

# Perancangan Model VMI (*Vendor Managed Inventory*) dengan Satu Pemasok dan Banyak *Retailer* yang Meminimasi Ongkos Total Rantai Pasok

Yosefa<sup>1\*</sup>, Carles Sitompul<sup>2</sup>, Alfian<sup>3</sup>

<sup>1,2,3</sup>) Fakultas Teknologi Industri, Jurusan Teknik Industri,

Universitas Katolik Parahyangan

Jl. Ciumbuleuit 94, Bandung 40141

email : carles@unpar.ac.id , alfian.tan@gmail.com

---

## Abstrak

Persaingan bisnis yang terjadi antar perusahaan yang ada di negara maju semakin ketat seiring dengan pertumbuhan ekonomi yang terjadi di negara tersebut. Untuk memenangkan kompetisi, perusahaan-perusahaan berlomba memperbaiki kinerja mereka. Salah satu cara yang dilakukan adalah menerapkan sistem persediaan barang yang lebih efisien yaitu dengan menerapkan manajemen *supply chain* yang akan memberikan keuntungan bagi pihak *supplier* maupun pihak *retailer*. VMI (*Vendor Managed Inventory*) merupakan salah satu metode dalam *supply chain* yang memberikan keuntungan bagi perusahaan melalui waktu *replenishment* yang singkat, perbaikan *customer service level*, total biaya rantai pasok yang lebih kecil dibanding metode persediaan tradisional. Penelitian-penelitian yang sudah dilakukan saat ini baru sebatas model VMI satu *supplier* dengan satu *retailer*. Padahal pada kenyataannya satu *supplier* dapat berhubungan dengan satu atau lebih *retailer* atau bahkan sebaliknya banyak *supplier* yang berhubungan dengan 1 *retailer*. Kondisi seperti ini menyebabkan perlunya pengembangan model yang melibatkan banyak *retailer* maupun *supplier*.

Dalam penelitian ini, dilakukan pengembangan model VMI yang dibatasi pada kondisi satu *supplier* dengan banyak *retailer*. Dilakukan penentuan parameter, variabel keputusan, fungsi tujuan dan koefisien pembatas untuk merancang model. Model yang dirancang kemudian diimplementasikan ke dalam bahasa pemrograman AMPL dan solusinya didapatkan dengan penggunaan *software* NEOS. Data yang digunakan merupakan data ilustrasi. Setelah didapat hasil dari pengimplementasian model, selanjutnya dibuat analisis berdasarkan hasil serta analisis sensitivitas untuk model. Hasil penggunaan model VMI (*Vendor Managed Inventory*) yang dirancang kemudian dibandingkan dengan hasil penggunaan metode lain, yaitu *Q System* serta *Wagner Within Algorithm*.

*Kata Kunci:* *Vendor Managed Inventory*, satu pemasok banyak *retailer*

---

## 1 Pendahuluan

Seiring pertumbuhan ekonomi yang terjadi di negaranegara maju, persaingan bisnis yang terjadi antar perusahaan pun semakin ketat. Berbagai cara dilakukan untuk memperbaiki kinerja perusahaan guna memenangkan kompetisi antar perusahaan. Salah satu perbaikan yang dilakukan adalah mencari cara agar dapat menerapkan sistem persediaan barang yang lebih efisien. Persediaan menjadi hal penting bagi perusahaan karena perusahaan tidak ingin

mengalami kerugian jika konsumen tidak jadi membeli barang karena barang tidak tersedia namun di sisi lain apabila perusahaan menyediakan barang dalam jumlah banyak, biaya persediaan yang dikeluarkan perusahaan akan semakin besar. Oleh karena itu, sistem persediaan menjadi salah satu hal penting yang harus diperhatikan oleh perusahaan.

Manajemen *supply chain* merupakan salah satu strategi yang dapat digunakan untuk mengatasi masalah pasokan barang. Menurut SimchiLevi dkk. (2000), strategi ini merupakan pendekatan yang digunakan untuk memastikan

---

\*Korespondensi Penulis

barang yang diproduksi dan didistribusikan berada dalam jumlah yang tepat, ke lokasi yang tepat dan waktu yang tepat untuk meminimasi biaya persediaan. Menurut Richardus Eko Indrajit dkk. (2002), *supply chain* sendiri merupakan sistem atau jaringan organisasi yang terhubung dengan berbagai macam proses dan kegiatan yang memberikan nilai pada produk dan jasa yang akan disalurkan dari produsen ke konsumen. Dalam manajemen *supply chain* terdapat hubungan antara pembeli dan penjual, dimana hubungan itu akan menguntungkan kedua belah pihak (*winwin relationship*). Hubungan tersebut biasanya disebut sebagai RSP (*Retailer Supplier Partnership*).

RSP merupakan strategi yang memerlukan kerjasama dan koordinasi yang baik antara pihak penjual dan pihak pembeli agar kedua belah pihak samasama mendapatkan keuntungan. Saat ini, ada tiga tipe RSP yang telah diterapkan dalam dunia industri, yaitu: QR (*Quick Response*), CR (*Continuous Replenishment* atau *Rapid Replenishment*) dan VMI (*Vendor Managed Inventory*). Penelitian ini hanya akan membahas mengenai VMI.

