

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Peramalan (*Forecasting*)**

##### **2.1.1 Pengertian Peramalan (*Forecasting*)**

Peramalan merupakan usaha untuk melihat situasi dan kondisi pada masa yang akan datang dengan cara memperkirakan pengaruh situasi dan kondisi pada masa yang akan datang terhadap perkembangan di masa yang akan datang (Ginting, 2007).

Tersine (1994) menjelaskan bahwa peramalan adalah sebuah prediksi, proyeksi, atau estimasi dari ketidakpastian masa depan. Pada umumnya kegiatan peramalan adalah sebagai berikut:

1. Sebagai alat bantu dalam perencanaan yang efektif dan efisien.
2. Untuk menentukan kebutuhan sumber daya di masa mendatang.
3. Untuk membuat keputusan yang tepat.

##### **2.1.2 Kegunaan dan Peran Peramalan**

Kegunaan dari peramalan terlihat pada saat pengambilan keputusan. Keputusan yang baik adalah keputusan yang didasarkan atas pertimbangan apa yang akan terjadi pada waktu keputusan itu dilaksanakan. Apabila kurang tepat ramalan yang kita susun, maka masalah peramalan juga merupakan masalah yang selalu kita hadapi. (Ginting, 2007).

##### **2.1.3 Jenis Peramalan**

Pada umumnya peramalan dapat dibedakan dari beberapa segi tergantung dan cara melihatnya. Apabila dilihat dari sifat penyusunannya, maka peramalan dapat dibedakan atas dua macam (Ginting, 2007):

1. Peramalan yang subjektif, yaitu peramalan yang didasarkan atas perasaan atau intuisi dari orang yang menyusunnya.
2. Peramalan yang objektif, yaitu peramalan yang didasarkan atas data yang relevan pada masa lalu, dengan menggunakan teknik-teknik dan metode-metode dalam penganalisaan data tersebut.

Disamping itu, jika dilihat dari jangka waktu ramalan yang disusun, maka peramalan dibedakan menjadi 2, yaitu (Ginting, 2007):

1. Peramalan jangka panjang, yaitu peramalan yang dilakukan untuk penyusunan lebih dari satu setengah tahun atau tiga semester.
2. Peramalan jangka pendek, yaitu peramalan yang dilakukan untuk penyusunan hasil ramalan dengan jangka waktu yang kurang dari satu setengah tahun atau tiga semester.

Berdasarkan sifat ramalan yang telah disusun, maka peramalan dapat dibedakan atas dua macam, yaitu (Ginting, 2007):

1. Peramalan kualitatif, yaitu peramalan yang didasarkan atas data kualitatif pada masa lalu. Hasil peramalan yang dibuat sangat tergantung pada orang yang menyusunnya.
2. Peramalan kuantitatif, yaitu peramalan yang didasarkan atas data kuantitatif pada masa lalu. Hasil peramalan yang dibuat sangat tergantung pada metode yang dipergunakan dalam peramalan tersebut.

#### **2.1.4 Langkah-Langkah Peramalan**

Peramalan yang baik adalah peramalan yang dilakukan dengan mengikuti langkah-langkah atau prosedur penyusunan yang baik. Pada dasarnya ada tiga langkah peramalan yang penting, yaitu (Ginting, 2007):

1. Menganalisa data yang lalu. Analisa ini dilakukan dengan cara membuat tabulasi dari data yang lalu. Dengan tabulasi tersebut dapat diketahui pola dari data tersebut.
2. Menentukan metode yang dipergunakan.
3. Memproyeksikan data yang lalu dengan menggunakan metode yang digunakan, dan mempertimbangkan adanya beberapa faktor perubahan.

#### **2.1.5 Jenis-Jenis Pola Data**

Langkah penting sebelum menentukan metode peramalan adalah menentukan pola data. Pola data dapat dibedakan menjadi empat, yaitu (Ginting, 2007):

1. **Pola Horizontal (H)**, terjadi bilamana nilai data berfluktuasi di sekitar nilai rata-rata yang konstan.
2. **Pola Musiman (S)**, terjadi bilamana suatu deret dipengaruhi oleh faktor musiman (misalnya kuartal tahun tertentu, bulanan, atau hari-hari pada minggu tertentu).
3. **Pola Siklis (C)**, terjadi bilamana datanya dipengaruhi oleh fluktuasi ekonomi jangka panjang seperti yang berhubungan dengan siklus bisnis.
4. **Pola Trend (T)**, terjadi bilamana terdapat kenaikan atau penurunan sekuler jangka panjang dalam data.

#### 2.1.6 Jenis-Jenis Metode Peramalan

Sejauh ini telah dikembangkan beberapa metode atau teknik-teknik peramalan untuk menghadapi bermacam-macam keadaan yang terjadi. Namun pada dasarnya metode peramalan kuantitatif dapat dibedakan atas (Ginting, 2007):

1. Metode peramalan yang didasarkan atas penggunaan analisa pola hubungan antara variabel yang akan diperkirakan dengan variabel waktu, yang merupakan deret waktu, atau *time-series*.
2. Metode peramalan yang didasarkan atas penggunaan analisa pola hubungan antara variabel yang akan diperkirakan dengan variabel lain yang mempengaruhinya, yang bukan waktu, yang disebut korelasi atau sebab akibat (*causal methods*).

Metode-metode peramalan dengan menggunakan analisa pola hubungan antara variabel yang akan diperkirakan dengan variabel waktu, terdiri dari (Ginting, 2007):

- a. Metode *smoothing*, yang mencakup metode data lewat (*past data*), metode rata-rata kumulatif, metode rata-rata bergerak (*moving averages*), dan metode *exponential smoothing*.
- b. Metode Box Jenkins, menggunakan dasar waktu dengan model matematis, agar kesalahan yang terjadi dapat sekecil mungkin.

- c. Metode proyeksi trend dan regresi, dasar garis trend untuk suatu persamaan matematis, sehingga dengan dasar persamaan tersebut dapat diproyeksikan hal yang diteliti untuk masa depan.

Metode-metode peramalan dengan menggunakan analisa pola hubungan antara variabel yang akan diperkirakan dengan variabel variabel lain yang mempengaruhinya, yang bukan waktu, yang disebut korelasi atau sebab akibat (*causal methods*), terdiri dari (Ginting, 2007):

- a. Metode regresi dan korelasi, didasarkan pada penetapan estimasi menggunakan teknik “*least square*”.
- b. Metode ekonometri, didasarkan atas peramalan pada sistem persamaan regresi yang diestimasi secara simultan.
- c. Metode input-output, dipergunakan untuk menyusun proyeksi trend ekonomi jangka panjang.

