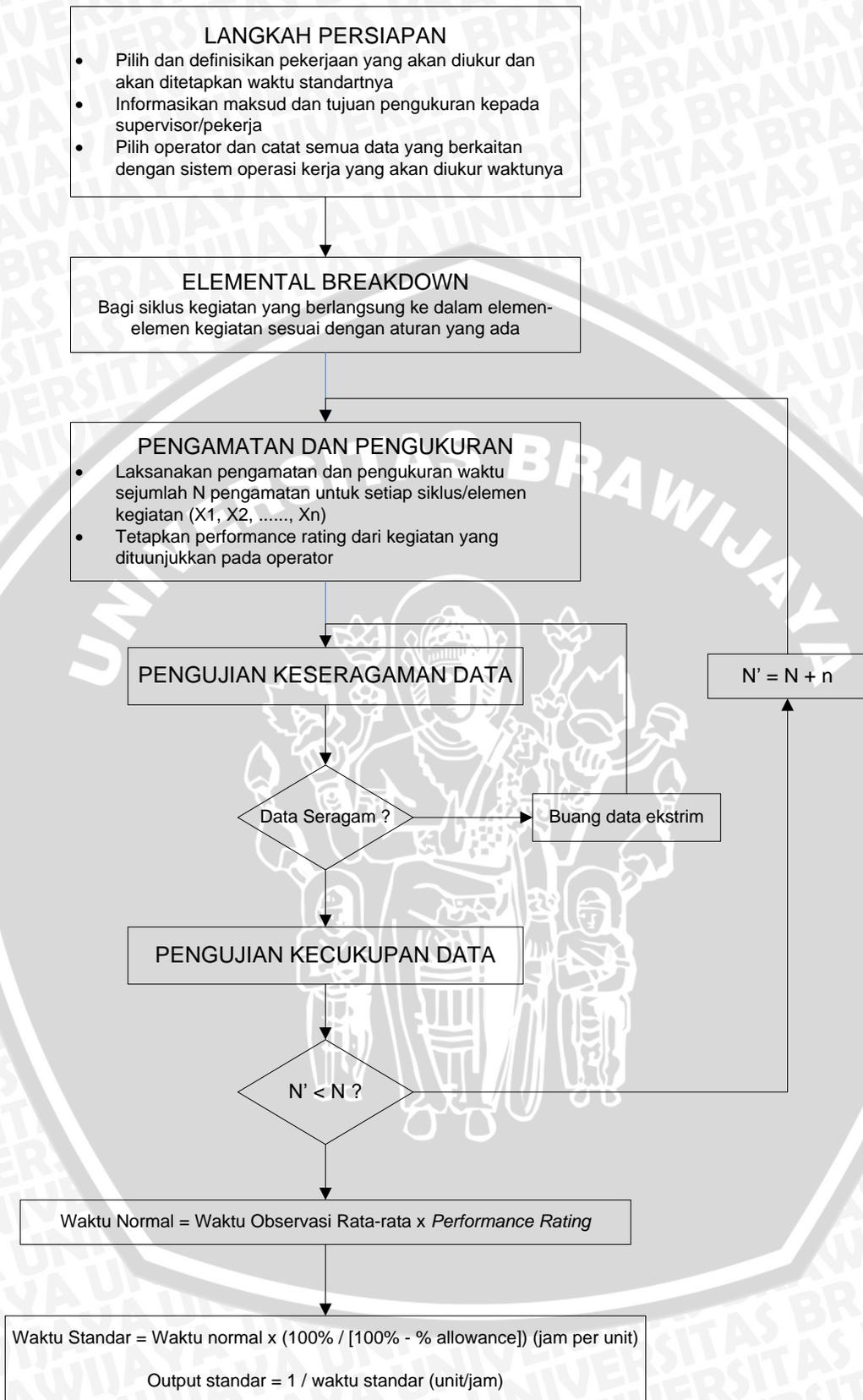
Pengukuran waktu kerja menggunakan metode jam henti diperkenalkan Frederick W. Taylor pada abad ke-19. Metode ini baik untuk diaplikasikan pada pekerjaan yang singkat dan berulang (*repetitive*). Dari hasil pengukuran akan diperoleh waktu baku untuk menyelesaikan suatu siklus pekerjaan yang akan dipergunakan sebagai waktu standar penyelesaian suatu pekerjaan bagi semua pekerja yang akan melaksanakan pekerjaan yang sama.

