

**OPTIMASI ALOKASI RUANG PRODUK PADA RAK *DISPLAY*  
GERAI RITEL *GROCERY* MENGGUNAKAN IMPLEMENTASI  
MODEL *SHELF SPACE ALLOCATION PROBLEM* DAN  
ALGORITMA GENETIKA SERTA ANALISIS KEBIJAKAN  
*IN-STORE SHELF REPLENISHMENT***

**SKRIPSI**

**RIZKI AMALIA  
0606077503**



**UNIVERSITAS INDONESIA  
FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK INDUSTRI  
DEPOK  
JULI 2010**

**OPTIMASI ALOKASI RUANG PRODUK PADA RAK *DISPLAY*  
GERAI RITEL *GROCERY* MENGGUNAKAN IMPLEMENTASI  
MODEL *SHELF SPACE ALLOCATION PROBLEM* DAN  
ALGORITMA GENETIKA SERTA ANALISIS KEBIJAKAN  
*IN-STORE SHELF REPLENISHMENT***

**SKRIPSI**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik**

**RIZKI AMALIA  
0606077503**



**UNIVERSITAS INDONESIA  
FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK INDUSTRI  
DEPOK  
JULI 2010**

## HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,  
dan semua sumber baik yang dikutip maupun yang dirujuk  
telah saya nyatakan dengan benar.**

**Nama : Rizki Amalia**

**NPM : 0606077503**

**Tanda tangan :**

**Tanggal : 5 Juli 2010**

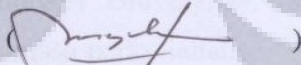
## HALAMAN PENGESAHAN

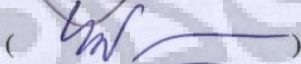
Skripsi ini diajukan oleh


Nama : Rizki Amalia  
NPM : 0606077503  
Program Studi : Teknik Industri  
Judul Skripsi : Optimasi Alokasi Ruang Produk pada Rak *Display* Gerai Ritel *Grocery* Menggunakan Implementasi Model *Shelf Space Allocation Problem* dan Metode Algoritma Genetika serta Analisis Kebijakan *in-Store Shelf Replenishment*

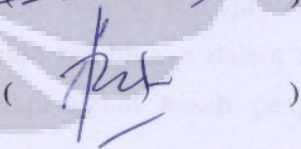
Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Industri, Fakultas Teknik, Universitas Indonesia

### DEWAN PENGUJI

Pembimbing : Ir. Amar Rachman, MEIM (  )

Penguji : Ir. Yadrifil M.Sc. (  )

Penguji : Armand Omar Moeis S.T., M.Sc. (  )

Penguji : Arian Dhini S.T., M.T. (  )

Ditetapkan di : Depok  
Tanggal : 30 Juni 2010

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, skripsi ini dapat diselesaikan dengan baik. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Program Studi Teknik Industri pada Fakultas Teknik Universitas Indonesia. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan samai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ir. Amar Rachman, MEIM, selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan skripsi ini;
2. Bapak Mujahidin Majid dan Bapak Irfan, selaku pihak Hypermart Depok yang telah banyak membantu dalam usaha memperoleh data yang dibutuhkan;
3. Kak Lismanto atas segala bantuan dan kerjasama yang tak terhingga dalam menyelesaikan skripsi ini;
4. Segenap karyawan Departemen Teknik Industri Universitas Indonesia yang telah banyak membantu penulis selama masa perkuliahan;
5. Orang tua dan keluarga yang selalu menyayangi dan mendoakan, memberikan perhatian, motivasi, masukan, dan dukungan baik material maupun moral;
6. Nicholas, Yanti Nitra, Evi Ferawaty, Leonardo, Asa V. R., Tuty Arsyida, Dira Ballerina, dan Irvan, rekan-rekan sesama bimbingan dalam skripsi, atas dukungan dan perhatiannya menghadapi keluh kesah pengerjaan skripsi;
7. Shinta Juliastri, Rika Arti, Ayu Kusumawardhani, Damayanti, Christie Gaol, Indah Cahyati, Renta Robasa, dan Nurulita atas segala kenangan persahabatan selama empat tahun terakhir ini yang tidak akan terlupakan. Canda, tawa, gosip, karaoke, movie marathon, sleepless nights, dan cerita-

cerita lainnya pasti akan selalu menjadi kenangan manis yang akan selalu diingat;

8. Teman-teman angkatan 2006 lainnya untuk segala waktu, obrolan, canda tawa dan bantuan yang telah diberikan selama masa perkuliahan;
9. Semua pihak yang terlibat dan telah membantu penulis sehingga skripsi ini dapat terselesaikan dengan baik.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu pengetahuan.

Depok, 5 Juli 2010

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI  
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

---

---

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Rizki Amalia  
NPM : 0606077503  
Program Studi : Teknik Industri  
Departemen : Teknik Industri  
Fakultas : Teknik  
Jenis karya : Skripsi

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :

**“Optimasi Alokasi Ruang Produk pada Rak *Display* Gerai Ritel *Grocery* Menggunakan Implementasi Model *Shelf Space Allocation Problem* dan Metode Algoritma Genetika serta Analisis Kebijakan *in-Store Shelf Replenishment*”**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta dan sebagai pemilih Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok  
Pada tanggal : 5 Juli 2010  
Yang menyatakan

(Rizki Amalia)



## ABSTRAK

Nama : Rizki Amalia  
Program Studi : Teknik Industri  
Judul Skripsi : Optimasi Alokasi Ruang Produk pada Rak *Display* Gerai Ritel *Grocery* Menggunakan Implementasi Model *Shelf Space Allocation Problem* dan Metode Algoritma Genetika serta Analisis Kebijakan *in-Store Shelf Replenishment*

Alokasi ruang merupakan suatu permasalahan yang signifikan bagi manajemen ritel, mengingat adanya sumber daya ruang rak *display* yang terbatas namun variasi produk yang terus-menerus bertambah. Di sisi lain, penelitian sebelumnya mengungkapkan bahwa jumlah ruang (*facing*) dapat mempengaruhi tingkat penjualan sebuah produk. Penelitian ini akan membahas mengenai optimasi alokasi ruang produk pada rak *display* gerai ritel *grocery* dengan mengimplementasikan model *Shelf Space Allocation Problem* (SSAP) yang dikembangkan Bai (2005) dengan metode penyelesaian algoritma genetika. Hasil penelitian menunjukkan metode yang digunakan dapat memberikan nilai keuntungan kategori produk teh kemasan 7,30% lebih baik dari yang ada saat ini. Analisis kebijakan *in-store shelf replenishment* akan ditinjau dengan menggunakan model umum *reorder point* (ROP).

Kata kunci:  
*Facing, shelf space allocation problem, algoritma genetika, teh kemasan*



## ABSTRACT

Name : Rizki Amalia  
Study Program : Industrial Engineering  
Title : Optimization of Product Space Allocation by Using Implementation of Shelf Space Allocation Model (SSAP) and Genetic Algorithm Method along with Analysis on the in-Store Shelf Replenishment Policy

Space allocation is a critical problem for retail management, considering the increasing variation of products in contrast with the limited resources of shelf space. Moreover, previous researches have concluded that the number of space (facing) allocated could affect the sales of a product. This research discusses optimization of product shelf space allocation in grocery retail store by implementing Shelf Space Allocation Problem (SSAP) model developed by Bai (2005) with genetic algorithm as solving method. Result has shown that this method produces 7,30% higher product category profit of tea drink than what it has today. Analysis on the in-store shelf replenishment is done by using the common reorder point model.

**Keywords:**

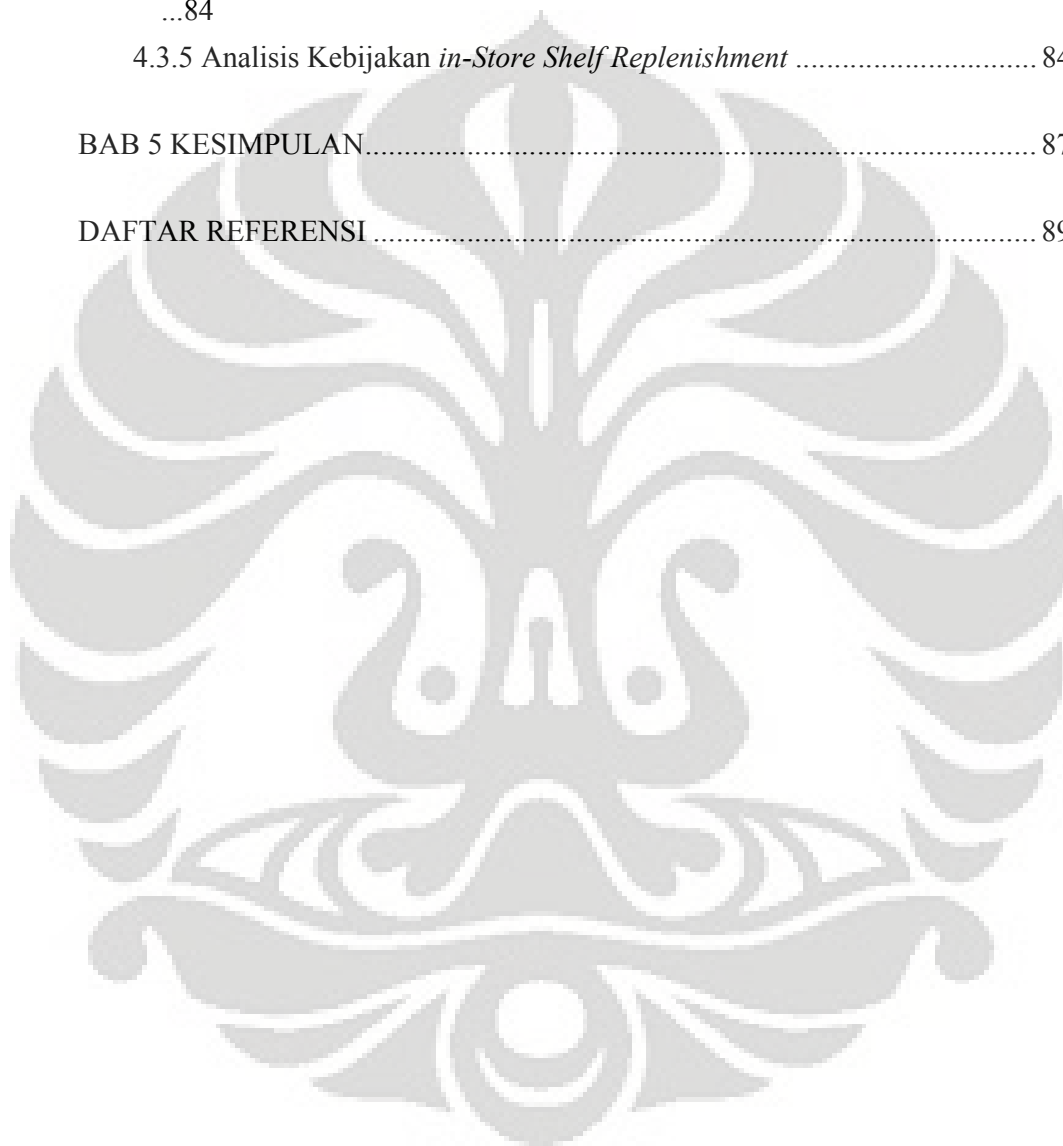
Facing, shelf space allocation problem, genetic algorithm, tea drink

## DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS .....	ii
HALAMAN PENGESAHAN .....	iii
KATA PENGANTAR .....	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS .....	vi
ABSTRAK .....	vii
ABSTRACT .....	viii
DAFTAR ISI .....	ix
DAFTAR TABEL .....	xii
DAFTAR GAMBAR .....	xivv
DAFTAR LAMPIRAN .....	xv
BAB 1 PENDAHULUAN .....	1
1.1 Latar Belakang Permasalahan .....	1
1.2 Diagram Keterkaitan Masalah .....	4
1.3 Perumusan Masalah .....	4
1.4 Tujuan Penelitian .....	4
1.5 Ruang Lingkup Permasalahan .....	6
1.6 Metodologi Penelitian .....	6
1.7 Sistematika Penelitian .....	10
BAB 2 DASAR TEORI .....	11
2.1 Konsep <i>Retailing</i> .....	11
2.1.1 Karakteristik Bisnis Ritel .....	11
2.1.2 Berbagai Tipe Bisnis Ritel .....	13
2.1.2.1 Tipe Bisnis Ritel Atas Kepemilikan (Ownership) .....	13
2.1.2.2 Tipe Bisnis Ritel Berdasarkan Merchandise Category .....	14
2.1.2.3 Tipe Bisnis Ritel Berdasarkan Luas Sales Area .....	14
2.1.2.4 Non-Store Retailer .....	15
2.2 Manajemen Ruang <i>Display</i> ( <i>Space management</i> ) .....	15
2.3 Kategori Produk Berdasarkan Hubungan <i>Sales-Space Elasticity</i> .....	16
2.4 <i>Shelf Space Allocation Problem</i> (SSAP) .....	16
2.4.1 Tujuan Umum Permasalahan Alokasi Ruang <i>Display</i> .....	18
2.4.2 Variabel Keputusan dalam SSAP .....	18

2.4.2.1 Facing.....	18
2.4.2.2 Lokasi.....	19
2.4.3 Kendala ( <i>Constraint</i> ) dalam SSAP.....	19
2.4.3.1 Kendala Fisik.....	20
2.4.3.2 Kendala Integrality.....	20
2.4.3.3 Ketentuan Display.....	20
2.4.4 Model Optimasi Permasalahan <i>Shelf Space Allocation Problem</i> .....	21
2.4.4.1 Model Kompleks SSAP.....	21
2.4.4.2 Simplifikasi Model SSAP.....	23
2.5 Algoritma Genetika.....	24
2.5.1 <i>Encoding</i> .....	25
2.5.2 Inisialisasi.....	25
2.5.3 <i>Crossover</i> dan Mutasi.....	26
2.5.4 Penanganan Kendala.....	26
2.5.5 <i>Fitness Function</i> dan Mekanisme <i>Scaling</i> .....	27
2.6 Kebijakan <i>in-Store Shelf Replenishment</i> dengan <i>Reorder Point Model</i> .....	27
 BAB 3 PENGUMPULAN DATA.....	 29
3.1 Pemilihan Jenis Produk.....	29
3.2 Data Penjualan dan Keuntungan Tiap Produk.....	30
3.3 Parameter <i>Space Elasticity</i> .....	33
3.4 Parameter Kendala Kapasitas dan Batas Alokasi <i>Facing</i> .....	34
 BAB 4 PENGOLAHAN DATA DAN ANALISIS.....	 35
4.1 Penyusunan Algoritma.....	35
4.1.1 Langkah-langkah Penyusunan Algoritma Genetika.....	35
4.1.2 Verifikasi dan Validasi Program.....	42
4.1.2.1 Hasil piranti lunak.....	44
4.1.2.2 Hasil perhitungan manual.....	44
4.2 Pengolahan Data.....	53
4.2.1 Parameter yang Digunakan dalam Algoritma Genetika.....	53
4.2.2 Data <i>Input</i> .....	53
4.2.3 Hasil Pengolahan Data SSAP.....	54
4.2.4 Usulan Kebijakan <i>in-Store Shelf-Replenishment</i> .....	57
4.3 Analisis.....	60
4.3.1 Analisis Model SSAP.....	60

4.3.2 Analisis Metode AG.....	63
4.3.3 Analisis Hasil Pengolahan Data.....	68
4.3.4 Analisis Sensitivitas Model SSAP .....	72
4.3.4.1 Analisis Sensitivitas terhadap Kendala Batas Alokasi Facing.....	72
4.3.4.2 Analisis Sensitivitas terhadap Parameter Koefisien Space Elasticity.....	84
4.3.5 Analisis Kebijakan <i>in-Store Shelf Replenishment</i> .....	84
 BAB 5 KESIMPULAN.....	 87
 DAFTAR REFERENSI .....	 89



## DAFTAR TABEL

<b>Tabel 3.1</b> Proporsi Penjualan Kategori Produk Minuman Ringan .....	29
<b>Tabel 3.2</b> Data SKU Sub-Kategori Minuman Teh Kemasan Botol.....	30
<b>Tabel 3.3</b> Data Penjualan, Keuntungan, Alokasi <i>Facing</i> Saat ini untuk Tiap Produk .....	31
<b>Tabel 3.4</b> Data <i>Unmodified Demand</i> .....	32
<b>Tabel 4.1</b> Data <i>Dummy</i> .....	43
<b>Tabel 4.2</b> Parameter Model SSAP untuk Data <i>Dummy</i> .....	43
<b>Tabel 4.3</b> Parameter GA Data <i>Dummy</i> .....	44
<b>Tabel 4.4</b> <i>Output</i> Hasil <i>Run</i> Program dengan Data <i>Dummy</i> .....	44
<b>Tabel 4.5</b> Pengkodean Bilangan Biner.....	45
<b>Tabel 4.6</b> Evaluasi <i>Fitness</i> Awal .....	45
<b>Tabel 4.7</b> Penalti Kapasitas untuk Tiap Kromosoml .....	45
<b>Tabel 4.8</b> Populasi Awal (dari Program).....	46
<b>Tabel 4.9</b> Nilai <i>Fitness</i> Akhir Tiap Kromosom.....	47
<b>Tabel 4.10</b> <i>Linear Fitness Ranking</i> .....	47
<b>Tabel 4.11</b> Rentang Proporsi tiap Kromosom .....	48
<b>Tabel 4.12</b> Pemilihan Solusi Terbaik dari Populasi Akhir.....	49
<b>Tabel 4.13</b> Alokasi <i>Facing</i> Produk Hasil Perhitungan Manual.....	49
<b>Tabel 4.14</b> Pasangan Pertama Kromosom Orang Tua .....	50
<b>Tabel 4.15</b> Kromosom Anak Hasil <i>Crossover</i> Pertama.....	50
<b>Tabel 4.16</b> Pasangan Kedua Kromosom Orang Tua.....	50
<b>Tabel 4.17</b> Kromosom Anak Hasil <i>Crossover</i> Kedua.....	51
<b>Tabel 4.18</b> Kromosom Anak Sebelum Mutasi .....	51
<b>Tabel 4.19</b> Kromosom Anak Setelah Mutasi.....	51
<b>Tabel 4.20</b> Populasi Kromosom Baru.....	52
<b>Tabel 4.21</b> Parameter Model AG SSAP .....	53
<b>Tabel 4.22</b> Hasil Pengolahan Data (10x <i>Run</i> ).....	54
<b>Tabel 4.23</b> Solusi Akhir SSAP .....	56
<b>Tabel 4.24</b> <i>Reorder Point</i> (ROP) untuk Tiap Produk .....	58
<b>Tabel 4.25</b> Nilai $Q^*$ untuk Tiap Produk.....	59
<b>Tabel 4.26</b> <i>Trial</i> Ukuran Populasi terhadap Kualitas Solusi .....	65
<b>Tabel 4.27</b> <i>Trial</i> Jumlah Generasi terhadap Kualitas Solusi .....	66
<b>Tabel 4.28</b> Solusi Akhir <i>Facing</i> Produk Minuman Teh Kemasan .....	70
<b>Tabel 4.29</b> Perbandingan Solusi Program dengan Alokasi <i>Facing</i> Saat ini.....	71

<b>Tabel 4.30</b> Alokasi <i>Facing</i> Berdasarkan Klasifikasi Produk (Min. <i>Facing</i> = 2; Max. <i>Facing</i> = 20 .....	73
<b>Tabel 4.31</b> Analisis Sensitivitas Kendala Minimal <i>Facing</i> terhadap Alokasi Ruang Produk .....	74
<b>Tabel 4.32</b> Analisis Sensitivitas Kendala Minimal <i>Facing</i> terhadap Nilai Keuntungan Kategori Produk .....	75
<b>Tabel 4.33</b> Grafik Perubahan Minimal <i>Facing</i> terhadap Keuntungan Produk.....	76
<b>Tabel 4.34</b> Analisis Sensitivitas Kendala Maksimal <i>Facing</i> terhadap Alokasi Ruang Produk .....	77
<b>Tabel 4.35</b> Analisis Sensitivitas Kendala Maksimal <i>Facing</i> terhadap Nilai Keuntungan Kategori Produk .....	78
<b>Tabel 4.36</b> Perbandingan Sensitivitas Maksimal <i>Facing</i> dengan Alokasi <i>Facing</i> Saat ini.....	78
<b>Tabel 4.37</b> Analisis Sensitivitas Kenaikan Parameter Maksimal dan Minimal <i>Facing</i> terhadap Proporsi Alokasi Ruang Produk dan Keuntungan Kategori Produk.....	80
<b>Tabel 4.38</b> Analisis Sensitivitas terhadap Kenaikan Minimal <i>Facing</i> dan Penurunan Maksimal <i>Facing</i> terhadap Alokasi Ruang Produk dan Keuntungan Kategori Produk.....	82
<b>Tabel 4.39</b> Analisis Sensitivitas Parameter <i>Space Elasticity</i> terhadap Alokasi Ruang Produk dan Nilai Keuntungan Kategori Produk.....	84

## DAFTAR GAMBAR

<b>Gambar 1.1</b> Diagram Keterkaitan Masalah .....	5
<b>Gambar 1.1</b> Diagram Keterkaitan Masalah (lanjutan) .....	6
<b>Gambar 1.2</b> Diagram Alir Metode Penelitian .....	7
<b>Gambar 1.2</b> Diagram Alir Metode Penelitian (lanjutan) .....	8
<b>Gambar 2.1</b> Fungsi Ritel dalam Distribusi .....	12
<b>Gambar 2.2</b> Karakteristik Bisnis Ritel .....	12
<b>Gambar 2.3</b> Ilustrasi Tingkat Penjualan dengan Mempertimbangkan Faktor Alokasi Ruang Rak Produk .....	21
<b>Gambar 4.1</b> Diagram Alur Algoritma Genetika Permasalahan SSAP .....	36
<b>Gambar 4.1</b> Diagram Alur Algoritma Genetika Permasalahan SSAP (lanjutan) .....	37
<b>Gambar 4.2</b> Tampilan Solusi Terbaik Program Matlab AG-SSAP .....	55
<b>Gambar 4.3</b> Grafik Solusi Terbaik ( <i>Run 2</i> ) .....	55
<b>Gambar 4.4</b> Perubahan Minimal <i>Facing</i> terhadap Alokasi Ruang Produk .....	75
<b>Gambar 4.5</b> Grafik Perubahan Minimal <i>Facing</i> terhadap Keuntungan Produk .....	76
<b>Gambar 4.6</b> Grafik Perubahan Maksimal <i>Facing</i> terhadap Alokasi Ruang .....	79
<b>Gambar 4.7</b> Grafik Perubahan Maksimal <i>Facing</i> terhadap Keuntungan Kategori Produk .....	79
<b>Gambar 4.8</b> Grafik Perubahan Kenaikan Minimal-Maksimal <i>Facing</i> terhadap Nilai Keuntungan Kategori Produk .....	80
<b>Gambar 4.9</b> Grafik Perubahan Minimal-Maksimal <i>Facing</i> terhadap Alokasi Ruang Produk .....	81
<b>Gambar 4.10</b> Grafik Perubahan Kenaikan Minimal <i>Facing</i> dan Penurunan Maksimal <i>Facing</i> terhadap Alokasi Ruang Produk .....	83
<b>Grafik 4.11</b> Grafik Perubahan Kenaikan Minimal <i>Facing</i> dan Penurunan Maksimal <i>Facing</i> terhadap Nilai Keuntungan Produk .....	83



## DAFTAR LAMPIRAN

Lampiran 1: *Source Code* MATLAB Algoritma Genetika



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang Permasalahan

Industri ritel modern Indonesia merupakan sektor area yang berkembang pesat dalam beberapa tahun terakhir ini. Badai krisis keuangan global yang terjadi belakangan tidak memberikan efek negatif yang berarti pada sektor ini, dimana industri ritel modern tetap mengalami pertumbuhan yang signifikan. Perkembangan bisnis ritel modern ditunjukkan dari segi omset yang tumbuh dari sekitar Rp 42 triliun pada tahun 2005, meningkat menjadi sekitar Rp 58 triliun pada tahun 2007, dan mencapai Rp 67 triliun pada tahun 2008 (*Peta Persaingan*, 2009). *Business Monitor International* (BMI) (2010) memprediksi sektor *Mass Grocery Retail* (MGR) Indonesia akan mengalami peningkatan penjualan signifikan hingga mencapai US\$10,44 miliar pada tahun 2014. Faktor-faktor yang diperkirakan dapat mendorong pertumbuhan tingkat penjualan pada industri ritel terletak pada pertumbuhan ekonomi yang kuat, jumlah populasi yang besar dan terus meningkat (terbesar keempat di dunia), meningkatnya pendapatan per kapita, serta adanya pembangunan berkelanjutan terhadap infrastruktur industri ritel (Business Monitor International, 2010).

Adanya pertumbuhan jumlah penduduk meningkatkan jumlah pelanggan potensial bagi industri ritel. Disamping itu, peningkatan GDP per kapita berdampak pada meningkatnya daya beli serta tingkat belanja konsumen. Gaya hidup masyarakat perkotaan yang senang belanja di ritel modern yang menjanjikan produk yang selalu segar, harga rendah, serta variasi produk yang lengkap juga menjadi pemicu suksesnya gerai-gerai ritel *grocery* modern, terutama ritel *hypermarket* yang dapat meraih omset hingga Rp 1 miliar per hari pada periode *peak season* (*Peluang Usaha*, n.d.). Sektor ritel merupakan jenis usaha yang dapat bertahan sepanjang masa, karena adanya potensi kebutuhan yang besar dari masyarakat terhadap produk-produk yang dijual serta keberadaannya yang tidak dipengaruhi tren pasar. Dengan kata lain, industri ritel akan selalu dibutuhkan oleh masyarakat dimanapun, kapanpun, dan dalam kondisi apa pun.

Pertumbuhan industri ritel *grocery* secara keseluruhan juga didorong oleh adanya ekspansi besar-besaran yang dilakukan oleh perusahaan-perusahaan utama yang berkecimpung di bidang ini dengan maraknya pembukaan gerai-gerai ritel *grocery* baru, baik yang berskala *hypermarket*, *supermarket*, maupun *minimarket*. Usaha ekspansi yang bertujuan untuk mendekati diri ke konsumen mendorong para pebisnis ritel untuk membangun gerai-gerainya di daerah yang lebih mudah dijangkau oleh konsumen dengan tujuan untuk meminimalkan faktor waktu dan biaya transportasi. Hal ini tentunya guna memenuhi kebutuhan masyarakat untuk mendapatkan barang dalam waktu yang cepat dan tepat. Alhasil, jumlah gerai ritel *grocery* berkembang pesat di daerah pinggiran kota dan daerah kota-kota satelit lainnya, dan menyebabkan persaingan yang sangat kompetitif antara gerai satu dengan gerai lainnya.

Ketatnya persaingan antar *retailer* menuntut para pelaku industri ritel untuk dapat menerapkan suatu sistem manajemen operasional yang efisien guna menarik pelanggan dan memaksimalkan keuntungan gerai. Keberagaman jenis produk yang ada saat ini merupakan tantangan tersendiri bagi para *retailer* untuk dapat memenuhi apa yang dibutuhkan konsumen. Tingkat loyalitas pelanggan tidak hanya terletak pada produk-produk yang dibelinya, namun juga pada gerai ritel dimana mereka mendapatkan produk tersebut. Bila suatu toko ritel telah dapat memenuhi kebutuhannya, maka pelanggan tidak akan segan untuk kembali berbelanja di toko tersebut. Pada jaman sekarang, ritel bukan hanya sekedar toko. Ada unsur kepuasan yang harus dijaga terus menerus, serta *brand experience* yang harus dirasakan oleh konsumen (Susanta, 2009). Seperti yang diungkapkan Levy dan Weitz dalam *Retailing Management*, selain adanya *merchandise pricing*, layanan servis, *advertising*, dan program promosi lainnya, taktik operasional yang dapat digunakan para *retailer* dalam mempengaruhi tingkat belanja pelanggan diantaranya meliputi *product assortment* (menentukan jenis produk apa yang akan dijual), tata letak toko, serta perencanaan alokasi ruang untuk tiap produk (Bai, 2005, p. 2).

Alokasi ruang rak *display* produk merupakan permasalahan yang patut diteliti lebih lanjut mengingat peluangnya yang dapat meningkatkan penjualan dari gerai ritel *grocery* itu sendiri. Dimana tata letak produk dalam toko ritel dapat

meningkatkan visual efek dari lingkungan berbelanja yang dirasakan pelanggan, implementasi alokasi ruang rak *display* yang tepat dan sesuai dapat meningkatkan produktivitas ruang dan mendorong pelanggan untuk melakukan pembelian (Ruibin Bai, 2005). Permasalahan mengenai alokasi ruang rak *display*, atau yang dikenal dengan *Shelf Space Allocation Problem* (SSAP) merupakan suatu hal yang menantang bagi pelaku ritel dimanapun. Kendala terbesar dari SSAP ini adalah adanya sumber daya rak *display* produk yang terbatas yang dimiliki gerai, sedangkan di sisi lain variasi jenis produk akan selalu bertambah dari waktu ke waktu. Kecilnya kemungkinan memperluas gerai ataupun menambah jumlah rak *display* menyebabkan optimasi alokasi ruang rak *display* untuk tiap produk merupakan salah satu bentuk keputusan yang penting untuk dipikirkan dalam manajemen ritel.

Telah banyak penelitian yang dilakukan mengenai model optimasi dari SSAP, maupun *shelf management* pada umumnya. Larson dan DeMarais (1990) mengungkapkan bahwa persediaan barang pada rak *display* dapat berfungsi untuk menstimulasi tingkat *demand*, dan oleh karenanya dianggap sebagai '*psychic stock*.' Mereka juga menerapkan aturan "*full-shelf merchandising*"; dimana area/ rak *display* dijaga agar selalu berada pada kondisi stok penuh, berdasarkan pada studi eksperimen yang menyimpulkan bahwa tingkat penjualan yang lebih tinggi dapat dicapai dengan menjaga rak *display* selalu dalam keadaan stok penuh (Urban, 1998). Nilai dari *space elasticity* juga perlu diestimasi, yaitu rasio perubahan relatif penjualan terhadap perubahan alokasi ruang rak pada suatu produk. Bagi sebuah gerai, optimasi alokasi ruang rak *display* produk akan sangat berguna terutama bagi kategori produk yang cenderung bersifat *unplanned purchases*, dimana pelanggan baru akan menentukan pembelian produk tertentu ketika berada di gerai dan dihadapkan pada rak *display* produk. Riset sebelumnya menunjukkan bahwa *unplanned purchases* menyumbang sekitar sepertiga dari seluruh transaksi di banyak gerai ritel *grocery* (Buttle, 1984).