Menurut SimchiLevi dkk. (2000), VMI merupakan strategi yang memerlukan waktu *replenishment* singkat dengan frekuensi dan pengiriman yang tepat waktu sehingga dapat mengurangi biaya persediaan. Selain itu, VMI juga dapat memperbaiki *customer service level* yang akan meningkatkan kelayakan *retailer* terhadap *supplier*. Peningkatan *customer service level* terjadi karena *supplier* dapat memenuhi permintaan *retailer* tepat pada waktunya. Keunggulan-keunggulan tersebut membuat VMI banyak diterapkan pada banyak perusahaan.

Dengan demikian, masalah yang dihadapi dalam penelitian ini dapat dirumuskan sebagai berikut: Bagaimana model VMI untuk satu pemasok dengan banyak *retailer* yang dapat meminimasi ongkos total.

## 2 Tinjauan Pustaka

Menurut SimchiLevi dkk. (2000), *supply chain management* adalah serangkaian pendekatan yang diterapkan untuk mengintegrasikan *supplier*, pengusaha, gudang (*warehouse*), dan tempat penyimpanan lainnya secara efisien sehingga produk dihasilkan dan didistribusikan dengan kuantitas yang tepat, lokasi tepat dan waktu yang tepat untuk memperkecil biaya dan memuaskan kebutuhan pelanggan.

Menurut SimchiLevi dkk. (2000), *strategic alliance* biasanya beragam, berorientasi pada tu-

juan, kemitraan jangka panjang antara dua perusahaan dimana resiko dan keuntungan ditanggung bersama. Ada tiga macam *strategic alliance* menurut SimchiLevi dkk. (2000), yaitu :

1. 3PL (*Third Party Logistics*)
2. RSP (*Retailer Supplier Partnership*)
3. DI (*Distributor Integration*)

Menurut SimchiLevi dkk. (2000), RSP (*Retailer Supplier Partnership*) adalah suatu strategi kolaborasi antara *supplier* dan *buyer* dimana dalam hubungan ini diperlukan kerjasama dan koordinasi yang baik antara kedua belah pihak untuk mendapatkan keuntungan yang dapat dirasakan bersama. Ada tiga tipe dari RSP, yaitu:

1. QR (*Quick Response*)

Pada QR, *supplier* menerima data POS dari *retailer* dan menggunakan informasi tersebut untuk disinkronkan dengan aktifitas produksi dan persediaan yang dimiliki *retailer* dengan penjualan pada *retailer*. Pada strategi ini, *retailer* masih mementingkan *individual order*, tetapi data POS digunakan oleh *supplier* untuk melakukan peramalan dan penjadwalan.

2. CR (*Continuous Replenishment* atau *Rapid Replenishment*)

Pada CR, *vendor* menerima data POS dan menggunakan data tersebut untuk mempersiapkan pengiriman dengan *interval* yang telah disepakati sebelumnya untuk mempertahankan tingkat persediaan.

3. VMI (*Vendor Managed Inventory*)

Pada VMI, *supplier* memiliki wewenang untuk menentukan *order quantity* yang akan dikirimkan ke *retailer* berdasarkan informasi data penjualan dan tingkat persediaan yang telah ditetapkan.

Ada dua langkah utama dalam mengimplementasikan VMI menurut SimchiLevi dkk. (2000), yaitu:

1. Ketentuan perjanjian kontrak harus dinegosiasikan. Dimana negosiasi tersebut termasuk keputusan mengenai kepemilikan, kapan barang akan dikirim, persyaratan kredit, tanggung jawab pemesanan dan ukuran performansi seperti tingkat layanan atau persediaan, jika diperlukan.
2. Ada tiga hal yang harus dieksekusi, yaitu:

- (a) Mengembangkan sistem informasi yang terintegrasi untuk *supplier* dan *retailer*. Sistem informasi tersebut harus mudah diakses oleh kedua belah pihak.

- (b) Mengembangkan teknik peramalan yang efektif untuk digunakan oleh *vendor* dan *retailer*.
- (c) Mengembangkan *decision support tools* untuk membantu dalam mengkoordinasikan kebijakan manajemen persediaan dan transportasi. Sistem yang dikembangkan tentu saja akan tergantung dari kemitraan.

Menurut SimchiLevi dkk. (2000), ada beberapa keuntungan yang bisa didapatkan jika menggunakan VMI, yaitu:

1. *Supplier* memiliki pengetahuan mengetahui *order quantity* dan kemampuan untuk mengontrol *bullwhip effect*.
2. Pada VMI, *retailer* akan memberikan informasi *demand* dan *supplier* akan membuat keputusan mengenai pemesanan yang akan mengontrol variasi dari *order quantity*. Hal ini dapat mengurangi biaya dalam sistem dan meningkatkan *service levels*. *Supplier* mendapatkan keuntungan dimana terjadi peningkatan *service level*, penurunan biaya manajerial dan penurunan biaya persediaan.

