

# RANCANG BANGUN OPTIMASI PERENCANAAN KEBUTUHAN BAHAN BAKU DENGAN ALGORITMA *SILVER-MEAL*

**Aulia Bahar, Sarwosri**

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember  
Kampus ITS Sukolilo, Surabaya  
Email: hell14hound@yahoo.com, sri@its-sby.edu

## ABSTRAK

Suatu usaha bisnis biasanya mengatur dengan sangat hati-hati sekali mengenai barang-barang inventori agar usahanya dapat berjalan lancar. Sebuah faktor penting dalam formulasi dan solusi dari suatu model inventori adalah permintaan (per satuan waktu) dari suatu item yang bersifat deterministik (bersifat tentu) atau bersifat probabilistik (ditentukan oleh distribusi kemungkinan). Material Requirement Planning (MRP) adalah situasi yang terjadi karena permintaan deterministik muncul. Kepopuleran sistem MRP dalam industri mengakibatkan meningkatnya minat terhadap topik dalam pengambilan keputusan dalam sistem manufaktur multistage. Akan tetapi, prosedur Lot-Sizing yang digunakan oleh sistem MRP yang sudah tersedia agak terbatas dalam kemampuannya untuk mengkoordinasi rencana produksi dari berbagai tahap dari proses manufaktur dan berbagai macam batas kapasitas. Masalah lot-sizing adalah menentukan ukuran lot produksi dari berbagai item dalam tahap produksi. Tujuan dari permasalahan ini adalah mengurangi biaya total manufakturing (termasuk biaya setup, biaya inventori holding, dan lain-lain) dengan tetap memperhatikan kebutuhan konsumen.

**Kata kunci:** MRP, Lot Sizing, Algoritma Genetika.

## 1. PENDAHULUAN

Kepopuleran sistem Material Requirement Planning (MRP) dalam industri mengakibatkan meningkatnya minat terhadap topik dalam pengambilan keputusan dalam sistem manufaktur multistage. Akan tetapi, prosedur Lot-Sizing yang digunakan oleh sistem MRP yang sudah tersedia agak terbatas dalam kemampuannya untuk mengkoordinasi rencana produksi dari berbagai tahap dari proses manufaktur dan berbagai macam batas kapasitas.

Masalah lot-sizing adalah menentukan ukuran lot produksi dari berbagai item dalam tahap produksi. Tujuan dari permasalahan ini adalah mengurangi biaya total manufakturing (termasuk biaya setup, biaya inventori holding, dan lain-lain) dengan tetap memperhatikan kebutuhan konsumen.

Langkah-langkah dalam MRP terdiri dari (1) *Netting*, yaitu menentukan kebutuhan bersih perusahaan akan bahan baku; (2) *Lotting*, yaitu menentukan besarnya pesanan per item yang optimal berdasarkan kebutuhan bersih; (3) *Offsetting*, yaitu pembuatan jadwal yang tepat untuk melakukan pemesanan sesuai dengan kebutuhan bersih; (4) *Exploding*, yaitu perhitungan kebutuhan kotor akan bahan baku. Salah satu teknik lot adalah algoritma *Silver-Meal*. Algoritma ini bisa dikatakan heuristik yang "hampir optimal" dimana membuat order

kuantitas dengan mengambil pendekatan analisis marginal. Jika dilihat pada pendekatan *Economic Order Quantity* (EOQ) yang sederhana, ukuran lot yang tetap akan membuat "leftover" dimana akan meningkatkan biaya total.

Sebuah pendekatan yang lebih baik adalah pemesanan stok "senilai semua periode". Meningkatkan kuantitas pembelian akan mengurangi jumlah biaya yang akan dikeluarkan karena biaya pemesanan dikurangi karena banyaknya pembelian akan berkurang. Walaupun pemesanan kuantitas yang besar akan mulai meningkatkan biaya total dan biaya holding akan meningkat. Pada penelitian sebelumnya penentuan kuantitas pemesanan dilakukan dengan algoritma *Wagner-Whitin* dan algoritma genetika.