Penelitian ini akan menekankan pada bagaimana mengalokasikan ruang rak *display* untuk tiap merk produk pada suatu kategori produk. Implementasi model matematis SSAP yang dikembangkan oleh Ruibin Bai (2005) dengan metode penyelesaian Algoritma Genetika akan digunakan untuk mencari solusi

yang optimal dari permasalahan ini. Lebih lanjut lagi, penelitian ini juga akan meninjau kebijakan *in-store shelf replenishment* dalam menjaga ketersediaan barang pada rak *display*. Kebijakan pengisian rak yang diterapkan manajemen ritel akan mempengaruhi kondisi *display* produk dan pada akhirnya berpengaruh pada kinerja penjualan produk tersebut. Metode umum *reorder point* (ROP) akan digunakan dalam penentuan kebijakan pengisian rak *display*. Hasil dari penelitian yang dilakukan diharapkan dapat menghasilkan produktivitas ruang rak yang optimal guna memaksimalkan keuntungan gerai dan meningkatkan kepuasan pelanggan.

## 1.2 Diagram Keterkaitan Masalah

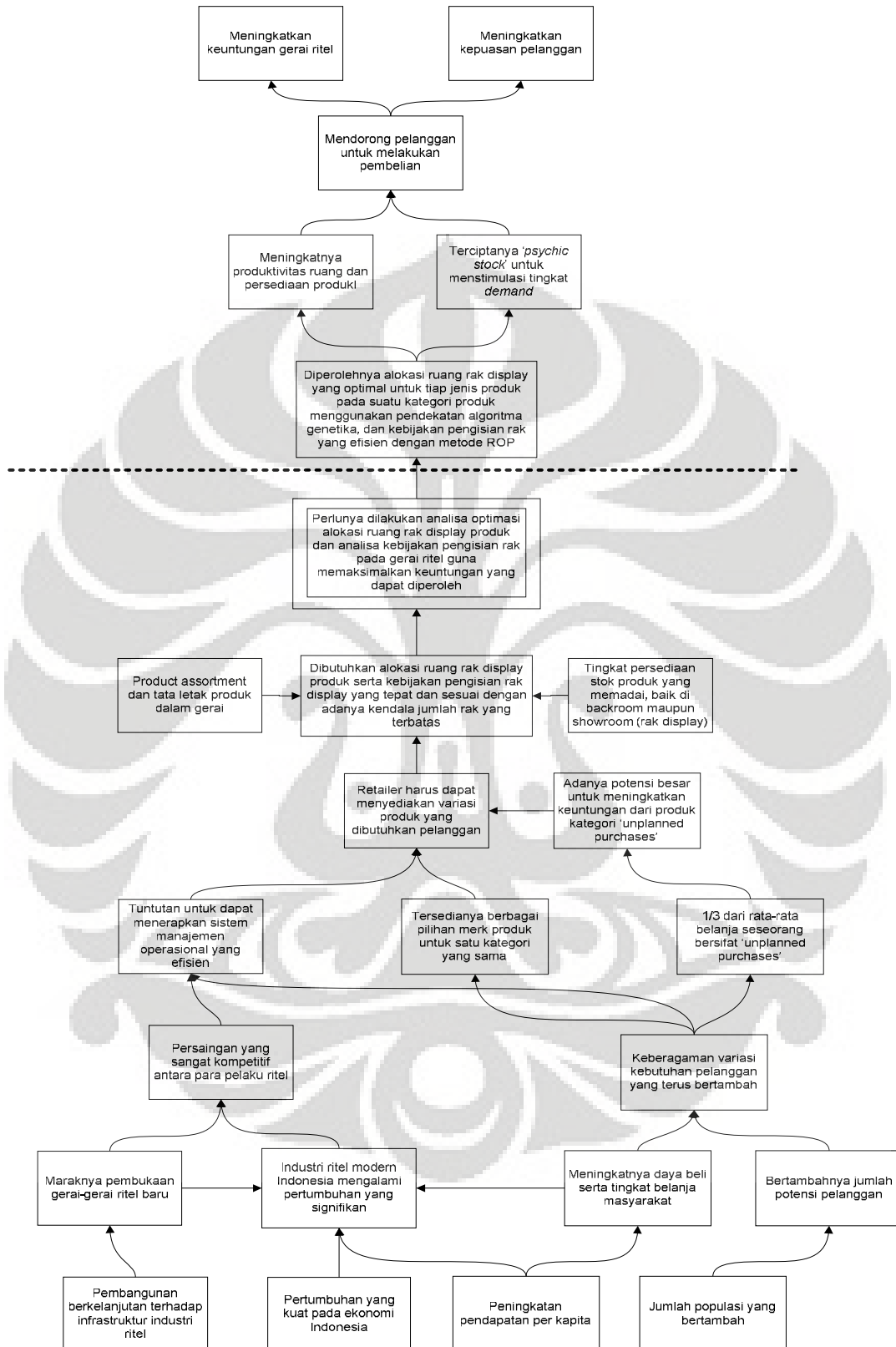
Berdasarkan latar belakang yang telah dijelaskan pada subbab sebelumnya, maka dapat dikonstruksikan sebuah diagram keterkaitan masalah yang menampilkan permasalahan secara visual dan sistematis. Diagram keterkaitan masalah dari penelitian ini dapat dilihat pada Gambar 1.1.

## 1.3 Perumusan Masalah

Pokok permasalahan yang menjadi fokus dalam penelitian ini adalah perlunya menentukan alokasi optimal ruang rak *display* untuk tiap jenis produk dalam suatu kategori produk. Variabel keputusan yang akan dihasilkan adalah berupa jumlah *facing* yang optimal untuk tiap produk.

## 1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk memperoleh konfigurasi alokasi ruang rak *display* yang optimal untuk tiap jenis produk yang ada dalam satu kategori dengan menggunakan implementasi model *Shelf Space Allocation Problem* (SSAP) metode Algoritma Genetika guna mencapai fungsi tujuan untuk memaksimalkan keuntungan yang dapat diperoleh dari kategori produk tersebut bagi gerail ritel, serta usulan kebijakan *in-store shelf replenishment* menggunakan metode ROP dalam menjaga ketersediaan produk pada rak *display*.



Gambar 1.1 Diagram Keterkaitan Masalah

### 1.5 Ruang Lingkup Permasalahan

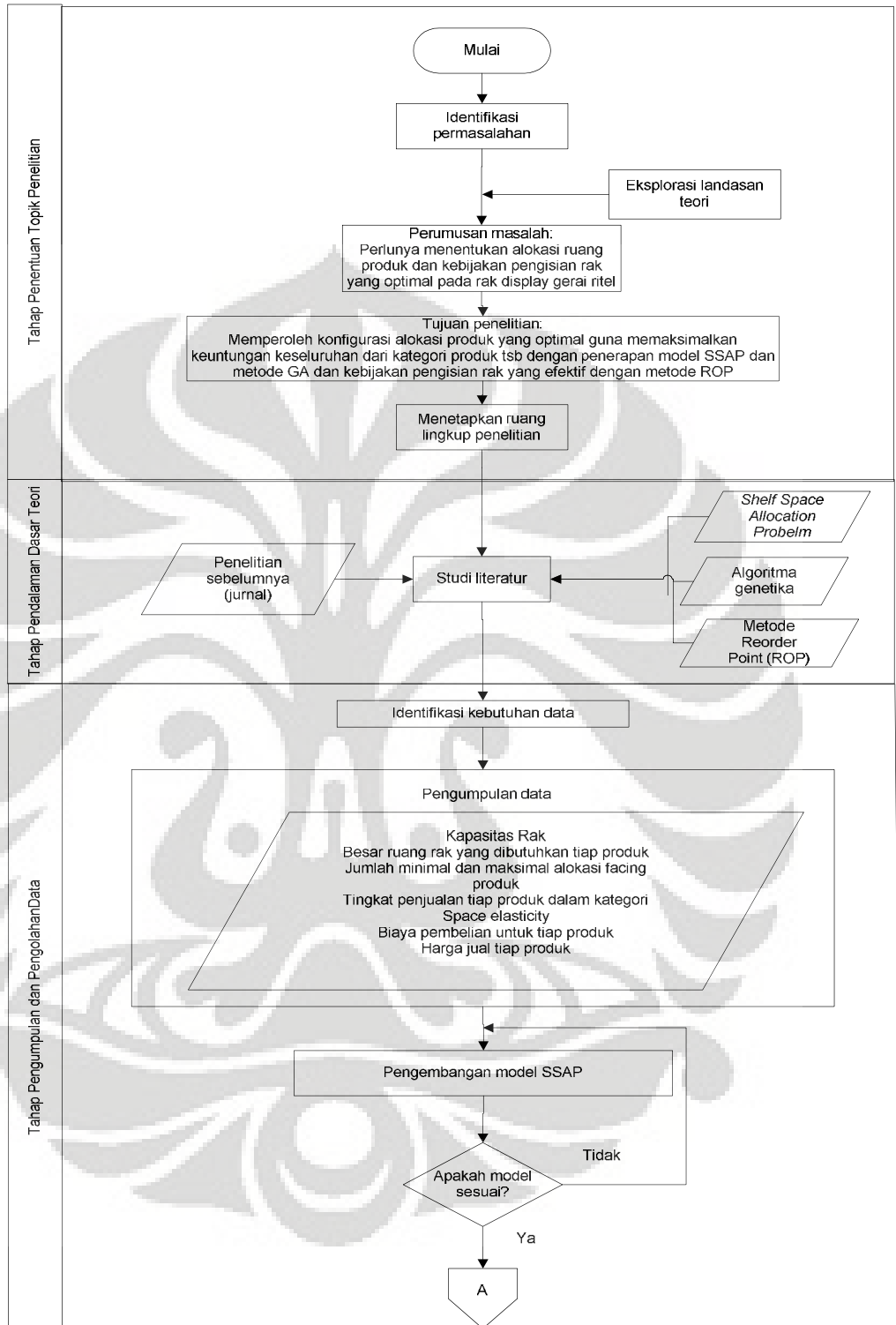
Ruang lingkup dari penelitian berfungsi untuk mengarahkan serta menentukan fokus utama mengenai masalah yang akan diteliti sesuai dengan apa yang direncanakan. Ruang lingkup serta asumsi yang digunakan dalam penelitian ini adalah sebagai berikut:

- penelitian akan difokuskan pada suatu grup kategori dengan melakukan optimasi alokasi ruang terhadap tiap jenis produk pada kategori tersebut
- kategori produk yang akan diteliti merupakan yang termasuk dalam kelas *unplanned* (tidak terencana) dalam kegiatan belanja seorang pelanggan, dimana pelanggan cenderung baru akan memutuskan merk produk apa yang akan dibeli ketika berhadapan dengan rak *display*
- analisis alokasi ruang dilakukan dari sisi manajemen ritel
- umur produk pada rak (*shelf life*) tidak mempengaruhi tingkat penjualan produk
- alokasi ruang yang akan diteliti dibatasi hanya pada jumlah *facing* untuk tiap produk, yaitu baris terdepan pada rak *display* yang dilihat langsung konsumen
- parameter harga serta unsur pemasaran lainnya dianggap tetap
- semua produk dimiliki oleh gerai ritel, dan oleh sebab itu keputusan mengenai alokasi ruang produk berada sepenuhnya pada manajemen gerai ritel
- efek kualitatif yang mencakup peletakan produk secara vertikal pada level rak tertentu maupun secara horizontal pada letak *aisle* tertentu tidak diperhatikan dalam penelitian ini, karena karakter-karakter tersebut membutuhkan studi lebih lanjut mengenai perilaku belanja konsumen dalam toko yang tidak dapat dicapai pada penelitian kali ini.

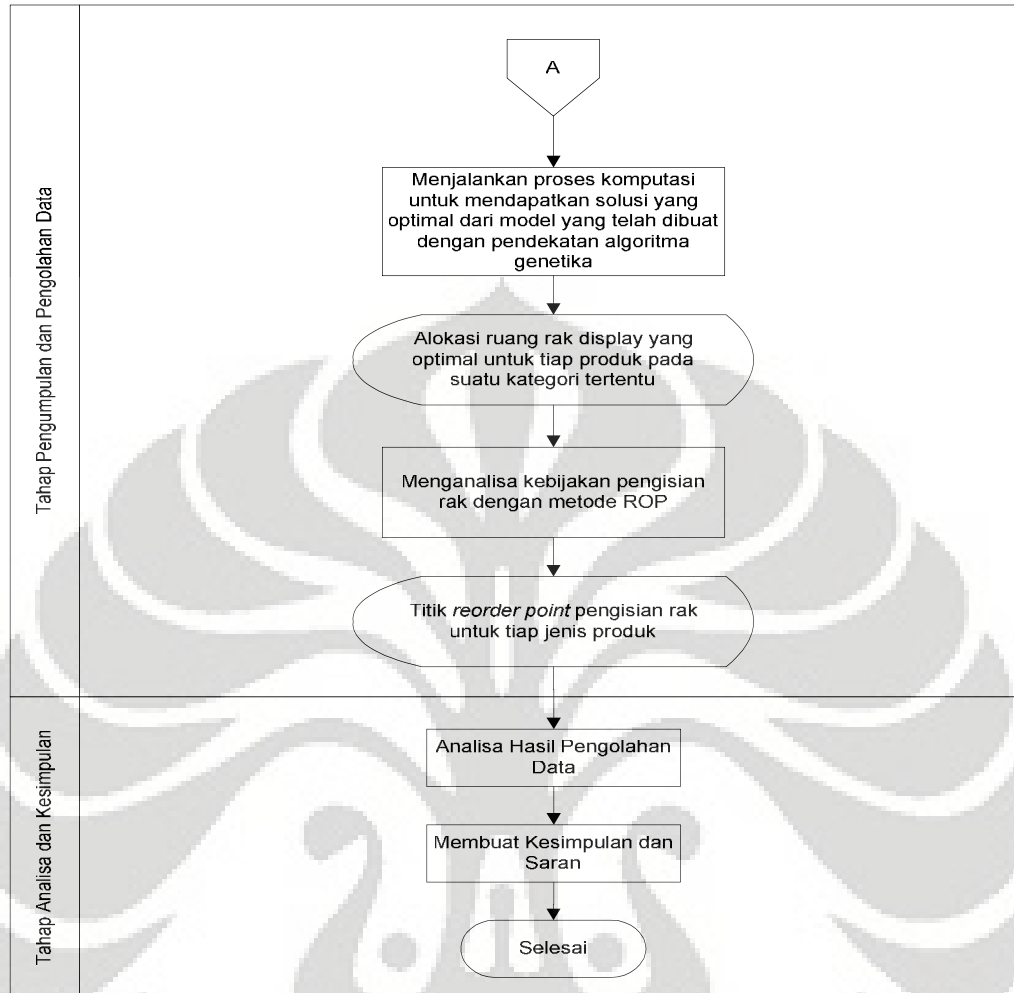
### 1.6 Metodologi Penelitian

Diagram alir metode penelitian dapat dilihat pada Gambar 1.2.





**Gambar 1.2** Diagram Alir Metode Penelitian



**Gambar 1.2** Diagram Alir Metode Penelitian (Lanjutan)

### 1. Penentuan topik penelitian

Pada tahap awal ini, akan ditetapkan topik utama yang akan menjadi fokus penelitian. Penentuan topik dilakukan berdasarkan latar belakang dari permasalahan yang akan diangkat dalam penelitian serta diskusi dengan dosen pembimbing serta pihak terkait lainnya. Topik dari penelitian ini adalah optimasi alokasi ruang *display* produk serta analisis kebijakan *in-store shelf replenishment* pada rak *display* gerai ritel *grocery*. Hasil akhir (*output*) serta ruang lingkup dari penelitian juga ditetapkan pada tahap awal ini agar penelitian dapat lebih terarah dan sesuai dengan yang diharapkan. Pada tahap ini juga telah dilakukan eksplorasi studi pustaka guna memberikan gambaran

umum mengenai dasar teori yang akan digunakan untuk mencari solusi permasalahan.

## 2. Pendalaman landasan teori

Tahap selanjutnya adalah melakukan pendalaman lebih lanjut mengenai dasar teori yang akan digunakan dalam penelitian. Landasan teori dapat berupa referensi jurnal dari penelitian-penelitian sebelumnya ataupun teori dasar dari metode-metode untuk proses pengolahan data. Adapun beberapa landasan teori yang terkait dengan penelitian ini adalah *Shelf Space Allocation Problem*, Algoritma Genetika, serta metode ROP.

## 3. Pengumpulan data

Tahap pengumpulan data merupakan tahap yang signifikan dalam penelitian. Tahap ini akan menentukan apakah penelitian dapat dilanjutkan dengan ketersediaan data yang ada. Penulis akan menentukan kebutuhan data terlebih dahulu dengan mengidentifikasi data apa yang diperlukan untuk menyelesaikan masalah ini. Secara umum, data yang dibutuhkan meliputi data kapasitas rak, besar ruang rak yang dibutuhkan untuk tiap jenis produknya, jumlah minimal dan maksimal produk yang harus distok pada rak *display*, *space elasticity*, biaya pembelian (*purchasing*), tingkat permintaan produk serta harga jual tiap produk.

## 4. Pengolahan data dan analisis

Setelah semua data yang dibutuhkan telah terkumpul, tahap selanjutnya adalah proses pengolahan data serta analisis terhadap hasil pengolahan data. Pengolahan data akan dimulai dengan membangun model SSAP yang sesuai dengan kasus permasalahan yang ada. Untuk menyelesaikan model matematis ini, akan dibuat sebuah program dengan software MATLAB untuk memudahkan proses komputasi data. Solusi optimal akan dikalkulasi dengan mengimplementasikan algoritma genetika pada model yang telah dibuat. Setelah solusi dari optimasi alokasi ruang telah diperoleh, maka dilakukan analisis kebijakan pengisian rak dengan metode ROP.

## 5. Penarikan kesimpulan

Setelah semua tahap telah dilakukan dengan baik, penarikan kesimpulan dilakukan berdasarkan analisis yang telah dibuat.

## 1.7 Sistematika Penelitian

Sistematika yang digunakan dalam penulisan penelitian ini mengikuti aturan standar baku penulisan tugas akhir skripsi. Penulisan tugas akhir penelitian ini terbagi ke dalam lima bab yang akan memberikan gambaran sistematis dari tahapan metodologi penelitian yang dilakukan sejak awal penelitian hingga tercapainya tujuan penelitian.

Bab pertama merupakan bab pendahuluan yang menjelaskan isi penelitian secara garis besar. Di dalam bab ini terdapat penjelasan mengenai latar belakang masalah, keterkaitan masalah dengan penelitian yang dilakukan, perumusan masalah, tujuan dan hasil yang diharapkan dari penelitian, ruang lingkup penelitian, metodologi penelitian, serta sistematika penulisan.

Bab kedua menjelaskan mengenai landasan teori yang digunakan dalam penelitian. Landasan teori akan dimulai dengan penjelasan umum mengenai industri ritel, kemudian difokuskan kepada permasalahan *shelf space allocation problem*, *space elasticity*, algoritma genetika, serta metode *replenishment*.

Bab ketiga secara umum berisi pengumpulan data yang dibutuhkan dalam melaksanakan penelitian. Proses pengumpulan data dilakukan dengan cara observasi, wawancara, dan pengumpulan dokumen di perusahaan.

Bab keempat merupakan penjelasan mengenai proses pengolahan data beserta analisisnya. Pengolahan data akan dimulai dengan penyusunan model matematis terkait dengan kasus yang dihadapi, kemudian menyelesaikan model tersebut menggunakan implementasi algoritma genetika menggunakan program komputer dengan MATLAB. *Output* yang diharapkan alokasi optimal ruang rak *display* untuk tiap produk pada suatu kategori produk. Selanjutnya, akan dilakukan analisis kebijakan pengisian rak *display* dengan menggunakan metode ROP.

Bab kelima merupakan bab kesimpulan dari pembahasan dan analisis secara keseluruhan dari apa yang telah dilakukan dalam penelitian ini.

## BAB 2 DASAR TEORI

### 2.1 Konsep *Retailing*

Secara harfiah kata ritel atau *retail* berarti eceran atau perdagangan eceran, dan peritel/ *retailer* diartikan sebagai pengecer atau pengusaha perdagangan eceran. Menurut kamus, kata *retail* ditafsirkan sebagai “*selling of goods and or services to the publics*”; atau “penjualan barang dan/ atau jasa kepada khalayak” (Manser, 1995). Dalam kaitannya dengan manajemen ritel, kata *retail* didefinisikan sebagai “*those business activities involved in the sale of goods and services to consumers for their personal, family, or household use*” atau “keseluruhan aktivitas bisnis yang menyangkut penjualan barang dan jasa kepada konsumen untuk digunakan oleh mereka sendiri, keluarga, atau rumah tangganya” (Berman dan Evans, 1992).

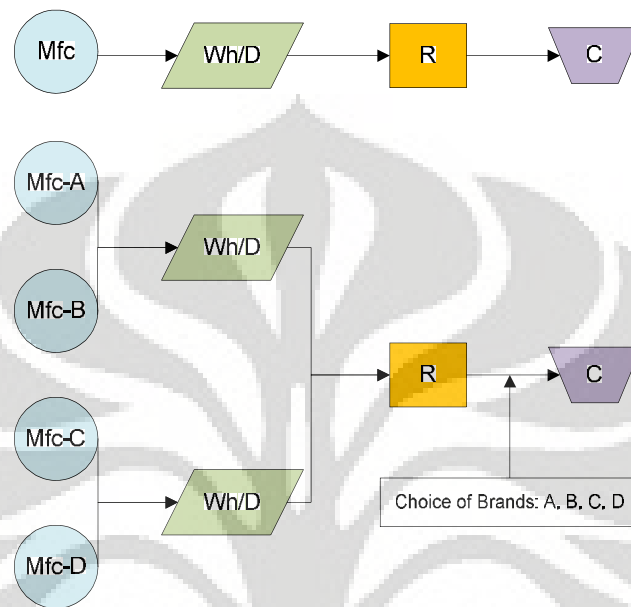
Dalam alur proses distribusi barang, bisnis ritel merupakan tahap akhir proses distribusi dengan dilakukannya penjualan langsung pada konsumen akhir. Gambar 2.1 memperlihatkan proses sederhana dalam distribusi barang/ jasa, dimana bisnis ritel merupakan suatu fungsi atau mata rantai proses distribusi sebagai perantara antara *distributor* (*wholesare* ataupun *importer*) dengan konsumen. *Retailer* juga berperan sebagai penghimpun berbagai kategori atau jenis barang yang menjadi kebutuhan konsumen, sehingga konsumen menjadikan toko ritel sebagai tempat rujukan untuk mendapatkan barang yang dibutuhkannya. Bisnis ritel juga merupakan penentu eksistensi barang dari *manufacturer* di pasar konsumsi, dan dengan demikian *manufacturer* dan *distributor* memiliki ketergantungan yang besar terhadap entitas bisnis ritel.

#### 2.1.1 Karakteristik Bisnis Ritel

Terdapat beberapa karakteristik suatu bisnis ritel yang membedakannya dengan entitas bisnis lainnya, sebagaimana terlihat pada Gambar 2.2. Karakteristik pertama adalah penjualan barang/ jasa dalam partai kecil, yaitu dalam jumlah secukupnya untuk dikonsumsi sendiri dalam periode waktu tertentu. Meskipun *retailer* mendapatkan barang dari *supplier* dalam bentuk kartonan

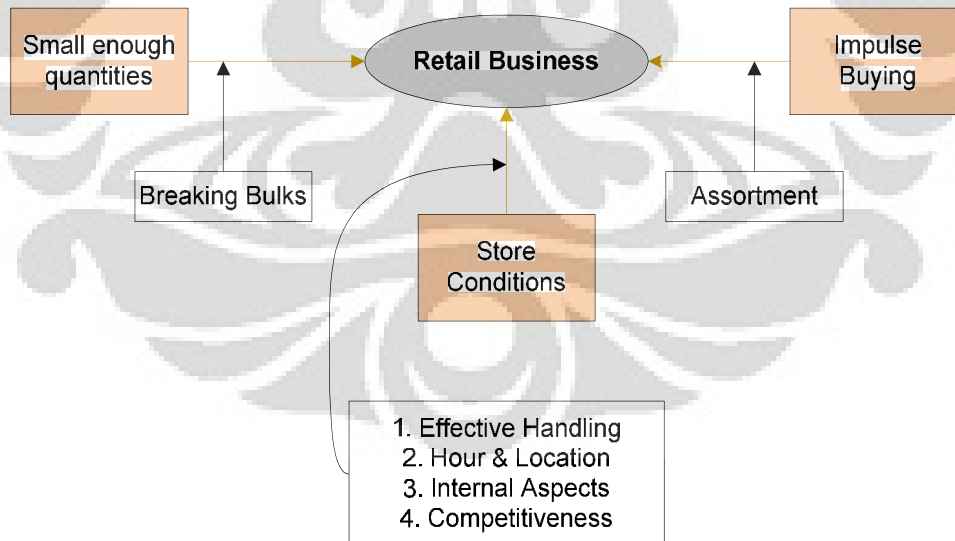
(cases), namun *retailer* men-*display* dan menjualnya dalam bentuk pecahan per unit.

### Fungsi Ritel dalam Distribusi



**Gambar 2.1** Fungsi Ritel dalam Distribusi

(Sumber: WPA ReSULTANT, 2002)



**Gambar 2.2** Karakteristik Bisnis Ritel

(Sumber: Berman & Evans, 1992)

Karakteristik kedua adalah “*impulse buying*,” yaitu kondisi yang tercipta dari ketersediaan barang dalam jumlah dan jenis yang sangat variatif sehingga menimbulkan banyaknya pilihan dalam proses belanja konsumen. Seringkali dalam proses belanja konsumen, keputusan yang diambil untuk membeli suatu barang adalah yang sebelumnya tidak tercantum dalam belanja barang (*unplanned purchases*). Keputusan ini muncul begitu saja terstimulasi oleh variasi bauran produk dan tingkat harga barang yang ditawarkan. Ciri penentu yang ketiga adalah *store condition* (kondisi lingkungan dan interior dalam toko) yang dipengaruhi oleh lokasi toko, efektivitas penanganan barang, *open hour* (jam buka toko), dan tingkat harga yang bersaing.

### 2.1.2 Berbagai Tipe Bisnis Ritel

Berdasarkan buku *Manajemen Ritel Modern* karangan Asep Sujana, bisnis ritel dapat diklasifikasikan berdasarkan (1) *ownership* (kepemilikan bisnis), (2) *merchandise category* (kategori barang dagangan), (3) luas area penjualan, dan (4) *non-store retailer* (peritel tanpa toko).

#### 2.1.2.1 Tipe Bisnis Ritel Atas Kepemilikan (Ownership)

- *Single-store Retailer*; merupakan tipe bisnis ritel yang paling banyak jumlahnya dengan ukuran toko umumnya di bawah 100 m<sup>2</sup>, mulai dari kios atau toko di pasar tradisional sampai dengan *minimarket* modern dengan kepemilikan secara individual.
- Rantai Toko Ritel; adalah toko ritel dengan banyak (lebih dari satu) cabang dan biasanya dimiliki oleh suatu institusi bisnis bukan perorangan, melainkan dalam bentuk perseroan (*company owned retail chain*). Bentuknya mulai dari rantai toko *minimarket* sampai dengan *hyperstore*.
- Toko Waralaba (*Franchise Store*); adalah toko ritel berdasarkan kontrak kerja waralaba (bagi hasil) antara terwaralaba (*franchisee*), yakni pengusaha investor perseorangan (*independent business person*) dengan pewaralaba (*franchisor*) yang merupakan pemegang lisensi toko.



### 2.1.2.2 Tipe Bisnis Ritel Berdasarkan Merchandise Category

- *Specialty Store* (Toko Khas); merupakan toko ritel yang menjual satu jenis kategori barang atau suatu rentang kategori barang (*merchandise category*) yang relatif sedikit. Contohnya adalah apotik, toko optik, galeri seni, toko perhiasan, toko buku, dan sebagainya.
- *Grocery Store* (Toko Serba Ada, Toserba); merupakan toko ritel yang menjual sebagian besar kategori barangnya adalah barang *groceries* (kebutuhan sehari-hari; *fresh-food, perishable, dry-food, beverages, cleanings, cosmetics, serta household items*). Contoh dari *grocery retailer* diantaranya Carrefour, Makro, Hero, Superindo dan sebagainya.
- *Department Store*; sebagian besar dari *assortments* yang dijual adalah merupakan bukan kebutuhan pokok (*non-basic items*), *fashionable* dan *branded items* dengan lebih dari 80% pola *consignment*.
- *Hyperstore*; menjual barang-barang dalam rentang kategori barang yang sangat luas, menjual hampir semua jenis barang kebutuhan setiap lapisan konsumen, mulai dari barang *grocery, household, textile, appliance, optical*, dan lainnya dengan konsep *one-stop-shopping*. Dibutuhkan setidaknya 10.000 m<sup>2</sup> luasan *sales area* untuk toko ritel jenis ini. Toko-toko ritel di Indonesia belum ada yang dapat dikategorikan dalam tipe *hyperstore*.

### 2.1.2.3 Tipe Bisnis Ritel Berdasarkan Luas Sales Area

- *Small Store/ Kios*; sebuah toko kecil (kios) yang umumnya merupakan toko ritel tradisional dioperasikan sebagai usaha kecil dengan *sales area* kurang dari 100m<sup>2</sup>.
- *Minimarket*; dioperasikan dengan luasan *sales area* antara 100 sampai dengan 1.000 m<sup>2</sup>.
- *Supermarket*; dioperasikan dengan luasan *sales area* antara 1.000 sampai dengan 5.000 m<sup>2</sup>.
- *Hypermarket*; dioperasikan dengan luasan *sales area* lebih dari 5.000 m<sup>2</sup>.

#### 2.1.2.4 Non-Store Retailer

- *Multi-Level-Marketing* (MLM); adalah suatu model penjualan barang secara langsung (*direct selling*) dengan sistem komisi penjualan berperingkat berdasarkan status keanggotaan dalam jalur distribusi.
- *Mail & Phone Order Retailer*; “toko pesan antar”, yakni perusahaan yang melakukan penjualan berdasarkan pesanan melalui surat dan/atau telepon.
- *Internet/Online Store (e-commerce)*; perkembangan teknologi informasi khususnya internet, telah memungkinkan berkembangnya “toko ritel” di dunia maya. Konsep yang diterapkan sama dengan *mail/phone order retailer*, yaitu melakukan penjualan berdasarkan pemesanan, namun pada *online store* pemesanan dilakukan dengan internet.

## 2.2 Manajemen Ruang *Display* (*Space management*)

Pada operasional gerai ritel *grocery* modern, *space management* merupakan bagian dari manajemen kategori produk. Manajemen kategori sendiri diartikan sebagai “suatu manajemen strategis dalam penanganan kelompok-kelompok barang tertentu melalui kerjasama perdagangan, yang bertujuan untuk memaksimalkan penjualan dan keuntungan dengan pemuasan kebutuhan-kebutuhan konsumen” (IGD.com, 2002). Dalam *space management* yang diterapkan saat ini, peritel modern selain menekankan pengaturan ruang-pajang atas dasar kontribusi penjualan tiap *item* barang, juga sangat menekankan aspek keserasian dan kepemimpinan merk dalam kategori (*captain category*).