Menurut Tersine (1994), tujuan dari manajemen persediaan adalah memiliki jumlah persediaan yang tepat di tempat yang tepat, waktu yang tepat, dan biaya yang rendah. Biaya-biaya persediaan terdiri dari:

1. *Purchase cost*  
*Purchase cost* merupakan biaya pembelian *item*, jika *item* tersebut didapatkan dari pihak di luar perusahaan. Jika *item* tersebut diproduksi sendiri oleh perusahaan maka *purchase cost* merupakan biaya produksi *item* tersebut.
2. *Setup cost*  
*Setup cost* adalah biaya yang dikeluarkan untuk melakukan pemesanan *item* ke *supplier*.
3. *Holding cost*  
*Holding cost* adalah biaya penyimpanan dan perawatan *item* yang ada dalam persediaan. Biaya modal, pajak, dan asuransi merupakan contoh dari *holding cost*.
4. *Stockout cost*  
*Stockout* terjadi apabila jumlah barang yang tersedia tidak dapat mencukupi *demand* konsumen.

Ada dua jenis sistem persediaan menurut Tersine (1994), yaitu: *fixed order size system* dan *fixed order interval system*. Tujuan utama dari

kedua sistem persediaan tersebut adalah untuk menentukan jumlah barang optimal yang harus dipesan dan periode pemesanan optimal.

*Fixed Order Size System, Continues Review System*, atau *Q System* merupakan sistem persediaan yang harus terus menerus memonitori posisi *inventory*. Jika posisi *inventory* mencapai titik pemesanan kembali (*reorder point*) maka dilakukan pemesanan barang kepada *supplier*. Rumus yang digunakan adalah sebagai berikut:

$$Q^* = \sqrt{\frac{2CR}{H}} \quad (1)$$

$$TC(Q) = PR + \frac{CR}{Q} + \frac{HQ}{2} \quad (2)$$

dimana  $Q^*$  adalah jumlah pemesanan optimal (unit), C biaya pesan per sekali pesan, R adalah permintaan tahunan (unit), H adalah biaya simpan per unit per tahun, P adalah biaya beli per unit, dan TC(Q) adalah biaya total tahunan.

WagnerWithin Algorithm merupakan pendekatan pemrograman dinamis yang digunakan untuk menghasilkan perhitungan biaya optimal. Ada 3 langkah yang dilakukan dalam perhitungan ini, yaitu:

1. Menghitung matriks biaya total *variable* untuk seluruh *alternative* pemesanan yang dapat dilakukan selama N periode. Biaya total *variable* sudah mencakup biaya pemesanan dan biaya penyimpanan, dimana  $Z_{ce}$  merupakan biaya *total variable* dari periode c hingga periode e yang memenuhi kebutuhan pada periode tersebut. Rumus yang digunakan sebagai berikut:

$$Z_{ce} = C + H \sum_{i=c}^e (Q_{ce} - Q_{ci}) \quad (3)$$

untuk  $1 \leq c \leq e \leq N$

$$Q_{ce} = \sum_{k=c}^e R_k \quad (4)$$

dimana C adalah biaya pesan per sekali pesan, H adalah biaya simpan per unit per tahun,  $R_k$  adalah tingkat permintaan pada periode k.

2. Mendefinisikan  $f_e$  sebagai biaya minimum yang dapat terjadi dari periode 1 hingga periode e, dimana tingkat persediaan pada akhir periode e adalah 0. Algoritma dimulai dengan  $f_0 = 0$  dan perhitungan  $f_1, f_2, \dots, f_N$  secara berturut-turut. Perhitungan  $f_e$  menggunakan rumus berikut:

$$f_e = \text{Min}(Z_{ce} + f_{c-1}) \quad (5)$$

untuk  $c = 1, 2, \dots, e$

- Untuk menerjemahkan solusi optimum ( $f_N$ ) yang diperoleh dari perhitungan algoritma menjadi jumlah pemesanan, diterapkan rumus berikut:

$$f_N = Z_{wN} + f_{w-1} \quad (6)$$

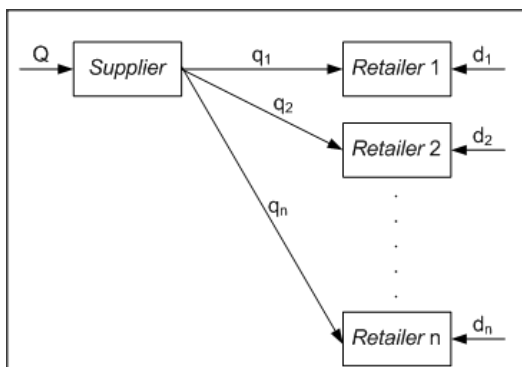
$$f_{w-1} = Z_{vw-1} + f_{v-1} \quad (7)$$

$$f_{u-1} = Z_{1u-1} + f_0 \quad (8)$$

Pemesanan akhir dilakukan pada periode  $w$  untuk memenuhi permintaan dari periode  $w$  hingga periode  $N$ . Pemesanan sebelum terjadi pemesanan akhir dilakukan pada periode  $v$  untuk memenuhi permintaan periode  $v$  hingga periode  $w-1$ . Pemesanan awal dilakukan pada periode 1 untuk memenuhi permintaan periode 1 hingga periode  $u-1$ .

### 3 Perancangan dan Pengujian Model VMI

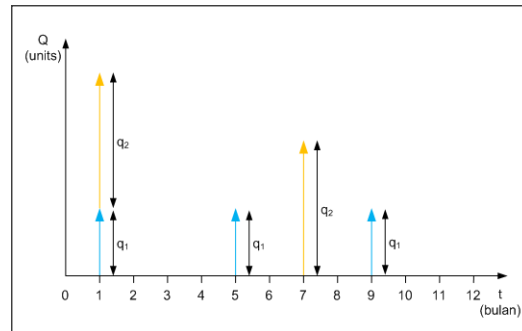
Kerangka pemodelan yang digunakan pada penelitian ini dapat dilihat pada Gambar 1, dimana huruf besar menyatakan pihak *supplier* sedangkan huruf kecil menyatakan pihak *retailer*. Notasi  $Q$  dan  $q$  merupakan *order quantity* sedangkan  $D$  dan  $d$  merupakan permintaan (*demand*).