Algoritma *Wagner-Whitin* cocok digunakan karena memberikan solusi optimal bagi persoalan ukuran pemesanan dinamis-deterministik pada periode waktu tertentu dimana kebutuhan seluruh periode harus terpenuhi. Algoritma *Wagner-Whitin* selalu menghasilkan gabungan minggu. Hasil penggabungan pada minggu pertama akan menghasilkan biaya minimum untuk seluruh minggu produksi.

Algoritma Genetika merupakan sebuah *heuristic search* yang menggunakan sekumpulan individu (populasi) untuk mencapai suatu solusi yang optimal. Algoritma dimulai dengan pembangkitan populasi awal secara acak, kemudian *fitness value* masing-masing

kromosom dievaluasi untuk mencari kromosom mana yang akan direproduksi dengan 2 operator yaitu *crossover* dan mutasi. Hasil dari algoritma Genetika ini menunjukkan bahwa algoritma ini mampu mereduksi total biaya produksi dengan menggabungkan beberapa periode pemesanan sehingga biaya pemesanan dapat ditekan.

Pada penelitian ini akan digunakan algoritma *Silver-Meal*. Algoritma *Silver-Meal* menukar biaya pemesanan dan biaya *holding* dengan menganalisa masalah "satu pembelian dalam satu waktu". Satu pembelian dalam satu waktu adalah pembelian pertama harus dapat memenuhi periode 1, periode 1 dan 2, periode 1, 2, dan 3, dan seterusnya. Untuk menjawab pertanyaan ini, prosedur mengenai potensi tiap kuantitas pembelian dan penghitungan biaya rata-rata per periode yang terpenuhi adalah hasil dari penjumlahan dari pemesanan dan biaya penyimpanan dinyatakan oleh potensi pembelian dibagi dengan jumlah periode yang harus terpenuhi.

## 2. MATERIAL REQUIREMENT PLANNING

*Material Requirement Planning* adalah suatu teknik atau prosedur untuk mengelola persediaan dalam suatu operasi manufaktur. Metode ini dikembangkan secara spesifik dengan tujuan berhadapan dengan kompleksitas penempatan waktu dan hubungannya dengan inventori. Pengendalian persediaan ini memperhatikan hubungan antar item persediaan, sehingga dapat meningkatkan efisiensi dalam menentukan hubungan setiap item secara cepat dan tepat.

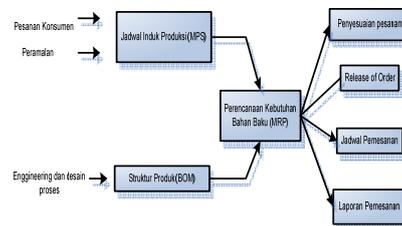
Hasil perencanaan akan memberikan *master production schedule* (MPS). Dengan menggunakan MPS akan diketahui jadwal kapan selesainya produk akhir. Jadwal ini akan sangat diperlukan untuk perencanaan produksi untuk mengetahui berapa jumlah produk akhir yang akan diproduksi

Konsep dalam perhitungan MRP (lihat Gambar 1) meliputi:

1. *Lot Size*: kuantitas dari item yang biasanya dipesan dari pabrik atau pemasok. Sering disebut juga sebagai kuantitas pesanan (*order quantity*) atau ukuran batch (*batch size*). Ada banyak algoritma yang dapat digunakan untuk menentukan ukuran lot, diantaranya adalah Algoritma *Silver Meal*.
2. *Gross Requirement* (Kebutuhan kotor): Total permintaan yang diharapkan dari suatu item atau bahan baku untuk masing-masing periode waktu.
3. *On Hand Inventory* (Persediaan awal): Menyatakan jumlah persediaan yang tersedia pada suatu periode waktu tertentu. Nilai

persediaan pada awal periode dimasukkan sesuai dengan jumlah persediaan saat itu.