Demi keserasian, dalam pemajangan barang senantiasa diperhatikan faktor-faktor keberagaman rasa, ukuran, dan warna. Sedangkan aspek kepemimpinan merk dalam kategori barang diperhatikan sebagai upaya untuk menarik konsumen mendatangi area pemajangan suatu kategori dengan menonjolkan pemajangan merek tertentu yang dapat mengasosiasikan kelompok kategori barang tersebut. Pada bagian inilah para *supplier* dapat mengambil manfaat dari kategorisasi dalam pemajangan barang. Kerjasama manajemen ruang-pajang dengan pihak peritel dirancang dan dilakukan untuk mendapatkan peluang ekspose terbesar dalam ruang pajang. Ekspose yang lebih besar dalam ruang pajang akan memberikan kesan kepemimpinan merk dalam kategori. Lebih

jauh lagi, tentunya hal ini akan membantu memperkuat citra merk produk tersebut.

### 2.3 Kategori Produk Berdasarkan Hubungan *Sales-Space Elasticity*

Brown dan Tucker (1961) menyatakan bahwa hubungan antara *sales-shelf elasticity* telah menciptakan tiga kelas produk, yang terdiri dari:

- *Unresponsive products* – barang-barang komoditas seperti gula dan garam;
- *General use products* – barang berjenis *staples* seperti produk untuk sarapan ataupun buah kalengan;
- *Occasional purchase products* – barang yang dibeli akibat adanya aspek ‘*impulse buying*’, contohnya seperti permen, minuman ringan, dan *snack* (Dreze et. al., 1994. p. 9).

‘*Impulse buying*’ didefinisikan sebagai fungsi dari faktor tempat (toko) dalam proses pembuatan keputusan untuk membeli (*in-store decision making*) (Leblanc-Maridor, 1989). ‘*Impulse buying*’ secara garis besar diukur berdasarkan perbedaan antara hasil belanja yang dilakukan seseorang dengan *intention* yang dimiliki sebelum memasuki toko, serta melalui kuesioner dengan pertanyaan untuk setiap barang yang dibeli, ‘kapan anda memutuskan untuk membeli produk ini?’ (Du Pont de Nemours, 1965; Point-of-Purchase Advertising Institute, 1963; Bellenger et. al., 1978). Dari penelitian ini disimpulkan bahwa produk-produk yang dibeli diluar niat sebelumnya yang dimiliki seseorang, atau juga yang disebut *unplanned purchases*, merupakan hasil dari adanya proses ‘*impulse buying*.’ Oleh karena itu, objek produk yang akan digunakan dalam penelitian kali ini adalah yang termasuk ke dalam kategori ‘*occasional purchase product*.’

### 2.4 *Shelf Space Allocation Problem (SSAP)*

Permasalahan alokasi ruang pada rak *display* gerai ritel *grocery* merupakan problema nyata yang dihadapi oleh banyak perusahaan ritel. Masalah muncul ketika terdapat jenis produk dalam jumlah besar untuk ditampilkan, namun ruang rak yang tersedia sangat terbatas. Penyelesaian permasalahan ini dengan menggunakan model *Shelf Space Allocation Problem (SSAP)* meliputi

pendistribusian ruang rak yang terbatas terhadap berbagai jenis produk yang dimiliki gerai ritel *grocery*.

Selama kurun waktu 50 tahun terakhir, variasi dari keberadaan produk telah meningkat secara dramatis, dan akan terus berlanjut untuk memenuhi berbagai keberagaman kebutuhan konsumen. Keberagaman ini dapat dikarenakan adanya berbagai fungsi, merk, jenis, warna, material dan bahkan ukuran, serta juga faktor-faktor lainnya. Walaupun supermarket secara berkala meningkatkan ukuran gerai mereka, proporsi peningkatan ini masih jauh lebih kecil daripada peningkatan variasi produk. Hal ini menimbulkan tantangan nyata bagi kebanyakan supermarket dalam menerapkan tata letak produk yang efektif sehingga beberapa tujuan dapat terpenuhi, seperti memaksimalkan keuntungan atau penjualan, meminimalkan biaya operasional, memaksimalkan kesetiaan pelanggan dsb.

Dalam prakteknya, alokasi ruang, dalam sebuah gerai ritel *grocery*, terbagi menjadi dua tingkatan: alokasi ruang antar kategori dan alokasi ruang untuk tiap SKU. Alasan dari pembagian ini adalah:

- Sebuah supermarket biasanya memiliki ribuan jenis produk. Kategori yang berbeda dapat memiliki kondisi dan ketentuan *display* yang berbeda. Menyelesaikan masalah untuk semua produk akan sangat tidak realistis, tidak hanya dikarenakan kesulitan dalam mengembangkan model yang cocok untuk semua produk dari berbagai kategori namun juga kebutuhan kemampuan komputasi yang tinggi.
- Mengelompokkan produk yang sejenis dalam suatu kategori memudahkan konsumen untuk saling membandingkan sebelum menentukan pilihan
- Manajemen kategori merupakan metode yang umum digunakan gerai ritel, terutama supermarket yang biasanya dikelompokkan ke dalam departemen, kategori, *brand*, dan *stock-keeping unit* (SKU) (Michael Levy dan Barton Weitz, 1992).

Terdapat beberapa taktik manipulasi yang dapat dilakukan untuk menstimulasi *demand*, yaitu perancangan *traffic flow* dalam gerai, lokasi penempatan sejumlah kategori dan tipe *brand*, alokasi ruang untuk tiap kategori

dan produk, promosi pada *point-of-sale* (POS) dan *display* khusus (Francis Buttle, 1984). *Retailer* pertama kali harus merancang *traffic flow* konsumen yang diarahkan dengan penempatan rak-rak yang sesuai sehingga setiap bagian gerai memiliki *exposure* yang maksimal. Di saat rak-rak tersebut telah ditempatkan dalam gerai, *retailer* harus menentukan alokasi ruang untuk berbagai kategori dan SKU dalam tiap kategori.

#### 2.4.1 Tujuan Umum Permasalahan Alokasi Ruang *Display*

Tujuan utama dari alokasi ruang rak *display* adalah untuk meminimalkan biaya operasional ataupun memaksimalkan tingkat keuntungan. Untuk meminimalkan biaya dapat digunakan model EOQ (*Economic Order Quantity*) dimana tingkat permintaan sebuah produk terbilang konstan dan alokasi ruang tidak mempengaruhi tingkat permintaan. Namun, jika tingkat permintaan sebuah produk tergantung pada variabel keputusan dari besarnya ruang yang dialokasikan, fungsi tujuan meminimalisasi biaya menjadi kurang tepat karena meminimalisasi biaya akan menghasilkan penurunan tingkat penjualan dan keuntungan karena model yang digunakan akan berusaha untuk mengurangi jumlah *facing* produk dengan tujuan untuk mengurangi biaya. Sebagai contoh ekstrimnya, ketika tidak ada ruang rak yang dialokasikan, biaya akan minimal. Namun, secara jelas tidak akan ada barang yang bisa dijual ketika barang tersebut tidak terdapat pada rak *display*. Oleh karena itu, memaksimalkan keuntungan agregat dipilih menjadi tujuan dalam permasalahan alokasi ruang rak *display* pada penelitian ini.

#### 2.4.2 Variabel Keputusan dalam SSAP

*Facing* dan lokasi adalah dua variabel keputusan yang sangat umum dalam permasalahan alokasi ruang rak *display*.

##### 2.4.2.1 Facing

*Facing* merupakan variabel yang sangat penting dalam alokasi ruang rak *display*. Jumlah *facing* dari sebuah SKU adalah banyaknya kuantitas sebuah barang yang dapat dilihat langsung pada rak oleh konsumen. Barang yang

diletakkan di belakang barang lainnya tidak dapat dilihat secara langsung dan oleh karenanya tidak dianggap sebagai *facing*. Banyaknya jumlah *facing* untuk suatu produk pada rak *display* dipercaya dapat mempengaruhi pembelian yang dilakukan konsumen. Riset telah menemukan bahwa lebih dari 33% dari pembelian bersifat tidak terencana (Buttle, 1984). Produk dengan *exposure* yang lebih baik memiliki peluang yang lebih besar untuk dibeli oleh konsumen. Namun, besarnya ruang rak yang dialokasikan dapat memiliki dampak yang berbeda terhadap penjualan satu produk dengan produk lainnya.

*Space elasticity* biasanya digunakan untuk mengukur tingkat *responsiveness* dari penjualan terhadap perubahan ruang rak yang dialokasikan. Curhan (Curhan, 1972) mendefinisikan *space elasticity* sebagai “*the ratio of relative change in unit sales to relative change in shelf space,*” atau rasio perubahan relatif penjualan unit barang terhadap perubahan relatif pada ruang rak *display*.

#### 2.4.2.2 Lokasi

Lokasi merupakan variabel lainnya yang dapat mempengaruhi tingkat permintaan dari sebuah produk. Secara umum, tingkatan rak pada *eye-level*, rak di akhir *aisle*, dan pada lokasi masuk gerai dipercaya sebagai posisi lokasi terbaik yang

Terdapat beberapa variabel *marketing* lainnya yang dapat digunakan untuk menstimulasi penjualan, seperti *advertising*, promosi, diskon dsb. Investigasi mengenai isu-isu ini merupakan di luar ruang lingkup dari penelitian yang akan dilakukan. Penelitian ini akan memfokuskan diri pada variabel *facing* dan lokasi.

#### 2.4.3 Kendala (*Constraint*) dalam SSAP

Terdapat beberapa kendala potensial dalam permasalahan SSAP. Meskipun gerai yang berbeda dapat memiliki kebutuhan *display* yang berbeda dan memiliki kendala yang berbeda pula, namun ada beberapa jenis kendala yang terbilang umum.



#### 2.4.3.1 Kendala Fisik

Kendala fisik merupakan kendala yang diaplikasikan pada setiap permasalahan SSAP. Total volume dari produk-produk yang dialokasikan pada sebuah rak tidak dapat melebihi total ruang rak yang tersedia. Kendala ini dapat bersifat satu dimensi (tidak menganggap adanya faktor ketinggian dan kedalaman rak) atau dua dimensi (tidak mengabaikan faktor kedalaman). Faktor kedalaman biasanya tidak dianggap karena kedalaman rak akan jauh lebih besar dari lebar SKU. *Retailer* akan menempatkan barang sebanyak-banyaknya di belakang produk yang berada pada baris depan untuk mengurangi jumlah waktu pengisian kembali, namun, keberadaan stok dibelakang *facing* tidak memiliki efek pada fungsi permintaan. Kendala ketinggian rak juga dapat tidak dianggap untuk beberapa jenis barang, seperti contohnya ketika menempatkan barang bertumpukan dengan barang lainnya tidak diperbolehkan (botol susu, botol *wine*). Selain itu, di banyak gerai ketinggian rak dapat disesuaikan. Hal ini dapat menyelesaikan permasalahan ketika tinggi sebuah produk melebihi ketinggian rak. Kendala fisik merupakan kendala yang bersifat *hard constraint*, yang artinya: pelanggaran terhadap kendala ini akan menghasilkan solusi yang tidak tepat (*infeasible*)

#### 2.4.3.2 Kendala Integrality

Besarnya ruang rak yang dialokasikan untuk sebuah produk haruslah merupakan perkalian integer dari ukuran produk tersebut, yang biasanya diukur dengan *facing*. Ini juga merupakan *hard constraint* dan harus dipenuhi. Tidaklah masuk akal untuk mengalokasikan 1,5 *facing* untuk sebuah produk.

#### 2.4.3.3 Ketentuan Display

Banyak *retailer* yang menentukan batas bawah dari banyaknya *facing* yang harus dialokasikan untuk sebuah produk untuk memastikan adanya *exposure* yang cukup bagi konsumen. Batas atas juga ditekankan agar banyaknya *facing* yang dialokasikan masih termasuk dalam jumlah yang wajar. Hal ini dilakukan agar tidak ada produk yang terlalu mendominasi dalam rak *display* tersebut.

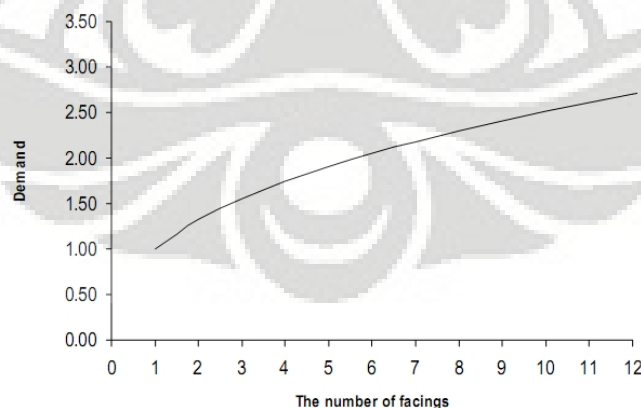


#### 2.4.4 Model Optimasi Permasalahan *Shelf Space Allocation Problem*

Telah banyak riset sebelumnya yang dilakukan dalam mengembangkan model optimasi *Shelf Space Allocation Problem* (SSAP). Penelitian ini akan mengimplementasikan model optimasi yang dikembangkan oleh Ruibin Bai, Beng, MSc. dalam karyanya “*An Investigation of Novel Approaches for Optimizing Retail Shelf Space Allocation.*” Model acuan yang digunakan merupakan model simplifikasi dari model kompleks SSAP.

##### 2.4.4.1 Model Kompleks SSAP

Hal utama dalam SSAP adalah mendefinisikan fungsi permintaan (*demand*) yang menampilkan hubungan antara besar ruang *display* dan banyaknya penjualan yang dapat diperoleh. Fungsi polynomial dengan *diminishing return* telah banyak digunakan oleh beberapa peneliti dalam literatur (Zufryden, 1986; Baker dan Urban, 1988; Dreze et al., 1994; Urban, 2002) untuk menjelaskan hubungan antara *display facing* dari sebuah produk dengan tingkat penjualan (permintaan) produk tersebut. Gambar 2.3 menampilkan sebuah ilustrasi dari fungsi ini. Dapat dilihat bahwa *demand* dari produk tersebut bertambah secara kontinu dengan adanya penambahan jumlah *facing* yang dialokasikan untuk produk tersebut. Namun, tingkat pertumbuhan akan melambat secara perlahan (*diminishing return*).



**Gambar 2.3** Ilustrasi Tingkat Penjualan dengan Mempertimbangkan Faktor Alokasi Ruang Rak Produk

(Sumber: Bai, 2005)

Model komprehensif telah dikembangkan dalam penelitian yang dilakukan Chen dan Yang (1999) dan Urban (2002), dimana fungsi *demand* diformulasikan dalam bentuk sebagai berikut:

$$= ( ) \quad (2.1)$$

dimana

- $n$  adalah banyaknya produk yang akan ditampilkan;
- $m$  adalah banyaknya jumlah rak untuk ditempatkan;
- $D_i$  adalah fungsi *demand* dari produk  $i$ ;
- $x_{ik}$  adalah banyaknya jumlah *facing* untuk produk  $i$  yang dialokasikan pada rak  $k$ ;
- $x_i$  adalah total alokasi *facing* dari sebuah produk,  $x_i = \sum_{k=1}^m (x_{ik})$ ;
- $\alpha_i$  adalah skala parameter penjualan untuk produk  $i$ ;
- $\beta_{ik}$  adalah *space elasticity* dari produk  $i$  pada rak  $k$ ;
- $\gamma_{iu}$  adalah *cross elasticity* antara produk  $i$  dan produk  $u$ ;
- $y_{it}$  adalah variabel pasar ke  $t$  ( $t = 1, \dots, R$ ) yang dapat mempengaruhi *demand*. Beberapa variabel pasar diantaranya harga, *advertising campaign*, kegiatan promosi dll.;
- $\delta_{it}$  adalah *elasticity value* dari variabel market ke- $t$  terhadap *demand value* dari produk  $i$ .

Tujuan dari model optimasi ini adalah untuk memaksimalkan total keuntungan yang sama dengan total keuntungan kotor dari penjualan dikurangi dengan biaya agregat untuk mencapai penjualan tersebut.

$$\max = - ( )^n \quad (2.2)$$

$$\text{subject to } \leq = 1, \dots, \quad (2.3)$$

$$\leq \leq = 1, \dots, \quad (2.4)$$

$$\in \{0, 1, 2, 3, \dots\} = 1, \dots, = 1, \dots, \quad (2.5)$$

dimana  $g_i$  adalah pendapatan per unit dari produk  $i$  dan  $\theta_i$  adalah koefisien biaya dari produk  $i$  dan  $\eta_i$  adalah elastisitas biaya dari produk  $i$  dengan pengaruhnya terhadap volume penjualan.  $T_j$  adalah kapasitas dari rak  $j$  dan  $l_i$  adalah ukuran *facing* dari produk  $i$ . Kendala pertama memastikan bahwa kendala keterbatasan kapasitas rak terpenuhi. Kendala kedua membatasi batas atas  $U_i$  dan batas bawah  $L_i$  dari banyaknya *facing* untuk produk  $i$  dan kendala ketiga merupakan ketentuan *integrality* untuk variabel *shelf space allocation*.

#### 2.4.4.2 Simplifikasi Model SSAP

Dalam penelitian ini, digunakan simplifikasi model SSAP yang dikembangkan oleh Bai Rubin (2005). Untuk menyederhanakan permasalahan SSAP ini, akan diasumsikan bahwa:

- 1) Total keuntungan untuk produk  $i$  adalah proporsional terhadap keuntungan per unit  $p_i$  (sama dengan  $\eta_i = 1$ ).
- 2) Nilai *cross elasticities* antar produk yang menunjukkan adanya pengaruh penjualan suatu produk terhadap alokasi ruang untuk produk lainnya memiliki nilai yang lebih kecil dari nilai *space elasticity* langsung, dan oleh sebab itu dapat diabaikan.
- 3) *Shelf-life* dari produk tidak mempengaruhi tingkat *demand* produk tersebut
- 4) Variabel pasar diabaikan dan dianggap tetap untuk tiap produknya.
- 5) Nilai *space elasticity* dianggap sama untuk tiap level rak, sehingga nilai *space elasticity* yang berlaku adalah untuk keseluruhan rak yang ada.

Penyamaan nilai *space elasticity* ini menyebabkan limitasi dari model simplifikasi ini, dimana solusi yang dihasilkan hanya terbatas pada jumlah alokasi *facing* untuk tiap produk, dan tidak mencapai alokasi penempatannya pada rak *display*.

Dengan notasi yang sama seperti yang dipaparkan pada sub bab sebelumnya, simplifikasi model SSAP berdasarkan asumsi-asumsi yang dibuat diekspresikan sebagai berikut:

$$\max \quad = \quad (2.6)$$

$$\text{subject to} \quad \leq \quad (2.7)$$

$$\leq \leq \quad (2.8)$$

$$\in \{0,1,2,3 \dots\} \quad = 1, \dots, \quad (2.9)$$

Variabel keputusannya merupakan  $x_i$ , yang merepresentasikan banyaknya *facing* dari produk  $i$  pada rak.  $\alpha_i$  adalah skala parameter dan  $\alpha_i > 0$ .  $\beta_i$  adalah nilai *space elasticity* dan  $0 \leq \beta_i \leq 1$ . Tujuan dari model ini adalah untuk memaksimalkan keseluruhan keuntungan tanpa melanggar kendala yang ada. Model ini merupakan permasalahan optimasi *non-linear* dan *multi-constraint*.

## 2.5 Algoritma Genetika

*Shelf Space Allocation Problem* (SSAP) memiliki kesamaan dengan *bin packing* dan *knapsack problem*, yang bersifat NP-Hard. Sampai saat ini tidak ada algoritma *polynomial-time bounded* yang diketahui yang dapat menyelesaikan tiap jenis permasalahan SSAP hingga mencapai titik optimalnya. Ketika menyelesaikan sebuah permasalahan optimasi, kita harus berusaha untuk menggunakan metode eksak untuk menemukan solusi optimal dari permasalahan tersebut. Namun, seperti yang telah dipaparkan sebelumnya, beberapa problem sangatlah sulit hingga algoritma untuk menemukan sebuah solusi optimal memiliki kompleksitas *exponential time*. Ketika dihadapkan pada permasalahan ukuran besar atau medium, maka akan sangat sulit untuk dapat melakukan komputasinya. Pada situasi seperti ini, kita dapat mengarah ke pendekatan perkiraan yang dapat menyelesaikan permasalahan dengan kualitas solusi yang memuaskan dengan waktu komputasi yang wajar. Pendekatan *heuristic* dan *metaheuristic* biasanya akan digunakan untuk mencapai tujuan ini.

Algoritma Genetika (AG) merupakan bentuk metode *metaheuristic* yang bersifat *population-based*. AG pertama kali diperkenalkan oleh Fraser dan Bremermann. Ide awal AG berasal dari prinsip seleksi alam *survival of the fittest*, yang mempercayai bahwa hanya individu yang terbaiklah yang akan bertahan melewati banyak generasi. Sejak diperkenalkan, AG telah banyak diaplikasikan

pada banyak problem optimasi dengan hasil yang dapat diterima. AG biasa digunakan pada permasalahan tata letak pada sebuah fasilitas maupun permasalahan alokasi *customer* pada gudang yang diterapkan. Selain itu, penelitian sebelumnya mengenai SSAP juga telah menggunakan AG dalam penyelesaian masalahnya (Hwang et. al., 2005; Timothy Urban, 1998). Dengan melihat banyaknya aplikasi AG yang berhasil diterapkan permasalahan optimasi yang serupa dengan SSAP, dan referensi dari penelitian yang telah ada, maka algoritma genetika diajukan untuk menghadapi *Shelf Space Allocation Problem* ini.

Algoritma genetika menerapkan populasi solusi yang berevolusi dari satu generasi ke generasi berikutnya. Untuk sebuah permasalahan optimasi, sebuah solusi (individu) dikodekan pada sebuah *string* yang disebut kromosom. Sejumlah individu, atau disebut populasi, dipelihara dan akan berevolusi dari satu generasi ke generasi lainnya. Sebuah populasi baru dihasilkan dengan menyalin beberapa individu yang lebih baik dari lainnya dalam sebuah populasi dan menyeleksi beberapa individu yang baru dibuat menggunakan operasi genetik, seperti *crossover* dan mutasi. Algoritma akan berhenti ketika kriteria terminasi terpenuhi. Untuk mengimplementasikan sebuah algoritma genetika, dibutuhkan keputusan mengenai skema *encoding*, operator (*crossover*, mutasi), dan sebuah metode seleksi serta beberapa nilai parameter, seperti ukuran populasi, banyaknya generasi, probabilitas *crossover*, dan probabilitas mutasi.

### 2.5.1 *Encoding*

Tahap pertama dalam mengembangkan algoritma genetika adalah dengan mengkodekan solusi ke dalam sebuah *finite-length string* yang disebut kromosom. Untuk permasalahan SSAP ini,  $X_i$  direpresentasikan sebagai 0-1 *binary digit* dan sebuah kromosom dikodekan sebagai  $(X_1, \dots, X_I)$ .

### 2.5.2 Inisialisasi

Tahap selanjutnya adalah pembuatan inisial populasi. Proses ini dapat dilakukan dengan membangkitkan secara random maupun penyelesaian awal menggunakan metode heuristik. Pada riset ini, set populasi inisial dihasilkan

secara random, dimana sebuah gen  $X_i$  dari kromosom dipilih secara random dalam interval  $[X_i^{min}, X_i^{max}]$ .

### 2.5.3 Crossover dan Mutasi

Kromosom anak biasanya dihasilkan dari populasi yang ada saat ini (kromosom induk) dengan menggunakan dua operasi genetik yang banyak digunakan, *crossover* dan mutasi. *Crossover* memainkan peran dalam pertukaran informasi antar kromosom. Hal ini biasanya mengarah ke sebuah kombinasi efektif dari solusi *partial* dan karenanya mengakselerasi prosedur pencarian di generasi awal. Salah satu metode dalam proses kawin silang yang juga digunakan pada penelitian kali ini adalah metode “*one-cut-point*”, dimana satu “*cut-point*” dipilih secara acak dan pertukaran dilakukan pada bagian kanan dua kromosom induk. Metode ini merupakan metode kawin silang yang paling sederhana. Untuk kromosom yang sangat panjang, misalkan 1000 gen, mungkin saja diperlukan beberapa titik potong. Pindah silang lebih dari satu titik potong disebut *n-point crossover*, dimana  $n$  titik potong dipilih secara random dan bagian-bagian kromosom dipilih dengan probabilitas tertentu dari salah satu orang tuanya. Skema pindah silang yang lain adalah *uniform crossover*, yang merupakan kasus khusus dari *n-point crossover* dimana  $n$  sama dengan jumlah gen dikurangi satu.

Proses mutasi dalam AG sangat sederhana. Untuk semua gen yang ada, jika bilangan random yang dibangkitkan kurang dari probabilitas mutasi  $P_{mut}$  yang ditentukan maka nilai gen akan diubah menjadi nilai kebalikannya (dalam *binary encoding*, 0 diubah 1 dan 1 diubah 0). Nilai  $P_{mut}$  adalah tetap selama proses evolusi.

### 2.5.4 Penanganan Kendala

Problem utama dari implementasi algoritma genetika untuk optimisasi berkendala adalah bagaimana menangani set kendala yang ada karena operator genetik yang digunakan untuk memanipulasi kromosom seringkali menghasilkan kromosom anak yang *infeasible*. Pada studi ini, digunakan strategi penalti dimana penalti akan dikenakan pada kromosom berdasarkan derajat pelanggaran kendala.



### 2.5.5 *Fitness Function* dan Mekanisme *Scaling*

Dalam algoritma genetika, *fitness* dikomputasi untuk tiap kromosom dalam populasi dan bertujuan untuk menemukan kromosom dengan nilai *fitness* maksimum. Pada penelitian ini, nilai total keuntungan produk akan digunakan sebagai nilai *fitness*. Karena performansi dari AG sangat sensitif terhadap nilai *fitness*, sangatlah penting untuk menskalakan *fitness* tidak hanya untuk menghindari *premature convergence* tapi juga untuk mendiversifikasi populasi.

### 2.6 Kebijakan *in-Store Shelf Replenishment* dengan *Reorder Point Model*

Metode *reorder point model* (ROP) merupakan bentuk metode umum yang digunakan dalam menentukan titik pengisian kembali untuk persediaan sebuah produk. Pengendalian persediaan dengan metode ROP mengasumsikan bahwa *demand* bersifat kontinu untuk setiap periode waktunya dan secara perlahan akan mengurangi tingkat persediaan yang ada. Ketika tingkat persediaan berkurang hingga sama dengan atau kurang dari suatu *level* tertentu yang disebut *reorder point*, maka sejumlah unit produk ( $Q^*$ ) akan didatangkan dari sumber *supply* untuk mengisi kembali persediaan yang kosong. Pada permasalahan persediaan produk pada rak *display* gerai ritel *grocery* digunakan model ROP untuk tingkat *demand* tidak pasti. Hal ini dikarenakan *demand* produk pada gerai ritel *grocery*, terutama produk yang bersifat *unplanned purchases*, cenderung bersifat fluktuatif tergantung dari waktu penjualan. Tingkat penjualan produk pada periode awal minggu akan cenderung lebih tinggi daripada penjualan produk pada akhir minggu karena volume kedatangan pelanggan yang lebih tinggi pada akhir pekan. Hal yang sama juga dijumpai pada periode awal bulan dan akhir bulan.

Pada metode ROP untuk *uncertain demand* ini digunakan nilai standar deviasi dari distribusi tingkat *demand* pada satu periode *lead time*. Standar deviasi untuk frekuensi distribusi *Demand During Lead Time* (DDLT) diperoleh dengan persamaan (Ballou, 2003):

$$s_{ddlt} = s_d \sqrt{LT} \quad (2.10)$$

Dengan  $s_d$  merupakan standar deviasi dari distribusi *demand* pada suatu periode tunggal dan  $LT$  adalah *lead time* yang dibutuhkan untuk mendatangkan produk.



Titik *reorder point* kemudian ditentukan dengan persamaan sebagai berikut.

$$= \quad + \quad ( \quad ) \quad (2.11)$$

Keterangan: ROP = *reorder point*

$d$  = rata-rata penjualan harian

$LT$  = waktu yang dibutuhkan untuk pengiriman (*lead time*)

$s'_d$  = standar deviasi selama periode penjualan ditambah  $LT$

$s_d$  = standar deviasi distribusi tingkat penjualan harian

$z$  = angka standar deviasi dari distribusi tingkat penjualan yang menunjukkan probabilitas barang dalam keadaan *in-stock* selama periode *lead time* (P)

Pada permasalahan *in-store shelf replenishment* ini tidak digunakan persamaan EOQ untuk menentukan nilai ekonomis jumlah produk yang harus dipesan ( $Q^*$ ). Hal ini dikarenakan pada level *in-store*, produk hanya dipindahkan dari gudang penyimpanan (*backroom*) ke ruang *display* (*frontroom*), sehingga estimasi biaya, baik *ordering cost* maupun *carrying cost* tidak perlu dilakukan. Sebagai alternatifnya, dalam menentukan  $Q^*$ , digunakan kebijakan *full-shelf merchandising policy* yang diterapkan oleh banyak manajemen gerai ritel *grocery*, dimana pengisian rak *display* diupayakan untuk dapat memenuhi seluruh alokasi ruang rak yang tersedia. Oleh karena itu, nilai  $Q^*$  akan ditentukan dengan mempertimbangkan titik ROP dan alokasi ruang *display* untuk tiap produknya.

### BAB 3

#### PENGUMPULAN DATA

Data untuk penelitian optimais alokasi ruang produk pada rak *display* gerai ritel ini diperoleh dari data sekunder salah satu gerai ritel terbesar di Indonesia, yaitu *hypermarket* Hypermart cabang Depok yang dimiliki oleh PT. Matahari Putra Prima Tbk.

#### 3.1 Pemilihan Jenis Produk

Dari kategori '*occasional purchase products*' yang telah dijelaskan pada bab sebelumnya, penelitian ini akan menggunakan produk minuman ringan kemasan sebagai objek studi kasus *Shelf Space Allocation Problem* (SSAP). Dari data historis yang diperoleh dari salah satu cabang gerai ritel Hypermart, proporsi penjualan dalam kategori produk ini dapat dilihat pada Tabel 3.1.

**Tabel 3.1** Proporsi Penjualan Kategori Produk Minuman Ringan

Produk	Total Penjualan	Proporsi
TEH KEMASAN BOTOL	11.143	35,61%
MILK-RELATED	920	2,94%
TEH KALENG	16	0,05%
TEH TETRAPACK	4.439	14,19%
COFFEE-DRINK	431	1,38%
JELLY DRINK	233	0,74%
ENERGY DRINK	5.964	19,06%
SODA DRINK	8.144	26,03%

(Sumber: Data Penjualan Hypermart Bulan Maret 2010)

Dari data penjualan pada tabel dapat dilihat bahwa dari 8 jenis produk yang ada dalam kategori produk minuman ringan, produk teh kemasan botol memiliki proporsi penjualan terbesar. Oleh karena itu, kita memperkecil lingkup penelitian ke dalam sub-kategori produk minuman teh kemasan botol.

Dalam kategori produk teh kemasan botol ini terdapat 28 jenis *Stock Keeping Units* (SKU) yang tersedia di gerai saat ini.