#### **2.1.7 Pemilihan Teknik dan Metode Peramalan**

Dalam pemilihan teknik dan metode peramalan, pertama kita perlu mengetahui ciri-ciri yang penting yang perlu diperhatikan bagi pengambilan keputusan dan analisa keadaan, dalam mempersiapkan peramalan. Ada 6 ciri utama yang perlu diperhatikan, yaitu (Ginting, 2007):

1. Horison waktu (*time horizon*). Horizon waktu umumnya dibagi ke dalam jangka pendek (satu sampai dengan tiga bulan), menengah (tiga sampai dengan setahun), dan jangka panjang (lebih dari satu setengah tahun).
2. Tingkat perincian (*level of detail*).
3. Jumlah produk.
4. Pengawasan versus perencanaan.
5. Stabilitas.
6. Prosedur perencanaan yang ada.

Banyak ciri-ciri lain yang penting dalam pemilihan teknik dan metode peramalan, akan tetapi enam hal di atas adalah yang terpenting. Sebagai tambahan dalam memperimbangan keadaan, pengambil

keputusan atau analisa harus pula mempertimbangkan ciri-ciri dari berbagai teknik dan metode peramalan. Adapun enam faktor utama yang dapat diidentifikasi sebagai teknik dan metode peramalan, yaitu (Ginting, 2007):

1. Horison waktu (*time horizon*). Ada dua aspek dari horizon waktu yang berhubungan dengan metode peramalan, yaitu cakupan waktu di masa yang akan datang dan jumlah periode untuk mana ramalan diinginkan.
2. Pola dari data. Dasar utama dari metode peramalan adalah anggapan bahwa macam dari pola yang didapati di dalam data yang diramalkan akan berkelanjutan.
3. Jenis dari model. Ada dua model dari keadaan yang diramalkan, yaitu model *time-series* dan *causal model*.
4. Biaya. Umumnya ada empat unsur biaya yang tercakup dalam penggunaan suatu prosedur peramalan, yaitu biaya-biaya pengembangan, penyimpanan data, operasi pelaksanaan dan kesempatan dalam penggunaan teknik-teknik dan metode lain.
5. Ketepatan (*accuracy*). Tingkat ketepatan yang dibutuhkan sangat erat hubungannya dengan tingkat perincian yang dibutuhkan dalam suatu peramalan.
6. Mudah tidaknya penggunaan atau aplikasinya.

### 2.1.8 Ukuran Akurasi Peramalan

Model-model peramalan yang dilakukan kemudian divalidasi menggunakan sejumlah indikator. Indikator-indikator yang umum digunakan adalah rata-rata penyimpangan absolut (*Mean Absolute Deviation*), rata-rata kuadrat terkecil (*Mean Square Error*), rata-rata persentase kesalahan absolut (*Mean Absolute Percentage Error*), validasi peramalan (*Tracking Signal*), dan pengujian kestabilan (*Moving Range*).

#### 1. *Mean Absolute Deviation* (MAD)

Metode untuk mengevaluasi metode peramalan menggunakan jumlah dari kesalahan-kesalahan yang absolut. *Mean*

*Absolute Deviation* (MAD) mengukur ketepatan ramalan dengan merata-rata kesalahan dugaan (nilai absolut masing-masing kesalahan). MAD berguna ketika mengukur kesalahan ramalan dalam unit yang sama sebagai deret asli.

2. *Mean Square Error* (MSE)

*Mean Squared Error* (MSE) adalah metode lain untuk mengevaluasi metode peramalan. Masing-masing kesalahan atau sisa dikuadratkan. Kemudian dijumlahkan dan ditambahkan dengan jumlah observasi. Pendekatan ini mengatur kesalahan peramalan yang besar karena kesalahan-kesalahan itu dikuadratkan. Metode itu menghasilkan kesalahan-kesalahan sedang yang kemungkinan lebih baik untuk kesalahan kecil, tetapi kadang menghasilkan perbedaan yang besar.

3. *Mean Absolute Percentage Error* (MAPE)

*Mean Absolute Percentage Error* (MAPE) dihitung dengan menggunakan kesalahan absolut pada tiap periode dibagi dengan nilai observasi yang nyata untuk periode itu. Kemudian, merata-rata kesalahan persentase absolut tersebut. Pendekatan ini berguna ketika ukuran atau besar variabel ramalan itu penting dalam mengevaluasi ketepatan ramalan. MAPE mengindikasikan seberapa besar kesalahan dalam meramal yang dibandingkan dengan nilai nyata.

4. *Tracking Signal*

Validasi peramalan dilakukan dengan *Tracking Signal*. *Tracking Signal* adalah suatu ukuran bagaimana baiknya suatu peramalan memperkirakan nilai-nilai aktual. Nilai *Tracking Signal* dapat dihitung dengan menggunakan rumus sebagai berikut.

*Tracking signal* yang positif menunjukkan bahwa nilai aktual permintaan lebih besar daripada ramalan, sedangkan *tracking signal* yang negatif berarti nilai aktual permintaan lebih kecil daripada ramalan. *Tracking signal* disebut baik apabila memiliki RSFE yang rendah, dan mempunyai *positive error* yang sama banyak atau seimbang dengan *negative error*, sehingga pusat dari *tracking signal* mendekati nol. *Tracking signal* yang telah dihitung dapat dibuat peta kontrol untuk

melihat kelayakan data di dalam batas kontrol atas dan batas kontrol bawah.

