

**PENENTUAN LOKASI *DISTRIBUTION CENTRE*
PADA SUATU PERUSAHAAN AGRIBISNIS**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh
gelar sarjana teknik**

**YOPI FERNANDES
04 05 07 05 85**



**UNIVERSITAS INDONESIA
FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INDUSTRI
DEPOK
JULI 2009**

PERNYATAAN ORISINALITAS

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

Nama : Yopi Fernandes

NPM : 0405070585

Tanda Tangan :

Tanggal : Juli 2009

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Yopi Fernandes
NPM : 0405070585
Program Studi : Teknik Industri
Judul Skripsi : Penentuan Lokasi *Distribution Centre* Pada Suatu
Perusahaan Agribisnis

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana S1 pada Program Studi Teknik Industri, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Armand Omar Moeis,ST.,MSc ()
Penguji : Ir. Amar Rachman,MEIM ()
Penguji : Ir. Sri Bintang P.,MSISE.,PhD ()
Penguji : Ir. Rahmat Nurcahyo,M.Eng.Sc ()

Ditetapkan di : Depok

Tanggal : Juli 2009

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan ke hadirat Allah SWT yang selalu memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Penyusunan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Departemen Teknik Industri pada Fakultas Teknik Universitas Indonesia. Penulis menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, sangatlah sulit bagi penulis untuk menyelesaikan skripsi ini. Untuk itu penulis mengucapkan terima kasih kepada :

1. Mama, Papa, dan Uni Tanti yang selalu menyayangi dan mendoakan, memberikan perhatian, motivasi, masukan dan inspirasi serta mendengarkan segala keluh kesah penulis. Mudah-mudahan ini bisa membuat mama, papa dan Uni bangga.
2. Bapak Armand Omar Moeis, S.T., M.Sc., selaku dosen pembimbing yang telah menyediakan waktu, tenaga dan pikiran untuk mengarahkan penulis dalam penyusunan skripsi ini.
3. Bapak Ir. Amar Rachman, MEIM, yang telah memberikan berbagai masukan penting dalam menyelesaikan skripsi ini.
4. Bang Komar yang telah dengan ikhlas memberikan masukan dan sumbangan pemikiran dalam menyelesaikan skripsi ini
5. Bapak Inyas dan Bapak Liberto yang telah memberikan kemudahan akses data yang diperlukan penulis.
6. Sekar Melati, yang selalu memberikan perhatian, dukungan dan semangat
7. Warman, Aan, ARC dan Cica, teman senasib dan seperjuangan yang selalu memberikan dukungan, hiburan dan masukan serta membuat penyusunan skripsi ini menjadi semakin mudah
8. Dadi, yang telah memberikan sumbangsih kamar dan obrolan-obrolan tak berguna
9. Dimi, Yuda, Guntur, Kily, dan Ucok, teman-teman persaudaraan dotA
10. Nangke dan Nyoman yang telah dengan ikhlas menjadi tempat pelampiasan stress penulis

11. Elice dan Carissa yang banyak memberikan masukan
12. Artur, Pipop, Keshia, dan Najwa, yang telah membuat algoritma DE menjadi mudah dimengerti
13. Teman-teman 2005 lainnya, untuk segala kekompakan, waktu, obrolan, canda tawa dan bantuan yang telah diberikan selama masa perkuliahan.
14. Teman-teman 2006, 2007 dan 2008
15. Pihak-pihak lain yang tidak bisa penulis sebutkan satu persatu.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan saudara-saudara semua. Dan semoga skripsi ini membawa manfaat bagi pengembangan ilmu pengetahuan

. Depok, Juli 2009

Penulis

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

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

Nama : Yopi Fernandes
NPM : 0405070585
Program Studi : Teknik Industri
Departemen : Teknik Industri
Fakultas : Teknik
Jenis karya : Skripsi

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

“Penentuan Lokasi *Distribution Centre* Pada Suatu Perusahaan Agribisnis” beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia / format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir 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 : Juni 2009

Yang menyatakan

(Yopi Fernandes)

ABSTRAK

Nama : Yopi Fernandes
Program Studi : Teknik Industri
Judul Skripsi : Penentuan Lokasi *Distribution Centre* Pada Suatu Perusahaan Agribisnis

Penelitian ini membahas pemilihan lokasi terbaik dari tiga lokasi yang ada untuk pendirian *distribution centre* pada sebuah perusahaan agribisnis. Tiga lokasi yang diteliti adalah Cibubur, Bintaro dan Pulo Gadung. Lokasi terbaik yang dipilih adalah lokasi dengan total biaya bulanan terkecil. Total Biaya dihitung berdasarkan estimasi biaya pengantaran barang selama 1 bulan dan biaya bulanan pembelian lahan. Estimasi biaya pengantaran barang dilakukan dengan terlebih dahulu melakukan estimasi jarak tempuh dengan menggunakan pendekatan *vehicle routing problem* algoritma *differential evolution*. Hasil penelitian menunjukkan bahwa tempat terbaik mendirikan DC adalah pada kandidat lokasi Cibubur, dengan total biaya per bulan sebesar Rp. 83.577.042,67.