4. *Net Requirement/NR<sub>t</sub>* (Kebutuhan Bersih): Jumlah kebutuhan yang sebenarnya yang dibutuhkan pada masing-masing periode waktu untuk memenuhi kebutuhan item pada *gross requirement*.
5. *Planned Order Receipts/POR* (Rencana Penerimaan): jumlah dari pemesanan yang direncanakan (belum tiba) dalam suatu periode. Rencana penerimaan pada periode *t* ini akan ada dengan sendirinya jika terdapat kebutuhan bersih (*NR<sub>t</sub>*) suatu item pada periode tertentu, dimana jumlah *POR* ini bergantung pada ukuran lot yang dipergunakan.
6. *Planned Order Release* (Rencana Pemesanan). Rencana pemesanan pada suatu level atau tingkat menentukan kebutuhan kotor pada level di bawahnya. Rencana pemesanan merupakan informasi terpenting dari sistem *Material Requirement Planning* yang menunjukkan item apa, berapa banyak, dan kapan dibutuhkan. Nilai dari rencana pemesanan tergantung dari teknik *lot sizing* yang digunakan dan nilainya sama dengan nilai *Planned Order Receipts* pada periode  $t + \text{lead time}$ .

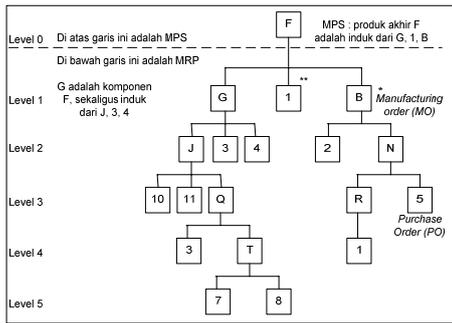


Gambar 1. Sistem MRP

### 2.1 Bill Of Materials

*Bill of Materials* (BOM) adalah suatu susunan bahan, material yang dibutuhkan untuk memproduksi produk akhir. Menyusun struktur BOM adalah proses yang mengatur rincian material dan menghasilkan subrakitan yang menuju perakitan dan kemudian menjadi bagian dari produk akhir. Setiap kaitan produk induk dengan komponen menghasilkan kaitan antara produk akhir dengan bagian-bagiannya atau antara dua atau lebih komponen.

Contoh BOM dapat dilihat pada Gambar 2. Tabel 1 menunjukkan data BOM yang digunakan pada penelitian ini.



Gambar 2. Bill Of Material

Tabel 1. Data BOM yang digunakan

ID	Nama	RC	SC	OC	HC	LT
23	Top	1	1000	25	3	2
30	Board	3	1500	20	5	1
66	Glue_A	1	800	5	2	1
67	Glue_B	1	1050	10	2	1
100	Table	1	2000	150	2	1
200	Base	1	1600	30	4	2
203	Legs	4	1100	50	3	1
220	Legs_Bolt	4	970	25	1	1
300	Frame	1	1250	95	3	2
411	Ends	2	800	65	4	1
533	Leg_Support	4	1300	55	2	1
622	Slides	2	1400	85	2	1

## 2.2 Auto Regressive (2)

Dalam melakukan suatu peramalan dipengaruhi beberapa aspek, antara lain: horison waktu peramalan, faktor yang menentukan hasil sebenarnya, tipe pola data dan berbagai aspek lainnya. Untuk menghadapi penggunaan yang luas tersebut, beberapa teknik telah dikembangkan. Teknik tersebut dibagi ke dalam 2 kategori utama yaitu metode kuantitatif dan metode kualitatif. Metode kuantitatif dapat dibagi ke dalam deret berkala (*time series*) dan metode kausal. Kedua model deret berkala (*time series*) dan kausal mempunyai keuntungan dalam situasi tertentu. Model deret berkala sering kali dapat digunakan dengan mudah untuk meramal, sedangkan model kausal dapat digunakan dengan keberhasilan yang lebih besar untuk pengambilan keputusan dan kebijaksanaan.