Gambar 1: Kerangka Pemodelan

Berdasarkan kerangka pemodelan tersebut diketahui bahwa *demand* yang terdapat pada *supplier* bersifat *lumpy*. Hal ini dikarenakan pada tiap periode *supplier* memiliki *demand* yang berbeda-beda. Contoh permintaan pada *supplier* yang mengirim pada 2 *retailer* dapat dilihat pada Gambar 2, dimana Frekuensi pemesanan *retailer*

1 adalah 3 kali dalam 1 tahun dan Frekuensi pemesanan *retailer* 2 adalah 2 kali dalam 1 tahun.



Gambar 2: Permintaan *Supplier*

Ada 3 metode yang digunakan untuk mendapatkan solusi kasus pada penelitian ini, yaitu: *Q System*, *WagnerWithin Algorithm*, dan Model Optimasi yang merupakan model dari VMI untuk satu pemasok dan banyak *retailer*. Rumus-rumus yang digunakan pada perhitungan *Q System* dapat dilihat pada (1) dan (2), sedangkan pada perhitungan *WagnerWithin Algorithm* menggunakan rumus *Q System* pada *retailer* dan (3) hingga (8) pada *supplier*. Hal ini dikarenakan *demand* pada *supplier* yang bersifat *lumpy* sedangkan pada *retailer* tidak.

Rumus pada model yang dirancang, yaitu Model Optimasi dapat dilihat sebagai berikut:

$$\text{Minimasi TBPR} = \sum_{t=1}^{12} \frac{H}{12} \cdot P_t + \sum_{i=1}^{12} C_i \cdot Y_i + \quad (9)$$

$$\sum_{i=1}^n \sum_{t=1}^{12} \frac{h_i}{12} \cdot P_{it} + \sum_{i=1}^n \sum_{t=1}^{12} C_i \cdot Y_{it} \quad (10)$$

s.t.

$$P_t = P_{t-1} + Q_t - \sum_{i=1}^n q_{it} \quad (11)$$

$$Q_t \leq M \cdot Y_t \quad (12)$$

$$Q_t \leq M \cdot Y_t \quad (13)$$

$$Y_{it} = \begin{cases} 0, & q_{it} = 0 \\ 1, & q_{it} \neq 0 \end{cases} \quad (14)$$

$$P_t, Q_t, P_{it}, q_{it} \geq 0 \quad (15)$$

dimana TBPR adalah total biaya rantai pasok,  $t$  adalah *discrete period* (1, 2, 3, ..., 12),  $H$  adalah biaya simpan *supplier*,  $h_i$  adalah biaya simpan *retailer*  $i$ ,  $P_t$  adalah jumlah *inventory supplier* pada periode  $t$ ,  $P_{(t-1)}$  adalah jumlah *inventory supplier*

pada periode  $t-1$ ,  $p_{it}$  adalah jumlah *inventory retailer*  $i$  pada periode  $t$ ,  $p_{(i-1)}$  adalah jumlah *inventory retailer*  $i$  pada periode  $t-1$ ,  $n$  adalah jumlah *retailer*,  $C$  adalah biaya pesan *supplier* per sekali pesan,  $c_i$  adalah biaya pesan *retailer*  $i$  per sekali pesan,  $Y_t$  adalah keputusan pemesanan *supplier* pada periode  $t$ ,  $y_{it}$  adalah keputusan pengiriman *retailer*  $i$  pada periode  $t$ ,  $Q_t$  adalah jumlah pemesanan *supplier* pada periode  $t$ ,  $q_{it}$  adalah jumlah pengiriman *retailer*  $i$  pada periode  $t$ ,  $M$  adalah bilangan bulat besar,  $d_i$  adalah jumlah permintaan *retailer*  $i$ .

Ada 10 studi kasus yang diuji pada penelitian ini. Datadatanya dapat dilihat pada Tabel 1. 6 studi kasus melibatkan 1 *supplier* dengan 2 *retailer*, 4 studi kasus melibatkan 1 *supplier* dengan 3 *retailer*.

Tabel 1: Rangkuman Soalsoal Studi Kasus

Kasus	Supplier			Retailer 1			Retailer 2			Retailer 3		
	C	R	H	$c_1$	$d_1$	$h_1$	$c_2$	$d_2$	$h_2$	$c_3$	$d_3$	$h_3$
1	15	160	3	55	100	4.4	15	60	4.5	0	0	0
2	25	180	2.5	50	80	5	30	100	9.6	0	0	0
3	20	160	4	55	100	4.4	20	60	24	0	0	0
4	7.5	180	3	15	60	4.5	12	120	3.2	0	0	0
5	16.5	120	4.4	15	60	4.5	20	60	24	0	0	0
6	15	240	4.5	12	120	3.2	20	120	12	0	0	0
7	15	240	4.5	50	80	5	15	60	4.5	30	100	9.6
8	15	240	4.5	60	120	4	15	60	4.5	20	60	24
9	15	240	4.5	50	80	5	30	100	9.6	20	60	24
10	15	240	4.5	15	60	4.5	12	120	3.2	20	60	24

Tabel 1 menunjukkan rangkuman soalsoal studi kasus. Solusi untuk studi kasus tersebut menggunakan *Q system*, *WagnerWithin Algorithm*, dan Model Optimasi. Hasil perhitungan studi kasus dapat dilihat pada Tabel 2.