**Tabel 3.2** Data SKU Sub-kategori Minuman Teh Kemasan Botol

NO.	PRODUK
1	FRESTEA FRUTCY APEL 500 ML
2	FRESTEA FRUTCY LMN 500 ML
3	FRESTEA FRUTCY MARKISA 500 ML
4	FRESTEA GREEN MY BODY ALOEVERA & ORANGE
5	FRESTEA GREEN MY BODY GINSENG & GINGER 3
6	FRESTEA GREEN NO CALORIE 500 ML
7	FRESTEA GREEN TEA PET 500 ML
8	FRESTEA JASMIN PET 500 ML
9	ARTEA APP GREEN TEA 500 ML
10	ARTEA HONEY GREEN TEA 500 ML
11	ZESTEA JASMINE GREEN TEA 500 ML
12	ZESTEA JASMINE GREEN TEA LESS SUGAR 500
13	ZESTEA PURE GREEN TEA 500 ML
14	NU GREEN TEA HONEY 500 ML
15	NU GREEN TEA LESS SUGAR 500 ML
16	NU GREEN TEA ORIGINAL 500 ML
17	NU GREEN TEA KURMA 500 ML
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML
21	JOY TEA GREEN COZY JASMINE 500 ML
22	C2 GREEN TEA APPLE 360 ML
23	C2 GREEN TEA CLASSIC 360 ML
24	FUTAMI OOLONG TEA 350 ML
25	FRUIT TEA FUSION 500 ML
26	FRUIT TEA BLKCURRENT 500 ML
27	FRUIT TEA STR 500 ML
28	FRUIT TEA APP 500 ML

(Sumber: Data Hypermart)

### 3.2 Data Penjualan dan Keuntungan Tiap Produk

Untuk menerapkan model SSAP, dibutuhkan data penjualan serta keuntungan dari tiap jenis produk yang ada.

**Table 3.3** Data Penjualan, Keuntungan, Alokasi *Facing* Saat ini untuk Tiap Produk

No.	Produk	Retail Price	Purchase Cost	Gross Keuntungan	Penjualan /hari	Facing saat ini
1	FRESTEA FRUTCY APEL 500 ML	4.300	3.447	853	52	9
2	FRESTEA FRUTCY LMN 500 ML	4.300	3.447	853	51	12
3	FRESTEA FRUTCY MARKISA 500 ML	4.300	3.447	853	40	14
4	FRESTEA GREEN MY BODY ALOEVERA & ORANGE	4.350	3.447	903	3	12
5	FRESTEA GREEN MY BODY GINSENG & GINGER 3	4.350	3.447	903	4	18
6	FRESTEA GREEN NO CALORIE 500 ML	4.250	3.390	860	17	15
7	FRESTEA GREEN TEA PET 500 ML	4.250	3.390	860	48	20
8	FRESTEA JASMIN PET 500 ML	4.250	3.390	860	31	15
9	ARTEA APP GREEN TEA 500 ML	3.350	2.673	677	1	7
10	ARTEA HONEY GREEN TEA 500 ML	3.350	2.673	677	4	7
11	ZESTEAS JASMINE GREEN TEA 500 ML	3.900	2.801	1.099	5	2
12	ZESTEAS JASMINE GREEN TEA LESS SUGAR 500	3.900	2.801	1.099	1	2
13	ZESTEAS PURE GREEN TEA 500 ML	3.900	2.801	1.099	1	12
14	NU GREEN TEA HONEY 500 ML	4.275	3.409	866	32	20
15	NU GREEN TEA LESS SUGAR 500 ML	4.275	3.409	866	6	15
16	NU GREEN TEA ORIGINAL 500 ML	4.275	3.409	866	19	15
17	NU GREEN TEA KURMA 500 ML	4.275	3.409	866	4	20
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	3.500	2.898	602	7	1
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	3.600	2.898	702	7	4
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	3.600	2.898	702	3	8
21	JOY TEA GREEN COZY JASMINE 500 ML	3.600	2.898	702	2	8
22	C2 GREEN TEA APPLE 360 ML	2.200	1.764	436	7	16
23	C2 GREEN TEA CLASSIC 360 ML	2.200	1.764	436	3	5
24	FUTAMI OOLONG TEA 350 ML	3.150	2.513	637	3	1
25	FRUIT TEA FUSION 500 ML	3.650	2.973	677	2	4
26	FRUIT TEA BLKCURRENT 500 ML	3.650	2.973	677	2	10
27	FRUIT TEA STR 500 ML	3.650	2.973	677	3	4
28	FRUIT TEA APP 500 ML	3.650	2.973	677	1	4

(Sumber: Data Hypermart)

Pada penelitian ini, tingkat penjualan suatu produk dianggap dipengaruhi oleh banyaknya jumlah *facing* yang dialokasikan kepada produk tersebut. Oleh karena itu, data penjualan yang diperoleh perlu untuk dikonversikan sebelumnya guna memperoleh data *unmodified demand*, yang menunjukkan tingkat penjualan dasar dari suatu produk yang tidak dipengaruhi oleh alokasi *facing*. Estimasi

*unmodified demand* dikalkulasi dengan persamaan berikut (Hubner & Kuhn, 2010):

$$\alpha(\cdot) = \frac{(\cdot)}{(\cdot, \cdot)^\beta} \quad (3.1)$$

dimana persamaan ini diperoleh dari fungsi deterministik dari *demand rate* yang digunakan dalam model SSAP.

$$(\cdot) = \alpha \cdot \beta \quad (3.2)$$

**Tabel 3.4** Data *Unmodified Demand*

No.	Produk	<i>Unmodified Demand</i>
1	FRESTEA FRUTCY APEL 500 ML	34
2	FRESTEA FRUTCY LMN 500 ML	31
3	FRESTEA FRUTCY MARKISA 500 ML	24
4	FRESTEA GREEN MY BODY ALOEVERA & ORANGE	2
5	FRESTEA GREEN MY BODY GINSENG & GINGER 3	2
6	FRESTEA GREEN NO CALORIE 500 ML	10
7	FRESTEA GREEN TEA PET 500 ML	26
8	FRESTEA JASMIN PET 500 ML	18
9	ARTEA APP GREEN TEA 500 ML	1
10	ARTEA HONEY GREEN TEA 500 ML	3
11	ZESTEA JASMINE GREEN TEA 500 ML	4
12	ZESTEA JASMINE GREEN TEA LESS SUGAR 500	1
13	ZESTEA PURE GREEN TEA 500 ML	1
14	NU GREEN TEA HONEY 500 ML	18
15	NU GREEN TEA LESS SUGAR 500 ML	3
16	NU GREEN TEA ORIGINAL 500 ML	11
17	NU GREEN TEA KURMA 500 ML	2
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	7
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	5
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	2
21	JOY TEA GREEN COZY JASMINE 500 ML	1
22	C2 GREEN TEA APPLE 360 ML	4
23	C2 GREEN TEA CLASSIC 360 ML	2
24	FUTAMI OOLONG TEA 350 ML	3
25	FRUIT TEA FUSION 500 ML	2
26	FRUIT TEA BLKCURRENT 500 ML	1
27	FRUIT TEA STR 500 ML	2
28	FRUIT TEA APP 500 ML	1

Data nilai penjualan dasar (*unmodified demand*) inilah yang akan digunakan dalam model SSAP.

### 3.3 Parameter *Space Elasticity*

Pada penelitian kali ini, digunakan asumsi estimasi nilai *space elasticity* berdasarkan penelitian-penelitian yang telah dilakukan sebelumnya. Hal ini dilakukan karena tidak tersedianya data historis yang dapat menampilkan perubahan alokasi *facing* dari tiap produk, serta ketidakcukupan waktu untuk melakukan observasi terhadap perubahan alokasi *facing* dari tiap produk.

R. Curhan dalam *paper*-nya ‘*The Relationship between Shelf Space and Unit Sales in Supermarkets*’ (Curhan, 1972) mengungkapkan hasil penelitiannya dalam menentukan estimasi *space elasticity* dengan melakukan eksperimen terhadap 500 produk *grocery* dengan 4 supermarket besar sebagai *test stores*. Pada produk-produk tersebut dilakukan pengurangan ataupun penambahan *shelf space* guna mengobservasi perubahan unit penjualan selama kurun waktu 5 hingga 12 minggu sebelum dan sesudah perubahan. Penelitian ini akan menggunakan data *space elasticity* rata-rata dari seluruh produk berdasarkan eksperimen yang dilakukan Curhan, yaitu sebesar **0.212**.

Hansen dan Heinsbroek (1979) mengungkapkan bahwa penggunaan data *space elasticity* rata-rata dapat dilakukan dalam suatu kasus SSAP, karena untuk memperoleh informasi mengenai nilai spesifik *space elasticity* dari tiap produk dibutuhkan proses dan eksperimen yang berbiaya besar, yang mungkin tidak dapat dilakukan oleh semua pihak. Penggunaan nilai *space elasticity* rata-rata menyebabkan model yang digunakan tidak menggambarkan karakteristik dari tiap produk yang ada, dimana sebenarnya tiap produk memiliki nilai *space elasticity* yang berbeda-beda. Pada penelitian dengan menggunakan nilai estimasi *space elasticity* rata-rata, didapatkan nilai total keuntungan yang lebih kecil 6,10% daripada ketika menggunakan nilai *space elasticity* spesifik untuk tiap produk (Hansen & Hensbroek, 1979). Oleh karena itu, dengan menggunakan penyamarataan nilai *space elasticity*, nilai objektif optimal yang dihasilkan masih akan cenderung berada di bawah estimasi sesungguhnya.

### 3.4 Parameter Kendala Kapasitas dan Batas Alokasi *Facing*

Pada penelitian ini, model SSAP yang digunakan memiliki dua kendala utama, yaitu kendala kapasitas maksimum rak *display* produk serta batas (*lower bound* dan *upper bound*) dari alokasi *facing* tiap produk. Berdasarkan hasil observasi data dan wawancara dengan pihak manajemen gerai ritel, diperoleh estimasi parameter kendala sebagai berikut.

- Kapasitas *facing* maksimum rak *display*: 1.672 cm;  
Kapasitas ini merupakan kapasitas gabungan dari 6 level rak yang dialokasikan untuk sub-kategori produk minuman teh kemasan botol.
- Batas minimal alokasi *facing* (*lower bound*): 2 *facing*/produk  
Batasan ini merupakan kendala yang ditetapkan secara manajerial yang membatasi minimal *exposure* yang dapat diterima oleh sebuah produk. Batasan minimal sejumlah 2 *facing* juga konsisten dengan jumlah minimal barang yang dikirimkan dari gudang. Jumlah barang minimal yang dikeluarkan dari *backroom* untuk ditampilkan pada *display room* adalah sebanyak 1 *bulk/case*. Dengan mengestimasi ukuran *bulk* rata-rata terdiri 24 botol/*case* dan banyaknya jumlah barang yang dapat disimpan pada alokasi 1 *facing* adalah 12 produk, maka diperoleh jumlah minimal *facing* sebanyak 2 (24/12).
- Batas maksimal alokasi *facing* (*upper bound*): 20 *facing*/produk  
Kendala ini juga merupakan batasan operasional yang diterapkan oleh manajemen guna menjaga keberagaman produk yang ditampilkan pada rak *display*.



## BAB 4

### PENGOLAHAN DATA DAN ANALISIS

#### 4.1 Penyusunan Algoritma

Model penyelesaian permasalahan optimasi alokasi ruang produk (SSAP) ini menggunakan pendekatan algoritma metaheuristik yaitu Algoritma Genetika (AG), seperti yang telah dijelaskan sebelumnya pada bab 2. Untuk membangun tahapan kerja algoritma tersebut, maka dibuatlah suatu program komputer dengan bantuan *software* MATLAB versi 2007a.

##### 4.1.1 Langkah-langkah Penyusunan Algoritma Genetika

Alur proses pengerjaan algoritma genetika dapat dilihat pada gambar 4.1. Prosedur penyusunan dan penyelesaian model SSAP dengan algoritma genetika dapat dilihat pada uraian berikut:

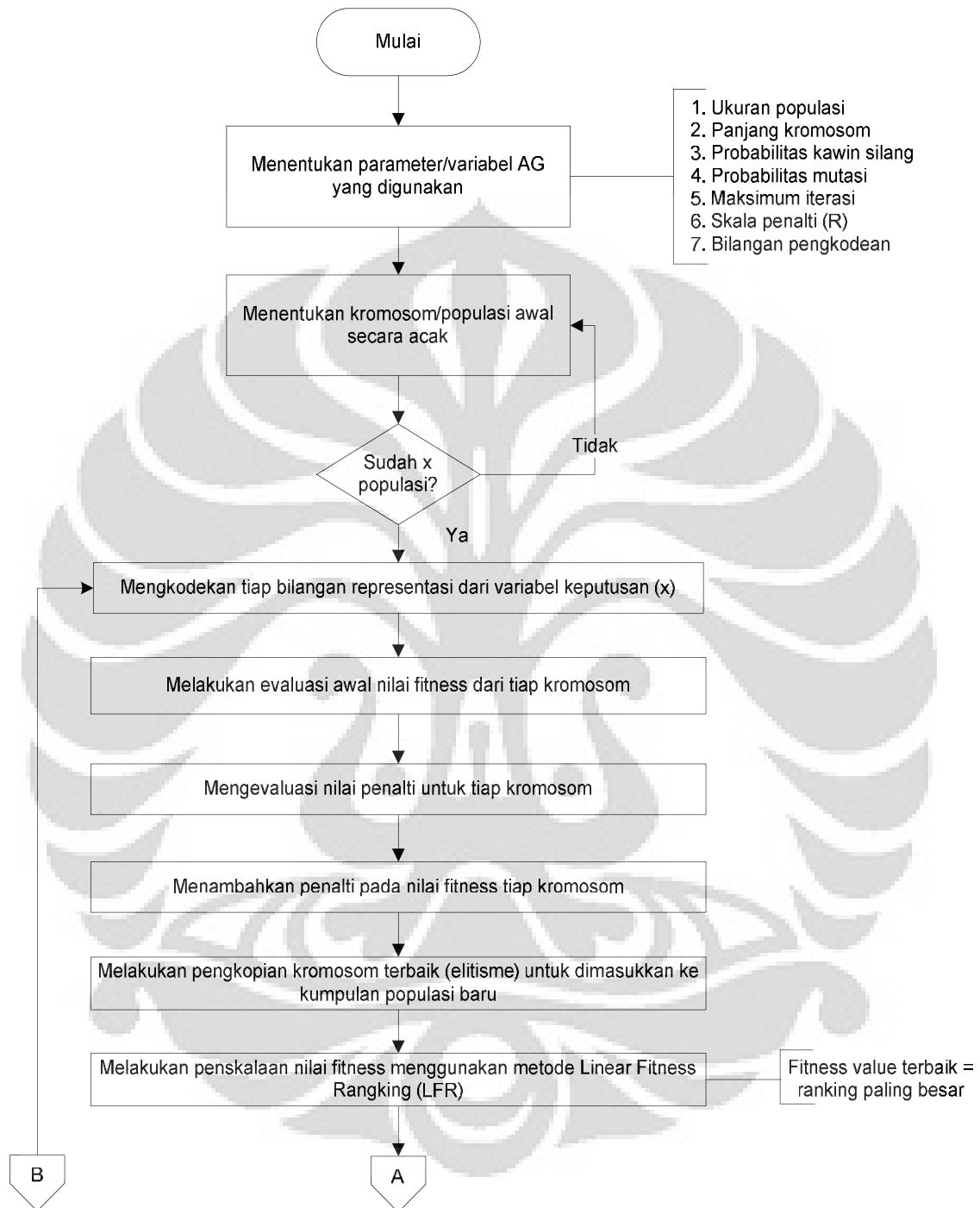
1. Menentukan parameter dan variabel yang akan digunakan yaitu:

- Ukuran populasi

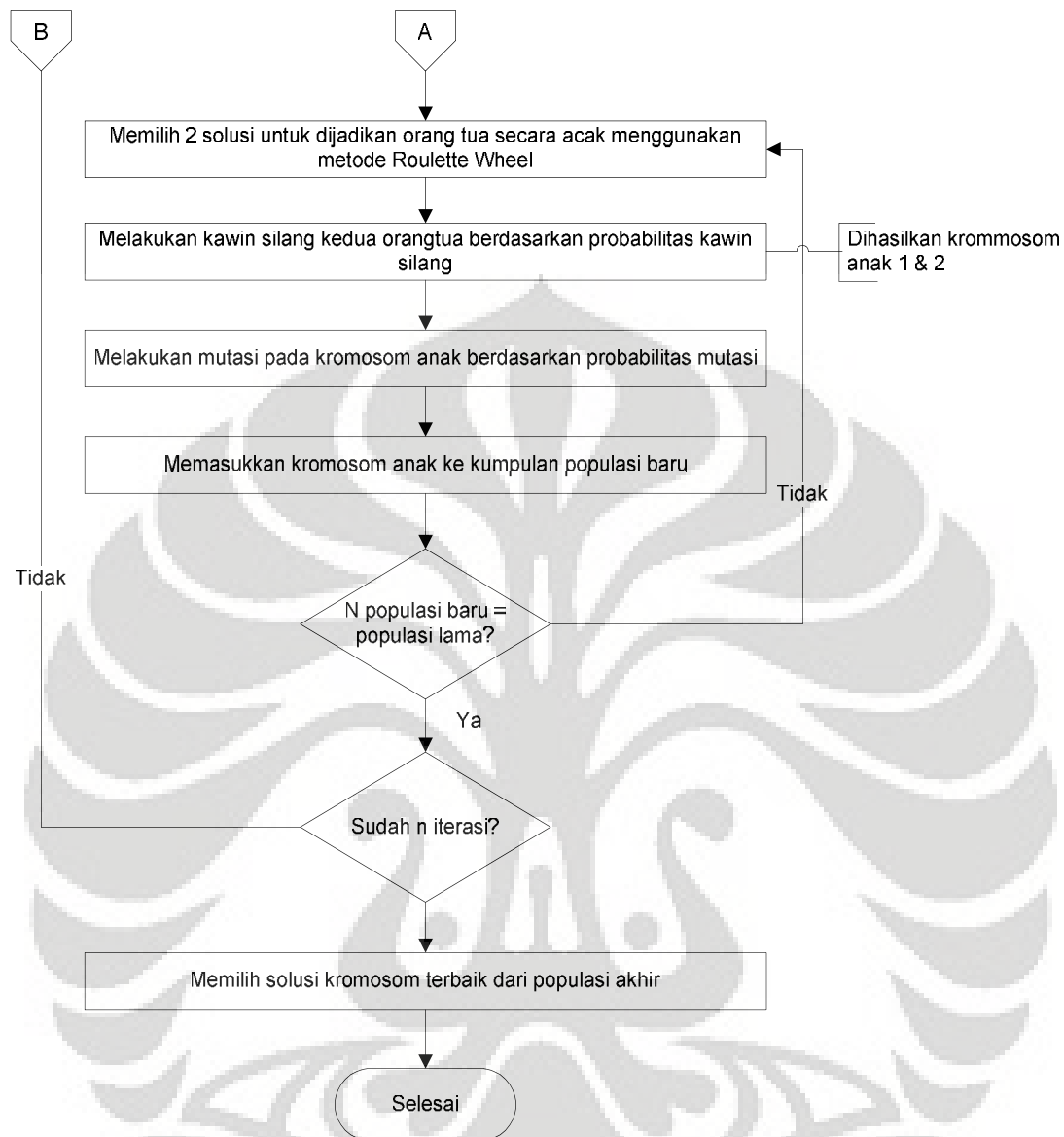
Algoritma genetika bekerja dengan membentuk suatu populasi solusi, bukan solusi tunggal, sehingga perlu ditetapkan suatu ukuran populasi permasalahan. Ukuran populasi yang optimal berkisar antara  $n$  hingga  $2n$ , dengan  $n$  merupakan panjang *string* kromosom dari model AG yang digunakan untuk merepresentasikan solusi (Alexander, 1929).

- Banyak iterasi

Algoritma genetika biasanya memiliki jumlah iterasi antara 50 hingga 500 atau lebih (Townsend, n. d.). Banyaknya iterasi yang dilakukan juga dapat mempengaruhi kualitas solusi yang dihasilkan. Semakin banyak jumlah iterasi yang digunakan, akan semakin baik pula kualitas solusi yang dihasilkan (Boyabatli & Sabuncuoglu, n. d.).



**Gambar 4.1** Diagram Alur Algoritma Genetika Permasalahan SSAP



**Gambar 4.1** Diagram Alur Algoritma Genetika Permasalahan SSAP (lanjutan)

- Probabilitas kawin silang  
Merupakan parameter penting dalam algoritma genetika karena sebuah kromosom yang mengarah pada kualitas solusi yang baik dapat diperoleh dari proses kawin silang. Probabilitas ini tergantung dari jenis masalah yang diselesaikan. Untuk problem SSAP, probabilitas kawin silang yang direkomendasikan adalah sebesar 0,25, yang artinya sekitar 25% dari kromosom dalam

populasi mengalami proses kawin silang (Hark Hwang, Bum Choi, & Min-Jin Lee, 2005).

- Probabilitas mutasi

Menurut penelitian yang telah dilakukan sebelumnya, probabilitas mutasi untuk permasalahan SSAP direkomendasikan sebesar 0,01 (Hark Hwang, Bum Choi, & Min-Jin Lee, 2005).

2. Menentukan representasi pengkodean solusi permasalahan dalam bentuk gen

Dalam permasalahan SSAP ini, digunakan *binary encoding*, dimana solusi potensial dari alokasi ruang untuk tiap produk direpresentasikan ke dalam sebuah set gen yang terdiri dari bilangan 0 atau 1. Satu variabel keputusan (jumlah *facing* untuk satu produk) direpresentasikan oleh 5 bit gen. Dengan 28 produk yang ada, maka sebuah kromosom akan memiliki 140 buah gen (28x5). Proses pengkodean dari bilangan biner ke bilangan sebenarnya yang merepresentasikan jumlah *facing* tiap produk menggunakan persamaan sebagai berikut (Suyanto, 2006):

$$= ( ) + \frac{( ( ) - ( ) )}{\sum 2} ( (1).2 + \dots + ( ).2 ) \quad (4.1)$$

Dimana  $r(a)$  dan  $r(b)$  merupakan batas atas (*upper bound*) dan batas bawah (*lower bound*) dari model SSAP yang digunakan, dan  $N$  adalah jumlah gen dalam kromosom yang merepresentasikan satu variabel. Dalam hal ini, berarti nilai  $N$  adalah 10, karena digunakan 10 buah gen untuk mengkodekan tiap variabel alokasi *facing* produk. Dengan menggunakan persamaan ini, maka kendala batas atas dan bawah dapat dikendalikan, karena tiap kromosom tidak akan melebihi dari batas kendala tersebut.

3. Menentukan kromosom populasi awal secara acak.

Populasi awal inilah yang nantinya akan diperbaiki dengan menggunakan skema reproduksi genetika kawin silang dan mutasi. Populasi awal dihasilkan dengan membangkitkan bilangan random antara 0 dan 1 pada semua gen untuk semua populasi.

4. Mengkodekan bilangan biner untuk tiap anggota populasi menjadi variabel  $x(i)$  sesuai dengan persamaan pengkodean pada *point* (2).

5. Melakukan evaluasi awal terhadap setiap kromosom dalam populasi awal.

Evaluasi awal dari populasi dilakukan dengan menghitung fungsi objektif dari model SSAP untuk tiap kromosomnya. Karena tujuan model adalah memaksimalkan fungsi objektif, maka nilai *fitness* akan sama dengan nilai fungsi objektif ( $fitness=F(x)$ ).

6. Menghitung penalti untuk tiap solusi kromosom.

Permasalahan SSAP merupakan bentuk *constrained optimization problem*. Fokus utama dalam mengaplikasikan AG dalam sebuah *constrained optimization* adalah bagaimana mengatasi kendala-kendala yang ada karena operator genetika yang digunakan untuk memanipulasi kromosom seringkali menghasilkan solusi kromosom yang *infeasible* (Gutierrez-Garcia & Kuri-Morales, n. d.). Pada penelitian ini, akan digunakan strategi penalti dimana penalti akan dikenakan pada nilai *fitness* kromosom sesuai dengan derajat tingkat pelanggaran yang ada. Permasalahan berkendala akan ditransformasikan menjadi problem non-kendala dengan penambahan nilai penalti tersebut, seperti yang terlihat pada fungsi di bawah ini:

$$F(x) = \begin{cases} f(x) & x \in \text{feasible region} \\ f(x) + \text{penalty}(x) & x \notin \text{feasible region} \end{cases}$$

Dalam menentukan nilai penalti yang dikenakan, digunakan metode penalti statis, dimana nilai penalti bergantung dari besarnya pelanggaran terhadap kendala yang ada. Skala parameter penalti bernilai tetap untuk setiap iterasinya. Fungsi penalti yang digunakan dalam model ini adalah sebagai berikut (Homaiffar, A., Lai, S. & Qi, C., 1994):

$$P(x) = R \cdot H(x) \quad (4.2)$$

Dengan R merupakan skala parameter penalti untuk kendala tersebut dan H(x) merupakan fungsi derajat besarnya pelanggaran terhadap kendala.

Model SSAP yang digunakan pada penelitian ini memiliki satu kendala yang akan ditransformasikan menjadi fungsi penalti, yaitu kendala batas maksimal kapasitas, dimana total alokasi ruang dari seluruh produk pada kategori tersebut tidak dapat melewati batas maksimal kapasitas yang dapat ditampung rak *display*. Kendala ini merupakan jenis *hard constraint* dimana kendala harus dapat dipenuhi.

Karena solusi dari kromosom yang melanggar kendala kapasitas ini akan bersifat *infeasible*, maka nilai parameter R ditentukan dengan estimasi bahwa nilai *fitness* dari solusi *infeasible* tersebut harus disejajarkan pada titik di bawah nilai *fitness* terkecil dari kemungkinan solusi yang ada, atau dengan kata lain R merupakan selisih dari nilai maksimal dan minimal *fitness* yang dapat dimiliki oleh sebuah kromosom. Nilai maksimal yang mungkin dimiliki oleh sebuah kromosom adalah jika semua gen pada sebuah kromosom bernilai 1, sehingga setiap variabel keputusan alokasi ruang pada kromosom tersebut memiliki nilai alokasi *facing* maksimum, sehingga akan diperoleh nilai *fitness* maksimal yang mungkin dimiliki oleh sebuah kromosom. Sedangkan nilai minimal didapatkan jika semua gen bernilai 0. Nilai R akan dikalikan -1 agar bernilai negatif.

Nilai  $H(x)$  yang merepresentasikan derajat besarnya pelanggaran yang dilakukan diperoleh dengan membandingkan total alokasi *facing* yang dihasilkan dengan maksimal kapasitas rak ( $l_i x_i / T$ ).

$$H(x) = \frac{\sum}{T} \quad (4.3)$$

Oleh karena itu, sebelumnya perlu dilakukan evaluasi total alokasi *facing* yang dihasilkan tiap kromosom untuk menentukan apakah kromosom solusi tersebut patut dikenakan penalti atau tidak. Jika total alokasi *facing* tidak melebihi atau sama dengan kapasitas rak, maka solusi kromosom tersebut tidak dikenakan penalti, karena masuk dalam kategori solusi *feasible*.

7. Menentukan evaluasi akhir dengan memasukkan nilai penalti untuk tiap kromosom.

Pada tahap ini, nilai evaluasi (*fitness*) awal diperbarahi dengan memasukkan nilai penalti (jika ada) pada tiap kromosom.

8. Melakukan elitisme (duplikasi) dari kromosom terbaik.

Jika ukuran populasi genap, maka elitisme dilakukan sebanyak dua kali pada kromosom terbaik. Jika ukuran populasi ganjil, maka hanya dilakukan satu kali duplikasi dari kromosom terbaik. Hasil elitisme ini akan langsung dimasukkan ke populasi baru untuk generasi berikutnya.

9. Untuk mencegah terjadinya *local optima*, maka perlu dilakukan penskalaan nilai *fitness* dengan menggunakan metode *Linear Fitness Ranking* (LFR)<sup>5</sup>. Nilai *fitness* yang telah dilinierkan/dinormalisasi disebut *linear fitness*. Fungsi *linear fitness* yang digunakan pada model ini adalah sebagai berikut:

$$R(i) = \frac{f(i) - f_{\min}}{f_{\max} - f_{\min}} \quad (4.4)$$

Dengan N merupakan jumlah individu dalam populasi dan R(i) merupakan peringkat *fitness* kromosom ke-i dalam populasi. Misal R(i) = 1, maka i adalah individu bernilai *fitness* tertinggi dan R(i) = N jika i adalah individu bernilai *fitness* terendah.

10. Setelah diperoleh nilai *linear fitness* untuk tiap individu kromosom, maka dilakukan proses seleksi terhadap kromosom untuk mendapatkan 2 kromosom yang disebut orang tua (*parent*) dengan menggunakan metode *roulette wheel*.

Metode ini menirukan permainan *roulette*, dimana masing-masing kromosom menempati potongan lingkaran pada roda *roulette* secara proporsional sesuai dengan nilai *linear fitness*-nya. Kromosom yang memiliki nilai *fitness* lebih besar menempati potongan lingkaran yang lebih besar dibandingkan dengan kromosom bernilai *fitness* rendah, dan tentu saja memiliki probabilitas terbesar untuk terpilih menjadi orangtua. Sebuah kromosom akan terpilih menjadi kromosom *parent* jika bilangan random yang dibangkitkan berada dalam interval kumulatifnya. Sebuah kromosom dapat diseleksi lebih dari satu kali untuk menjadi kromosom *parent*.

11. Selanjutnya, dilakukan operasi genetika yang pertama yaitu kawin silang (*crossover*).

Operasi kawin silang dilakukan dengan menggunakan metode *one-cut-point* dimana akan dibangkitkan sebuah angka random antara 1 hingga jumlah gen dalam kromosom untuk menentukan satu titik potong. Kromosom Anak(1) akan terdiri dari gen-gen bagian kiri dari titik potong kromosom orang tua(1) dan gen-gen bagian kanan dari titik potong kromosom orang tua(2), begitu juga sebaliknya untuk kromosom anak(2).



Misalnya: kromosom **1110 0001 0011** dan **1000 0110 0111** dapat dikawinsilangkan pada titik potong gen ke-4 pada tiap kromosom dan menghasilkan dua kromosom anak: **1110 0110 0111** dan **1000 0001 0011**. Proses kawin silang juga dilakukan berdasarkan probabilitas kawin silang. Jika angka random yang dibangkitkan lebih kecil dari probabilitas kawin silang, maka proses kawin silang akan terjadi. Namun jika angka random lebih besar dari probabilitas kawin silang, maka kawin silang tidak terjadi dan kromosom anak dianggap sama dengan kromosom orang tua.

12. Melakukan operasi genetik mutasi pada tiap kromosom anak.

Pada operasi ini, akan dibangkitkan angka random untuk tiap gen dalam kromosom anak. Jika angka tersebut lebih kecil dari probabilitas muasi yang dimiliki, maka mutasi akan terjadi pada gen tersebut. Proses mutasi dilakukan secara sederhana, yaitu mengganti dari yang sebelumnya bilangan 0 menjadi 1 dan sebaliknya. Proses seleksi hingga mutasi dilakukan hingga ukuran populasi baru mencapai ukuran populasi awal.