Aplikasi ini menggunakan metode *Auto Regressive (2)*. *Auto Regressive (AR)* merupakan salah satu metode *forecast* (peramalan) yang bersifat *time series* (berurutan) dalam sebuah serial waktu. Metode tersebut adalah alat kunci untuk mengidentifikasi pola dasar dan menetapkan model yang sesuai untuk sebuah serial waktu. Rumusnya sebagai berikut:

$$F_t = \sigma + (\phi_1 * F_{t-1}) + (\phi_2 * F_{t-2}) + e \quad (1)$$

$F_t$  adalah jumlah suatu unit pada waktu  $t$ .  $\sigma$  adalah nilai rata-rata (*mean*).  $\phi_1$  adalah nilai untuk *tetha1*.  $\phi_2$  adalah nilai untuk *tetha2*.  $F_{t-1}$  &  $F_{t-2}$  adalah nilai untuk  $t-1$  dan  $t-2$ .

## 2.3 PERAMALAN EXPONENTIAL SMOOTHING: HOLT'S METHOD

*Exponential Smoothing* adalah salah satu cara untuk melakukan estimasi nilai berikutnya dari sebuah *time series*. Teknik *Forecasting* ini dalam bentuk dasarnya sebaiknya hanya digunakan pada *non-seasonal time series* yang menunjukkan tidak adanya pola yang sistematis.

Rumus *exponential smoothing* dapat diperluas menjadi *linear exponential smoothing* sehingga bisa melakukan *forecasting* data dengan *trends*. *Holt's Linear exponential smoothing* menggunakan dua *smoothing constant*,  $\alpha$  dan  $\beta$  (dengan nilai antara 0 dan 1), dan ada 3 persamaan sebagai berikut:

$$L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + b_{t-1}) \quad (2)$$

$$b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \quad (3)$$

$$F_{t+m} = L_t + b_{tm} \quad (4)$$

$L_t$  adalah estimasi kenaikan dari *series* pada waktu  $t$ ;  $b_t$  adalah estimasi penurunan dari *series* pada waktu  $t$ . Persamaan (2) mengatur  $L_t$  untuk menjadi *trend* dari periode sebelumnya,  $b_t - 1$ , dengan menambahkannya ke nilai terakhir yang dihaluskan,  $L_{t-1}$ . Persamaan (3) dan (4) adalah update dari *trend*, dimana diekspresikan sebagai selisih antara dua nilai terakhir yang dihaluskan (*smoothed*). Rumus ini digunakan karena ada *trend* dalam data, nilai baru seharusnya lebih tinggi atau lebih rendah dari data sebelumnya. Contoh perhitungan, dengan  $\alpha = 0,501$  dan  $\beta = 0,072$ :

$$F_{23} = L_{22} + b_{22}$$

$$\begin{aligned} L_{22} &= 0,501 Y_{22} + 0,499(L_{21} + b_{21}) \\ &= 0,501(242) + 0,499(227,33 + 5,31) \\ &= 237,33 \end{aligned}$$

$$\begin{aligned} b_{22} &= 0,072(L_{22} - L_{21}) + 0,928 b_{21} \\ &= 0,072(237,33 - 227,33) + \\ &= 0,928(5,31) \\ &= 5,64 \end{aligned}$$

$$\text{Maka, } F_{23} = 237,33 + 5,64 = 242,97.$$

Nilai peramalan untuk periode ke 23 adalah 242,97.

### 3. ALGORITMA SILVER-MEAL

Algoritma *Silver-meal* adalah heuristik yang "hampir optimal" dimana membuat order kuantitas dengan mengambil pendekatan analisis marginal. Optimal adalah nilai *total cost* yang rendah. Model *Silver-Meal* hanya bisa digunakan pada inventori yang mempunyai biaya produksi tetap dan identik untuk tiap periode.

Jika dilihat pada pendekatan EOQ yang sederhana, ukuran lot yang tetap akan membuat "leftover" dimana akan meningkatkan biaya total. Sebuah pendekatan yang lebih baik adalah pemesanan stok "senilai semua periode". Meningkatkan kuantitas pembelian akan mengurangi jumlah biaya yang akan dikeluarkan karena biaya pemesanan dikurangi karena banyaknya pembelian akan berkurang.

Walaupun pemesanan kuantitas yang besar akan mulai meningkatkan biaya total dan biaya penyimpanan akan meningkat. Algoritma *Silver-Meal* menukar biaya pemesanan dan biaya *holding* dengan menganalisa masalah "satu pembelian dalam satu waktu".