#### 5. *Moving Range* (MR)

Peta *Moving Range* dirancang untuk membandingkan nilai permintaan aktual dengan nilai peramalan. Data permintaan aktual dibandingkan dengan nilai peramalan pada periode yang sama. Peta tersebut dikembangkan ke periode yang akan datang hingga dapat dibandingkan data peramalan dengan permintaan aktual. Peta *Moving Range* digunakan untuk pengujian kestabilan sistem sebab-akibat yang mempengaruhi permintaan. Langkah-langkah dalam membuat peta rentang bergerak ini adalah sebagai berikut:

##### 1. Hitung rentang bergerak untuk setiap periode

$$MR = [(F_t - A_t) - (F_{t-1} - A_{t-1})]$$

Dimana:

MR = *Moving Range*

$F_t$  = nilai ramalan periode t

$A_t$  = nilai aktual periode t

$F_{t-1}$  = nilai ramalan periode t-1

$A_{t-1}$  = nilai aktual periode t-1

##### 2. Hitung rata-rata rentang bergerak

$$\overline{MR} = \frac{MR}{(n - 1)}$$

##### 3. Buat peta rentang bergerak dengan ketentuan:

- Sumbu Y adalah  $(F_t - A_t)$
- Sumbu X adalah periode n
- Batas Kendali Atas = 2,66 MR
- Batas Kendali Bawah = - 2,66 MR

##### 4. Plot $(F_t - A_t)$ untuk setiap periode

##### 5. Tentukan:

- Daerah A, yaitu daerah diluar  $|1,77 MR|$
- Daerah B, yaitu daerah diluar  $|0,89 MR|$

- Daerah C, yaitu daerah diatas dan dibawah garis tengah peta rentang bergerak
6. Kondisi Out Of Control terjadi apabila:
    - a. Dari 3 titik yang berurutan, 2 titik atau lebih di daerah A
    - b. Dari 5 titik yang berurutan. 4 titik atau lebih di daerah B
    - c. Seluruh titik berada atau di bawah center line
    - d. Satu titik di luar batas kontrol

Jika ditemukan satu titik yang berada diluar batas kendali pada saat peramalan diverifikasi maka harus ditentukan apakah data harus diabaikan atau mencari peramal baru. Jika ditemukan sebuah titik berada diluar batas kendali maka harus diselidiki penyebabnya. Penemuan itu mungkin saja membutuhkan penyelidikan yang ekstensif. Jika semua titik berada di dalam batas kendali, diasumsikan bahwa peramalan permintaan yang dihasilkan telah cukup baik. Jika terdapat titik yang berada di luar batas kendali, jelas bahwa peramalan yang didapat kurang baik dan harus direvisi (*Gaspersz, 1998*).

Kegunaan peta *Moving Range* ialah untuk melakukan verifikasi hasil peramalan *least square* terdahulu. Jika peta *Moving Range* menunjukkan keadaan diluar kriteria kendali. Hal ini berarti terdapat data yang tidak berasal dari sistem sebab-akibat yang sama dan harus dibuang maka peramalan pun harus diulangi lagi.

## **2.2 Pemrograman Linear**

### **2.2.1 Pengertian Pemrograman Linear**

Model *pemrograman linear* adalah model matematis perumusan masalah umum pengalokasian sumber daya untuk berbagai kegiatan. Model ini merupakan bentuk dan susunan dalam menyajikan masalah-masalah yang akan dipecahkan dengan teknik *pemrograman linear* (*Siswanto, 2007*).

*Program linear* merupakan salah satu cara yang digunakan dalam proses optimasi dari suatu persoalan yang dapat diinformasikan dalam

bentuk model matematis. Beberapa masalah yang sering dipecahkan dengan *pemrograman linear*, diantaranya adalah (Puryani dan Ristono, 2012):

- Penentuan kombinasi beberapa macam barang yang akan diproduksi.
- Penentuan kombinasi beberapa macam barang yang akan dijual/dipasarkan.
- Penentuan kombinasi beberapa campuran bahan mentah.
- Penentuan penjadwalan produksi yang paling baik (meminimumkan ongkos produksi).
- Penentuan pola pengangkutan barang yang paling baik (meminimumkan total ongkos angkut).
- Menentukan pola penugasan beberapa tugas pada beberapa operator (mesin) yang paling baik, dan lain-lain.

### 2.2.2 Bentuk Model *Program Linear*

Bentuk umum dari model *program linear* adalah sebagai berikut (Megasari, 2010):

Memaksimumkan atau Meminimumkan

$$Z = \sum_{j=1}^n C_j x_j \quad \dots(2.1)$$

Dengan kendala:

$$\sum_{i=1}^m a_{ij} x_j \{<, =, >\} b_i \text{ untuk } i = 1, 2, 3, \dots, m \quad \dots(2.2)$$

$$x_j \geq 0 \text{ untuk } j = 1, 2, 3, \dots, n \quad \dots(2.3)$$

Keterangan:

$Z$  : nilai fungsi tujuan.

$C_j$  : sumbangan per unit kegiatan, untuk masalah maksimisasi  $C_j$  menunjukkan keuntungan atau penerimaan per unit, sementara dalam kasus minimasi menunjukkan biaya per unit.

$x_j$  : banyaknya kegiatan  $j$ , dengan  $j = 1, 2, 3, \dots, n$ .

$a_{ij}$  : banyaknya sumber daya  $i$  yang dikonsumsi kegiatan  $j$ .

$b_i$  : jumlah sumber daya  $i$  ( $i = 1, 2, \dots, m$ ).

Persamaan (2.1) dinamakan fungsi tujuan, yaitu fungsi matematis dari variable-variabel keputusan yang menunjukkan hubungan dengan nilai sisi kanannya. Persamaan (2.2) dinamakan kendala utama, yaitu fungsi matematis dari variabel-variabel simpangan yang menyatakan kombinasi sebuah objektif. Persamaan (2.3) dinamakan kendala non negatif, yaitu tujuan yang tidak boleh dilanggar dengan pengertian mempunyai penyimpangan positif dan atau negatif bernilai nol.

### 2.2.3 Sifat *Program Linear*

Sifat-sifat dari program linear adalah sebagai berikut (Puryani dan Ristono, 2012):

- a. Fungsi tujuan dan fungsi-fungsi pembatas (kendala) semuanya merupakan fungsi linear.
- b. Semua variabel keputusan yang terlibat dalam masalah adalah tidak negatif. Pemrograman linear hanya berhubungan dengan masalah nyata dimana harga variabel keputusan negatif adalah tidak logis.
- c. Kriteria pemilihan nilai terbaik dari variabel keputusan dapat ditentukan dengan fungsi linear dari variabel-variabel tersebut. Fungsi kriteria ini disebut fungsi tujuan.
- d. Aturan operasi yang mengatur proses dapat digambarkan sebagai satu set persamaan atau kesamaan linear. Set ini disebut set (himpunan) pembatas (kendala).

### 2.2.4 Manipulasi *Program Linear*

Pada model *program linear* dapat dilakukan perubahan bentuk tanpa mengubah maksud yang terkandung di dalamnya. Perubahan tersebut dilakukan untuk suatu kepentingan tertentu, misalnya dalam pemecahan menggunakan simpleks atau teknik lain (Puryani dan Ristono, 2012).