Dalam pengukuran kerja, hal-hal penting yang harus diketahui dan ditetapkan adalah untuk apa hasil pengukuran (dalam hal ini tentu saja waktu baku) tersebut digunakan dalam kaitannya dengan proses produksi. Aktivitas pengukuran kerja dengan jam henti umumnya diaplikasikan pada industri manufaktur yang memiliki karakteristik kerja yang berulang, terspesifikasi jelas, dan menghasilkan *output* yang relatif sama.

Secara garis besar, langkah-langkah dalam melaksanakan pengukuran waktu kerja antara lain sebagai berikut (Wingjosoebroto, 2003):

1. Mendefinisikan pekerjaan yang akan diteliti untuk diukur waktunya dan menginformasikan pada operator yang diukur dan diamati akan maksud dan tujuan pengukuran ini.
2. Mencatat semua informasi yang berkaitan erat dengan penyelesaian pekerjaan seperti spesifikasi mesin atau peralatan kerja lain yang digunakan.
3. Membagi operasi kerja dalam elemen-elemen kerja sedetail-detailnya tetapi masih dalam batas-batas kemudahan untuk pengukuran waktunya.
4. Mengamati, mengukur dan mencatat waktu yang dibutuhkan oleh operator untuk menyelesaikan elemen-elemen kerja tersebut.
5. Menetapkan jumlah siklus kerja yang diukur dan dicatat. Melakukan *test* keseragaman data yang diperoleh.
6. Menetapkan *rate of performance* dari operator saat melaksanakan aktivitas kerja yang diukur dan dicatat waktunya tersebut. *Rate of Performance* ini ditetapkan untuk setiap elemen kerja yang ada dan hanya ditujukan untuk *performance* operator. Untuk elemen kerja yang secara penuh dilaksanakan oleh mesin maka *performance* dianggap normal (100%)
7. Menyesuaikan waktu pengamatan berdasarkan *performance* yang ditunjukkan oleh operator tersebut sehingga akhirnya akan diperoleh waktu kerja normal.
8. Menetapkan waktu longgar (*allowance time*) untuk memberikan fleksibilitas waktu yang diberikan pada operator dalam kondisi-kondisi seperti kebutuhan personal yang bersifat pribadi dan faktor kelelahan pada operator.
9. Menetapkan waktu kerja baku (*standart time*) yaitu jumlah total antara waktu normal dan waktu longgar.

Pada Gambar 2.3 berikut ini dijelaskan mengenai langkah-langkah sistematis dalam pengukuran kerja dengan metode Jam Henti (*stopwatch time study*) yang disajikan dalam bentuk diagram alir.



Gambar 2.3 Langkah-Langkah Sistematis Pengukuran Kerja dengan Jam Henti  
Sumber : Wignjosoebroto (2003)

### 2.4.2 Perhitungan Waktu Normal

Waktu normal merupakan waktu kerja yang dibutuhkan oleh operator secara wajar atau semestinya yang artinya pekerjaan tersebut diselesaikan tidak terlalu cepat dan tidak terlalu lambat. Waktu normal didapatkan dari waktu siklus dikalikan dengan faktor penyesuaian (*performance rating*). Menurut Wignjosoebroto (2003) rumus perhitungan waktu siklus dapat dilihat pada persamaan (2-4).

$$W_s = \frac{\sum x}{N} \quad (2-4)$$

Dimana :

$W_s$  = waktu siklus

$\sum x$  = jumlah waktu pengamatan

$N$  = banyaknya pengamatan

Menurut Wignjosoebroto (2003) rumus perhitungan waktu normal dapat dilihat pada persamaan (2-5).

$$W_n = W_s \times (1 + p) \quad (2-5)$$

Dimana :

$W_n$  = waktu normal

$W_s$  = waktu siklus

$p$  = *performance rating*

Metode yang digunakan dalam menentukan *performance rating* dalam penelitian ini adalah dengan metode *Westing House*, dimana terdapat empat faktor yang menjadi penentu nilai *performance rating*, yaitu : kinerja operator (*performance*), ketrampilan (*skill*), usaha (*condition*), dan konsistensi (*consistency*). Tabel dari *performance rating* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Tabel *Performance Rating* dengan Sistem *Westing House*