13. Tahap terakhir yaitu penggantian populasi dengan sistem *generational replacement*, dimana seluruh kromosom pada populasi baru akan menggantikan kromosom pada populasi sebelumnya tanpa mengevaluasi nilai *fitness* dari tiap kromosom.

14. Prosedur evaluasi *fitness* awal hingga penggantian populasi merupakan proses untuk satu generasi, dan prosedur ini akan berulang terus hingga tercapainya *stopping criteria*, yaitu maksimum iterasi sesuai dengan jumlah yang telah ditentukan. Kemudian, akan diambil kromosom terbaik yang akan menjadi solusi untuk alokasi ruang produk pada rak *display*.

#### 4.1.2 Verifikasi dan Validasi Program

Sebelum program yang disusun digunakan untuk menyelesaikan permasalahan SSAP, terlebih dahulu dibutuhkan proses verifikasi dan validasi dari program tersebut. Verifikasi merupakan proses pemeriksaan pada program apakah logika operasional model sesuai dengan konseptual model yang kita inginkan. Parameter model program dikatakan telah terverifikasi apabila telah

berjalan sesuai konseptual model. Beberapa proses verifikasi yang dilakukan dalam proses pembuatan program ini antara lain:

- Memastikan kebenaran logika pemikiran, yaitu kesesuaian penulisan *source code* dengan konsep algoritma genetika, serta dapat dijalankan dengan baik
- Memastikan bahwa program sudah memiliki semua batasan kendala masalah yang ada
- Memastikan fungsi tujuan telah berjalan dengan benar

Setelah proses verifikasi, selanjutnya adalah proses validasi. Validasi dilakukan untuk dapat memastikan bahwa program berjalan sesuai dengan fungsinya dan *output* yang dihasilkan benar dan sesuai. Validasi terhadap program dilakukan dengan memasukkan data *dummy* (Tabel 4.1). Hasil *run* program dengan data *dummy* kemudian dibandingkan dengan perhitungan manual untuk memperoleh validasi tersebut.

Data *dummy* yang digunakan yaitu data kategori yang terdiri dari 5 buah produk dengan nilai *demand*, keuntungan, *space elasticity* serta parameter dan ketentuan lainnya dapat dilihat pada tabel-tabel berikut.

**Tabel 4.1** Data *Dummy*

Product No.	Keuntungan	<i>Demand</i>	Panjang <i>facing</i>	<i>Space Elasticity</i>
1	853	23	6	0,212
2	860	10	6	0,212
3	860	17	6	0,212
4	1099	4	6	0,212
5	702	5	6	0,212

**Tabel 4.2** Parameter Model SSAP untuk Data *Dummy*

Parameter Model	Nilai
Kapasitas Rak	150 cm
Batas Min <i>Facing</i>	2
Batas Max <i>Facing</i>	10
Skala Penalti R	40.413

**Tabel 4.3** Parameter GA Data *Dummy*

Parameter GA	Nilai
Ukuran Populasi	6
Probabilitas Kawin Silang	0,99
Probabilitas Mutasi	0,99
Maksimum Iterasi	1

#### 4.1.2.1 Hasil piranti lunak

Hasil *run* program menghasilkan maksimum keuntungan **Rp118.414** dengan kapasitas *facing* yang ditempati 150 cm. Rincian *facing* untuk tiap produk dapat dilihat pada Tabel 4.4.

**Tabel 4.4** *Output* Hasil *Run* Program dengan Data *Dummy*

Product No.	<i>Facing</i>
1	4
2	3
3	6
4	7
5	5

#### 4.1.2.2 Hasil perhitungan manual

Untuk memvalidasi hasil yang dikeluarkan program, dilakukan juga perhitungan manual sebagai perbandingan dengan hasil keluaran program. Langkah-langkah yang dilakukan sesuai dengan diagram alur pada Gambar 4.1 adalah sebagai berikut:

1. Pembentukan populasi solusi awal (inisialisasi populasi)

Pembentukan inisial populasi secara acak didapatkan dari hasil yang dikeluarkan program. Penggunaan hasil pembangkitan bilangan *random* dari program dimaksudkan guna memastikan kesamaan variasi populasi awal. Populasi awal yang akan digunakan dapat dilihat pada Tabel 4.7.

2. Pengkodean bilangan biner untuk tiap anggota populasi menjadi variabel  $x(i)$

Pengkodean dilakukan menggunakan persamaan yang telah dijelaskan pada subbab 4.1.1.

**Tabel 4.5** Pengkodean Bilangan Biner

	Variabel				
	1	2	3	4	5
<b>Kromosom 1</b>	2	3	7	6	3
<b>Kromosom 2</b>	9	8	5	2	9
<b>Kromosom 3</b>	3	5	5	4	9
<b>Kromosom 4</b>	3	7	9	4	9
<b>Kromosom 5</b>	4	3	6	7	5
<b>Kromosom 6</b>	5	4	2	3	6

3. Evaluasi *fitness* awal untuk tiap kandidat solusi kromosom  
Evaluasi nilai *fitness* dilakukan dengan menghitung fungsi objektif dari dari tiap kromosom.

**Tabel 4.6** Evaluasi *Fitness* Awal

Kromosom No.	Fitness Awal
1	112.476
2	128.771
3	116.328
4	122.824
5	118.414
6	112.603

4. Penghitungan penalti untuk tiap kromosom  
Penalti dikenakan pada solusi kromosom yang total alokasi *facing*-nya melebihi kapasitas rak. Nilai penalti untuk tiap kromosom adalah:

**Tabel 4.7** Penalti Kapasitas untuk Tiap Kromosom

Kromosom No.	Kapasitas	Penalti
1	126	-
2	198	(70.416)
3	156	(43.711)
4	192	(66.213)
5	150	-
6	120	-

Tabel 4.8 Populasi Awal (dari Program)

Kromosom No.	Gen Variabel																																																		
	1										2										3										4										5										
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	
1	0	0	0	0	1	1	0	1	0	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	1	0	1	
2	1	1	1	0	0	0	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	1	1	0	1	
3	0	0	1	0	0	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	
4	0	0	1	0	0	1	1	1	0	1	1	0	1	0	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
5	0	1	0	1	1	1	0	1	1	1	0	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	
6	0	1	1	0	0	1	0	0	1	0	1	0	1	0	0	0	1	1	0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	

5. Penghitungan nilai *fitness* akhir dari tiap kromosom  
*Fitness* akhir merupakan nilai *fitness* awal yang telah ditambahkan nilai penalti.

**Tabel 4.9** Nilai *Fitness* Akhir Tiap Kromosom

Kromosom No.	Fitness Akhir
1	112.476
2	58.354
3	72.617
4	56.611
5	118.414
6	112.603

6. Elitisme kromosom terbaik  
 Dilihat dari Tabel 4.9, nilai *fitness* akhir terbaik terdapat pada kromosom 5, sehingga kromosom inilah yang akan dilakukan elitisme dengan mengkopinya langsung ke dalam kumpulan populasi baru. Karena populasi berjumlah genap, maka elitisme dilakukan sebanyak dua kali pada kromosom yang sama.
7. Penskalaan nilai *fitness* (*Linear Fitness Ranking*)  
 Nilai *fitness* setelah dilakukan proses LFR adalah sebagai berikut.

**Tabel 4.10** *Linear Fitness Ranking*

Kromosom No.	Final Fitness	Rank	Linear Fitness
5	118.414	1	118.414
6	112.603	2	106.053
1	112.476	3	93.693
3	72.617	4	81.332
2	58.354	5	68.972
4	56.611	6	56.611

8. Seleksi dua kromosom orang tua  
 Bilangan angka random yang digunakan untuk menentukan kromosom orang tua merupakan angka yang sama yang dihasilkan oleh program. Hal ini dilakukan guna memastikan perbandingan yang akurat antara

perhitungan manual dengan perhitungan program. Rentang proporsi untuk tiap kromosom dapat dilihat pada Tabel 4.11.

**Tabel 4.11** Rentang Proporsi tiap Kromosom

Kromosom No.	Rank	Linear Fitness	Proporsi	Kum. Range
5	1	118.414	0,226	0,226
6	2	106.053	0,202	0,427
1	3	93.693	0,178	0,606
3	4	81.332	0,155	0,761
2	5	68.972	0,131	0,892
4	6	56.611	0,108	1,000

Dengan angka random 0,1032 dan 0,5266, maka dua kromosom orang tua yang dipilih adalah kromosom 1 dan 3. Karena jumlah populasi berjumlah 6, dibutuhkan dua kali proses seleksi orang tua untuk dapat memenuhi populasi baru. Untuk seleksi orang tua tahap dua, angka random yang dihasilkan adalah 0,3718 dan 0,0213, maka kromosom orang tua yang dipilih adalah kromosom 2 dan 1.

9. Operasi genetika kawin silang (*crossover*) terhadap kromosom orang tua  
 Bilangan random untuk menentukan titik potong pada proses kawin silang juga diperoleh dari angka yang sama yang dihasilkan program. Angka titik potong acak yang dihasilkan program adalah 41 untuk pasangan orang tua pertama dan 48 untuk pasangan orang tua kedua. Probabilitas kawin silang yang besar yaitu 0,99 menyebabkan proses kawin silang hampir pasti terjadi. Adapun bilangan *random* yang dihasilkan program untuk menentukan apakah kawin silang dilakukan atau tidak adalah 0,656 dan 0,449. Hasil kromosom anak untuk tiap proses kawin silang dapat dilihat pada Tabel 4.14-4.17.
10. Operasi genetika mutasi pada kromosom anak  
 Proses mutasi dilakukan berdasarkan probabilitas mutasi yang telah ditetapkan (0,99) pada tiap gen pada tiap kromosom anak. Karena tingginya probabilitas mutasi yang ditetapkan, semua bilangan random yang dihasilkan program menunjukkan proses mutasi layak untuk



dilakukan pada tiap gen di kromosom anak. Kromosom anak sebelum dan sesudah mutasi dapat dilihat pada Tabel 4.18 dan 4.19.

11. Penggantian populasi lama dengan populasi baru hasil operasi genetika (*generational replacement*)

Populasi baru yang dihasilkan, seperti terlihat pada Tabel 4.20 terdiri dari kromosom anak hasil *crossover* dan mutasi serta dua kromosom hasil elitisme.

12. Pemilihan solusi terbaik dari populasi akhir

Setelah satu iterasi terpenuhi, dilakukan pemilihan solusi terbaik dari populasi akhir dengan melakukan kembali evaluasi nilai *fitness* dari tiap kromosom.

**Tabel 4.12** Pemilihan Solusi Terbaik dari Populasi Akhir

Kromosom No.	Final Fitness
1	118.414
2	118.414
3	74.654
4	78.149
5	111.392
6	73.247

Dari hasil evaluasi nilai *fitness* pada populasi baru untuk menentukan solusi akhir, didapatkan bahwa kromosom ke-1 dengan nilai *fitness* 118.414 merupakan kromosom solusi terbaik, dengan alokasi *facing* untuk tiap produk adalah sebagai berikut.

**Tabel 4.13** Alokasi *Facing* Produk Hasil Perhitungan Manual

Product No.	Facing
1	4
2	3
3	6
4	7
5	5

Tabel 4.14 Pasangan Pertama Kromosom Orang Tua

Kromosom Orang Tua No.	Gen Variabel															S																																					
	1					2					3						4																																				
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50			
1	0	0	0	0	0	1	1	0	1	0	1	1	0	0	1	1	0	0	1	0	1	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	1	
2	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	1	1	0	0	1	0	1	0	1	1	0	1	0	0	1	1	0	1	0	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4. 15 Kromosom Anak Hasil *Crossover* Pertama

Kromosom Anak No.	Gen Variabel															S																																					
	1					2					3						4																																				
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50			
1	0	0	0	0	1	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	0	1	0	0	1	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0		
2	0	0	1	0	0	1	1	0	0	1	0	0	1	0	0	1	1	0	0	1	0	1	0	1	1	0	1	0	0	1	1	0	1	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4.16 Pasangan Kedua Kromosom Orang Tua

Kromosom Orang Tua No.	Gen Variabel															S																																						
	1					2					3						4																																					
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50				
1	1	1	1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	1	0	0	1	1	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	1	1	0	1	1	0	1	0	1				
1	0	0	0	0	1	1	0	1	0	1	0	0	1	0	0	1	1	0	0	1	0	1	0	1	1	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4.17 Kromosom Anak Hasil Crossover Kedua

Kromosom Anak No.	Gen Variabel				
	1	2	3	4	5
1	1	1	1	1	1
2	1	1	1	1	1
3	1	1	1	1	1
4	1	1	1	1	1

Tabel 4.18 Kromosom Anak Sebelum Mutasi

Kromosom Anak No.	Gen Variabel				
	1	2	3	4	5
1	1	1	1	1	1
2	1	1	1	1	1
3	1	1	1	1	1
4	1	1	1	1	1

Tabel 4.19 Kromosom Anak Setelah Mutasi

Kromosom Anak No.	Gen Variabel				
	1	2	3	4	5
1	1	1	1	1	1
2	1	1	1	1	1
3	1	1	1	1	1
4	1	1	1	1	1

Tabel 4.20 Populasi Kromosom Baru

Kromosom No.	1										2										3										4										5										
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0	0	1	0	0	1	0	1	0	1	1	0	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1	1	1	0	1	0	
2	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0	0	1	0	0	1	0	1	0	1	1	0	1	1	1	1	1	1	0	0	1	0	1	1	1	1	1	1	1	1	0	1	0
3	1	1	1	1	0	0	1	0	1	1	1	0	1	1	1	0	0	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	0	1	1	1	0	1	1	1	0	0	0	0	1	1	1	1	1	1	1	0	0	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	0	1	0	0	1	0	1	0	1	0	1	0	
5	0	0	0	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	0	0	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Dari perbandingan perhitungan manual dengan keluaran yang dihasilkan program, diperoleh solusi yang sama, baik nilai fungsi objektif maupun alokasi *facing* untuk tiap produk. Oleh karena itu, dapat dikatakan bahwa program MATLAB untuk permasalahan SSAP ini telah tervalidasi.

## 4.2 Pengolahan Data

### 4.2.1 Parameter yang Digunakan dalam Algoritma Genetika

Berikut adalah rangkuman parameter-parameter yang digunakan dalam pengolahan data model AG SSAP ini.

**Tabel 4.21** Parameter Model AG SSAP

Parameter Model AG SSAP	
Ukuran Populasi	280
Maks. Generasi	1000
Prob. Crossover	0,25
Prob. Mutasi	0,01
N bit	5
Batas Atas Facing (Ra)	20
Batas Bawah Facing (Rb)	2
Skala Penalti (R)	129.401

Nilai skala penalti (R) diperoleh dari selisih kemungkinan nilai maksimum *fitness* dan minimal *fitness*. Nilai skala R akan secara otomatis terkalkulasi dalam program.

$$= \dots - \dots \quad (4.5)$$

### 4.2.2 Data Input

Data yang dibutuhkan dalam algoritma ini merupakan data dari kategori minuman teh kemasan yang telah dijabarkan pada Bab 3, yaitu data penjualan (*unmodified demand*), keuntungan tiap produk (*gross* keuntungan), kapasitas rak, serta batas minimal dan maksimal *facing* untuk tiap produk.

### 4.2.3 Hasil Pengolahan Data SSAP

Program AG SSAP dijalankan sebanyak 10 kali *run* dengan solusi akhir diambil dari hasil solusi terbaik dari kesepuluh *run* tersebut. Tabel 4.22 menampilkan hasil pengolahan data program untuk setiap *run*-nya.

**Tabel 4.22** Hasil Pengolahan Data (10x *Run*)

	RUN									
	1	2	3	4	5	6	7	8	9	10
<b>FITNESS</b>	321.260	321.510	320.500	321.300	321.010	320.820	321.020	320.990	319.820	319.840
<b>PRODUCT NO.</b>	FACING									
1	20	20	20	20	20	20	20	20	20	20
2	20	20	20	20	20	20	20	20	20	20
3	20	20	20	20	20	20	20	20	20	20
4	6	10	11	5	8	10	3	3	3	4
5	6	3	6	7	7	6	7	6	6	6
6	20	20	16	20	20	15	17	20	10	10
7	20	20	19	20	20	20	20	20	20	20
8	20	20	20	20	20	20	20	20	20	20
9	3	2	6	3	2	2	4	3	2	6
10	6	8	7	7	8	5	6	11	9	7
11	10	19	10	17	10	10	10	10	15	13
12	2	4	3	3	3	2	6	4	3	4
13	3	4	6	9	4	4	3	5	3	6
14	20	20	20	20	20	20	20	20	20	20
15	8	10	11	8	7	11	7	10	19	8
16	20	20	20	19	19	20	20	17	20	18
17	4	3	3	5	11	6	9	11	4	11
18	10	10	15	10	10	13	19	14	9	17
19	13	13	10	9	18	13	13	10	10	10
20	8	3	4	6	3	10	6	3	5	6
21	4	3	3	2	3	3	3	3	2	2
22	10	8	4	4	6	10	3	5	11	8
23	3	2	5	3	2	2	5	3	3	6
24	10	4	6	6	3	7	6	8	8	6
25	4	2	3	4	2	2	4	2	5	2
26	2	2	2	3	6	2	2	4	2	3
27	4	5	6	6	3	2	3	4	6	3
28	2	3	2	2	2	3	2	2	3	2
<b>CAP</b>	1668	1668	1668	1668	1662	1668	1668	1668	1668	1668
<b>RUN TIME (s)</b>	391,583	373,349	374,393	376,522	371,018	371,923	377,945	378,050	386,302	382,396

*Run* ke-dua menunjukkan solusi terbaik dengan *fitness* akhir sebesar 321.250.

```

Masukkan batas atas: 20
Masukkan batas bawah: 2
Masukkan ukuran populasi: 280
Masukkan generasi: 1000

ans =

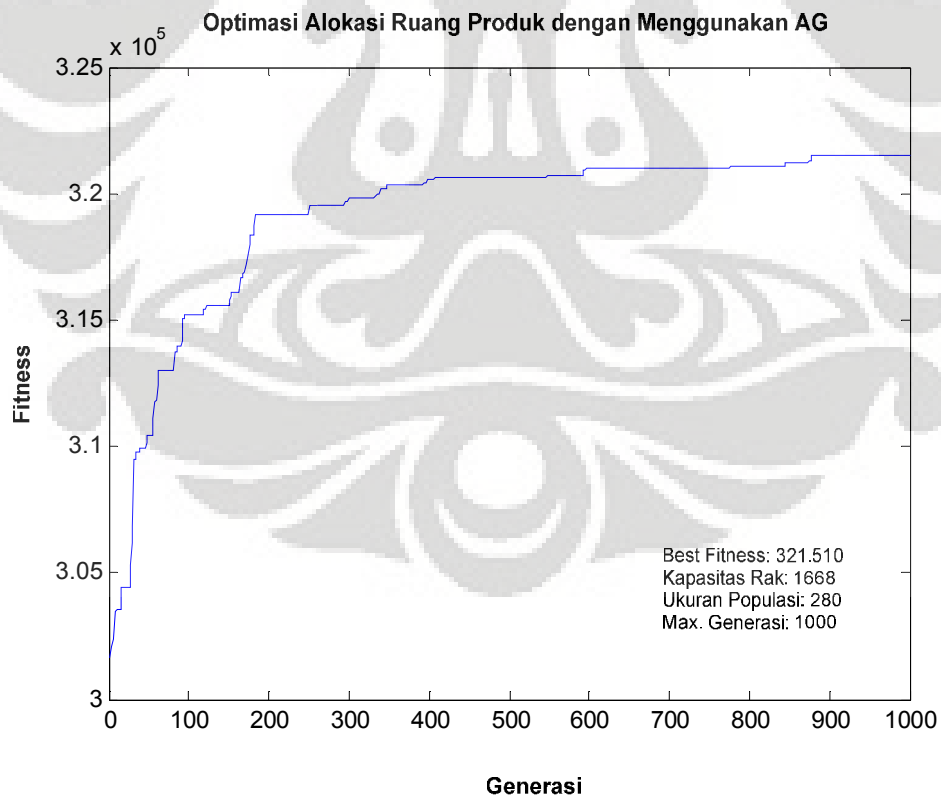
Columns 1 through 17
    20    20    20    10     3    20    20    20     2     8    19     4     4    20    10    20     3

Columns 18 through 28
    10    13     3     3     8     2     4     2     2     5     3

Elapsed time is 373.348557 seconds.
>> |

```

**Gambar 4.2** Tampilan Solusi Terbaik Program Matlab AG-SSAP



**Gambar 4.3** Grafik Solusi Terbaik (*Run 2*)



Solusi alokasi *facing* dari permasalahan optimasi SSAP untuk kategori produk minuman teh kemasan dapat dilihat pada Tabel 4.23.

**Tabel 4.23** Solusi Akhir SSAP

<b>SOLUSI SSAP KATEGORI PRODUK MINUMAN TEH KEMASAN</b>		
<b>NO.</b>	<b>PRODUCT NAME</b>	<b>FACING</b>
1	FRETEA FRUTCY APEL 500 ML	20
2	FRETEA FRUTCY LMN 500 ML	20
3	FRETEA FRUTCY MARKISA 500 ML	20
4	FRETEA GREEN MY BODY ALOEVERA & ORANGE	10
5	FRETEA GREEN MY BODY GINSENG & GINGER 3	3
6	FRETEA GREEN NO CALORIE 500 ML	20
7	FRETEA GREEN TEA PET 500 ML	20
8	FRETEA JASMIN PET 500 ML	20
9	ARTEA APP GREEN TEA 500 ML	2
10	ARTEA HONEY GREEN TEA 500 ML	8
11	ZESTEAM JASMINE GREEN TEA 500 ML	19
12	ZESTEAM JASMINE GREEN TEA LESS SUGAR 500	4
13	ZESTEAM PURE GREEN TEA 500 ML	4
14	NU GREEN TEA HONEY 500 ML	20
15	NU GREEN TEA LESS SUGAR 500 ML	10
16	NU GREEN TEA ORIGINAL 500 ML	20
17	NU GREEN TEA KURMA 500 ML	3
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	10
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	13
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	3
21	JOY TEA GREEN COZY JASMINE 500 ML	3
22	C2 GREEN TEA APPLE 360 ML	8
23	C2 GREEN TEA CLASSIC 360 ML	2
24	FUTAMI OOLONG TEA 350 ML	4
25	FRUIT TEA FUSION 500 ML	2
26	FRUIT TEA BLKCURRENT 500 ML	2
27	FRUIT TEA STR 500 ML	5
28	FRUIT TEA APP 500 ML	3
<b>KEUNTUNGAN KATEGORI PRODUK/ HARI</b>		<b>Rp321.510</b>

#### 4.2.4 Usulan Kebijakan *in-Store Shelf Replenishment*

Setelah solusi untuk masalah optimasi alokasi ruang rak *display* produk telah diperoleh dengan algoritma genetika, peninjauan terhadap manajemen operasional dalam menjaga tingkat persediaan barang di rak juga perlu dilakukan. Dalam hal ini, akan dianalisis mengenai usulan kebijakan *in-store shelf replenishment* yang dapat diterapkan dalam gerai. Kebijakan pengisian rak dibatasi hanya dalam batas ruang lingkup antara gudang penyimpanan (*backroom*) dengan ruang gerai *display* (*frontroom*). Kebijakan *shelf replenishment* berdasar pada model *reorder point* (ROP) yang umum digunakan (Ronald Ballou, 2003):

$$ROP = d \cdot LT + z \cdot s'_d \quad (4.6)$$

Dimana  $s'_d$  adalah

$$s'_d = \sqrt{d \cdot (s_d^2 + LT^2)} \quad (4.7)$$

Keterangan: ROP = *reorder point*  
 $d$  = rata-rata penjualan harian  
 $LT$  = waktu yang dibutuhkan untuk pengiriman (*lead time*)  
 $s'_d$  = standar deviasi selama periode penjualan ditambah  $LT$   
 $s_d$  = standar deviasi distribusi tingkat penjualan harian  
 $z$  = angka standar deviasi dari distribusi tingkat penjualan yang menunjukkan probabilitas barang dalam keadaan *in-stock* selama periode *lead time* (P)

Model ROP pada persamaan di atas digunakan untuk menentukan titik jumlah persediaan barang di rak *display* untuk kemudian dilakukan proses pengisian kembali. Dalam operasional gerai ritel ini, digunakan waktu *lead time* selama 1 hari, dengan asumsi bahwa proses pengecekan tingkat persediaan barang di rak dilakukan pada akhir hari operasi, untuk kemudian proses pengisian dilakukan sebelum gerai dibuka keesokan harinya. Maka, skenario terburuk adalah jika titik ROP jatuh di waktu awal operasional gerai, proses pengisian rak baru akan dilakukan pada hari berikutnya. Dengan menggunakan data distribusi penjualan harian gerai ritel Hypermart untuk kategori produk minuman teh kemasan selama bulan Maret 2010, nilai ROP untuk tiap produk teh kemasan adalah:

**Tabel 4.24** *Reroder Point (ROP)* untuk Tiap Produk

Product No.	Avg. Demand( <i>d</i> )	<i>S<sub>d</sub></i>	<i>S'<sub>d</sub></i>	$SS = z * S'_d$	<i>ROP</i>
1	52	56,08	56,08	91,98	144
2	51	58,51	58,51	95,95	147
3	40	47,12	47,12	77,28	117
4	3	4,41	4,41	7,24	10
5	4	4,22	4,22	6,92	11
6	17	11,89	11,89	19,50	37
7	48	60,47	60,47	99,17	147
8	31	24,67	24,67	40,45	71
9	1	2,57	2,57	4,22	5
10	4	5,10	5,10	8,36	12
11	5	5,70	5,70	9,35	14
12	1	1,65	1,65	2,71	4
13	1	1,01	1,01	1,66	3
14	32	50,87	50,87	83,43	115
15	6	9,36	9,36	15,35	21
16	19	17,08	17,08	28,00	47
17	4	5,95	5,95	9,76	14
18	7	20,38	20,38	33,42	40
19	7	3,71	3,71	6,08	13
20	3	3,48	3,48	5,71	9
21	2	3,71	3,71	6,08	8
22	7	7,67	7,67	12,58	20
23	3	5,51	5,51	9,04	12
24	3	4,91	4,91	8,05	11
25	2	2,90	2,90	4,75	7
26	2	4,63	4,63	7,59	10
27	3	3,56	3,56	5,84	9
28	1	2,35	2,35	3,86	5

Nilai  $z$  yang digunakan adalah sebesar 1,64 yang menunjukkan probabilitas ketersediaan barang selama periode *lead time* sebesar 95%. Angka probabilitas ini merupakan kebijakan yang diterapkan manajemen perusahaan gerai ritel.

Nilai  $Q^*$  atau jumlah produk yang harus diisikan ke rak diperoleh dengan mempertimbangkan aspek kapasitas ruang untuk tiap produk. Kapasitas ruang diperoleh dengan mengalikan jumlah *facing* yang diperoleh tiap produk dengan daya tampung satu baris *facing*, yaitu 12 produk. Kebijakan *full-shelf merchandising* yang diterapkan manajemen gerai ritel *grocery* mengarahkan

proses pengisian rak harus dapat memenuhi kapasitas ruang rak yang dialokasikan untuk tiap produk. Oleh karena itu, nilai  $Q^*$  disesuaikan untuk memenuhi kapasitas ruang *display*, yaitu mengurangi kapasitas maksimum ruang rak *display* untuk tiap produk dengan ROP.

$$* = - \quad (4.8)$$

Tabel 4.25 Nilai  $Q^*$  untuk Tiap Produk

Product No.	Facing	Kapasitas	$Q^*$	Bulk	$Q^*$ (kardus)	Perkiraan Waktu Pengisian (hari)
1	20	240	96	24	4	2
2	20	240	93	24	4	2
3	20	240	123	24	5	3
4	10	120	110	24	5	37
5	3	36	25	24	1	6
6	20	240	203	24	8	12
7	20	240	93	24	4	2
8	20	240	169	24	7	5
9	2	24	19	24	1	19
10	8	96	84	24	3	21
11	19	228	214	24	9	43
12	4	48	44	24	2	44
13	4	48	45	24	2	45
14	20	240	125	24	5	4
15	10	120	99	24	4	16
16	20	240	193	24	8	10
17	3	36	22	24	1	6
18	10	120	80	24	3	11
19	13	156	143	24	6	20
20	3	36	27	24	1	9
21	3	36	28	24	1	14
22	8	96	76	24	3	11
23	2	24	12	24	1	4
24	4	48	37	24	2	12
25	2	24	17	24	1	9
26	2	24	14	24	1	7
27	5	60	51	24	2	17
28	3	36	31	24	1	31

Dikarenakan jumlah *inventory* yang dikeluarkan dari *backroom* harus dalam satuan *bulk* (dus), maka nilai  $Q^*$  dibulatkan ke dalam satuan dus untuk tiap produknya. Pada Tabel 4.25 juga ditampilkan perkiraan waktu pengisian untuk tiap produk dalam satuan hari. Angka ini diperoleh dari pembagian jumlah produk yang diisi dalam satu periode pengisian dengan tingkat *demand* rata-rata. Estimasi waktu ini dapat digunakan untuk memperkirakan kapan produk akan mencapai titik ROP-nya..

$$= \frac{Q^*}{D} \quad (4.9)$$

### 4.3 Analisis

*Shelf Space Allocation Problem* (SSAP) merupakan permasalahan yang ditemui pada setiap gerai ritel *grocery* dimana terdapat sumber daya ruang rak *display* yang terbatas yang harus dialokasikan untuk menampilkan berbagai jenis variasi produk yang dimiliki gerai. Untuk itu dibutuhkan sebuah metode yang aplikatif dan relevan yang dapat digunakan dalam menyelesaikan masalah ini.

Usulan penyelesaian adalah dengan menerapkan salah satu algoritma metaheuristik, yaitu Algoritma Genetika (AG). Dalam penelitian ini, algoritma tersebut diadopsi ke dalam bahasa pemrograman MATLAB guna menjalankan program optimasi SSAP. Analisis terhadap hasil pengolahan data dengan metode tersebut akan dibagi menjadi tiga bagian, yaitu analisis model SSAP, metode, analisis parameter AG, analisis hasil, serta analisis sensitivitas. Selain itu, juga akan dijabarkan analisis mengenai usulan *in-store shelf replenishment*.

#### 4.3.1 Analisis Model SSAP

Pada penelitian ini digunakan simplifikasi model optimasi SSAP yang dikembangkan oleh Bai Rubin (Rubin, 2005). Model simplifikasi SSAP ini memiliki fungsi objektif yang terdiri nilai *gross* keuntungan, tingkat penjualan, *facing*, dan nilai *space elasticity* dari tiap produknya. Permasalahan ini tergolong ke dalam kategori *non-convex optimization problem* karena fungsi objektif yang tidak linear serta adanya kendala *integrality* dalam menentukan variabel solusi. Kendala yang digunakan dalam model merupakan kendala kapasitas rak serta batas minimal dan maksimal *facing* yang dapat dialokasikan pada satu produk.