Maksudnya adalah, pembelian pertama harus dapat memenuhi periode 1, periode 1 dan 2, periode 1, 2, dan 3, dan seterusnya. Untuk menjawab pertanyaan ini, prosedur mengenai potensi tiap kuantitas pembelian dan penghitungan biaya rata-rata per periode yang terpenuhi adalah hasil dari penjumlahan dari pemesanan dan biaya. Rumus dari algoritma *Silver-Meal* adalah:

$$TCU_{(i,t)} = \frac{TC(i,t)}{t-i+1} \quad (5)$$

dimana *i* adalah periode dimulainya iterasi (contoh: iterasi 1 dimulai pada periode 1 maka *i*=1, iterasi 2 dimulai pada periode 4 maka *i*=4), *t* adalah banyak periode dalam satu iterasi.

Tujuan dari model ini adalah untuk meminimalisir biaya *setup* dan biaya *holding*. Tabel 2 menunjukkan contoh perhitungan *Silver-Meal*.

**Tabel 2. Contoh perhitungan Silver-Meal**

Gabungan periode	Trail Lot Sizing	Kumulatif Cost	Rata-rata total cost
2	30	200 + (30 * 4) = 320	200/1 = 200
2,3	30+40=70	(200 + (40 * 4)) + 40*2=440	280/2 = 140
2,3,4	30+40+0=70	200 + (0 * 4) + 40*2=280	280/3 = 93,33
2,3,4,5	30+40+0+10=80	(200 + (10 * 4)) + (50+10+10)*2 = 380	340/4 = 85
2,3,4,5,6	30+40+0+10+40=120	(200 + (40 * 4)) + (50+50+50+40)*2 = 660	660/5 = 132
Periode 2,3,4,5, dikombinasikan didapatkan total biaya inventory minimum			
6	40	200 + (40 * 4) =	200/1 = 200
6,7	70	200+30*2 =260	260/2 = 130
6,7,8	70	200 + 30*2 = 260	260/3 = 86,67

### 4. HASIL UJI COBA

Agar dapat mengetahui hasil dari sistem optimasi kebutuhan bahan baku dengan menggunakan algoritma *Silver-Meal*, maka diperlukan data-data untuk melakukan uji. Data

uji tersebut akan menggunakan masukan dari *user*. Setiap data uji coba untuk sistem ini mempunyai beberapa komponen seperti *mean*, *Y1*, *Y2*, *tetha1*, *tetha2*, *beta*, *periode*, *mean error* dan *standard deviation*. Tabel 3 menunjukkan data uji coba sistem yang digunakan.

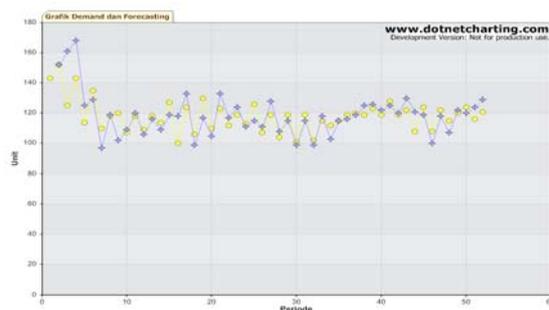
**Tabel 3. Data uji coba sistem**

Data	1	2	3	4	5	6
Mean	60	70	50	5	200	150
Y1	143	143	143	140	140	140
Y2	152	152	152	80	80	80
Tetha1	-0.2	-0.2	-0.2	-0.4	-0.4	-0.4
Tetha2	0.6	0.6	0.6	0.7	0.7	0.7
Periode	52	52	52	52	52	52
Mean Error	10	10	10	30	25	13
Standar_dev	5	5	5	20	13	6