Tabel 2: Perbandingan Total Biaya Keseluruhan

Kasus	Q System Pada Supplier dan Retailer	Q System Pada Retailer dan WagnerWithin Algorithm Pada Supplier	Persentase Perubahan Nilai dari Kolom 2 ke Kolom 3	Model Optimasi	Persentase Perubahan Nilai dari Kolom 3 ke Kolom 5
1	480	370	13.95%	317.917	14.08%
2	590	540	8.47%	464.583	13.97%
3	620	580	6.45%	461.667	20.81%
4	276	231	16.30%	185.25	19.81%
5	462	429	7.14%	330.25	23.02%
6	516	456	11.63%	340	25.44%
7	710	635	10.56%	525.833	17.19%
8	750	660	12.00%	553.75	16.10%
9	860	800	6.98%	643.333	19.38%
10	606	546	9.90%	410.25	24.86%

Pada penelitian ini dilakukan 3 uji sensitivitas pada Model Optimasi, yaitu:

1. Uji Sensitivitas Terhadap Nilai  $c_1$  dan Total Biaya Rantai Pasok
2. Uji Sensitivitas Terhadap Nilai  $c_2$  dan Total Biaya Rantai Pasok
3. Uji Sensitivitas Terhadap Periode Pengiriman dan Total Biaya Rantai Pasok

Hasil perhitungan sensitivitas dapat dilihat pada Tabel 3 hingga Tabel 5 dan Gambar 3 hingga Gambar 5.

Tabel 3: Perubahan Nilai  $c_1$  dan Total Biaya Rantai Pasok

Perubahan	Gradien	$c_1$ (\$)	Total Biaya Rantai Pasok (\$)
-2	4	10	332
-1.9	4	10.1	332.8
-1.7	4	10.3	333.2
-1.6	4	10.4	333.6
-1.5	4	10.5	334
-1.4	4	10.6	334.4
-1.3	4	10.7	334.8
-1.2	4	10.8	335.2
-1.1	4	10.9	335.6
-1	4	11	336
-0.9	4	11.1	336.4
-0.8	4	11.2	336.8
-0.7	4	11.3	337.2
-0.6	4	11.4	337.6
-0.5	4	11.5	338
-0.4	4	11.6	338.4
-0.3	4	11.7	338.8
-0.2	4	11.8	339.2
-0.1	4	11.9	339.6
0	0	12	340

Pada Model Optimasi, dilakukan penyelesaian kasus yang melibatkan 2 hingga 10 *retailer* dengan periode pengiriman 12 bulan. Hasil komputasi dapat dilihat pada Tabel 6 dan Tabel 7.

Tabel 4: Perubahan Nilai  $c_2$  dan Total Biaya Rantai Pasok

Perubahan	Gradien	$c_2$ (\$)	Total Biaya Rantai Pasok (\$)
-2	4	18	332
-1.9	4	18.1	332.4
-1.8	4	18.2	332.8
-1.7	4	18.3	333.2
-1.6	4	18.4	333.6
-1.5	4	18.5	334
-1.4	4	18.6	334.4
-1.3	4	18.7	334.8
-1.2	4	18.8	335.2
-1.1	4	18.9	335.6
-1	4	19	336
-0.9	4	19.1	336.4
-0.8	4	19.2	336.8
-0.7	4	19.3	337.2
-0.6	4	19.4	337.6
-0.5	4	19.5	338
-0.4	4	19.6	338.4
-0.3	4	19.7	338.8
-0.2	4	19.8	339.2
-0.1	4	19.9	339.6
0	0	20	340

Tabel 6 menunjukkan hasil komputasi waktu *run* Model Optimasi, sedangkan Tabel 7 menunjukkan hasil komputasi waktu *crash* Model Optimasi.

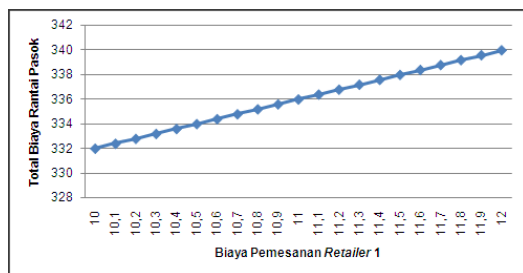
## 4 Analisis

Pada *Q System* dilakukan perhitungan optimal oleh masing-masing pihak. Hasil optimal yang didapatkan kemudian dijumlahkan untuk mendapatkan total biaya rantai pasok. Total biaya tersebut belum tentu optimal. Berdasarkan kenyataan, *supplier* tidak memiliki permintaan yang konstan dikarenakan permintaan pada *supplier* didapatkan dari penjumlahan pesanan dari beberapa *retailer* pada periode tertentu. Dapat dikatakan bahwa permintaan *supplier* bersifat *lumpy*. Tetapi jika menggunakan *Q System*, permintaan *supplier* diasumsikan konstan. Hal ini dikarenakan permintaan yang masuk ke *supplier* merupakan total dari seluruh permintaan *retailer* pada periode waktu tertentu, dalam penelitian ini 12 bulan. Contoh hasil perhitungan *Q System* pada contoh kasus dapat dilihat pada Gambar 6.