Model simplifikasi ini disederhanakan dengan menghilangkan beberapa parameter yang terdapat pada model komprehensif SSAP berdasarkan penelitian sebelumnya:

- Nilai *cross elasticities*, yang menunjukkan hubungan interdependensi antara dua produk (Corstjens dan Doyle, 1981). Nilai *cross elasticities* berada pada *range*  $[-1,1]$ , dengan nilai positif untuk dua produk yang komplementer dan negatif untuk produk yang bersifat substitusi terhadap produk lainnya. Walaupun nilai *cross elasticities* berkontribusi dalam menunjukkan hubungan antar produk yang berbeda, sangatlah sulit untuk mendapatkan estimasi yang *reliable* untuk sejumlah nilai *cross elasticities* ( $n \times n$  untuk  $n$  produk), karena adanya hubungan *merchandising* yang kompleks. Banyaknya nilai *cross elasticities* yang harus diestimasi serta kecenderungan bahwa nilai-nilai tersebut akan jauh lebih kecil dari nilai *space elasticity* langsung, menyebabkan banyak periset masa kini menghilangkan parameter *cross elasticities* pada model yang mereka kembangkan (Desmet dan Renaudin, 1998; Timothy Urban, 2002).
- Variabel *marketing*, seperti harga produk, kegiatan promosi, *advertising campaign* dianggap tetap karena kategori produk yang akan diteliti adalah yang berada pada *regular shelf*, sehingga parameter dan *elasticity value* untuk variabel market tersebut dapat diabaikan.
- Dengan mengasumsikan total keuntungan untuk sebuah produk adalah proporsional terhadap keuntungan per unit produk tersebut, maka nilai elastisitas biaya,  $\eta_i$ , dianggap bernilai 1, sehingga fungsi objektif dari model SSAP dapat disederhanakan, yaitu dengan memasukkan langsung nilai keuntungan untuk tiap produk yang diperoleh dari pengurangan pendapatan untuk produk tersebut (harga jual) dengan biaya pengadaan produk (*purchasing cost*).
- Penggunaan nilai *space elasticity* rata-rata untuk tiap produk juga berdampak pada penyamaan nilai *space elasticity* untuk tiap level rak yang merupakan parameter penempatan lokasi produk pada rak *display* seperti yang digambarkan pada model komprehensif SSAP.

Hal ini menyebabkan adanya limitasi dari model simplifikasi SSAP yang digunakan, dimana solusi yang dihasilkan hanya terbatas pada alokasi jumlah *facing* untuk tiap produk tidak sampai pada alokasi penempatan produk pada rak *display*. Limitasi ini juga disebabkan adanya keterbatasan waktu dan ruang penelitian yang dilakukan, dimana untuk mengestimasi tiap nilai *space elasticity* untuk tiap produk di tiap raknya, ataupun nilai efektifitas tiap level rak terhadap penjualan produk dibutuhkan studi eksperimental yang komprehensif yang membutuhkan dana dan dukungan yang besar dari manajemen gerai ritel. Beberapa penelitian sebelumnya dari aspek ergonomi mengenai efektifitas level rak *display* supermarket hanya terbatas hingga *point of attention* tidak *point of purchase*, sedangkan untuk dapat digunakan dalam model SSAP ini, parameter efektifitas mengacu tingkatan level rak yang memberikan kontribusi penjualan terbesar ketika suatu produk diletakkan pada tingkat rak tersebut. Penyamaraan nilai *space elasticity* juga menyebabkan model SSAP yang digunakan tidak dapat menggambarkan karakteristik yang unik dari tiap produk, dimana sebenarnya tiap produk tersebut memiliki nilai *space elasticity* yang berbeda-beda.

Salah satu kekurangan dari banyaknya model SSAP yang telah dikembangkan selama ini adalah banyaknya parameter yang digunakan dalam model, dan sulit untuk menggunakan model-model tersebut pada kasus nyata dikarenakan kesulitan dalam mendapatkan nilai estimasi parameter yang *reliable*. Yang (2001) berargumen bahwa untuk model yang akan digunakan secara komersial, kriteria yang paling penting dalam menentukan metode untuk alokasi ruang *display* adalah kesederhanaan serta kemudahan dalam mengoperasikan metode tersebut (Ruibin Bai, 2005). Oleh karena itu, penyederhanaan model SSAP layak untuk dilakukan guna memudahkan dalam penggunaannya pada kasus nyata. Selain itu, karena banyaknya variasi jenis produk dan gaya bisnis yang dilakukan dalam industri ritel, akan banyak terdapat jenis permasalahan untuk alokasi ruang *display*. Dalam penelitian ini, digunakanlah jenis model SSAP yang paling umum yang telah banyak menjadi subjek penelitian dalam riset-



riset akademis sebelumnya, yang memfokuskan pada model SSAP untuk gerai ritel *grocery* (supermarket).

#### 4.3.2 Analisis Metode AG

Penyelesaian permasalahan SSAP menggunakan pendekatan metode *population-based search*, yaitu Algoritma Genetika (AG). Permasalahan SSAP yang bersifat NP-hard menyebabkan sulitnya menemukan prosedur solusi penyelesaian eksak dengan batasan *polynomial time bounded* yang dapat menyelesaikan setiap jenis permasalahan secara optimal. *Dynamic programming* awalnya digunakan untuk mengoptimasikan model *shelf space allocation*. Namun, metode ini membutuhkan waktu komputasi yang sangat tinggi untuk jenis permasalahan besar. Permasalahan berjenis NP dapat diselesaikan dalam *polynomial time* dengan menggunakan algoritma *nondeterministic* (Ruibin Bai, 2005). Pada dasarnya, algoritma *nondeterministic* terdiri dari dua tahapan. Tahap pertama dari algoritma ini secara sederhana mengira-ngira struktur  $S$  untuk problem  $I$ , dimana selanjutnya akan digunakan sebagai *input* untuk tahap selanjutnya yaitu mengecek apakah struktur  $S$  tersebut merupakan solusi untuk permasalahan jenis  $I$ . Pendekatan pencarian solusi dengan menggunakan metode metaheuristik *population-based search* memiliki prinsip yang sama dengan penjabaran algoritma *nondeterministic*. Oleh karena itu, algoritma genetika yang merupakan salah satu bentuk metode metaheuristik *population-based search* dinilai tepat untuk digunakan dalam penyelesaian masalah SSAP ini. AG merupakan bentuk algoritma metaheuristik yang sederhana dan dapat menghasilkan kualitas solusi yang mendekati optimal. Selain itu, metode AG juga terbukti efektif dalam menyelesaikan problem SSAP berdasarkan penelitian yang telah dilakukan sebelumnya.

Dalam algoritma genetika, prinsip dasar yang digunakan adalah evolusi biologi Darwin, dimana hanya individu yang terkuatlah yang akan mampu bertahan hingga banyak generasi, sedangkan individu yang lemah akan musnah. Algoritma genetika diawali dengan pembuatan populasi awal dengan membangkitkan bilangan random 0 atau 1 sebanyak jumlah kromosom dalam satu populasi. Selanjutnya, individu yang akan digunakan dalam proses kawin silang

dan mutasi juga dipilih secara acak dengan membandingkan bilangan random yang dibangkitkan dengan proporsi *fitness* yang dimiliki tiap individu. Penentuan dilakukannya kedua proses genetika tersebut juga dilakukan dengan membangkitkan bilangan random, dimana jika bilangan tersebut lebih kecil dari probabilitas kawin silang atau probabilitas mutasi, maka individu tersebut berhak untuk dilakukan kawin silang/ mutasi. Bilangan random untuk proses mutasi dibangkitkan untuk tiap gen pada tiap kromosom dalam satu populasi. Titik potong untuk proses kawin silang juga ditentukan dengan membangkitkan satu bilangan acak yang terdapat dalam rentang 1 hingga banyaknya jumlah gen dalam satu kromosom.

Banyaknya proses dalam algoritma genetika yang berdasarkan pada bilangan random yang dibangkitkan, menyebabkan hasil solusi yang ditampilkan pada tiap *run* program akan berbeda, namun tetap dalam *range* nilai *fitness* yang berdekatan. Selain itu, walaupun hasil nilai *fitness* yang dihasilkan pada tiap *run*-nya berbeda, namun program AG akan mencari solusi yang paling mendekati maksimum tanpa melanggar kendala yang ada. Hal ini dapat dilihat dari samanya jumlah kapasitas ruang *display* yang menjadi solusi akhir dari program, yaitu 1668 cm. Model SSAP yang digunakan memiliki kendala batasan kapasitas ruang rak *display* sebesar 1672 cm. Hal ini menunjukkan bahwa program AG akan berusaha untuk selalu mencari solusi terbaik dari ruang solusi yang ada.

Kombinasi parameter AG yang digunakan, seperti jumlah populasi, panjang kromosom (jumlah bit yang merepresentasikan vairabel keputusan), maksimal generasi probabilitas kawin silang, dan probabilitas mutasi akan berpengaruh terhadap kualitas solusi yang akan diperoleh. Oleh karena itu, paramater AG yang digunakan harus ditentukan dengan tetap dan sesuai dengan permasalahan yang dihadapi agar dapat menghasilkan solusi yang optimal.

Ukuran populasi dalam algoritma genetika biasanya berkisar antara  $n$  hingga  $2n$ , dengan  $n$  merupakan panjang kromosom. Dari studi eksperimen yang dilakukan sebelumnya disimpulkan bahwa banyaknya ukuran populasi akan mempengaruhi kualitas solusi yang dihasilkan, dimana semakin banyak jumlah individu dalam populasi, maka solusi yang dihasilkan akan lebih baik. Hal ini dikarenakan semakin banyak jumlah individu, maka akan semakin banyak pula

ruang solusi yang dapat dipilih oleh program, sehingga peluang untuk memperoleh solusi dengan nilai *fitness* lebih baik akan semakin besar pula, dibandingkan dengan jika memiliki individu dalam jumlah yang lebih kecil. Hasil yang sama juga ditampilkan dalam proses *trial* yang dilakukan untuk menentukan jumlah populasi yang akan digunakan dalam pengolahan data.

**Tabel 4.26** *Trial* Pengaruh Ukuran Populasi terhadap Kualitas Solusi

Ukuran Populasi	Computation Time (detik)	Hasil (Keuntungan)	Rata-rata Computation Time (detik)	Rata-rata Hasil (Keuntungan)
<b>140</b>	114,513	318.680	110,7936	<b>319.872</b>
	109,847	319.740		
	110,248	320.410		
	110,894	319.610		
	108,466	320.920		
<b>185</b>	179,697	320.310	178,826	<b>320.088</b>
	179,861	319.080		
	176,623	320.670		
	178,454	320.460		
	179,495	319.920		
<b>235</b>	272,466	320.930	269,634	<b>320.718</b>
	273,338	321.350		
	270,096	320.940		
	267,245	319.420		
	265,025	320.950		
<b>280</b>	367,920	321.180	368,4786	<b>320.980</b>
	368,366	321.560		
	364,564	320.980		
	370,159	319.760		
	371,384	321.420		

Dengan menjaga parameter lainnya konstan, diperoleh kesimpulan yang sama dengan hasil yang diperoleh Alexander (1992) dalam percobaannya. Semakin besar ukuran populasi, maka kualitas solusi yang dihasilkan akan semakin baik pula. Perlu diketahui bahwa panjang kromosom untuk permasalahan ini adalah sebanyak 140 gen ( $N$  variabel  $\times N$  bit). Oleh karena itu, dipilihlah ukuran populasi sebanyak  $2n$  (280) untuk digunakan dalam metode penyelesaian masalah SSAP ini.

Hal yang sama juga berlaku untuk parameter maksimal generasi. Algoritma genetika biasanya memiliki jumlah generasi antara 50 dan 500 atau lebih. Semakin banyak generasi yang dilakukan, peluang untuk memperoleh hasil yang lebih baik akan semakin besar pula. Kesimpulan yang sama juga didapatkan dari proses *trial* yang dilakukan berikut ini. Maka, dipilihlah angka maksimal generasi yang besar (1000) guna menjamin kualitas solusi yang dihasilkan program.

**Tabel 4.27** *Trial* Pengaruh Jumlah Generasi terhadap Kualitas Solusi

Generasi	<i>Computation Time</i> (detik)	Hasil (Keuntungan)	Rata-rata <i>Computation Time</i> (detik)	Rata-rata Hasil (Keuntungan)
50	19,178	307.260	18,9036	<b>309.552</b>
	19,161	306.840		
	18,593	307.720		
	18,731	311.200		
	18,855	314.740		
200	76,367	317.860	75,7962	<b>317.792</b>
	75,922	316.830		
	74,968	319.630		
	75,396	319.770		
	76,328	314.870		
350	132,716	319.380	132,564	<b>319.758</b>
	132,813	319.970		
	131,270	320.210		
	130,618	319.340		
	135,403	319.890		
500	188,258	319.660	188,715	<b>320.160</b>
	193,393	320.940		
	185,969	319.690		
	186,398	320.950		
	189,558	319.560		

Dalam menentukan panjang kromosom sebagai representasi solusi, digunakan jumlah bit sebanyak 5 untuk digunakan merepresentasikan tiap variabel keputusan yang ada. Dengan menggunakan jumlah bit 5, maka kombinasi dari kode yang dapat dihasilkan untuk alokasi *facing* tiap produk sudah dapat mencakup keseluruhan kemungkinan alokasi *facing* dari *range* batas minimal

*facing* (2) hingga batas maksimalnya (20). Panjang *string* kromosom untuk permasalahan SSAP dengan 28 produk adalah sebanyak 140 gen ( $28 \times 5$ ).

Untuk parameter lainnya seperti probabilitas kawin silang dan mutasi, kombinasi angka probabilitas yang optimal untuk permasalahan SSAP diperoleh dari penelitian yang telah dilakukan sebelumnya, yaitu 0,25 untuk probabilitas kawin silang dan 0.01 untuk probabilitas mutasi. Nilai probabilitas 0,25 menunjukkan bahwa hanya 25% dari kromosom awal (kromosom orang tua) yang mengalami proses kawin silang sebelum menjadi individu populasi baru. Sedangkan nilai probabilitas mutasi 0,01 menunjukkan bahwa hanya sekitar 1 gen dari tiap kromosomnya yang mengalami proses mutasi. Kecilnya angka probabilitas kawin silang dan mutasi yang digunakan dimaksudkan agar individu yang berpotensi menjadi solusi, yaitu bernilai *fitness* tinggi tidak rusak akibat dilakukannya proses genetika. Hal ini dilakukan untuk menjaga kualitas individu yang dihasilkan dari proses algoritma genetika ini. Perlu diketahui bahwa kombinasi probabilitas kawin silang dan mutasi dapat bervariasi tergantung dari jenis permasalahan yang dihadapi.

Metode penggantian populasi yang digunakan pada algoritma genetika ini ialah penggantian populasi dengan skema *generational replacement*, yaitu penggantian seluruh individu pada populasi awal/ lama dengan individu baru hasil proses genetika. Alasan pemilihan metode ini dibandingkan dengan metode *steady-state reproduction* ialah:

- Waktu komputasi yang lebih singkat dengan metode *generational replacement*. Pada metode *generational replacement* hanya dilakukan satu kali evaluasi individu yaitu di awal iterasi, sedangkan pada *steady-state reproduction* dilakukan dua kali evaluasi individu yaitu pada awal iterasi, serta pada individu baru setelah dilakukannya proses genetika kawin silang dan mutasi. Proses komputasi pada *steady-state reproduction* tentunya akan memakan waktu lebih lama daripada metode *generational replacement*.
- Metode *generational replacement* dikombinasikan dengan proses elitisme untuk menjaga agar individu terbaik dari populasi sebelumnya tidak hilang. Elitisme merupakan proses duplikasi

individu terbaik di populasi awal untuk kemudian dimasukkan secara langsung menjadi individu populasi baru. Dengan adanya proses elitisme ini, maka tidak perlu dilakukan evaluasi individu kembali setelah proses genetika kawin silang dan mutasi, karena walaupun nilai *fitness* dari individu baru tidak lebih baik dari individu lama, individu terbaik dari populasi lama tetap akan terjaga sehingga nilai *fitness* terbaik dari populasi baru setidaknya akan sama dengan nilai *fitness* terbaik dari populasi lama.

Program algoritma genetika untuk penyelesaian permasalahan SSAP dalam perangkat MATLAB dapat berjalan dengan baik. Kekurangan dari program AG yang dikembangkan adalah waktu komputasi yang cukup lama, yaitu sekitar 6 menit untuk ukuran problem yang diselesaikan, yaitu 28 produk. Komputasi ini akan memakan waktu lebih banyak jika program dihadapkan pada ukuran problem yang lebih besar. Lamanya waktu komputasi tentunya juga disebabkan dari besarnya nilai parameter AG yang digunakan, seperti ukuran populasi dan maksimal generasi. Seperti yang dapat dilihat pada Tabel 4.26 dan 4.27, waktu komputasi akan lebih lama untuk ukuran parameter jumlah populasi dan maksimum generasi yang lebih besar. Namun, walaupun terbilang cukup lama, waktu komputasi untuk program AG ini masih dalam batas wajar.

#### 4.3.3 Analisis Hasil Pengolahan Data

Untuk penyelesaian permasalahan SSAP untuk kategori produk minuman teh kemasan, program AG yang dikembangkan menghasilkan solusi dengan nilai keuntungan kategori produk tersebut sebesar Rp 321.510. Seperti yang telah ditampilkan sebelumnya, rincian alokasi *facing* untuk tiap produk dapat dilihat pada Tabel 4.28.

Dari tabel solusi alokasi *facing* terlihat bahwa terdapat 8 produk yang memiliki alokasi *facing* maksimum, yaitu 20 *facing*. Kedelapan produk tersebut merupakan produk-produk unggulan yang memiliki nilai *demand* yang tinggi, yaitu berkisar di atas 17 unit/hari. Selain itu, nilai keuntungan/unit dari tiap produk tersebut juga tergolong tinggi dengan kisaran Rp800/unit sehingga hasil



pengolahan data yang dijalankan program akan mengarahkan pada alokasi maksimal *facing* untuk produk-produk tersebut.

Untuk produk lainnya, alokasi *facing* untuk tiap produk terlihat lebih dinamis dalam mempertimbangkan hubungan antara tingkat penjualan dan nilai keuntungan untuk tiap produk. Hal ini salah satunya terlihat pada solusi alokasi *facing* untuk produk Sosro Teh Botol dengan Joy Tea Green Jasmine. Kedua produk tersebut memiliki tingkat penjualan yang sama, yaitu 7 unit/hari, namun program AG menghasilkan alokasi yang lebih tinggi untuk produk Joy Tea Green Jasmine yaitu sebanyak 13 *facing*, daripada untuk produk Sosro Teh Botol yang dialokasikan sebanyak 10 *facing*. Hal ini dikarenakan Joy Tea Green Jasmine memiliki tingkat keuntungan/unit produk yang lebih besar daripada produk Sosro Teh Botol sehingga program AG yang dibuat akan cenderung untuk mengalokasikan *facing* lebih banyak untuk produk Joy Tea Green Jasmine.

Selain itu, dari solusi yang dihasilkan, dapat dilihat pula sebuah produk dapat dialokasikan jumlah *facing* lebih banyak walaupun tingkat penjualannya lebih kecil dari produk lain, seperti yang terjadi pada produk Zestea Jasmine Green Tea. Tingkat penjualan produk ini tergolong rendah, hanya 5 unit/hari, namun produk tersebut mendapatkan alokasi *facing* yang besar karena produk memiliki nilai keuntungan/unit tertinggi dalam kategori tersebut. Produk ini dianggap dapat mendatangkan keuntungan yang lebih besar bagi kategori produk minuman teh kemasan akibat *resulted demand* dari jumlah alokasi *facing* yang besar. Namun, hal yang sama tidak terjadi untuk dua *brand* Zestea lainnya, karena dua produk lainnya memiliki tingkat penjualan terendah dalam kategori produk minuman kemasan, sehingga masih akan lebih menguntungkan untuk menambah alokasi *facing* yang memiliki tingkat penjualan lebih tinggi daripada kedua produk Zestea tersebut walaupun nilai keuntungan/unit-nya lebih kecil dari produk Zestea.

Jika membandingkan dengan alokasi *facing* untuk produk teh kemasan yang digunakan saat ini di gerai ritel Hypermart seperti yang ditampilkan pada Tabel 4.29, program AG menghasilkan solusi alokasi *facing* yang lebih baik. Hal ini dapat terlihat dari perbedaan nilai keuntungan kategori produk dimana terjadi kenaikan sebesar 7,30% yang dapat dicapai dengan alokasi *facing* hasil keluaran



program AG. Pada tabel tersebut dapat dilihat pula terdapat beberapa produk yang memiliki perbedaan alokasi *facing* yang cukup besar antar alokasi yang digunakan saat ini dengan usulan alokasi *facing* yang dikeluarkan program. Perbedaan inilah yang mempengaruhi peningkatan nilai keuntungan dari yang dapat diraih saat ini dengan perkiraan yang dapat diraih dengan menggunakan solusi yang dikeluarkan program. Implementasi model SSAP dengan algoritma genetika terbukti dapat menghasilkan solusi yang cukup optimal untuk permasalahan ini.

**Tabel 4.28** Solusi Akhir Alokasi *Facing* Produk Minuman Teh Kemasan

SOLUSI SSAP KATEGORI PRODUK MINUMAN TEH KEMASAN				
NO.	PRODUCT NAME	FACING	DEMAND/ DAY	KEUNTUNGAN
1	FRESTEA FRUTCY APEL 500 ML	20	52	853,03
2	FRESTEA FRUTCY LMN 500 ML	20	51	853,03
3	FRESTEA FRUTCY MARKISA 500 ML	20	40	853,03
4	FRESTEA GREEN MY BODY ALOEVERA & ORANGE	10	3	903,03
5	FRESTEA GREEN MY BODY GINSENG & GINGER 3	3	4	903,03
6	FRESTEA GREEN NO CALORIE 500 ML	20	17	859,85
7	FRESTEA GREEN TEA PET 500 ML	20	48	859,85
8	FRESTEA JASMIN PET 500 ML	20	31	859,85
9	ARTEA APP GREEN TEA 500 ML	2	1	677,27
10	ARTEA HONEY GREEN TEA 500 ML	8	4	677,27
11	ZESTEA JASMINE GREEN TEA 500 ML	19	5	1098,58
12	ZESTEA JASMINE GREEN TEA LESS SUGAR 500	4	1	1098,58
13	ZESTEA PURE GREEN TEA 500 ML	4	1	1098,58
14	NU GREEN TEA HONEY 500 ML	20	32	865,91
15	NU GREEN TEA LESS SUGAR 500 ML	10	6	865,91
16	NU GREEN TEA ORIGINAL 500 ML	20	19	865,91
17	NU GREEN TEA KURMA 500 ML	3	4	865,91
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	10	7	602,3
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	13	7	702,3
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	3	3	702,27
21	JOY TEA GREEN COZY JASMINE 500 ML	3	2	702,27
22	C2 GREEN TEA APPLE 360 ML	8	4	436,36
23	C2 GREEN TEA CLASSIC 360 ML	2	2	436,36
24	FUTAMI OOLONG TEA 350 ML	4	3	636,82
25	FRUIT TEA FUSION 500 ML	2	1	676,52
26	FRUIT TEA BLKCURRENT 500 ML	2	1	676,52
27	FRUIT TEA STR 500 ML	5	2	676,52
28	FRUIT TEA APP 500 ML	3	1	676,52

**Tabel 4.29** Perbandingan Solusi Program dengan Alokasi *Facing* Saat ini

NO.	PRODUCT NAME	FACING HASIL PROGRAM	FACING SAAT INI
1	FRESTEA FRUTCY APEL 500 ML	20	9
2	FRESTEA FRUTCY LMN 500 ML	20	12
3	FRESTEA FRUTCY MARKISA 500 ML	20	14
4	FRESTEA GREEN MY BODY ALOEVERA & ORANGE	10	12
5	FRESTEA GREEN MY BODY GINSENG & GINGER 3	3	16
6	FRESTEA GREEN NO CALORIE 500 ML	20	15
7	FRESTEA GREEN TEA PET 500 ML	20	20
8	FRESTEA JASMIN PET 500 ML	20	15
9	ARTEA APP GREEN TEA 500 ML	2	7
10	ARTEA HONEY GREEN TEA 500 ML	8	7
11	ZESTEA JASMINE GREEN TEA 500 ML	19	2
12	ZESTEA JASMINE GREEN TEA LESS SUGAR 500	4	2
13	ZESTEA PURE GREEN TEA 500 ML	4	12
14	NU GREEN TEA HONEY 500 ML	20	20
15	NU GREEN TEA LESS SUGAR 500 ML	10	14
16	NU GREEN TEA ORIGINAL 500 ML	20	14
17	NU GREEN TEA KURMA 500 ML	3	20
18	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	10	2
19	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	13	4
20	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	3	8
21	JOY TEA GREEN COZY JASMINE 500 ML	3	8
22	C2 GREEN TEA APPLE 360 ML	8	16
23	C2 GREEN TEA CLASSIC 360 ML	2	5
24	FUTAMI OOLONG TEA 350 ML	4	2
25	FRUIT TEA FUSION 500 ML	2	4
26	FRUIT TEA BLKCURRENT 500 ML	2	10
27	FRUIT TEA STR 500 ML	5	4
28	FRUIT TEA APP 500 ML	3	4
<b>KEUNTUNGAN KATEGORI PRODUK/ HARI</b>		<b>Rp321.510</b>	<b>Rp299.643</b>
<b>% KENAIKAN KEUNTUNGAN</b>		<b>7,30%</b>	

Prinsip dasar program AG yang merupakan metode penyelesaian solusi dengan pencarian acak menyebabkan solusi yang dihasilkan dapat berbeda-beda dalam setiap *run*-nya. Dalam permasalahan SSAP, untuk beberapa produk yang tidak menonjol dalam hal tingkat penjualan maupun nilai keuntungan/unit-nya, hasil solusi alokasi *facing* yang dihasilkan dapat berfluktuasi dari *run* yang satu dengan *run* lainnya. Namun, program ini akan terus bekerja untuk menghasilkan

solusi dengan nilai fungsi objektif yang maksimal dari ruang solusi yang dicarinya. Metode algoritma genetika dinilai dapat bekerja secara efektif dalam menyelesaikan permasalahan SSAP dan tentunya metode ini dapat diimplementasikan pada sistem manajemen ritel yang ada saat ini.

#### 4.3.4 Analisis Sensitivitas Model SSAP

Analisis sensitivitas merupakan sebuah metode yang digunakan untuk menganalisis pengaruh perubahan data terhadap solusi optimal. Analisis sensitivitas biasanya dilakukan melalui eksperimen dimana beberapa nilai parameter yang berbeda digunakan untuk menyelesaikan permasalahan optimasi untuk melihat bagaimana perilaku model merespon terhadap perubahan nilai-nilai parameter tersebut. Pada model SSAP ini, analisis sensitivitas dilakukan terhadap kendala minimal dan maksimal *facing* yang menentukan tingkat *exposure* yang dapat diterima oleh sebuah produk, serta nilai koefisien *space elasticity*.

##### 4.3.4.1 Analisis Sensitivitas terhadap Kendala Batas Alokasi Facing

Pada model SSAP untuk permasalahan alokasi ruang produk kategori minuman teh kemasan, batas minimal *facing* yang digunakan adalah sebesar 2 *facing*, sedangkan batas maksimal adalah 20 *facing* untuk sebuah produk. Dengan mengkategorikan produk ke dalam klasifikasi ABC, pada Tabel 4.29 dapat terlihat bahwa 35,97% dari total *facing* dialokasikan untuk produk kelas A, 42,45% untuk produk kelas B, dan 21,58% untuk produk kelas C, serta nilai kategori keuntungan sebesar 321.510.

Analisis sensitivitas dilakukan dengan merubah parameter minimal *facing* dan maksimal *facing* pada model SSAP. Perilaku model dari perubahan parameter minimal *facing* dengan menjaga nilai maksimal *facing* tetap ( $R_a=20$ ) dapat dilihat pada Tabel 4.31. Secara garis besar nilai kategori keuntungan yang dapat diperoleh akan semakin berkurang seiring dengan meningkatnya nilai minimal *facing* yang digunakan. Hal ini dikarenakan terjadinya pergeseran alokasi *facing* dari produk-produk dengan tingkat *demand* dan keuntungan yang lebih tinggi kepada produk-produk dengan tingkat yang lebih rendah guna memenuhi batas minimal *facing* yang ditentukan. Ketika menggunakan batas minimal *facing*

yang rendah ( $R_b = 2$ ), beberapa produk dalam kelas C dialokasikan memiliki jumlah *facing* minimal karena nilai *demand* dan keuntungannya tergolong sangat rendah. Namun, ketika kendala batas minimal *facing* dinaikkan, maka program harus dapat memenuhi kendala tersebut terlebih dahulu sebelum mengalokasikan ekstra *facing* kepada produk-produk lain yang lebih menguntungkan.