- 1) Konversi fungsi tujuan dari maksimasi menjadi minimasi dan sebaliknya.

$$\text{Maksimasi } \sum_{j=1}^n c_j x_j = - \text{Minimasi } \sum_{j=1}^n c_j x_j$$

- 2) Bentuk ketidaksamaan ( $\leq, =, \geq$ ) dapat diubah arahnya menjadi berlawanan, dengan cara mengalikan kedua ruas ketidaksamaan dengan bilangan -1.

Misal:  $a_{11}x_1 + a_{12}x_2 \geq b_1$  adalah ekuivalen dengan:

$$-a_{11}x_1 - a_{12}x_2 \leq -b_1$$

- 3) Sebuah bentuk persamaan dapat diganti dengan dua buah bentuk ketidaksamaan yang berlawanan arah.

Misal:  $a_{11}x_1 + a_{12}x_2 = b_1$  adalah ekuivalen dengan:

$$a_{11}x_1 + a_{12}x_2 \geq b_1 \text{ dan } a_{11}x_1 + a_{12}x_2 \leq b_1$$

- 4) Sebuah bentuk ketidaksamaan dengan ruas kirinya berada dalam bentuk harga mutlak dapat diubah ke dalam dua bentuk ketidaksamaan biasa.
- 5) Sebuah variabel keputusan yang tidak dibatasi tandanya (boleh negatif, nol, atau positif) adalah ekuivalen dengan selisih antara 2 variabel keputusan yang tidak negatif.

Misalnya:  $x_1$  tandanya tidak dibatasi (*un-constrained in sign*)

Adalah ekuivalen dengan:  $x_1 = (x_1^+ - x_1^-)$

Dimana  $x_1^+ \geq 0$  dan  $x_1^- \geq 0$

### 2.2.5 Menentukan Model Matematika dari Kalimat Verbal

Model matematika adalah suatu cara penyelesaian masalah dengan cara mengubah bentuk kalimat verbal menjadi suatu model yang selanjutnya diselesaikan dengan pendekatan matematika (Puryani dan Ristono, 2007).

Langkah-langkah dalam melakukan formulasi matematika adalah sebagai berikut (Puryani dan Ristono, 2007):

1. Memahami masalah.
2. Identifikasi variabel keputusan.
3. Penentuan fungsi tujuan sebagai kombinasi linear dari variabel keputusan.
4. Menentukan konstrain sebagai kombinasi linear dari variabel keputusan.
5. Identifikasi batas atas atau bawah dari variabel keputusan.

Suatu persoalan disebut persoalan program linear apabila memenuhi hal-hal berikut (Puryani dan Ristono, 2007):

1. Tujuan (*objective*)

Apa yang menjadi tujuan permasalahan yang dihadapi yang ingin dipecahkan dan dicari jalan keluarnya. Tujuan ini harus jelas dan tegas yang disebut fungsi tujuan (*objective function*). Fungsi tujuan tersebut dapat berupa dampak positif yang berupa manfaat-manfaat, atau dampak negatif yang berupa kerugian, resiko, biaya, jarak maupun waktu yang ingin diminimumkan.

2. Alternatif Perbandingan

Harus ada sesuatu atau alternatif yang ingin diperbandingkan, misalnya antara kombinasi waktu tercepat dan biaya tertinggi dengan waktu terlambat dan biaya terendah, atau alternatif padat modal dengan padat karya, proyeksi permintaan tinggi dengan rendah, dan seterusnya.

3. Sumber Daya

Sumber daya yang dianalisis harus berada dalam keadaan terbatas. Misalnya keterbatasan tenaga, bahan mentah terbatas, modal terbatas, ruangan penyimpanan barang terbatas, dan lain-lain.

Pembatasan harus dalam ketidaksamaan linear (*linear inequality*). Keterbatasan dalam sumber daya tersebut dinamakan sebagai fungsi kendala atau syarat ikatan.

4. Perumusan Kuantitatif

Fungsi tujuan dan kendala tersebut harus dapat dirumuskan secara kuantitatif dalam model matematika.

5. Keterikatan Peubah

Peubah-peubah yang membentuk fungsi tujuan dan fungsi kendala tersebut harus memiliki hubungan keterikatan atau hubungan fungsional.

## 2.3 *Goal Programming*

### 2.3.1 *Pengertian Goal Programming*

Model *goal programming* merupakan perluasan dari model *pemrograman linear* yang dikembangkan oleh A. Charles dan W. M. Cooper pada tahun 1956. Sehingga seluruh asumsi, notasi, formulasi matematika, prosedur perumusan model dan penyelesaian tidak berbeda. Perbedaannya hanya terletak pada kehadiran sepasang variabel deviasional yang muncul di fungsi tujuan dan fungsi kendala (Siswanto, 2007). *Pemrograman linear* sendiri adalah sebuah model matematis yang dipergunakan untuk menemukan suatu penyelesaian optimal dengan cara memaksimalkan atau meminimumkan fungsi tujuan terhadap satu kendala susunan. Model *goal programming* mempunyai tiga unsur utama, yaitu variabel keputusan, fungsi tujuan dan fungsi kendala.

Beberapa asumsi dasar yang diperlukan dalam *goal programming* adalah sebagai berikut (Harjiyanto, 2014):

a. *Linieritas*

Asumsi ini menunjukkan perbandingan antara input yang satu dengan input yang lain atau suatu input dengan output besarnya tetap dan terlepas pada tingkat produksi. Hubungannya bersifat linear.

b. *Proporsionalitas*

Asumsi ini menyatakan bahwa jika peubah pengambil keputusan berubah, maka dampak perubahannya akan menyebar dalam proporsi yang sebanding dengan fungsi tujuan dan juga fungsi kendalanya. Jadi tidak berlaku hukum kenaikan hasil yang semakin berkurang.

c. *Aditivitas*

Asumsi ini menyatakan nilai parameter suatu kriteria optimisasi merupakan jumlah dari nilai individu-individu. Dampak total terhadap kendala ke-i merupakan jumlah dampak individu terhadap peubah pengambilan keputusan.

d. *Divisibilitas*

Asumsi ini menyatakan bahwa peubah pengambilan keputusan jika diperlukan dapat dibagi ke dalam pecahan-pecahan.

e. Deterministik

Asumsi ini menghendaki agar semua parameter tetap dan diketahui atau ditentukan secara pasti.