Tabel 5: Perubahan Periode Pengiriman dan Total Biaya Rantai Pasok

Periode	Total Biaya Rantai Pasok (\$)	Gradien
3	369	-29.000
4	340	7.000
6	354	-2.333
12	340	

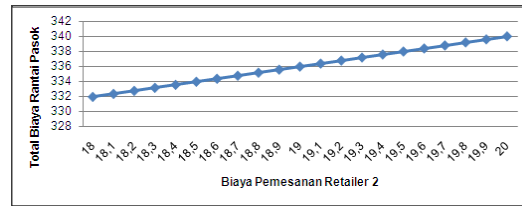
Meskipun terlihat pada Gambar 6 bahwa permintaan *retailer* tidak dapat dipenuhi oleh *supplier*, tetapi jika dilakukan perhitungan tingkat permintaan pada pihak *supplier* dan *retailer* diketahui bahwa tingkat permintaan pihak *supplier* memenuhi tingkat permintaan pihak *retailer* 1 dan *retailer* 2 pada tiap periode. Hal ini terlihat dapat dilihat pada Tabel 8.



Gambar 3: Grafik Perubahan Nilai  $c_1$  dan Total Biaya Rantai Pasok

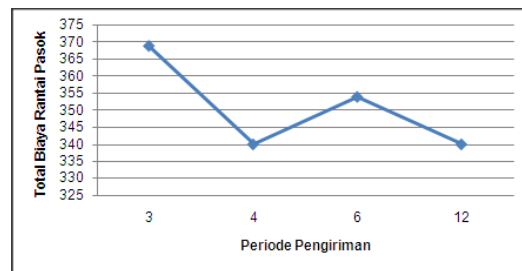
Pada *WagnerWithin Algorithm*, perhitungan pada *retailer* dilakukan dengan menggunakan *Q System*. Karena diasumsikan bahwa permintaan *retailer* bersifat konstan, maka *Q System* merupakan metode perhitungan total biaya persediaan tradisional bagi *retailer* yang optimal. Data *retailer* yang didapatkan dari hasil perhi-

tungan tersebut kemudian dimasukkan dalam perhitungan total biaya *supplier* dengan menggunakan *WagnerWithin Algorithm*.



Gambar 4: Grafik Perubahan Nilai  $c_2$  dan Total Biaya Rantai Pasok

*Supplier* menggunakan *WagnerWithin Algorithm* dikarenakan permintaan yang masuk pada *supplier* bersifat *lumpy*, dimana tiap bulan *supplier* harus mengirim barang ke semua *retailer* dengan jumlah yang berbedabeda tiap bulannya berdasarkan hasil perhitungan *Q System* yang dilakukan oleh tiap *retailer*. Hal ini sesuai dengan situasi nyata, dimana *supplier* harus memperhitungkan jumlah pengiriman yang sesuai untuk memenuhi permintaan *retailer* tiap bulan agar *retailer* tetap mau memesan pada *supplier*. Selain itu, *WagnerWithin Algorithm* merupakan metode heuristik yang paling efisien (menghasilkan total biaya persediaan paling kecil) dibanding metode lainnya. Tidak seperti Model Optimasi, *WagnerWithin Algorithm* dapat digunakan untuk kasus dengan 7 *retailer* atau lebih.



Gambar 5: Grafik Perubahan Periode Pengiriman dan Total Biaya Rantai Pasok

Pada Model Optimasi, keputusan akan diambil oleh *supplier* berdasarkan data yang diberikan oleh seluruh *retailer*. Hal ini dikarenakan Model Optimasi dirancang untuk permasalahan VMI (*Vendor Managed Inventory*), dimana *supplier* adalah pengambil keputusan bagi kedua belah pihak (pihak *supplier* maupun pihak *retailer*). Pada tahap perancangan model dilakukan penentuan parameter, variabel keputusan, fungsi tujuan dan koefisien pembatas.

Parameter yang digunakan adalah total biaya rantai pasok yang dihasilkan model optimasi lebih kecil dari pada metode lain (*Q Sys-*

Tabel 6: Hasil Komputasi Waktu *Run Model Optimasi*

Jumlah Retailer	Rata-rata Waktu <i>Run</i> (detik)	Rata-rata Waktu <i>Run</i> (menit)	Rata-rata Waktu <i>Run</i> (jam)	Solusi
2	12.83	0.214	0.004	Ada
3	313.5	5.225	0.087	Ada
4	952	15.867	0.264	Ada
5	8390	139.833	2.331	Ada
6	10093	168.217	2.804	Ada