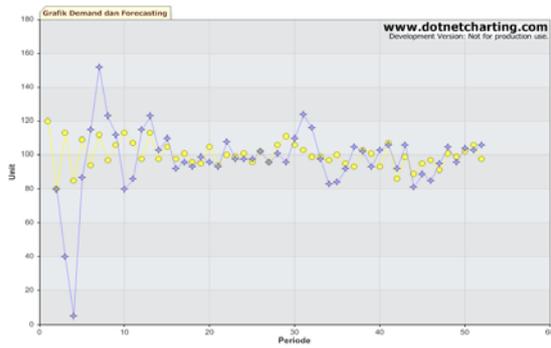
Berdasarkan pada Tabel 3 dapat diketahui bahwa untuk melakukan suatu pembangkitan data, diperlukan beberapa masukan standar seperti *mean* yang nilainya sebaiknya berada diantara nilai 10% sampai 50 % dari Y1. Nilai Y2 harus lebih besar dari Y1. Nilai kesalahan rata-rata (*mean error*) harus mempunyai nilai maksimal 10% dari Y1. Untuk *standard deviation* harus sebanyak setengah atau kurang dari *mean error*. Jumlah periode yang harus dimasukkan adalah 52. Proses pembangkitan data uji coba dijalankan dengan jumlah pembangkitan sebanyak 52 data pembangkitan sesuai dengan masukan periode.

Untuk mendapatkan *actual data*, data-data tersebut akan dihitung dengan menggunakan rumus (1), *actual data* kemudian dijadikan masukan untuk proses selanjutnya yaitu peramalan. Proses peramalan akan menggunakan rumus (2,3,4). Grafik hasil uji coba dapat dilihat pada Gambar 3-8.

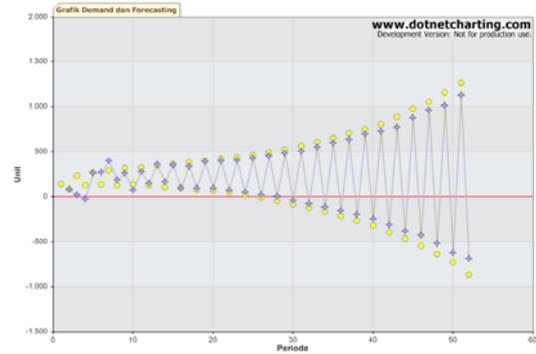
Hasil peramalan data kemudian disimpan menjadi *Master Production of Schedule* (MPS). MPS adalah salah satu masukan untuk *proses lot-sizing* (*Silver-Meal*) yang menggunakan rumus (5). Dari perhitungan *lot-sizing* diperoleh biaya total untuk semua produk. Tabel 4-6 menunjukkan biaya total dari berbagai uji coba.