Ada beberapa istilah yang dipergunakan dalam *goal programming* yaitu (Harjiyanto, 2014):

- 1) **Variabel keputusan (*decision variables*)**, adalah seperangkat variabel yang tidak diketahui yang berada di bawah kontrol pengambilan keputusan, yang berpengaruh terhadap solusi permasalahan dan keputusan yang akan diambil. Biasanya dilambangkan dengan  $X_j$  ( $j=1,2,3, \dots, n$ ).
- 2) **Nilai sisi kanan (*right hand sides values*)**, merupakan nilai-nilai yang biasanya menunjukkan ketersediaan sumber daya (dilambangkan dengan  $b_i$ ) yang akan ditentukan kekurangan atau penggunaannya.
- 3) **Koefisien teknologi (*technology coefficient*)**, merupakan nilai-nilai numerik yang dilambangkan dengan  $a_{ij}$  yang akan dikombinasikan dengan variabel keputusan, dimana akan menunjukkan penggunaan terhadap pemenuhan nilai kanan.
- 4) **Variabel deviasional (*penyimpangan*)**, adalah variabel yang menunjukkan kemungkinan penyimpangan-penyimpangan negatif dan positif dari nilai sisi kanan fungsi tujuan. Variabel penyimpangan negatif berfungsi untuk menampung penyimpangan yang berada di bawah sasaran yang dikehendaki, sedangkan variabel penyimpangan positif berfungsi untuk menampung penyimpangan yang berada di atas sasaran. Dalam *Goal Programming* dilambangkan  $d_i^-$  dengan penyimpangan negatif dan  $d_i^+$  untuk penyimpangan positif dari nilai sisi kanan tujuan.
- 5) **Fungsi tujuan**, adalah fungsi matematis dari variabel-variabel keputusan yang menunjukkan hubungan dengan nilai sisi kanannya, fungsi tujuan dalam *Goal Programming* adalah meminimumkan variabel deviasional.
- 6) **Fungsi pencapaian**, adalah fungsi matematis dari variabel-variabel simpangan yang menyatakan kombinasi sebuah objektif.

- 7) **Fungsi tujuan mutlak (non negatif)**, merupakan tujuan yang tidak boleh dilanggar dengan pengertian mempunyai penyimpangan positif dan atau negatif bernilai nol. Prioritas pencapaian dari fungsi tujuan ini berada pada urutan pertama, solusi yang dapat dihasilkan adalah terpenuhi atau tidak terpenuhi.
- 8) **Prioritas**, adalah suatu sistem urutan dari banyaknya tujuan pada model yang memungkinkan tujuan-tujuan tersebut disusun secara ordinal dalam *Goal programming*. Sistem urutan tersebut menempatkan sasaran-sasaran tersebut dalam susunan dengan seri.
- 9) **Pembobotan**, merupakan timbangan matematis yang dinyatakan dengan angka ordinal yang digunakan untuk membedakan variabel simpangan  $i$  dalam suatu tingkat prioritas  $k$ .

Dalam *goal programming* terdapat tiga unsur utama yaitu fungsi tujuan, kendala tujuan, dan kendala non negatif. Penjelasannya sebagai berikut (Harjiyanto, 2014):

1. Fungsi Tujuan

Fungsi tujuan dalam *goal programming* pada umumnya adalah masalah minimisasi, karena dalam fungsi tujuan terdapat variabel simpangan yang harus diminimumkan. Fungsi tujuan dalam *goal programming* adalah meminimumkan total penyimpangan tujuan yang ingin dicapai.

2. Kendala Tujuan

Menurut Harjiyanto (2014), dalam *goal programming* ada enam jenis kendala tujuan yang berlainan. Tujuan dari setiap jenis kendala itu ditentukan oleh hubungannya dengan fungsi tujuan. Enam jenis kendala tersebut dapat dilihat pada tabel 2.1. Dari tabel 2.1 terlihat bahwa setiap kendala tujuan memiliki satu atau dua variabel simpangan yang keduanya atau salah satunya ditempatkan pada fungsi tujuan.

**Tabel 2.1** Tabel Jenis Kendala dalam *Goal Programming*

No.	Kendala Tujuan	Variabel Deviasi dalam Fungsi Tujuan	Kemungkinan Simpangan	Penggunaan Nilai RHS yang diinginkan
1	$C_{ij}x_{ij} + d_i^- = b_i$	$d_i^-$	Negatif	$= b_i$
2	$C_{ij}x_{ij} + d_i^+ = b_i$	$d_i^-$	Positif	$= b_i$
3	$C_{ij}x_{ij} + d_i^- - d_i^+ = b_i$	$d_i^-$	Negatif atau Positif	$b_i$ atau lebih
4	$C_{ij}x_{ij} + d_i^- - d_i^+ = b_i$	$d_i^-$	Negatif atau Positif	$b_i$ atau kurang
5	$C_{ij}x_{ij} + d_i^- - d_i^+ = b_i$	$d_i^-$ dan $d_i^+$	Negatif atau Positif	$= b_i$
6	$C_{ij}x_{ij} - d_i^+ = b_i$	$d_i^+$ ( <i>artifisial</i> )	Tidak Ada	$= b_i$

### 3. Kendala Non Negatif

Kendala non negatif dalam *goal programming* adalah semua variabel-variabel bernilai positif atau sama dengan nol. Jadi variabel keputusan dan variabel deviasi dalam masalah *goal programming* bernilai positif atau sama dengan nol. Pernyataan non negatif dilambangkan  $x_j, d_i^-, d_i^+ \geq 0$

#### 2.3.2 Model Umum *Goal Programming*

Model umum dari *goal programming* tanpa faktor prioritas di dalam strukturnya adalah sebagai berikut (Harjiyanto, 2014).

Minimumkan:  $Z = \sum_{i=1}^m (d_i^+ + d_i^-)$

Dengan kendala tujuan:

$$C_{11}x_1 + C_{12}x_2 + \dots + C_{1n}x_n + d_1^- - d_1^+ = b_1$$

$$\begin{aligned}
C_{21}x_1 + C_{22}x_2 + \dots + C_{2n}x_n + d_2^- - d_2^+ &= b_2 \\
&\vdots \\
C_{m1}x_1 + C_{m2}x_2 + \dots + C_{mn}x_n + d_m^- - d_m^+ &= b_m
\end{aligned}$$

Kendala non negatif:  $x_j, d_i^-, d_i^+ \geq 0$

Keterangan:

$Z$  : nilai fungsi tujuan.

$C_{ij}$  : koefisien teknologi fungsi kendala tujuan, yaitu yang berhubungan dengan tujuan peubah pengambilan keputusan ( $x_j$ ).