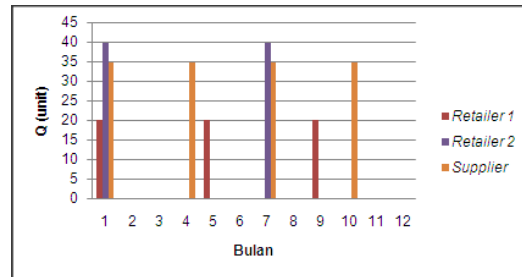
tem dan *WagnerWithin Algorithm*). Model yang dirancang merupakan model *VMI (Vendor Managed Inventory)* yang melihat total biaya yang dibebankan kepada pihak *supplier* dan *retailer* dimana diinginkan bahwa total biaya yang dihasilkan model lebih kecil dari pada total biaya yang dihasilkan oleh metode *Q System* dan *WagnerWithin Algorithm*. Variabel keputusan yang digunakan adalah jumlah *inventory* ( $P_i, p_{ii}$ ), jumlah pemesanan ( $Q_i, q_{ii}$ ) dan keputusan pemesanan ( $Y_i, y_{ii}$ ). Hal ini dikarenakan untuk mengetahui biaya penyimpanan perlu diketahui jumlah *inventory* dan untuk mengetahui biaya pemesanan perlu diketahui jumlah pemesanan optimal beserta kapan pemesanan optimal dilakukan. Fungsi tujuan pada penelitian ini adalah meminimasi total biaya rantai pasok yang terdiri dari biaya penyimpanan dan biaya pemesanan yang dibebankan kepada pihak *supplier* dan *retailer*. Koefisien pembatas merupakan batasan-batasan yang digunakan pada model optimasi yang disesuaikan dengan kendala pada kasus nyata. Oleh karena itu model dapat dikatakan *valid*.

Tabel 7: Hasil Komputasi Waktu *Crash Model Optimasi*

Jumlah Retailer	Rata-rata Waktu <i>Crash</i> (detik)	Rata-rata Waktu <i>Crash</i> (menit)	Rata-rata Waktu <i>Crash</i> (jam)	Solusi
7	7451	124.183	2.070	Tidak Ada
8	5688	94.800	1.580	Tidak Ada
9	5388	89.800	1.497	Tidak Ada
10	3268	54.467	0.908	Tidak Ada

Ada 3 batasan yang digunakan, yaitu: *inventory balanced constraint* (10) dan (13), *binary constraint* (11), (12), (14) dan (15) serta *non-negativity constraint* (16). *Inventory balanced constraint* digunakan karena sesuai dengan kenyataan bahwa jumlah *inventory* pada tiap periode bergantung pada jumlah *inventory* di periode sebelumnya, jumlah barang yang dipesan pada periode yang bersangkutan serta jumlah permintaan pada periode tersebut. *Binary constraint* digunakan untuk menentukan kapan terjadi pemesanan dimana *constraint* ini akan bernilai 1 ketika pada periode tersebut terjadi pemesanan dan bernilai 0 ketika pada periode tersebut tidak terjadi pemesanan. *Non-negativity constraint* digunakan karena sesuai dengan kenyataan bahwa jumlah *inventory* dan jumlah barang yang dipesan tidak

akan bernilai negatif. Jika ada jumlah *inventory* yang bernilai negatif maka dapat dikatakan telah terjadi *stock out*. Pada penelitian ini tidak diperhitungkan mengenai *stock out*.



Gambar 6: Grafik Perhitungan *Q System* Pada Contoh Kasus

Model yang telah dibuat kemudian diterjemahkan kedalam bahasa pemrograman *AMPL* dan kemudian diselesaikan dengan menggunakan program *NEOS* yang dapat diakses pada website <http://www.neos-server.org/neos/solvers/milp:MINTO/AMPL.html> dengan memasukkan data *AMPL* yang telah dibuat. *AMPL* merupakan bahasa pemrograman yang ringkas sehingga digunakan pada penelitian ini. Penelitian ini juga menggunakan program *NEOS* dikarenakan program tersebut mudah digunakan dan mudah diakses oleh pengguna.

Tabel 8: Tingkat Permintaan Pihak *Supplier* dan *Retailer*

Bulan	Retailer 1		Retailer 2		Supplier			
	Q'	Tingkat Permintaan	Bulan	Q'	Tingkat Permintaan	Bulan	Q'	Tingkat Permintaan
1	20	5	1	40	6.667	1	35	11.667
2	0	5	2	0	6.667	2	0	11.667
3	0	5	3	0	6.667	3	0	11.667
4	0	5	4	0	6.667	4	35	11.667
5	20	5	5	0	6.667	5	0	11.667
6	0	5	6	0	6.667	6	0	11.667
7	0	5	7	40	6.667	7	35	11.667
8	0	5	8	0	6.667	8	0	11.667
9	20	5	9	0	6.667	9	0	11.667
10	0	5	10	0	6.667	10	35	11.667
11	0	5	11	0	6.667	11	0	11.667
12	0	5	12	0	6.667	12	0	11.667

Model Optimasi memiliki kelemahan dimana model tidak dapat digunakan untuk *supplier* yang memiliki 7 atau lebih *retailer* dengan periode waktu pengiriman 12 bulan. Hal ini dikarenakan batasan program. Oleh karena itu, untuk kasus tersebut dapat digunakan perhitungan lain seperti *WagnerWithin Algorithm*. Lama penyelesaian kasus tergantung dari memori yang dibutuhkan oleh program *NEOS* dalam penyelesaian. Semakin banyak data yang diinput, semakin banyak pula memori yang dibutuhkan. Oleh karena itu, semakin bertambah jumlah *retailer* yang diuji, semakin lama waktu yang dibutuhkan untuk penyelesaiannya. Pada penyelesaian 7 hingga 10 *retailer*, memori yang

dibutuhkan sudah melebihi kapasitas program sehingga program tidak dapat menyelesaikan kasus tersebut. Semakin banyak data yang di-*input*, semakin cepat waktu yang dibutuhkan *software* untuk menjadi *crash*.