SKILL			EFFORT		
+ 0,15	A1	Superskill	+ 0,13	A1	Superskill
+ 0,13	A2		+ 0,12	A2	
+ 0,11	B1	Excellent	+ 0,10	B1	Excellent
+ 0,08	B2		+ 0,08	B2	
+ 0,06	C1	Good	+ 0,05	C1	Good
+ 0,03	C2		+ 0,02	C2	
0,00	D	Average	0,00	D	Average
-0,05	E1	Fair	-0,04	E1	Fair
-0,10	E2		-0,08	E2	
-0,16	F1	Poor	-0,12	F1	Poor
-0,22	F2		-0,17	F2	
CONDITION			CONSISTENCY		
+ 0,06	A	Ideal	+ 0,04	A	Ideal
+ 0,04	B	Excellent	+ 0,03	B	Excellent
+ 0,02	C	Good	+ 0,01	C	Good

Lanjutan Tabel 2.2 Tabel *Performance Ratings* dengan Sistem *Westinghouse*

CONDITION			CONSISTENCY		
0,00	D	Average	0,00	D	Average
-0,03	E	Fair	-0,02	E	Fair
-0,07	F	Poor	-0,04	F	Poor

Sumber: Wignjosoebroto (2003)

### 2.4.3 Perhitungan Waktu Baku

Waktu baku merupakan waktu yang dibutuhkan secara wajar oleh pekerja yang bekerja secara normal dengan memperhatikan faktor kelonggaran (*allowance*) untuk melakukan suatu pekerjaan dimana *allowance* yang diberikan kepada operator di masing-masing stasiun kerja berdasarkan rekomendasi ILO yang disajikan pada Tabel 2.3. Menurut Ginting (2009) rumus perhitungan waktu baku dapat dilihat pada persamaan (2-6).

$$Wb = Wn \times \frac{100\%}{100\% - allowance(\%)} \quad (2-6)$$

Dimana :

 $Wb$  = waktu baku $Wn$  = waktu normal*allowance*(%) = prosentase faktor kelonggaranTabel 2.3 Rekomendasi *Allowance* ILO

Allowance	Percentage
Constant allowances:	
1. Personal Allowance	5%
2. Basic fatigue Allowance	4%
Variable allowances:	
1. Standing allowance	2%
2. Abnormal position allowance	
a. Slightly awkward	0%
b. Awkward (bending)	2%
c. Very awkward (lying, stretching)	7%
3. Use of force, or muscular energy (lifting, pulling, or pushing)	
5	0%
10	1%
15	2%
20	3%
25	4%
30	5%
35	7%
40	9%
45	11%
50	13%
60	17%
70	22%

Lanjutan Tabel 2.3 Rekomendasi Allowance ILO

Allowance	Percentage
4. Bad light:	
a. Slightly below recommended	0%
b. Well below	2%
c. Quite inadequate	5%
5. Atmospheric conditions (heat and humidity)	0-100%
6. Close attention:	
a. Fairly fine work	0%
b. Fine or exacting	2%
c. Very fine or very exacting	5%
7. Noise level:	
a. Continous	0%
b. Intermittent - loud	2%
c. Intermittent – very loud	5%
d. High-pitched – loud	5%
8. Metal strain:	
a. Fairly complex process	1%
b. Complex or wide span of attention	4%
c. Very complex	8%
9. Monotony:	
a. Low	0%
b. Medium	1%
c. High	4%
10. Tediousness:	
a. Rather tedious	0%
b. Tedious	2%
c. Very tedious	5%

Sumber: Niebel dan Freivalds (2000)

#### 2.4.4 Pengujian Keseragaman Data

Uji keseragaman data dilakukan agar data yang digunakan dalam penelitian bukan merupakan data yang ekstrim (data yang nilainya terlalu besar atau terlalu kecil dan menyimpang jauh dari nilai *trend* rata-ratanya). Menurut Wignjosoebroto (2003) batas atas dan batas bawah yang digunakan dalam pembuatan peta kontrol dapat dilihat pada persamaan (2-7) dan (2-8).

$$BKA = X + (3.SD) \quad (2-7)$$

$$BKB = X - (3.SD) \quad (2-8)$$

Dimana :

*BKA* = batas kontrol atas

*BKB* = batas kontrol bawah

*X* = nilai rata-rata

*SD* ( $\sigma$ ) = standar deviasi

### 2.4.5 Pengujian Kecukupan Data

Uji kecukupan data dilakukan untuk melihat apakah data yang diambil sudah mewakili data populasi dan sesuai dengan tingkat ketelitian yang diinginkan atau tidak. Menurut Groover (2007) rumus perhitungan kecukupan data dapat dilihat pada persamaan (2-9).