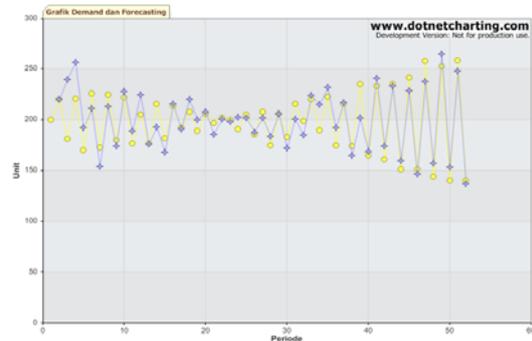
**Gambar 3. Hasil uji coba 1**



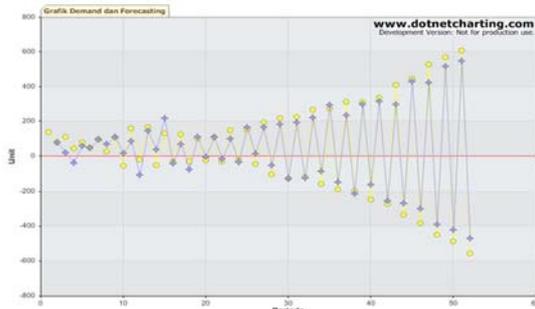
Gambar 4. Hasil uji coba 2



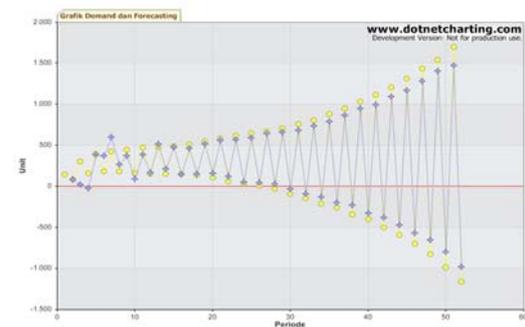
Gambar 8. Hasil uji coba 6



Gambar 5. Hasil uji coba 3



Gambar 6. Hasil uji coba 4



Gambar 7. Hasil uji coba 5

Tabel 4. Biaya total uji 1 & 2

Uji	Id	TC	Uji	Id	TC
1	100	44680	2	100	40696
1	23	35618	2	23	31949
1	30	85695	2	30	73845
1	66	26250	2	66	24784
1	67	30870	2	67	28562
1	200	52024	2	200	46968
1	203	64868	2	203	62056
1	300	40199	2	220	41748
1	411	45704	2	300	36134
1	533	66472	2	411	42904
1	622	49332	2	533	60976
1	220	44250	2	622	44868

Tabel 5. Biaya total uji 3 & 4

Uji	Id	TC	Uji	Id	TC
3	100	54472	4	30	37065
3	23	40323	4	23	23069
3	30	12448	4	66	18446
3	66	31540	4	67	22118
3	67	38820	4	100	32566
3	200	60164	4	200	34836
3	203	85808	4	203	37000
3	220	54160	4	220	24134
3	300	46849	4	300	27177
3	411	66336	4	411	19768
3	533	74728	4	533	32268
3	622	56332	4	622	33932

## 5. KESIMPULAN

Dari hasil uji coba dapat dihasilkan beberapa simpulan sebagai berikut:

- 1) Apabila data yang dimasukkan tidak sesuai dengan masukan standar, maka besar kemungkinannya akan menghasilkan nilai negatif seperti pada gambar 4-6.

**Tabel 6. Biaya total uji 5 & 6**

Uji	Id	TC	Uji	Id	TC
5	30	96015	6	30	54150
5	23	23275	6	23	22808
5	66	18050	6	66	16660
5	67	22118	6	67	19410
5	100	35096	6	100	27228
5	200	36100	6	200	33344
5	203	35756	6	203	41120
5	220	27836	6	220	27076
5	300	28025	6	300	25540
5	411	24408	6	411	24408
5	533	35548	6	533	43064
5	622	32300	6	622	31144

- 2) Peramalan data dengan menggunakan *Exponential Smoothing: Holt's Method* menghasilkan *mean absolute percentage error* (MAPE) 7,7%, 5% dan 7,4%. Semakin kecil nilai MAPE, nilai *forecasting data* dengan nilai *actual data* akan semakin mirip.
- 3) Semakin besar nilai *Fixed Setup Cost* maka semakin banyak periode yang dapat digabungkan sehingga nilai *Total Cost* dapat menjadi lebih rendah.
- 4) Dari uji coba 1 sampai uji coba 6 diperoleh 95% *Confidence Interval* untuk *total cost*

dengan menggunakan *t-distribution*, yaitu:  
 $338307 < TC < 654469$ .

## 6. DAFTAR PUSTAKA

- [1] Richard J. Tersine. "Principles of Inventory and Materials Management".
- [2] Prof. Duane S. Boning. Manufacturing System. Lecture 11: Forecasting.
- [3] Logistic System Fall. 2003. Finite Planning Horizon Inventory Problems.
- [4] Jaenuddin ST. 2006. Belajar Sendiri .net dengan Visual C# 2005
- [5] Mohd Omar, Mustafa Mat Deris. 2001. The Silver Meal Heuristic Method for Deterministic Time Varying Demand, Departement Computer Science & IT University of College Terengganu, Institut of Mathematical Science University of Malaya, Malaysia.
- [6] Prajakta S. Kalekar. December 6 2004. Time Series Forecasting using Holt's Winter Exponential Smoothing. Kanwal Rekhi School of Information Technology.
- [7] Box, G. E. P. , Jenkins, G. M. dan Reinsel, G. C. 1970, Time Series Analysis Forecasting and Control, Prentice Hall/Pearson
- [8] Hamdy, A. Taha. 2003. Operations Research. United States of America: Prentice-Hall International.