Biaya pemesanan *retailer* pada VMI (*Vendor Managed Inventory*) lebih kecil dari pada Q *System*. Hal ini dikarenakan pada VMI (*Vendor Managed Inventory*), *retailer* hanya dikenakan biaya penyaluran informasi yang berisi data yang dimiliki oleh *retailer* dan biaya transportasi sedangkan pada Q *System*, *retailer* dikenakan biaya telepon, biaya *fax*, biaya penyaluran informasi, biaya transportasi, biaya penerimaan dan pemeriksaan material, biaya administrasi yang berhubungan dengan transaksi pembelian, dan lain-lain. Pada penelitian ini, biaya pemesanan *retailer* yang digunakan dalam perhitungan merupakan biaya pemesanan dalam Q *System*. Oleh karena itu dapat dikatakan bahwa total biaya rantai pasok Model Optimasi sesungguhnya lebih kecil lagi dibandingkan dengan total biaya yang didapatkan pada perhitungan. Hal tersebut telah dibuktikan dengan melakukan uji sensitivitas.

## 5 Kesimpulan

Berdasarkan hasil perancangan, pegujian dan analisis yang dilakukan pada model dapat disimpulkan bahwa model optimasi yang dikembangkan merupakan model optimasi untuk meminimasi total biaya rantai pasok yang terdiri dari biaya di sisi *supplier* dan *retailer* yang melibatkan biaya penyimpanan dan biaya pemesanan. Solusi yang dihasilkan model optimasi lebih baik dibandingkan solusi yang dihasilkan oleh Q *System* maupun *Wagner Within Algorithm*. Model yang dikembangkan telah *valid* dikarenakan batasan-batasan yang digunakan pada model sesuai dengan kendala yang ada pada kasus nyata. Batasan tersebut dapat diklasifikasi menjadi 3 jenis, yaitu: *inventory balanced constraint*, *binary constraint* serta *nonnegativity constraint*. Saransaran yang dapat diberikan untuk pengembangan penelitian ini antara lain: Model Optimasi yang telah dirancang dapat diverifikasi dengan menguji model pada kasus nyata yang sederhana. Selain itu, model optimasi dapat dikembangkan sehingga dapat digunakan untuk kasus yang memiliki 7 *retailer* atau lebih dengan periode waktu 12 bulan. Penelitian ini dapat dikembangkan lebih lanjut untuk masalah negosiasi antara pihak *supplier* dan pihak *retailer* dalam pembagian keuntungan total biaya rantai pasok.

## Daftar Pustaka

- Disney, S. M. dan Towill, D. R. 2003, *Vendor-Managed Inventory and Bullwhip Reduction in a TwoLevel Supply Chain*. *International Journal of Operations & Production Management*, vol. 23, iss. 5, p. 625 651.
- Fogarty, Donald W., Blackstone dan Thomas R. Hoffman. 1991, *Production and Inventory Management, 2nd edition*, South Western Publishing, Cincinnati.
- Hartini, Sri dan Kamal, Andre S. 2010, Penentuan Kebijakan Pemenuhan Pesanan dengan Model *Vendor Managed Inventory*. *Jurnal Teknik Industri*, vol. 11, no. 2, Agustus 2010, p. 95 100
- <http://jobs.indonetasia.com/id/lowongan-kerja-sma-pt-nutrifood-indonesia/> [2013, 3 Januari]
- <http://www.neos-server.org/neos/solvers/milp:MINTO/AMPL.html> [2013, 6 Desember]
- <http://www.waytodeal.com/web/ultrajaya/about> [2013, 3 Januari]
- Indrajit, Richardus Eko dan Djokopranoto, Richardus. 2002, *Konsep Manajemen Supply Chain: Cara Baru Memandang Mata Rantai Penyediaan Barang*, Grasindo, Jakarta.
- Kim, Bowon dan Park, Chulsoon. 2010, *Supply Chain Coordination between Supplier and Retailer in a VMI (VendorManaged Inventory) Relationship*. *The Business Review*, Cambridge, vol. 15, no. 2, Summer 2010, p. 165 170.
- Savasaneril, Secil dan Erkip, Nesim. 2010, *An Analysis of Manufacturer Benefits Under Vendor Managed Systems*. *IIE Transactions*, vol. 42, no. 7, Juli 2010, p.455 477.
- Simchi Levi, D., Kaminsky, P., dan Simchi Levi, E. 2000, *Designing and Managing The Supply Chain: Concepts, Strategies, and Case Studies*, 1st edition, McGraw-Hill, New York.
- Tersine, Ricahrd J. 1994, *Principles Of Inventory and Materials Management*, 4thedition, Prentice Hall, New Jersey.
- Yao, Y., Dong, Y., dan Dresner, Martin E. 2007, *Analizing InformationEnabled Stockout Management Under VendorManaged Inventory*. *Inf Thecnol Manage*, vol. 8, p. 133 145.



Yao, Y., Evers, Philip T., dan Dresner, Martin E.  
2007, *Supply Chain Integration In VendorMan-  
aged Inventory*. Decision Support Systems, 43,  
p. 663 674.