**Tabel 4.30** Alokasi *Facing* Berdasarkan Klasifikas Produk (Min. *Facing* = 2; Max. *Facing* = 20)

KELAS	PRODUK	FACING	TOTAL FACING	PERSENTASE FACING
A	FRESTEA FRUTCY APEL 500 ML	20	100	35,97%
	FRESTEA FRUTCY LMN 500 ML	20		
	FRESTEA GREEN NO CALORIE 500 ML	20		
	FRESTEA FRUTCY MARKISA 500 ML	20		
	NU GREEN TEA HONEY 500 ML	20		
B	FRESTEA JASMIN PET 500 ML	20	118	42,45%
	NU GREEN TEA ORIGINAL 500 ML	20		
	FRESTEA GREEN NO CALORIE 500 ML	20		
	NU GREEN TEA LESS SUGAR 500 ML	10		
	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	13		
	SOSRO TEHBOTOL JASMINE LESS SUGAR 500ML	10		
	ZESTEA JASMINE GREEN TEA 500 ML	19		
	FRESTEA GREEN MY BODY GINSENG & GINGER 3	3		
	NU GREEN TEA KURMA 500 ML	3		
C	C2 GREEN TEA APPLE 360 ML	8	60	21,58%
	ARTEA HONEY GREEN TEA 500 ML	8		
	FRESTEA GREEN MY BODY ALOEVERA & ORANGE	10		
	FRUIT TEA STR 500 ML	5		
	JOY TEA GREEN COZY JASMINE 500 ML	3		
	FUTAMI OOLONG TEA 350 ML	4		
	FRUIT TEA FUSION 500 ML	2		
	FRUIT TEA BLKCURRENT 500 ML	2		
	JOY TEA GREEN CHEERFUL MADU LEMON 500 ML	3		
	C2 GREEN TEA CLASSIC 360 ML	2		
	ZESTEA JASMINE GREEN TEA LESS SUGAR 500	4		
	ZESTEA PURE GREEN TEA 500 ML	4		
	FRUIT TEA APP 500 ML	3		
	ARTEA APP GREEN TEA 500 ML	2		
	<b>TOTAL FACING</b>			
<b>KATEGORI KEUNTUNGAN</b>			321.510	

**Tabel 4.31** Analisis Sensitivitas Kendala Minimal *Facing* terhadap Alokasi Ruang Produk

KELAS	PRODUK	FACING					
		Rb = 2	Rb = 3	Rb = 4	Rb = 5	Rb = 6	Rb = 7
A	FRETEA FRUTCY APEL 500 ML	20	20	20	20	20	20
	FRETEA FRUTCY LMN 500 ML	20	20	20	20	20	20
	FRETEA GREEN NO CALORIE 500 ML	20	20	20	20	20	20
	FRETEA FRUTCY MARKISA 500 ML	20	20	20	20	16	18
	NU GREEN TEA HONEY 500 ML	20	20	20	20	19	17
B	FRETEA JASMIN PET 500 ML	20	20	11	20	15	19
	NU GREEN TEA ORIGINAL 500 ML	20	17	20	16	15	9
	FRETEA GREEN NO CALORIE 500 ML	20	19	20	20	15	11
	NU GREEN TEA LESS SUGAR 500 ML	10	6	8	8	9	9
	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	13	11	7	11	8	8
	SOSRO TEHBOTOL JASMINE LESS SUGAR	10	15	15	6	15	7
	ZESTE JASMINE GREEN TEA 500 ML	19	13	11	8	7	7
	FRETEA GREEN MY BODY GINSENG & GINGER	3	6	5	7	7	7
NU GREEN TEA KURMA 500 ML	3	5	7	5	6	7	
C	C2 GREEN TEA APPLE 360 ML	8	5	7	5	6	7
	ARTEA HONEY GREEN TEA 500 ML	8	6	5	6	6	7
	FRETEA GREEN MY BODY ALOEVERA	10	6	7	6	7	7
	FRUIT TEA STR 500 ML	5	5	4	7	6	7
	JOY TEA GREEN COZY JASMINE 500 ML	3	6	5	6	7	7
	FUTAMI OOLONG TEA 350 ML	4	8	4	6	6	7
	FRUIT TEA FUSION 500 ML	2	3	4	5	6	7
	FRUIT TEA BLKCURRENT 500 ML	2	5	4	5	6	7
	JOY TEA GREEN CHEERFUL MADU LEMON	3	3	8	5	6	8
	C2 GREEN TEA CLASSIC 360 ML	2	3	6	6	6	7
	ZESTE JASMINE GREEN TEA LESS SUGAR 500	4	5	7	5	6	7
	ZESTE PURE GREEN TEA 500 ML	4	4	4	5	6	7
	FRUIT TEA APP 500 ML	3	3	5	5	6	7
	ARTEA APP GREEN TEA 500 ML	2	4	4	5	6	7

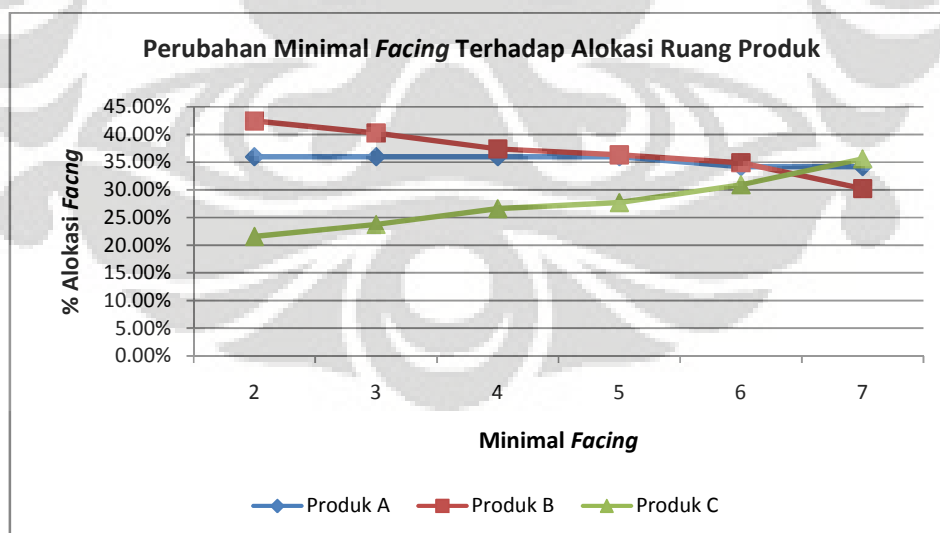
Pergeseran alokasi *facing* cenderung dilakukan pada produk-produk dalam kelas B untuk memenuhi batas minimal *facing* pada produk-produk di kelas C. Hal ini terlihat dari proporsi produk kelas B yang semakin mengecil pada nilai Rb 3, 4, dan 5 sedangkan proporsi untuk produk kelas A tetap. Produk kelas A merupakan produk-produk yang memiliki karakteristik tingkat *demand* dan keuntungan yang sangat kuat, sehingga produk-produk tersebut menjadi prioritas

untuk memperoleh alokasi *facing* yang maksimal. Ketika nilai batas minimal *facing* semakin dinaikkan, pergeseran alokasi ruang juga mulai dilakukan dari produk A, terlihat dari proporsi alokasi *facing* produk kelas A yang berkurang dari sebelumnya.

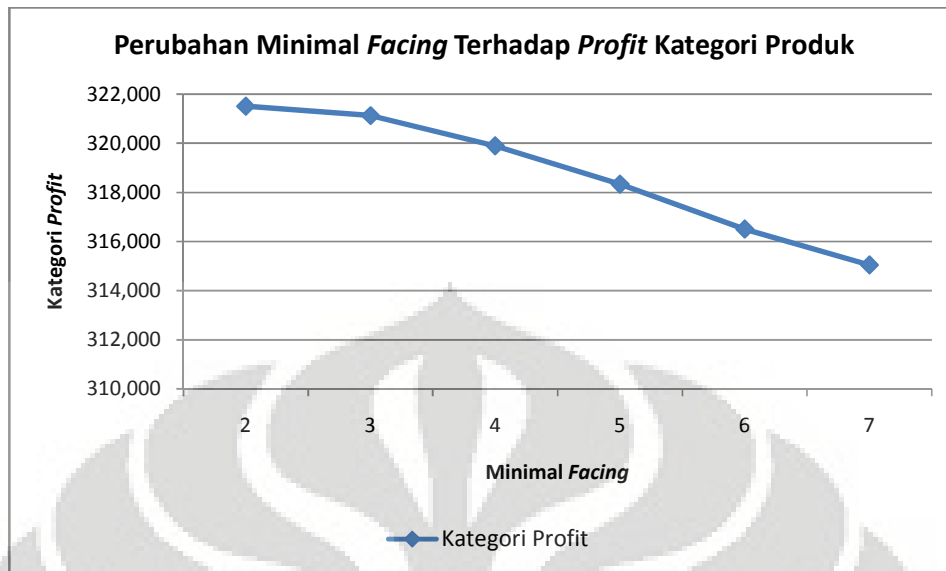
Jika dibandingkan dengan alokasi *facing* yang digunakan saat ini, kenaikan batas minimal *facing* hingga  $R_b = 7$  masih dapat menghasilkan nilai kategori keuntungan yang lebih tinggi 5,14% daripada kategori keuntungan saat ini.

**Tabel 4.32** Analisis Sensitivitas Kendala Minimal *Facing* terhadap Nilai Keuntungan Kategori Produk

	Kendala Minimal Facing					
	2	3	4	5	6	7
<b>Alokasi Facing</b>						
Produk A	35,97%	35,97%	35,97%	35,97%	34,17%	34,17%
Produk B	42,45%	40,29%	37,41%	36,33%	34,89%	30,22%
Produk C	21,58%	23,74%	26,62%	27,70%	30,94%	35,61%
Kategori Keuntungan	321.510	321.130	319.900	318.340	316.510	315.050
Total Facing	278	278	278	278	278	278
% Perubahan Keuntungan		-0,118%	-0,501%	-0,986%	-1,555%	-2,009%



**Gambar 4.4** Perubahan Minimal *Facing* Terhadap Alokasi Ruang Produk



**Gambar 4.5** Grafik Perubahan Minimal *Facing* Terhadap Keuntungan Produk

**Tabel 4.33** Perbandingan Sensitivitas Minimal *Facing* dengan Alokasi *Facing* Saat Ini

	Minimal Facing						
	2 (Current)	2 (Program)	3	4	5	6	7
<b>Alokasi Facing</b>							
Produk A	26,98%	35,97%	35,97%	35,97%	35,97%	34,17%	34,17%
Produk B	36,69%	42,45%	40,29%	37,41%	36,33%	34,89%	30,22%
Produk C	36,33%	21,58%	23,74%	26,62%	27,70%	30,94%	35,61%
Kategori Keuntungan	299.643	321.510	321.130	319.900	318.340	316.510	315.050
Total Facing	278	278	278	278	278	278	278
% Perubahan Keuntungan		7,30%	7,17%	6,76%	6,24%	5,63%	5,14%

Analisis sensitivitas terhadap batas maksimal *facing* dilakukan dengan mengurangi dan menaikkan nilai batas parameter tersebut dengan menjaga parameter minimal *facing* tetap ( $R_b = 2$ ). Dari hasil eksperimen yang diperoleh, terlihat bahwa produk kelas A mendapatkan alokasi *facing* yang maksimal pada setiap *run* percobaan dengan nilai batas maksimal yang berbeda. Hal ini menunjukkan keunggulan dari produk-produk tersebut yang memiliki tingkat *demand* dan keuntungan yang tinggi. Di sisi lain, ketika kita mengurangi batas maksimal *facing*, rentang antara batas minimal dan maksimal akan semakin



mengecil. Hal ini menyebabkan alokasi *facing* menjadi lebih merata untuk seluruh produk. Pada nilai maksimal *facing* 14, tidak terdapat produk yang memperoleh alokasi *facing* minimal 2, dan juga persebaran alokasi *facing* terlihat lebih merata.

**Tabel 4.34** Analisis Sensitivitas Kendala Maksimal *Facing* terhadap Alokasi Ruang Produk

KELAS	PRODUK	FACING					
		Ra = 14	Ra = 16	Ra = 18	Ra = 20	Ra = 22	Ra = 25
A	FRESTEA FRUTCY APEL 500 ML	14	16	18	20	22	25
	FRESTEA FRUTCY LMN 500 ML	14	16	18	20	22	25
	FRESTEA GREEN NO CALORIE 500 ML	14	16	18	20	22	25
	FRESTEA FRUTCY MARKISA 500 ML	14	16	18	20	22	25
	NU GREEN TEA HONEY 500 ML	14	16	18	20	22	25
B	FRESTEA JASMIN PET 500 ML	14	16	18	20	22	25
	NU GREEN TEA ORIGINAL 500 ML	14	16	18	20	20	18
	FRESTEA GREEN NO CALORIE 500 ML	14	16	18	20	16	13
	NU GREEN TEA LESS SUGAR 500 ML	14	6	5	10	6	7
	JOY TEA GREEN JASMINE LESS SUGAR 500 ML	14	15	13	13	9	12
	SOSRO TEHBOTOL JASMINE LESS SUGAR	14	16	18	10	6	6
	ZESTEA JASMINE GREEN TEA 500 ML	13	16	18	19	16	11
	FRESTEA GREEN MY BODY GINSENG & GINGER	12	7	5	3	12	3
NU GREEN TEA KURMA 500 ML	6	11	5	3	3	3	
C	C2 GREEN TEA APPLE 360 ML	8	8	10	8	3	2
	ARTEA HONEY GREEN TEA 500 ML	12	8	13	8	5	10
	FRESTEA GREEN MY BODY ALOEVERA	10	8	9	10	11	5
	FRUIT TEA STR 500 ML	8	12	3	5	5	3
	JOY TEA GREEN COZY JASMINE 500 ML	6	5	3	3	3	5
	FUTAMI OOLONG TEA 350 ML	9	8	5	4	3	7
	FRUIT TEA FUSION 500 ML	3	5	2	2	3	4
	FRUIT TEA BLKCURRENT 500 ML	3	2	2	2	2	2
	JOY TEA GREEN CHEERFUL MADU LEMON	7	2	3	3	3	3
	C2 GREEN TEA CLASSIC 360 ML	8	2	3	2	2	4
	ZESTEA JASMINE GREEN TEA LESS SUGAR 500	7	3	5	4	6	2
	ZESTEA PURE GREEN TEA 500 ML	5	9	5	4	7	4
	FRUIT TEA APP 500 ML	4	5	2	3	3	2
ARTEA APP GREEN TEA 500 ML	3	2	5	2	2	2	

Nilai fungsi objektif menunjukkan bahwa model SSAP ini bersifat sensitif terhadap perubahan batasan kendala *facing* maksimal. Hal ini terlihat dari

semakin meningkatnya nilai kategori keuntungan dengan bertambahnya batasan alokasi maksimal *facing*. Dari grafik pada Gambar 4.6 dapat terlihat ketika nilai maksimal *facing* ditingkatkan, proporsi *shifting* alokasi ruang lebih besar terjadi pada produk kelas C untuk dialokasikan pada produk kelas A untuk memperoleh nilai keuntungan kategori produk yang lebih baik.

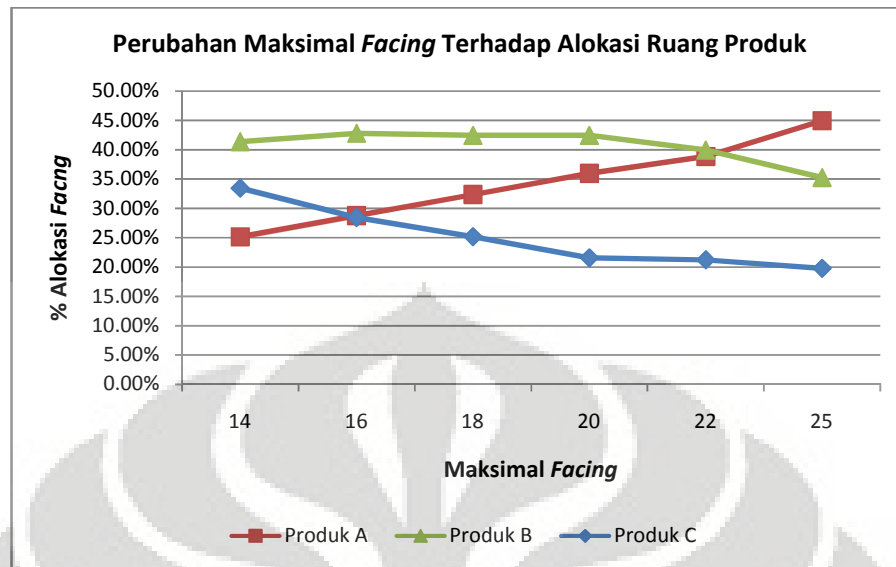
Jika membandingkan dengan alokasi *facing* saat ini, nilai kategori keuntungan yang dihasilkan program masih lebih besar 2,00% ketika nilai maksimal *facing* diperkecil hingga  $Ra = 14$ . Hal ini memperkuat bahwa program AG untuk model SSAP dapat menghasilkan solusi yang lebih optimal daripada solusi yang ada saat ini, meskipun batas maksimal alokasi *facing* yang digunakan jauh lebih kecil.

**Tabel 4.35** Analisis Sensitivitas Kendala Maksimal *Facing* terhadap Nilai Keuntungan Kategori Produk

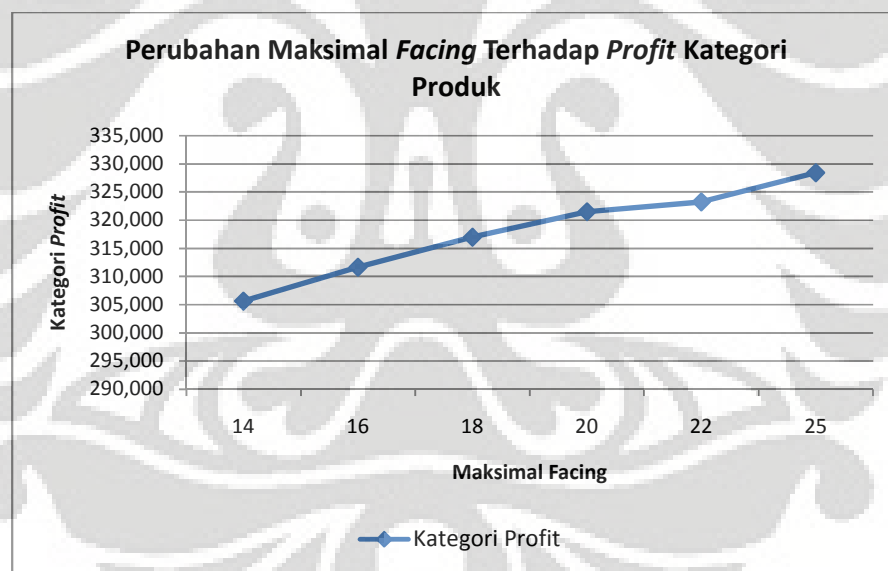
	Kendala Maksimal Facing					
	14	16	18	20	22	25
<b>Alokasi Facing</b>						
<b>Produk A</b>	25,18%	28,78%	32,37%	35,97%	38,85%	44,96%
<b>Produk B</b>	41,37%	42,81%	42,45%	42,45%	39,93%	35,25%
<b>Produk C</b>	33,45%	28,42%	25,18%	21,58%	21,22%	19,78%
<b>Kategori Keuntungan</b>	305.640	311.660	316.990	321.510	323.270	328.440
<b>Total Facing</b>	278	278	278	278	278	278
<b>% Perubahan Keuntungan</b>	-4,936%	-3,064%	-1,406%		0,547%	2,144%

**Tabel 4.36** Perbandingan Sensitivitas Maksimal *Facing* dengan Alokasi *Facing* Saat ini

	Kendala Maksimal Facing						
	14	16	18	20 (Current)	20 (Program)	22	25
<b>Alokasi Facing</b>							
<b>Produk A</b>	25,18%	28,78%	32,37%	26,98%	35,97%	38,85%	44,96%
<b>Produk B</b>	41,37%	42,81%	42,45%	36,69%	42,45%	39,93%	35,25%
<b>Produk C</b>	33,45%	28,42%	25,18%	36,33%	21,58%	21,22%	19,78%
<b>Kategori Keuntungan</b>	305.640	311.660	316.990	299.643	321.510	323.270	328.440
<b>Total Facing</b>	278	278	278	278	278	278	278
<b>% Perubahan Keuntungan</b>	2,00%	4,01%	5,79%		7,30%	7,89%	9,61%



**Gambar 4.6** Grafik Perubahan Maksimal *Facing* Terhadap Alokasi Ruang Produk



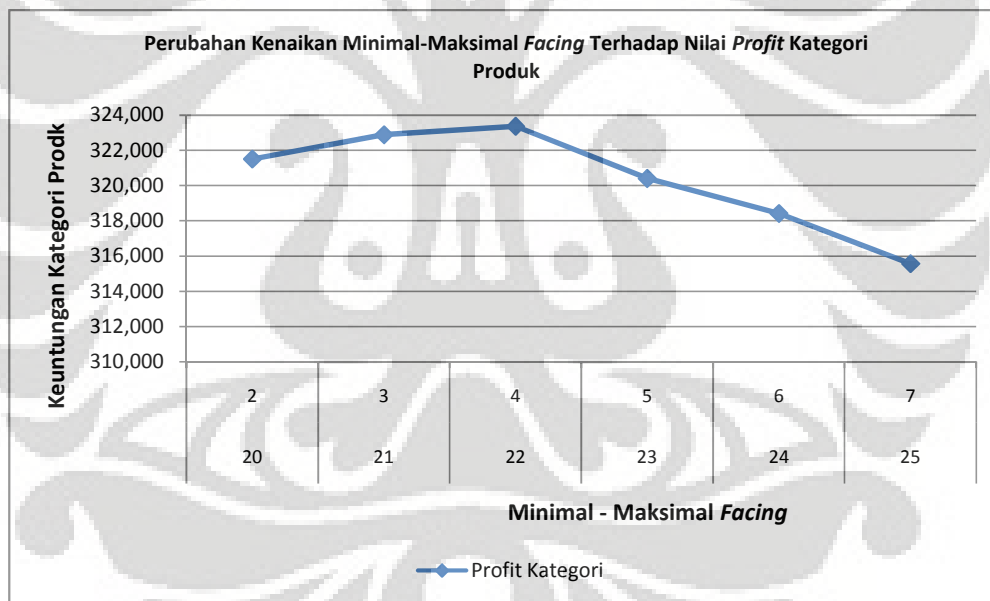
**Gambar 4.7** Grafik Perubahan Maksimal *Facing* Terhadap Keuntungan Kategori Produk

Setelah meninjau perubahan nilai objektif terhadap masing-masing kendala maksimal dan minimal *facing*, selanjutnya dilakukan juga analisis sensitivitas dari kombinasi kedua kendala tersebut. Dengan menaikkan batas kendala untuk kedua parameter maksimal dan minimal *facing*, hasil analisis

sensitivitas yang diperoleh adalah seperti yang terlihat pada Tabel 4.36 serta Gambar 4.8 dan 4.9.

**Tabel 4.37** Analisis Sensitivitas Kenaikan Parameter Maksimal dan Minimal *Facing* Terhadap Proporsi Alokasi Ruang Produk dan Keuntungan Kategori

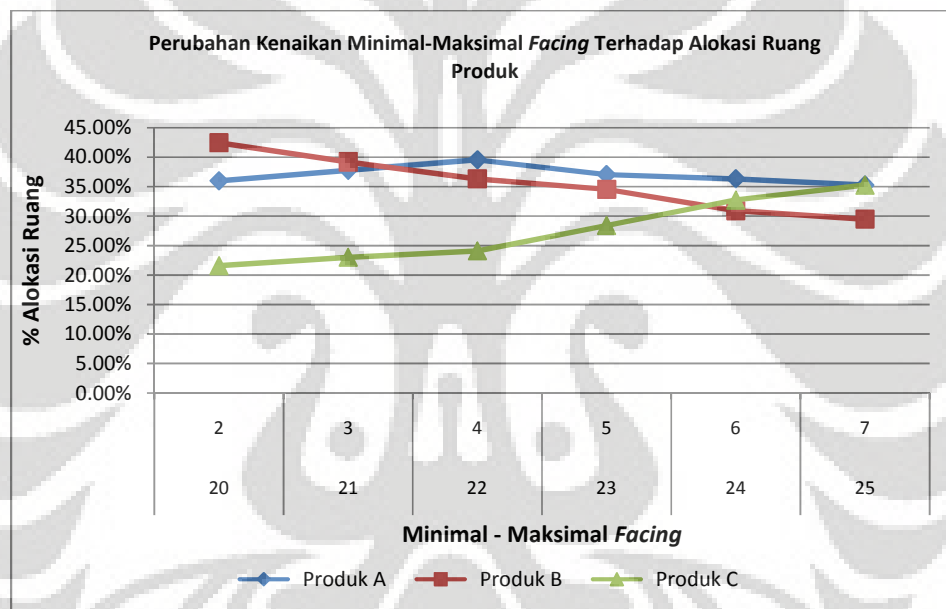
Maks. <i>Facing</i>	20	21	22	23	24	25
Min. <i>Facing</i>	2	3	4	5	6	7
<b>Alokasi <i>Facing</i></b>						
Produk A	35,97%	37,77%	39,57%	37,05%	36,33%	35,25%
Produk B	42,45%	39,21%	36,33%	34,53%	30,94%	29,50%
Produk C	21,58%	23,02%	24,10%	28,42%	32,73%	35,25%
Keuntungan Kategori	321.510	322.890	323.370	320.410	318.420	315.570
Total <i>Facing</i>	278	278	278	278	278	278
% Perubahan Keuntungan		0,43%	0,58%	-0,34%	-0,96%	-1,87%



**Gambar 4.8** Grafik Perubahan Kenaikan Minimal-Maksimal *Facing* Terhadap Nilai Keuntungan Kategori Produk

Dari data analisis sensitivitas yang ditampilkan, dapat dilihat bahwa nilai keuntungan kategori produk akan meningkat hingga batas tertentu dan kemudian akan menurun. Kapasitas rak yang dialokasikan untuk kategori produk tersebut mempengaruhi batasan model untuk meningkatkan hasil fungsi objektif

keuntungan produk. Ketika kenaikan nilai minimal dan maksimal *facing* dilakukan, model akan menghasilkan hasil keuntungan kategori yang lebih tinggi asalkan kapasitas rak yang ada masih dapat menampung kenaikan alokasi *facing*, terutama untuk produk kelas A. Namun, ketika kapasitas rak telah terpakai untuk memenuhi batas minimal *facing*, maka nilai keuntungan kategori produk akan cenderung menurun. Pergeseran alokasi ruang terjadi antara produk kelas B dan produk kelas C. Dari analisis sensitivitas ini dapat disimpulkan bahwa kendala minimal *facing* lebih menjadi prioritas daripada kendala maksimal dalam penyelesaian permasalahan alokasi ruang produk.



**Gambar 4.9** Grafik Perubahan Minimal-Maksimal *Facing* Terhadap Alokasi Ruang Produk

Dari data analisis sensitivitas yang ditampilkan, dapat dilihat bahwa nilai keuntungan kategori produk akan meningkat hingga batas tertentu dan kemudian akan menurun. Kapasitas rak yang dialokasikan untuk kategori produk tersebut mempengaruhi batasan model untuk meningkatkan hasil fungsi objektif keuntungan produk. Ketika kenaikan nilai minimal dan maksimal *facing* dilakukan, model akan menghasilkan hasil keuntungan kategori yang lebih tinggi asalkan kapasitas rak yang ada masih dapat menampung kenaikan alokasi *facing*,

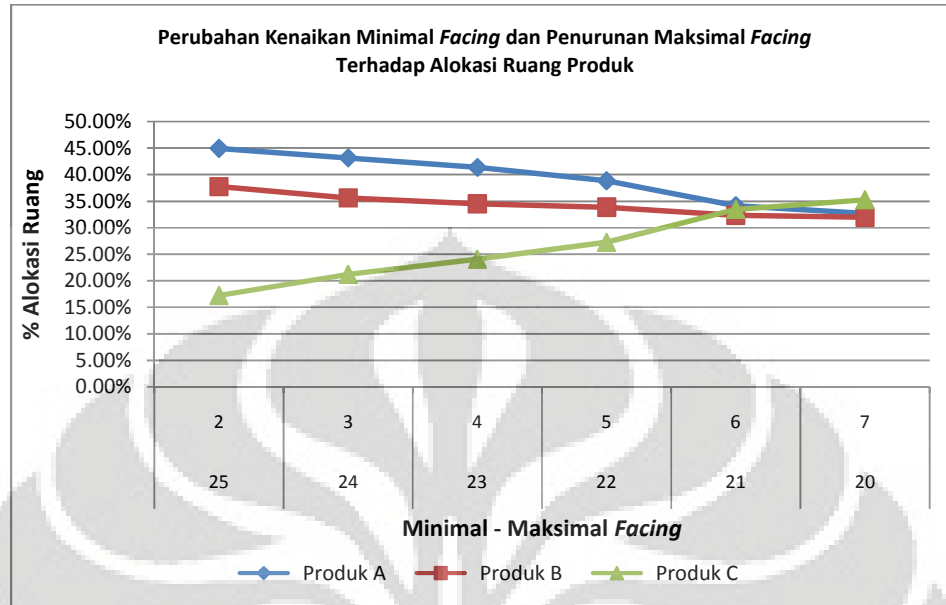
terutama untuk produk kelas A. Namun, ketika kapasitas rak telah terpakai untuk memenuhi batas minimal *facing*, maka nilai keuntungan kategori produk akan cenderung menurun. Pergeseran alokasi ruang terjadi antara produk kelas B dan produk kelas C. Dari analisis sensitivitas ini dapat disimpulkan bahwa kendala minimal *facing* lebih menjadi prioritas daripada kendala maksimal dalam penyelesaian permasalahan alokasi ruang produk.

Analisis sensitivitas selanjutnya juga dilakukan pada kombinasi kenaikan pada salah satu parameter batas *facing* (minimal *facing*) dan penurunan pada parameter batas lainnya (maksimal *facing*). Hasil analisis sensitivitas dapat dilihat pada Tabel 3.7 serta Gambar 4.10 dan 4.11.

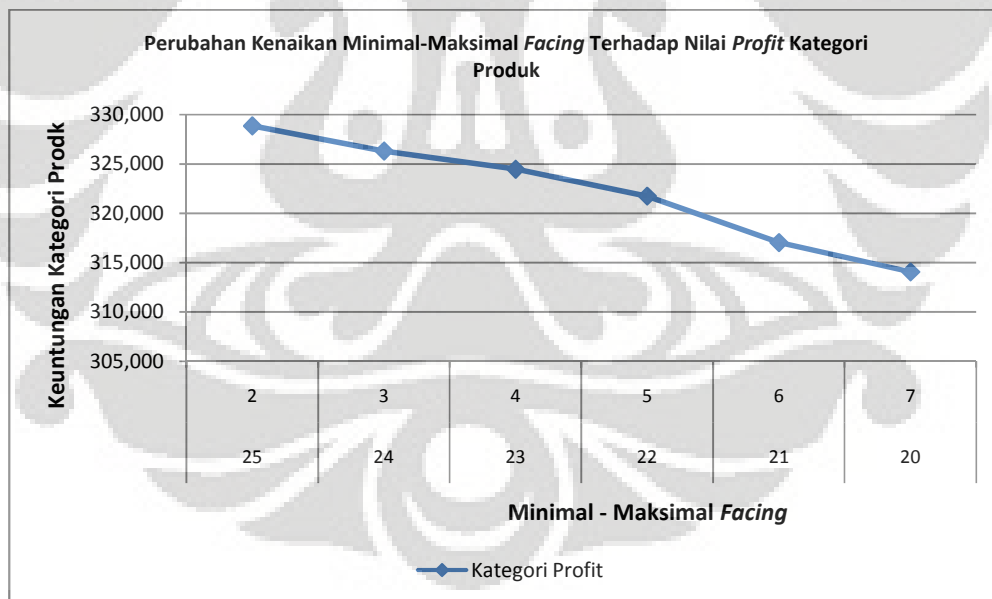
**Tabel 4.38** Analisis Sensitivitas Terhadap Kenaikan Minimal *Facing* dan Penurunan Maksimal *Facing* Terhadap Alokasi Ruang Produk dan Keuntungan Kategori Produk

Maks. <i>Facing</i>	25	24	23	22	21	20
Min. <i>Facing</i>	2	3	4	5	6	7
<b>Alokasi <i>Facing</i></b>						
<b>Produk A</b>	44,96%	43,17%	41,37%	38,85%	34,17%	32,73%
<b>Produk B</b>	37,77%	35,61%	34,53%	33,89%	32,37%	32,01%
<b>Produk C</b>	17,27%	21,22%	24,10%	27,26%	33,45%	35,25%
<b>Kategori <i>Profit</i></b>	328.860	326.310	324.460	321.740	317.030	314.050
<b>Total <i>Facing</i></b>	278	278	278	278	278	278
<b>% Perubahan <i>Profit</i></b>		-0,78%	-1,34%	-2,17%	-3,60%	-4,50%

Hasil analisis sensitivitas menunjukkan bahwa nilai keuntungan kategori produk akan mengalami penurunan ketika nilai minimal *facing* dinaikkan dan maksimal *facing* diturunkan. Hal ini dikarenakan karena ruang produk lebih banyak dialokasikan untuk memenuhi batas minimal *facing* untuk tiap produk. Selain itu, dengan diturunkannya batas maksimal *facing*, maka model juga otomatis akan menghasilkan nilai keuntungan yang lebih rendah, terutama untuk produk-produk pada kelas A yang pada umumnya akan dialokasikan secara maksimal. Pergeseran alokasi *facing* terjadi pada produk A dan B kepada produk kelas C guna memenuhi batas minimal *facing* yang ditetapkan.