$$n = \left[ \frac{Z \times S}{e \times T} \right]^2 \quad (2-9)$$

Dimana :

$n$  = banyaknya data yang harus diambil

$z$  = tingkat kepercayaan (95%) sehingga  $Z_{\alpha/2}=1,96$

$s$  = standar deviasi

$$S(\sigma) = \sqrt{\frac{\sum(X_i - \bar{X})^2}{n-1}}$$

$T$  = waktu siklus atau *cycle time*

$$\text{cycle time} = \frac{\sum x}{\sum n}$$

### 2.5 Pemrograman Linier (*Linear Programming*)

Pemrograman linier (*linear programming*) adalah suatu metode yang digunakan dalam menentukan alokasi sumber dengan cara yang optimal (Taha, 1996). Menurut Gaspersz (1998) *linear programming* merupakan teknik riset oprasional yang telah dipergunakan secara luas dalam berbagai jenis masalah manajemen. Dalam hal ini pada proses perencanaan yang mengalokasikan sumber-sumber daya manufaktur seperti: mesin, tenaga kerja, modal, waktu, dan bahan baku digunakan dalam kombinasi tertentu yang paling optimum untuk menghasilkan produk (barang atau jasa).

Gaspersz (1998) menyebutkan terdapat lima karakteristik utama dalam masalah program linier (*linear programming*) adalah sebagai berikut :

1. Masalah *linear programming* berkaitan dengan upaya memaksimalkan (pada umumnya keuntungan) atau meminimumkan (pada umumnya biaya). Upaya optimasi (maksimum atau minimum) ini disebut sebagai fungsi tujuan (*objective function*) dari *linear programming*. Fungsi tujuan ini terdiri dari variabel-variabel keputusan (*desicion variable*).
2. Terdapat kendala-kendala atau keterbatasan, yang membatasi pencapaian tujuan yang dirumuskan dalam *linear programming*. Kendala-kendala ini dirumuskan dalam fungsi-fungsi kendala (*constraint's function*), terdiri dari variabel-variabel

keputusan yang menggunakan sumber-sumber daya yang terbatas itu. Dengan demikian yang akan diselesaikan dalam *linear programming* adalah mencapai fungsi tujuan (maksimum keuntungan atau minimum biaya) dengan memperhatikan fungsi-fungsi kendala (keterbatasan atau kendala) sumber-sumber daya yang ada.

3. Memiliki sifat linearitas. Sifat linearitas ini berlaku untuk semua fungsi tujuan dan fungsi-fungsi kendala.
4. Memiliki sifat homogenitas. Sifat homogenitas ini berkaitan dengan kesamaan sumber-sumber daya yang digunakan dalam proses produksi.
5. Memiliki sifat *divisibility*. Sifat *divisibility* diperlukan, karena *linear programming* mengasumsikan bahwa nilai dari variabel-variabel keputusan maupun penggunaan sumber-sumber daya dapat dibagi ke dalam pecahan-pecahan. Jika pembagian ini tidak mungkin dilakukan terhadap variabel keputusan, misalnya industri mobil, furnitur, dan lain-lain, karena nilai kuantitas produksi diukur dalam bilangan bulat, maka modifikasi terhadap LP harus dilakukan. Bentuk modifikasi dari LP ini disebut dengan *integer linear programming*.

Secara matematik, model umum dari pemograman linier yang terdiri dari sekumpulan variabel keputusan  $X_1, X_2, \dots, X_n$  dapat dirumuskan sesuai dengan persamaan (2-10) sebagai berikut :

$$\text{Maksimum (atau Minimum) } Z = C_1X_1 + C_2X_2 + \dots + C_nX_n \quad (2-10)$$

Dengan kendala :

$$A_{11}X_1 + A_{12}X_2 + \dots + A_{1n}X_n \leq B_1$$

$$A_{21}X_1 + A_{22}X_2 + \dots + A_{2n}X_n \leq B_2$$

$$A_{m1}X_1 + A_{m2}X_2 + \dots + A_{mn}X_n \leq B_m$$

$$X_1 \leq D_1$$

$$X_n \leq D_n$$

$$X_1, X_2, \dots, X_n \geq 0$$

Dimana pada penelitian ini dalam mengelola dan meningkatkan kapasitas stasiun kerja yang tidak mencukupi guna meminimasi SOP dengan penamaan tiap-tiap variabel sebagai berikut:

$X_1, X_2, \dots, X_n$  = jumlah masing-masing tipe produk

$C_1, C_2, \dots, C_n$  = target *order* produksi

$A_{m1}, A_{m2}, \dots, A_{mn}$  = waktu proses masing-masing tipe produk pada tiap SK

$B_1, B_2, \dots, B_m$  = kapasitas yang tersedia di masing-masing SK (menit)

$D_1, D_2, \dots, D_n$  = permintaan masing-masing tipe produk

### 2.5.1 Integer Linear Programming (ILP)

Menurut Sitompul (2006) pemograman linier *integer* (*integer linear programming*) pada intinya berkaitan dengan program linier dimana beberapa atau semua variabel memiliki nilai integer (bulat). Program *integer* dibagi atas tiga jenis, yaitu:

1. Program *integer* murni (*Pure Integer Programming*), semua variabel keputusannya adalah integer.
2. Program *integer* campuran (*Mixed Integer Programming*), sebagian keputusannya adalah integer.
3. Program *integer* 0-1 (*Zero One Integer Programming*), variabel keputusannya hanya memiliki nilai 0 dan 1.

Model matematis untuk pemograman linier *integer* serupa dengan model pemograman linier, perbedaannya hanya pada penambahan 1 kendala bahwa variabelnya harus berupa bilangan bulat. Pada dasarnya *integer programming* merupakan analisis pasca optimal pemrograman linier. Jika program linier menghasilkan bilangan pecahan maka untuk mendapatkan bilangan bulat yang optimal dilakukan dengan menggunakan *integer linier programming* (ILP).

Metode yang biasa diterapkan adalah metode percabangan dan pembatasan (*branch and bound*) serta algoritma bidang pemotong (*cutting plane*). Menurut Taha (1996) metode *cutting plane* tidak dapat digunakan secara efektif untuk memecahkan masalah integer umum tetapi gagasan dari metode itu dapat meningkatkan efektivitas jenis teknik pemecahan lainnya. Jadi, bila terdapat pilihan antara metode *cutting plane* dan metode *branch and bound*, maka metode *branch and bound* umumnya terbukti lebih baik.

### 2.5.2 Metode Branch and Bound

Menurut Siswanto (2006) metode *branch and bound* adalah sebuah metode untuk menghasilkan penyelesaian optimal pemrograman linier yang menghasilkan variabel-variabel keputusan bilangan bulat. Sesuai dengan namanya, metode ini membatasi penyelesaian optimal yang akan menghasilkan bilangan pecahan dengan cara membuat cabang batas atas dan bawah bagi masing-masing variabel keputusan yang bernilai pecahan agar bernilai bulat sehingga setiap pembatasan akan menghasilkan cabang baru.

Langkah-langkah penyelesaian masalah program linier menggunakan metode cabang batas, yaitu :

1. Metode ini diawali dengan metode simpleks sampai terdapat penyelesaian optimal. Kemudian untuk basis  $X_j^*$  variabel yang nilainya real dirubah menjadi *integer*  $X_j$  yang batasnya  $X_j^* \leq X_j \leq X_j^* + 1$ . Tetapi karena range tersebut tidak memberikan penyelesaian *integer*, maka konsekuensinya nilai *integer*  $X_j$  harus memenuhi salah satu syarat dibawah ini:  $X_j \geq X_j^*$  atau  $X_j \leq X_j^* + 1$ .
2. Kemudian persoalan pemrograman linier yang awal dengan kendala tambahan  $X_j \geq X_j^*$  atau  $X_j \leq X_j^* + 1$  diselesaikan sampai diperoleh keadaan optimum. Dengan demikian setiap  $X_j$  akan menghasilkan dua cabang yang berbeda, dengan nilai basis dan nilai fungsi tujuan optimum yang berbeda. Basis yang sudah *integer* tetapi menghasilkan nilai fungsi tujuan yang lebih rendah dari basis yang *integer* di cabang lain, harus dibuang dan cabang tersebut tidak perlu dilanjutkan penyelusurannya.
3. Nilai  $X_j$  yang *integer* lalu dimasukkan ke dalam basis sampai semua variabel basis yang diinginkan menjadi *integer*, setiap  $X_j$  yang baru akan menghasilkan dua cabang yang baru, kecuali cabang yang tidak fisibel. Cabang yang tidak fisibel langsung dapat dibuang.  
Dengan cara demikian akan dapat diketahui semua nilai variable basis yang *integer* dan memberikan penyelesaian optimum yang fisibel.