$x_j$  : peubah pengambilan keputusan atau kegiatan yang kini dinamakan sebagai sub tujuan.

$b_i$  : tujuan atau target yang ingin dicapai.

$d_i^+$  : jumlah unit deviasi yang kelebihan (+) terhadap tujuan ( $b_i$ )

$d_i^-$  : jumlah unit deviasi yang kelebihan (-) terhadap tujuan ( $b_i$ )

Model untuk persoalan tujuan ganda dengan struktur timbangan prioritas (*pre-emptive weights*) adalah sebagai berikut (Harjiyanto, 2014).

Minimumkan:  $Z = P_1 d_i^- + \dots + P_l d_i^- + P_{l+1} d_i^+ + \dots + P_k d_i^+$

Dengan kendala tujuan:

$$\begin{aligned}
C_{11}x_1 + C_{12}x_2 + \dots + C_{1n}x_n + d_1^- - d_1^+ &= b_1 \\
C_{21}x_1 + C_{22}x_2 + \dots + C_{2n}x_n + d_2^- - d_2^+ &= b_2 \\
&\vdots \\
C_{m1}x_1 + C_{m2}x_2 + \dots + C_{mn}x_n + d_m^- - d_m^+ &= b_m
\end{aligned}$$

Kendala non negatif:  $x_j, d_i^-, d_i^+ \geq 0$

Keterangan:

$Z$  : nilai fungsi tujuan.

$C_{ij}$  : koefisien teknologi fungsi kendala tujuan, yaitu yang berhubungan dengan tujuan peubah pengambilan keputusan ( $x_j$ ).

$x_j$  : peubah pengambilan keputusan atau kegiatan yang kini dinamakan sebagai sub tujuan.

$b_i$  : tujuan atau target yang ingin dicapai.

$d_i^+$  : jumlah unit deviasi yang kelebihan (+) terhadap tujuan ( $b_i$ )

$d_i^-$  : jumlah unit deviasi yang kelebihan (-) terhadap tujuan ( $b_i$ )

$P_k$  : faktor prioritas pada tujuan ke-k

Berdasarkan perumusan model *goal programming*, pencapaian tingkat sasaran atau target dilakukan dengan cara meminimumkan peubah deviasi. Ada dua tipe program sasaran, yaitu program sasaran yang setiap sasarnya memiliki prioritas yang sama dan program sasaran yang mengurutkan sasarnya menurut tingkat prioritas dari sasarnya. Untuk sasaran yang diurutkan berdasarkan tingkat prioritasnya diberikan faktor pembobot. Faktor pembobot adalah suatu nilai numerik yang tidak berdimensi dan digunakan untuk menunjukkan tingkat prioritas relatif dari suatu sasaran. Besar kecilnya nilai faktor pembobot dari setiap sasaran diperoleh dari hasil manipulasi pendapat para ahli atau pengambil keputusan (Harjiyanto, 2014).

Jika faktor pembobot fungsi sasaran prioritas ke- $i$  dilambangkan dengan  $W_i$ , maka secara matematik dapat bersifat:

$$0 < W_i < 1, \text{ dan}$$

$$\sum_{i=1}^k W_i = 1$$

Apabila ada pernyataan  $W_c$  lebih besar dari  $W_y$ , menunjukkan bahwa sasaran ke- $c$  lebih penting dari sasaran ke- $y$  dan jika  $W_c$  sama dengan  $W_y$  maka sasaran ke- $c$  dan sasaran ke- $y$  mempunyai urutan prioritas yang sama.

### 2.3.3 Perumusan Masalah *Goal Programming*

Harjiyanto (2014) menyatakan langkah perumusan permasalahan *Goal Programming* adalah sebagai berikut:

- 1) Penentuan variabel keputusan, merupakan dasar dalam pembuatan model keputusan untuk mendapatkan solusi yang dicari. Semakin tepat penentuan variabel keputusan akan mempermudah pengambilan keputusan yang dicari
- 2) Penentuan fungsi kendala, yaitu kendala-kendala yang membatasi tujuan yang ingin dicapai oleh perusahaan.

- 3) Perumusan fungsi tujuan, dimana setiap sasaran pada sisi kirinya ditambahkan dengan variabel simpangan, baik simpangan positif maupun simpangan negatif. Dengan ditambahkan variabel simpangan, maka bentuk dari fungsi sasaran menjadi:

$$f_i(x_i) + d_i^- - d_i^+ = b_i$$

- 4) Penentuan prioritas utama. Pada langkah ini dibuat urutan dari sasaran-sasaran. Penentuan sasaran ini tergantung pada hal-hal berikut:
- a. keinginan dari pengambil keputusan
  - b. keterbatasan sumber-sumber yang ada
- 5) Penentuan pembobotan. Pada tahap ini merupakan kunci dalam menentukan urutan dalam suatu tujuan dibandingkan dengan tujuan yang lain.
- 6) Penentuan fungsi pencapaian. Dalam hal ini, yang menjadi kuncinya adalah memilih variabel simpangan yang benar untuk dimasukkan dalam fungsi pencapaian. Dalam memformulasikan fungsi pencapaian adalah menggabungkan setiap tujuan yang berbentuk minimisasi variabel penyimpangan sesuai prioritasnya.
- 7) Penyelesaian model *Goal Programming*.

#### 2.3.4 Metode Pemecahan Masalah

Algoritma simpleks dapat digunakan untuk menyelesaikan masalah *Goal Programming* dengan menggunakan variabel keputusan lebih dari dua. Harjiyanto (2014) menyatakan langkah-langkah penyelesaian *Goal Programming* dengan metode algoritma simpleks adalah:

- a. Membentuk tabel simpleks awal.
- b. Pilih kolom kunci dimana  $C_j - Z_j$  memiliki negatif terbesar. Kolom kunci ini disebut kolom pivot.
- c. Pilih baris kunci yang berpedoman pada  $b_i/a_{ij}$  dengan rasio terkecil dimana  $b_i$  adalah nilai sisi kanan dari setiap persamaan. Baris kunci ini disebut baris pivot.
- d. Mencari sistem kanonik yaitu sistem dimana nilai elemen pivot bernilai 1 dan elemen lain bernilai nol dengan cara mengalikan baris pivot

dengan -1 lalu menambahkannya dengan semua elemen di baris pertama. Dengan demikian, diperoleh tabel simpleks iterasi I.