**Gambar 4.10** Grafik Perubahan Kenaikan Minimal *Facing* dan Penurunan Maksimal *Facing* Terhadap Alokasi Ruang Produk



**Gambar 4.11** Grafik Perubahan Kenaikan Minimal *Facing* dan Penurunan Maksimal *Facing* Terhadap Nilai Keuntungan Kategori Produk



#### 4.3.4.2 Analisis Sensitivitas terhadap Parameter Koefisien Space Elasticity

Parameter *space elasticity* merupakan parameter utama yang menentukan peluang *resulted demand* yang dapat diperoleh suatu produk. Semakin besar nilai *space elasticity* yang dimiliki, maka akan semakin besar pula potensi keuntungan yang dapat diraih produk tersebut. Hal ini tercermin dari eksperimen sensitivitas yang dilakukan terhadap beberapa nilai parameter *space elasticity* yang berbeda. Nilai kategori keuntungan meningkat secara signifikan ketika nilai *space elasticity* yang digunakan meningkat. Di sisi lain, komposisi proporsi alokasi *facing* tiap kelas produk cenderung sama rata walaupun parameter nilai *space elasticity* diubah, hal ini dikarenakan penggunaan nilai *space elasticity* yang sama untuk semua produk dan perubahan nilai dilakukan juga untuk semua produk.

**Tabel 4.39** Analisis Sensitivitas Parameter *Space Elasticity* terhadap Alokasi Ruang Produk dan Nilai Keuntungan Kategori Produk

	Parameter <i>Space Elasticity</i>				
	0,1	0,15	0,212	0,25	0,3
<b>Alokasi Facing</b>					
<b>Produk A</b>	35,97%	35,97%	35,97%	35,97%	35,97%
<b>Produk B</b>	42,81%	43,88%	42,45%	41,01%	40,29%
<b>Produk C</b>	21,22%	20,14%	21,58%	23,02%	23,74%
<b>Kategori Keuntungan</b>	234.370	269.660	321.510	357.750	411.840
<b>Total Facing</b>	278	278	278	278	278
<b>% Perubahan Keuntungan</b>	-27,103%	-14,493%		11,27%	15,12%

#### 4.3.5 Analisis Kebijakan *in-Store Shelf Replenishment*

Usulan kebijakan untuk proses pengisian rak *display* diperoleh dengan menggunakan metode ROP untuk *uncertain demand*. Nilai standar deviasi dari distribusi persebaran *demand* sangat menentukan nilai *safety stock* (SS) dan tentunya titik ROP suatu produk dalam kalkulasi. Nilai *safety stock* yang dibutuhkan untuk tiap produk tergolong besar, jika dibandingkan dengan nilai rata-rata *demand*. Hal ini dikarenakan tingkat *demand* yang sangat fluktuatif menyebabkan harus dialokasikannya jumlah *safety stock* yang cukup besar untuk satu periode. Sebagian besar dari produk memiliki nilai standar deviasi yang lebih

besar daripada nilai *demand* rata-ratanya, yang menyebabkan persediaan *safety stock* harus dialokasikan dalam jumlah yang cukup besar.

Nilai *safety stock* ini juga ditentukan dengan besarnya nilai  $z$ , yang menunjukkan angka tingkat ketersediaan produk yang harus dimiliki selama periode *lead time*. Dengan nilai  $z = 1,64$ , kebijakan manajerial menetapkan tingkat persediaan barang berada pada level 95% dalam satu periode. Hal ini sangatlah masuk akal untuk industri ritel yang rata-rata menjual produk *fast moving consumer goods*. Ketika konsumen tidak dapat melihat produk yang diinginkannya pada rak *display*, maka konsumen tidak akan dapat membeli produk tersebut, dan pada akhirnya produk tidak dapat memberikan kontribusi penjualan terhadap toko, walaupun mungkin sebenarnya persediaan produk tersebut masih tersedia di gudang. Konsumen dapat beralih untuk membeli produk lain yang sejenis, namun juga terdapat peluang bagi konsumen untuk membatalkan pembeliannya. *Lost sales* akibat manajemen operasional yang kurang efektif merupakan aspek yang sangat dihindari oleh manajemen ritel karena selain dapat mengurangi tingkat keuntungan yang dapat diperoleh, juga dapat berdampak buruk terhadap tingkat kepuasan pelanggan.

Dari ROP yang telah diperoleh untuk tiap produk, selanjutnya dapat ditentukan pula nilai  $Q^*$ , yaitu jumlah barang yang perlu diisikan dalam setiap periode pengisian. Nilai  $Q^*$  dibulatkan ke dalam satuan kardus (*bulk*) karena bentuk satuan inilah yang digunakan dalam gudang, sehingga jumlah barang yang masuk ataupun keluar dari gudang harus dalam satuan kardus. Perkiraan waktu nilai barang mencapai ROP dapat ditentukan dengan melihat jumlah barang yang diisikan dengan tingkat rata-rata *demand*. Dari perkiraan waktu pengisian yang diperoleh, terdapat beberapa produk yang memiliki perkiraan waktu yang terlampaui panjang (lebih dari 30 hari). Hal ini disebabkan karena produk tersebut memiliki tingkat *demand* yang rendah namun alokasi ruangnya cukup besar, sehingga jika ruang untuk produk tersebut dipenuhi, maka akan memakan waktu lama untuk stok produk berkurang dan mencapai ROP. Selain itu, dalam menganalisis kebijakan pengisian rak digunakan alokasi ruang optimal hasil penghitungan program, namun data historis *demand* produk yang digunakan masih berpaku pada alokasi *facing* yang digunakan oleh manajemen selama ini.

Dalam penyelesaian permasalahan alokasi ruang dengan program algoritma genetika, terdapat beberapa produk yang memiliki alokasi ruang yang cukup besar dibandingkan dengan tingkat *demand* per harinya. Dalam model SSAP dimana dianggap bahwa alokasi *facing* dapat mempengaruhi tingkat *demand* sebuah produk, maka terdapat peluang untuk produk-produk tersebut mengalami kenaikan tingkat *demand* setelah alokasi *facing*-nya ditambah.

Oleh karena itu, untuk ke depannya ROP serta nilai jumlah produk yang diisikan ke dalam rak sebaiknya ditinjau kembali setelah beberapa periode waktu. Hal ini dikarenakan nilai *resulted demand* dapat berubah akibat adanya perubahan alokasi *facing* yang diterapkan dari sebelumnya. Namun, jika pada periode ke depan ternyata tingkat *demand* tidak mengalami perubahan yang signifikan setelah penambahan alokasi ruang, hal ini dapat menjadi pertimbangan bagi manajemen perilaku yang sebaiknya diterapkan bagi produk tersebut, apakah harus diupayakan kegiatan promosi yang lebih gencar untuk mendorong penjualan ataupun tidak mempertahankan produk tersebut dalam *assortment* produk yang dijual di gerai ritel.

Selain itu, perkiraan waktu pengisian juga dapat berubah lebih cepat dari perkiraan mengingat jumlah produk yang dapat diisikan ke dalam rak tergantung tingkat persediaan yang ada di gudang. Jika jumlah persediaan tidak mencukupi untuk memenuhi alokasi ruang untuk produk tersebut pada proses pengisian, maka akan sangat besar kemungkinannya titik ROP akan jatuh lebih awal. Oleh karena itu, tingkat persediaan di gudang juga harus dapat dijaga agar selalu berada dalam *level* yang memadai.

## BAB 5

### KESIMPULAN

Berdasarkan penelitian penyelesaian permasalahan alokasi ruang produk pada rak *display* gerai ritel *grocery*, dapat diperoleh kesimpulan sebagai berikut.

- Model SSAP dengan metode penyelesaian algoritma genetika telah berhasil menemukan solusi terbaik alokasi *facing* untuk ke-28 produk minuman teh kemasan, dengan nilai keuntungan kategori produk sebesar Rp 321.510. Nilai keuntungan ini lebih baik dari nilai kategori keuntungan yang diperoleh dari alokasi *facing* yang digunakan saat ini, yaitu Rp299.643 atau naik sekitar 7,30%.
- ROP untuk tiap produk juga telah dapat ditentukan sebagai acuan dalam melakukan proses pengisian rak. Titik pengisian berkisar antara jumlah stok 4 hingga 157 dengan jumlah pengisian berkisar antara 1 hingga 9 kardus dalam sekali pengisiannya.
- Hasil analisis sensitivitas menunjukkan bahwa alokasi ruang produk kelas A cenderung tidak peka terhadap kenaikan parameter kendala batas minimal *facing*. Di sisi lain, peningkatan minimal *facing* menyebabkan peningkatan alokasi *facing* untuk produk kelas C yang diikuti penurunan alokasi *facing* produk kelas B. Namun, kenaikan keuntungan produk C yang diakibatkan bertambahnya alokasi *facing* masih lebih kecil daripada penurunan keuntungan yang terjadi pada produk kelas B. Sehingga, secara keseluruhan nilai keuntungan kategori produk akan cenderung menurun.
- Kenaikan parameter batasan maksimal *facing* menyebabkan kenaikan pada alokasi *facing* untuk produk A yang diikuti penurunan alokasi *facing* untuk produk C. Alokasi *facing* untuk produk B tidak mengalami perubahan yang signifikan. Hal ini tentunya menyebabkan kenaikan pada nilai keuntungan kategori
- Kendala minimal *facing* merupakan kendala yang menjadi prioritas utama untuk dipenuhi. Hal ini terlihat dari hasil analisis sensitivitas terhadap kombinasi kenaikan nilai kedua parameter minimal dan maksimal *facing*. Walaupun nilai maksimal *facing* dinaikkan, namun alokasi *facing* untuk

produk A lama kelamaan akan menurun karena adanya peningkatan alokasi *facing* produk C guna memenuhi kendala minimal batas *facing*.

- Nilai *space elasticity* juga terbukti bersifat sensitif terhadap nilai kategori keuntungan yang dapat diperoleh, terlihat dari kecenderungan peningkatan nilai fungsi objektif ketika parameter *space elasticity* tersebut dinaikkan.
- Metode yang digunakan terbukti dapat menghasilkan solusi yang jauh lebih baik dari solusi alokasi *facing* yang ada saat ini. Hal ini dapat dilihat dari tetap meningkatnya nilai keuntungan kategori walaupun nilai batas minimal dinaikkan hingga 7 *facing*. Kategori keuntungan yang dihasilkan senilai Rp315.050 masih lebih baik 5,14% daripada kategori keuntungan yang ada saat ini. Penurunan batas atas *facing* juga tetap menghasilkan solusi yang lebih baik dari solusi yang ada saat ini. Kategori keuntungan senilai Rp305.640 untuk batas maksimal 14 *facing* masih lebih baik 2,00% daripada solusi yang ada saat ini.

Model SSAP dengan metode penyelesaian algoritma genetika terbukti dapat menyelesaikan permasalahan alokasi ruang produk dengan cukup optimal dan cepat. Untuk kedepannya, penelitian ini memiliki peluang yang besar untuk dapat dikembangkan lebih lanjut. Studi dalam menentukan parameter-parameter koefisien yang digunakan dalam model SSAP perlu untuk dilakukan agar dapat menghasilkan solusi yang lebih tepat dan optimal.

## DAFTAR REFERENSI

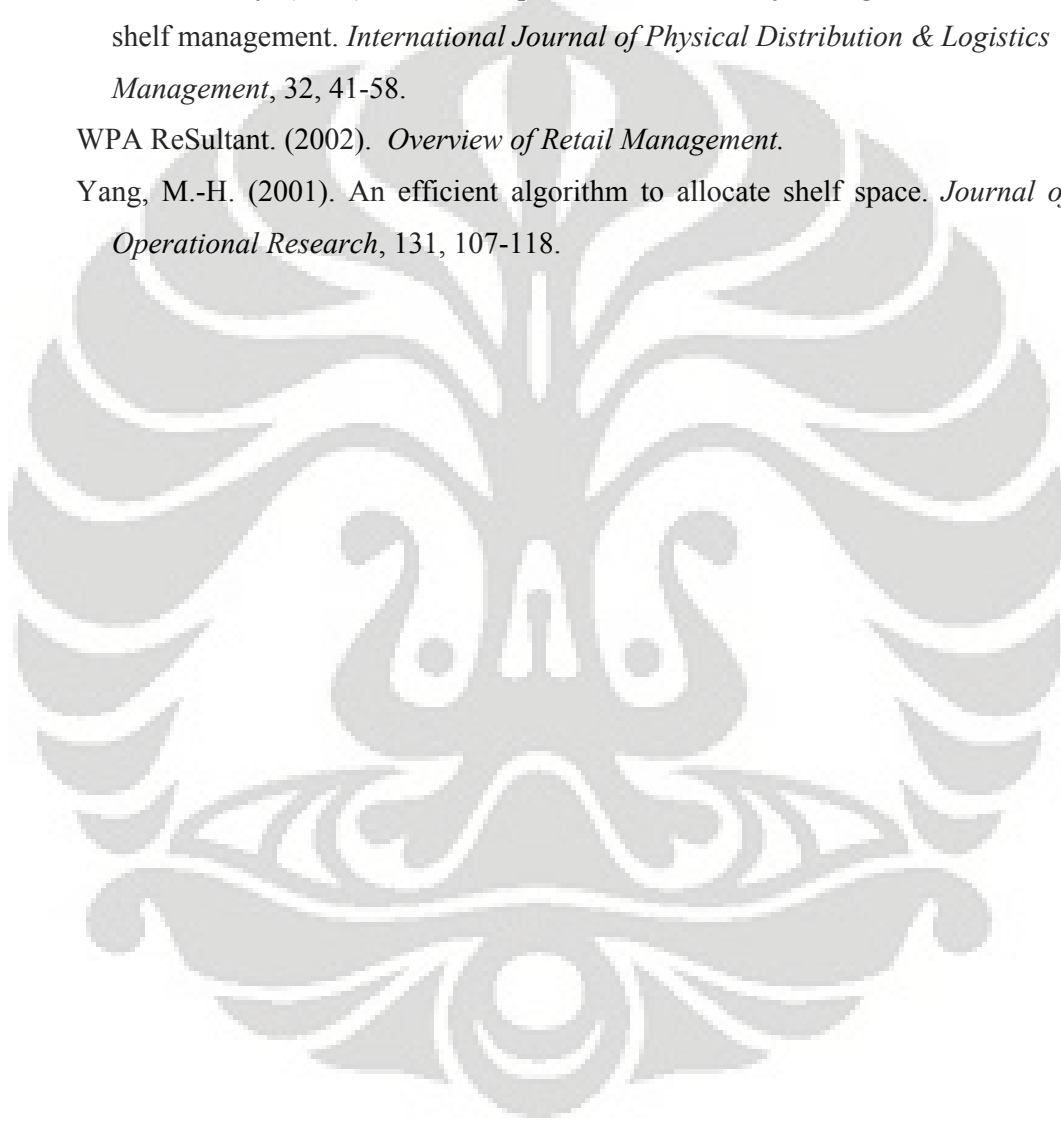
- Alexander, J. T. (1992). On optimal population size of genetic algorithm. *Proceedings of ComEuro*, 92, 65-70.
- Bai, Ruibin. (September, 2005). *An Investigation of Novel Approaches for Optimizing Retail Shelf Space Allocation*. Thesis submitted to the University of Nottingham for the Degree of Doctor of Philosophy, Nottingham.
- Ballou, Ronald. (2003). *Business Logistics Management (5th Edition)*. New Jersey: Prentice Hall Inc.
- Bellenger, R. N., Robertson, D. H., & Hirschman, E. C. (1978). Impulse buying varies by product. *Journal of Advertising Research*, 18(6), 15-18.
- Berman, B. & Evans, J. R. (2010). *Retailing Management: A Strategic Approach*. Prentice Hall.
- Boyabatli, O. & Sabuncuoglu, I. Parameter selection in genetic algorithms. *Systemics, Cybernetics and Informatics*, Vol. 2(4).
- Brown, W. & Tucker, W. (1961). The marketing center: vanishing shelf space. *Atlanta Economic Review*, 4, 9-13.
- Business Monitor International (BMI). (2010). *Indonesia Retail Report Q2 2010*.
- Buttle, F. (1984). Merchandising. *European Journal of Marketing*, 18, 104-123.
- Chen, W. C., & Yang, M. H. (1999). A study on shelf space allocation and management. *International Journal of Production Economics*, 60, 309-317.
- Curhan, Ronald. (1972). The relationship between shelf space and unit sales in supermarkets. *Journal of Marketing Research*, 9(4), 406-412.
- Desmet, P. & Renaudin, V. (1998). Estimation of product category sales responsiveness to allocated shelf space. *International Journal of Research in Marketing*, 15, 443-457.
- Dreze, X., Hoch, Stephen J., & Purk, M. (1994). *Shelf Management and Space Elasticity*. Graduate School of Business The University of Chicago.
- Du Pont de Nemours and Compagny. (1995). *Consumer buying habits studies*. Wilmington, DE: du Pont de Nemours and Compagny.



- Gutierrez-Garcia, J. & Kuri-Morales, A. F. Penalty function methods for constrained optimization with genetic algorithms: a statistical analysis. *Instituto Tecnológico Autónomo de México*.
- Hansen, P. & Heinsbroek, H. (1979) Product selection and space allocation in supermarkets. *European Journal of Operation Research*, 3, 474-484.
- Homaiffar, A., Lai, S. & Qi, C. (1994). Constrained optimization via genetic algorithm. *Simulation*, 242-254.
- Hubner, A. & Kuhn, H. (2010). Retail shelf space management model with integrated inventory elastic demand and consumer-driven substitution effects. *Working Paper Catholic University Eichstatt-Ingolstadt Germany*.
- Hwang, H., Choi, B. & Lee, M. (2005). A model for shelf space allocation and inventory control considering location and inventory level effects on demand. *International Journal of Production Economics*, 97, 185-195.
- IGD.com. (2002). March 20, 2010. *What's Category Management*.  
[http://www.igd.com/freeinfo/IGD-CatMgt\\_What's.htm](http://www.igd.com/freeinfo/IGD-CatMgt_What's.htm)
- Larson, P.D. & DeMarais, R.A. (1990). Psychic stock: an independent variable category of inventory. *International Journal of Physical Distribution and Logistics Management*, 20 (7), 28-34.
- Leblanc-Maridor, F. (1989). L'achat impulsif: doctrine et realite. *Revue Francaise du Marketing*, 123(3), 51-70.
- Levy, M. & Weitz, B. (1992). *Retailing Management*. Homewood, IL: McGraw-Hill.
- Manser, Martin H. (1987). *Wordmaster Dictionary*. Harmondsworth: Penguin.
- Peluang Usaha Retail*. (n.d.). February 28, 2010.  
<http://bannerpointritel.com/peluang-usaha-ritel.html>
- Peta Persaingan Bisnis Retail Modern di Indonesia*. (2009). PT Media Data Riset.
- Point-of-Purchase Advertising Institute. (1963). *Drugstore brand switching and impulse buying*. Point-of-Purchase Advertising Institute, New York.
- Sujana, Asep. *Paradigma Baru dalam Modern Retail Management*. Jakarta: WPA
- ReSULTANT
- Susanta, Rahmat. (2009, July 17). Kolom Redaksional. *Marketing*.



- Suyanto. (2006). *Algoritma Genetika dalam MATLAB*. Yogyakarta: Penerbit ANDI.
- Townsend, A. A. R. *Genetic Algorithm: a Tutorial*.
- Urban, Timothy L. (1998). An Inventory-Theoretic Approach to Product Assortment and Shelf-Space Allocation. *Journal of Retailing*, 74(1), 15-35.
- Urban, Timothy. (2002). The interdependence of inventory management and retail shelf management. *International Journal of Physical Distribution & Logistics Management*, 32, 41-58.
- WPA ReSultant. (2002). *Overview of Retail Management*.
- Yang, M.-H. (2001). An efficient algorithm to allocate shelf space. *Journal of Operational Research*, 131, 107-118.



**1) Inisialisasi Populasi**

```
function Populasi = InisialisasiPopulasi (UkPop, JumGen)
for k=1:UkPop
    for g=1:JumGen
        a = rand;
        if (a < 0.5)
            Populasi(k,g) = 0;
        else
            Populasi(k,g) = 1;
        end
    end
end
```

**2) Dekodekan Kromosom**

```
function x = DekodekanKromosom(Populasi, Nvar, Nbit, Ra, Rb, UkPop)
JumPenyebut = 0;
for i = 1:Nbit
    JumPenyebut = JumPenyebut + 2^(-i);
end
for baris = 1:UkPop
    for i=1:Nvar
        x(baris,i) = 0;
        for j=1:Nbit
            x(baris,i) = x(baris,i) + Populasi(baris, (i-1)*Nbit+j)*2^(-j);
        end
        x(baris,i) = floor(Rb + ((Ra-Rb)/JumPenyebut)*x(baris,i));
    end
end
```

**3) Evaluasi Individu Awal**

```
%Mengevaluasi individu sehingga didapatkan nilai fitness-nya
%Masukan
% x : individu
%Keluaran
% fitness : nilai fitness

function Fitness =
EvaluasiIndividu(x, Profit, Demand, SpaceElasticity, UkPop, Nvar)
for j = 1:UkPop;
    Fitness(j) = 0;
    for i = 1:Nvar
        Fitness(j) = Fitness(j) +
        (Profit(i)*Demand(i)*(x(j,i)^SpaceElasticity));
    end
end
Fitness = Fitness';
```

#### 4) Evaluasi Penalti

- **Evaluasi Maksimum *Fitness***

```
function MaxValue =  
MaxParameter (Ra, Profit, Demand, SpaceElasticity, UkPop, Nvar  
);  
MaxValue = 0;  
for i = 1:Nvar  
    MaxValue = MaxValue +  
(Profit(i)*Demand(i)*(Ra^SpaceElasticity));  
end
```

- **Evaluasi Minimum *Fitness***

```
function MinValue =  
MinParameter (Rb, Profit, Demand, SpaceElasticity, UkPop, Nvar  
);  
MinValue = 0;  
for i = 1:Nvar  
    MinValue = MinValue +  
(Profit(i)*Demand(i)*(Rb^SpaceElasticity));  
end
```

- **Skala Penalti**

```
function R = SkalaPenalti (MaxValue, MinValue)  
R = MaxValue - MinValue;
```

- **Kalkulasi Penalti**

```
function  
Penalti=EvaluasiPenalti(x, Fitness, L, Cap, R, UkPop, Nvar);  
for j = 1:UkPop;  
    Penalti(j) = 0;  
    TotalKap(j) = 0;  
    for i = 1:Nvar  
        TotalKap(j) = TotalKap(j) + L(i)*x(j,i);  
    end  
    if TotalKap(j) > Cap  
        Penalti(j) = -R*(TotalKap(j)/Cap);  
    else  
        Penalti(j) = 0;  
    end  
end
```

#### 5) Evaluasi *Fitness* Akhir

```
function FitnessAkhir =  
EvaluasiFitnessAkhir (Fitness, Penalti, UkPop)  
for j=1:UkPop  
    FitnessAkhir(j) = Fitness(j) + Penalti(j);  
end
```

#### 6) Pemilihan Kromosom Terbaik dan Elitisme

```
function NewPopulasi=  
KromosomTerbaik (UkPop, FitnessAkhir, Populasi)  
[UkPop, JumGen]=size (Populasi);  
for j = 1:UkPop  
    Maximum = max(FitnessAkhir);
```

```

        if FitnessAkhir(j) == Maximum
            Letak = j;
            break;
        end
    end
end
if mod(UkPop,2)==0        %Ukuran populasi genap
    for i=1:JumGen
        NewPopulasi(1,i) = Populasi(Letak,i);
        NewPopulasi(2,i) = Populasi(Letak,i);
    end
else        %Ukuran populasi ganjil
    for i=1:JumGen
        NewPopulasi(1,i) = Populasi(Letak,i);
    end
end
end

```

### 7) Linear Fitness Ranking

```

function [LFR,FitnessTerurut,KromosomTerbaik] =
ScaleFitness(FitnessAkhir,Populasi,UkPop)
FitnessTerurut = -sort(-FitnessAkhir);
for i = 1:UkPop
    Terurut = FitnessTerurut(i);
    for j = 1:UkPop
        if Terurut == FitnessAkhir(j);
            Letak = j;
            KromosomTerbaik(i,:) = Populasi(j,:);
        end
    end
end
FMax = max(FitnessAkhir);
FMin = min(FitnessAkhir);
for i = 1:UkPop
    LFR(i) = FMax - ((FMax-FMin)*((i-1)/(UkPop-1)));
end

```

### 8) Seleksi Kromosom Orang Tua

```

function [ParentKromosom1,ParentKromosom2] =
SeleksiParent(FitnessTerurut,KromosomTerbaik,UkPop)
Total = sum(FitnessTerurut);
for i = 1:UkPop
    Proporsi(i) = FitnessTerurut(i)/Total;
    if i > 1
        CumProporsi(i) = Proporsi(i) + CumProporsi(i-1);
    else
        CumProporsi(i) = Proporsi(i);
    end
end
Pilih1 = rand();
for i = 1:UkPop
    if Pilih1 <= CumProporsi(i)
        ParentKromosom1 = KromosomTerbaik(i,:);
        break
    end
end
Pilih2 = rand();
for i = 1:UkPop

```

```

        if Pilih2 <= CumProporsi(i)
            ParentKromosom2 = KromosomTerbaik(i,:);
            break
        end
    end
end

```

## 9) Pindah Silang

```

function Child =
PindahSilang(ParentKromosom1,ParentKromosom2,JumGen,Psilang)
if (rand < Psilang)
    TP = 1 + fix(rand*(JumGen-1));
    Child(1,:) = [ParentKromosom1(1:TP)
ParentKromosom2(TP+1:JumGen)];
    Child(2,:) = [ParentKromosom2(1:TP)
ParentKromosom1(TP+1:JumGen)];
else
    Child(1,:) = ParentKromosom1(1,:);
    Child(2,:) = ParentKromosom2(1,:);
end
end

```

## 10) Mutasi

```

function ChildMut = Mutasi(Child,JumGen,Pmutasi)
for baris =1:2
    for i = 1:JumGen
        prob=rand();
        if (prob < Pmutasi && Child(baris,i)==0)
            ChildMut(baris,i)=Child(baris,i)+1;
        elseif (prob < Pmutasi && Child(1,i)==1)
            ChildMut(baris,i)=Child(baris,i)-1;
        else
            ChildMut(baris,i)=Child(baris,i);
        end
    end
end
end

```

## 11) Program Utama Algoritma Genetika

```

clc;clear all;
%=====Data Input=====
Ra = input('\nMasukkan batas atas: ');
Rb = input('\nMasukkan batas bawah: ');
UkPop = input('\nMasukkan ukuran populasi: ');
MaxG = input('\nMasukkan generasi: ');
tic;

%=====Data Produk=====
Nvar = 28;
Nbit = 5;
Cap = 1672;
Profit = [853.03 853.03 853.03 903.03 903.03 859.85 859.85
859.85 677.27 677.27 1098.58 1098.58 1098.58 865.91 865.91
865.91 865.91 602.3 702.3 702.3 702.3 436.36 436.36 636.82
676.52 676.52 676.52 676.52];

```

```

Demand = [33 30 23 2 2 10 25 17 1 3 4 1 1 17 3 11 2 7 5 2 1 4
2 3 1 1 2 1];
SpaceElasticity = 0.212;
L = [6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6];

%=====Parameter GA=====
JumGen = Nvar*Nbit;
Psilang=0.25;
Pmutasi=0.01;

fprintf('\n');

%=====Inisialisasi Populasi=====
Populasi = InisialisasiPopulasi (UkPop, JumGen);

x = DekodekanKromosom(Populasi, Nvar, Nbit, Ra, Rb, UkPop); %berisi
x1 s.d x28 yang berupa decision variable

Fitness =
EvaluasiIndividu(x, Profit, Demand, SpaceElasticity, UkPop, Nvar);

MaxValue =
MaxParameter(Ra, Profit, Demand, SpaceElasticity, UkPop, Nvar);

MinValue =
MinParameter(Rb, Profit, Demand, SpaceElasticity, UkPop, Nvar);

R = SkalaPenalti(MaxValue, MinValue);

Penalti = EvaluasiPenalti(x, Fitness, L, Cap, R, UkPop, Nvar);

FitnessAkhir = EvaluasiFitnessAkhir(Fitness, Penalti, UkPop);

for generasi = 1:MaxG %Generasi mulai

%=====Pemilihan Kromosom Terbaik=====
for j = 1:UkPop
Maximum = max(FitnessAkhir);
if FitnessAkhir(j) == Maximum
Letak = j;
break;
end
end

%=====Elitisme=====
if mod(UkPop,2)==0 %Ukuran populasi genap
for i=1:JumGen
NewPopulasi(1,i) = Populasi(Letak,i);
NewPopulasi(2,i) = Populasi(Letak,i);
end
else %Ukuran populasi ganjil
for i=1:JumGen
NewPopulasi(1,i) = Populasi(Letak,i);
end
end
end

```

```

%=====Linear Fitness Ranking=====
[LFR,FitnessTerurut,KromosomTerbaik] =
ScaleFitness(FitnessAkhir,Populasi,UkPop);
IterasiMulai = 3;
for j = IterasiMulai:2:UkPop,

%=====Operasi Genetika=====
[ParentKromosom1,ParentKromosom2] =
SeleksiParent(FitnessTerurut,KromosomTerbaik,UkPop);
Child =
PindahSilang(ParentKromosom1,ParentKromosom2,JumGen,Psilang);
ChildMut = Mutasi(Child,JumGen,Pmutasi);
NewPopulasi(j,:) = ChildMut(1,:);
NewPopulasi(j+1,:) = ChildMut(2,:);
end

Populasi = NewPopulasi; %Generational Replacement

%=====Evaluasi Individu=====
x = DekodekanKromosom(Populasi,Nvar,Nbit,Ra,Rb,UkPop);%
berisi x1 s.d x28 yang berupa decision variable

Fitness =
EvaluasiIndividu(x,Profit,Demand,SpaceElasticity,UkPop,Nvar);

MaxValue =
MaxParameter(Ra,Profit,Demand,SpaceElasticity,UkPop,Nvar);

MinValue =
MinParameter(Rb,Profit,Demand,SpaceElasticity,UkPop,Nvar);

R = SkalaPenalti(MaxValue,MinValue);

Penalti = EvaluasiPenalti(x,Fitness,L,Cap,R,UkPop,Nvar);

FitnessAkhir =
EvaluasiFitnessAkhir(Fitness,Penalti,UkPop);

absis(generasi)=generasi;
BestFitness(generasi)=max(FitnessAkhir);

if generasi==MaxG
    break;
end
end

% Membuat grafik pergerakan dari nilai Fitness terbaik
plot(absis,BestFitness)
for i=1:UkPop
    if FitnessAkhir(i)==max(FitnessAkhir)
        letak=i;
        break;
    end
end
x(letak,:)
toc;

```