- e. Pemeriksaan optimasi, yaitu melihat apakah solusi layak atau tidak. Solusi dikatakan layak bila variabel adalah positif atau nol.

Berikut adalah tabel awal model *Goal Programming*:

**Tabel 2.2** Tabel Awal *Goal Programming*

	$C_j$	0	0	...	0	$\omega_1 P_1$	$\omega_1 P_1$	...	$\omega_m P_m$	$\omega_m P_m$	$b_i$	$R_i$
$\bar{C}_i$	$\bar{X}_i/X$	$X_1$	$X_2$	...	$X_m$	$d_1^-$	$d_1^+$	...	$d_m^-$	$d_m^+$		
$\omega_1 P_1$	$d_1^-$	$a_{11}$	$a_{12}$	...	$a_{1m}$	1	-1	...	0	0	$b_1$	$R_1$
$\omega_1 P_1$	$d_2^-$	$a_{21}$	$a_{22}$	...	$a_{2m}$	0	0	...	0	0	$b_2$	$R_2$
...	...	...	...	...	...			...			...	...
$\omega_m P_m$	$d_m^-$	$a_{m1}$	$a_{m2}$	...	$a_{mm}$	0	0		1	-1	$b_m$	$R_m$
	$Z_j$	...	...	...	...	...	...	...	...	...	Z	
	$Z_j - C_j$	...	...	...	...	...	...	...	...	...	Z	

Keterangan:

$\bar{X}_i$  : variabel basis

$\bar{C}_i$  : koefisien dari  $\bar{X}_i$

$$Z_j = \sum_{i=1}^m \bar{C}_i a_{ij}$$

$$Z = \sum_{i=1}^m \bar{C}_i b_i, \text{ nilai fungsi tujuan}$$

$R_i$  rasio antara  $b_i$  dan  $a_{ij}$  jika  $X_k$  terpilih menjadi variabel basis.

Setelah model *goal programming* tersebut diselesaikan dengan metode simpleks maka diperoleh nilai dari variabel  $X_1, \dots, X_n$  yang mengoptimalkan fungsi tujuan. Selain itu, juga diperoleh nilai variabel-variabel simpangan yang diartikan sebagai besarnya penyimpangan dari tujuan, tetapi dijamin simpangan yang diperoleh tetap paling minimal.

## 2.4 LINGO

LINGO merupakan program komputer yang digunakan untuk aplikasi *pemrograman linear*. Aplikasi *pemrograman linear* adalah suatu

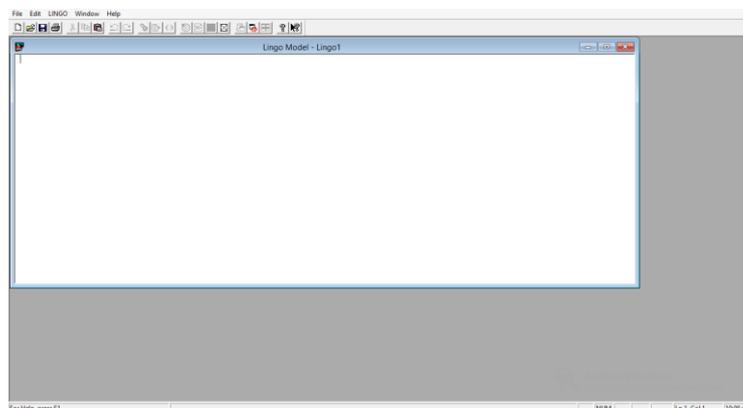
pemodelan matematika yang digunakan untuk mendapatkan suatu solusi optimal dengan kendala yang ada (Harjiyanto, 2014).

LINGO adalah perangkat lunak yang dapat digunakan untuk menyelesaikan masalah-masalah pemrograman linear, non-linear dan integer. LINGO sudah banyak digunakan oleh perusahaan-perusahaan untuk membantu membuat perencanaan produksi yang bertujuan untuk mendapatkan keuntungan yang optimum dan biaya yang minimum. Selain itu, LINGO juga digunakan dalam pengambilan keputusan untuk perencanaan produksi, transportasi, keuangan, alokasi saham, penjadwalan, inventarisasi, pengaturan model, alokasi daya dan lain-lain (Harjiyanto, 2014).

LINGO telah menjadi *software* optimasi selama lebih dari 20 tahun. Sistem LINGO telah menjadi pilihan utama dalam penyelesaian yang cepat dan mudah, terutama dalam masalah optimasi persamaan matematika. Selain itu struktur bahasa yang digunakan dalam memformulasikan masalahnya sederhana, yaitu persamaan linear. Versi terbaru dari LINGO adalah LINGO 16.0. Tampilan awal LINGO 16.0 dapat dilihat pada gambar 2.1 (Harjiyanto, 2014).

Untuk menggunakan *software* LINGO ada beberapa tahapan yang perlu dilakukan, yaitu (Harjiyanto, 2014):

1. Merumuskan masalah dalam kerangka program linear.
2. Menuliskan dalam persamaan matematika.
3. Merumuskan rumusan ke dalam LINGO dan mengeksekusinya.
4. Interpretasi keluaran LINGO.



**Gambar 2.1** Tampilan awal program LINGO 16.0

Cara untuk menginput *script* dilakukan seperti mengetik tulisan biasa, bedanya hanya terdapat pada setiap akhir perintah diakhiri dengan tanda titik koma (;). Contoh *script* untuk menyelesaikan permasalahan contoh kasus 2.1 dengan *goal programming* sebagai berikut:

$$\text{min}=\text{d11}+\text{d12}+\text{d22}+\text{d21}+\text{d31}+\text{d41};$$

$$-\text{d12}+6*\text{x1}+5*\text{x2}+\text{d11}=60;$$

$$-\text{d22}+2*\text{x1}+\text{x2}+\text{d21}=40;$$

$$\text{X1}+\text{d31}=9;$$

$$\text{x2}+\text{d41}=8;$$

$$\text{x1}\geq 0;$$

$$\text{x2}\geq 0;$$

End

Untuk mengeksekusi perintah dilakukan dengan menekan “Solve” pada submenu LINGO, maka hasil output akan dikeluarkan oleh *software* seperti gambar 2.2

Total variables:	8
Nonlinear variables:	0
Integer variables:	0
Total constraints:	7
Nonlinear constraints:	0
Total nonzeros:	20
Nonlinear nonzeros:	0

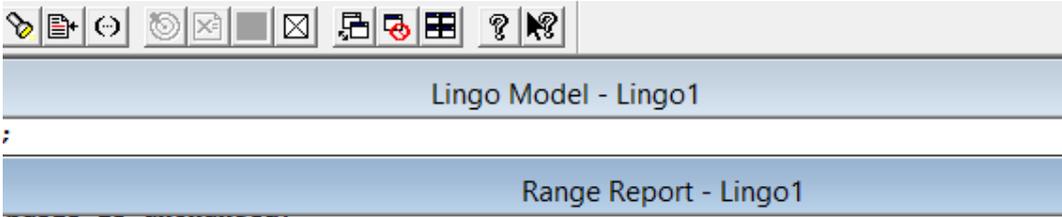
Variable	Value	Reduced Cost
D11	0.000000	1.400000
D12	0.000000	0.600000
D22	0.000000	2.000000
D21	20.800000	0.000000
D31	0.000000	0.600000
D41	6.800000	0.000000
X1	9.000000	0.000000
X2	1.200000	0.000000

Row	Slack or Surplus	Dual Price
1	27.600000	-1.000000
2	0.000000	0.400000
3	0.000000	-1.000000
4	0.000000	-0.400000
5	0.000000	-1.000000
6	9.000000	0.000000
7	1.200000	0.000000

**Gambar 2.2** Output LINGO contoh kasus 2.1

Dari hasil output LINGO, dapat dilihat kecocokan hasil yang diberikan antara perhitungan menggunakan tabel simpleks dengan *software* LINGO. Untuk mencari *range* atau uji analisis sensitivitas perintah dilakukan dengan menekan “*Range*” pada submenu LINGO, maka hasil output akan dikeluarkan oleh *software* seperti pada gambar 2.3.



Objective Coefficient Ranges:

Variable	Current Coefficient	Allowable Increase	Allowable Decrease
D11	1.000000	INFINITY	1.400000
D12	1.000000	INFINITY	0.600000
D22	1.000000	INFINITY	2.000000
D21	1.000000	3.000000	0.750000
D31	1.000000	INFINITY	0.600000
D41	1.000000	0.500000	7.000000
X1	0.000000	0.600000	INFINITY
X2	0.000000	7.000000	0.500000

Righthand Side Ranges:

Row	Current RHS	Allowable Increase	Allowable Decrease
2	60.00000	34.00000	6.00000
3	40.00000	INFINITY	20.80000
4	9.00000	1.00000	5.666667
5	8.00000	INFINITY	6.80000
6	0.00000	9.00000	INFINITY
7	0.00000	1.20000	INFINITY

**Gambar 2.3** Output range contoh 2.1

## 2.5 Penelitian Pendahulu

Penelitian pendahulu dilakukan oleh Kartika Megasari dalam SKRIPSI yang berjudul *Goal Programming Untuk Perencanaan Produksi Agregat Dengan Kendala Sumber Daya* pada tahun 2010. Dalam tugas akhir tersebut, diterapkan pendekatan *goal programming* agar dapat menghasilkan keluaran berupa pilihan alternatif-alternatif dari skenario

perencanaan produksi. Selain itu *goal programming* digunakan untuk mempertimbangkan kelancaran *supply* pasar dan pertimbangan keuangan yaitu dengan memaksimalkan laba, memaksimalkan pemanfaatan mesin produksi dan meminimumkan biaya produksi. Data permintaan diperoleh dari metode peramalan ARIMA. Permintaan kecap manis bulan 1 dan 2 sebesar 200.574,7 dan 200.952,9 dapat tercapai dengan model yang ada saat ini yaitu 200.574,7 dan 2009.952,9. Permintaan sambal kecap bulan 1 dan 2 (29.936,2 dan 30.255) tercapai dengan model saat ini (29.936,2 dan 30.255,78). Profit perusahaan sebesar Rp 1.230.000.000 tercapai dengan model yang ada yaitu Rp 1.300.890.000. Biaya produksi Rp 3.948.000.000 tercapai dengan model saat ini Rp 3.798.213.360. Dan penggunaan mesin untuk memproduksi kecap manis sebesar 6.900 dapat tercapai dengan model sekarang yaitu 8.030.

Selanjutnya pada tahun 2013, dalam Jurnal yang disusun oleh Pupy Ajiningtiyas, dkk dengan judul “***Penerapan Metode Goal Programming untuk Perencanaan Produksi pada Produk Olahan Tebu (Studi Kasus: PG. XXX, Jawa Timur)***”. Dalam penelitian ini dibahas mengenai penerapan metode *goal programming* untuk perencanaan produksi untuk meminimumkan kendala yang dihadapi perusahaan agar produksi dan penjualan produk optimal serta penggunaan metode *time series* model *Double Exponential Smoothing: metode linear* satu parameter dari Brown. Berdasarkan analisa dan pengolahan data yang telah diperoleh, didapatkan hasil bahwa tidak semua target perusahaan terpenuhi untuk target penjualan produk, keuntungan penjualan terpenuhi, minimal biaya produksi terpenuhi, minimal pemakaian bahan baku dan maksimal jam kerja mesin terpenuhi.

Kemudian, Eka A. Surbakti, dkk (2013) melakukan penelitian dengan menggunakan 2 prioritas, 2 produk dan 2 periode pada JURNAL ILMIAH dengan judul ***Penerapan Metode Goal Programming untuk Optimalisasi Perencanaan Produksi (Studi Kasus Susu Nandhi Murni pada KUD “BATU”)***. Pada jurnal ini dibahas mengenai *goal programming* dalam menyelesaikan permasalahan perencanaan produksi dalam hal transportasi dan pemenuhan jumlah pemesanan susu Nandhi murni pada

KUD “Batu”. Peramalan jumlah pesanan menggunakan metode ARIMA. Jumlah peramalan pesanan yang diperoleh pada bulan Juli 2013 untuk produk I dan II sebesar 5.849,6624 (5.850) dan 4.472,4519 (4.472). sementara itu, hasil peramalan pesanan yang diperoleh pada bulan Agustus 2013 untuk produk I dan II adalah 5.648,5382 (5.649) dan 4.472,5015 (4.473). berdasarkan hasil yang diperoleh dengan menggunakan LINDO, jumlah produk yang dipesan dapat dipenuhi dan biaya transportasi yang perlu dikeluarkan pada dua periode lebih minimum daripada biaya yang ditetapkan. Biaya yang perlu dikeluarkan yaitu sebesar Rp 1.156.080 pada periode pertama dan Rp 1.141.760 pada periode kedua.