Kata kunci:

Distribution centre, vehicle routing problem, differential evolution, biaya bulanan

ABSTRACT

Name : Yopi Fernandes
Study Program : Industrial Engineering
Title : Distribution Centre Location Problem in an Agribusiness Company

This research focus on determining the best of three places to construct a new distribution centre (DC) in an agribusiness company. The three locations examined are Cibubur, Bintaro and Pulo Gadung. The best location is location with lowest monthly cost. The monthly cost itself is calculated based on the estimation of distribution cost for 1 month and the monthly cost of purchasing land. The estimation of distribution cost is done by estimating the total distribution distance from each place using vehicle routing problem (VRP), differential evolution (DE) algorithm approach. The result of this research proves that the best place to construct DC is Cibubur, with Rp. 83,577,042.67 monthly cost.

Keywords:

Distribution centre, vehicle routing problem, differential evolution, monthly cost

DAFTAR ISI

HALAMAN JUDUL.....	Error! Bookmark not defined.
HALAMAN_PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
HALAMAN UCAPAN TERIMA KASIH	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vi
ABSTRAK.....	vii
DAFTAR ISI	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR LAMPIRAN.....	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Permasalahan.....	1
1.2 Diagram Keterkaitan masalah.....	3
1.3 Perumusan Permasalahan.....	3
1.4 Tujuan Penelitian.....	5
1.5 Ruang Lingkup Permasalahan.....	5
1.6 Metodologi Penelitian.....	5
1.7 Sistematika Penulisan	7
BAB 2 TINJAUAN PUSTAKA	10
2.1 Facility Location Problem.....	10
2.1.1 Definisi dan Karakteristik.....	10
2.1.2 Metode Penyelesaian.....	12
2.2 Vehicle Routing Problem.....	13
2.2.1 Vehicle Routing Problem with <i>Time Windows</i>	17
2.2.1.1 Model Matematika VRPTW	17
2.2.2 Metode Penyelesaian.....	19
2.2.2.1 Pendekatan Eksak	19
2.2.2.2 Heuristik Klasik.....	20
2.2.2.3 Metaheuristik.....	22
2.3 Algoritma <i>Differential Evolution</i>	24
2.3.1 Sejarah.....	24

2.3.2 Definisi	25
2.3.3 Tahapan Penyelesaian	26
2.4 Annual Cash Flow Analysis	33
BAB 3 PENGUMPULAN DATA	34
3.1 Data Untuk Menghitung Biaya Operasional	34
3.1.1 Depot	34
3.1.2 Kandidat Lokasi Pendirian DC	35
3.1.3 Konsumen	35
3.1.4 Armada Pengiriman	37
3.1.5 Biaya Pengiriman Barang	38
3.1.6 Permintaan	40
3.1.7 Waktu dan Kecepatan	49
3.1.8 Jarak	49
3.2 Data Untuk Menghitung Biaya Pembelian Tanah Bulanan	51
3.2.1 Biaya Pembelian Tanah	51
3.2.2 Suku Bunga Kredit Investasi	51
BAB 4 PENGOLAHAN DATA DAN ANALISA	52
4.1 Pengolahan Data	52
4.1.1 Perhitungan Jarak Antara Tiap Kandidat ke Konsumen	52
4.1.1.1 Langkah Pembuatan Program MATLAB	53
4.1.1.2 Verifikasi dan Validasi Program	59
4.1.1.3 Input Data	61
4.1.2 Perhitungan Data Jarak antara Depot ke Tiap Kandidat	64
4.1.3 Perhitungan Biaya	64
4.2 Hasil Pengolahan Data	65
4.2.1 Jarak Masing-masing Kandidat untuk Melayani Pelanggan	65
4.2.2 Jarak Depot-Kandidat	68
4.2.3 Konversi Biaya	68
4.2.3.1 Biaya Operasional	68
4.2.3.2 Biaya Pembelian Tanah	70
4.2.3.3 Biaya Total	Error! Bookmark not defined.
4.3 Analisis	70
4.3.1 Analisis Metode Perhitungan Jarak	70

4.3.2 Analisis Program.....	71
4.3.2.1 Analisis Penggunaan Algoritma DE.....	71
4.3.2.2 Analisis Jalannya Program.....	72
4.3.3 Analisis Hasil.....	76
4.3.3.1 Analisis Jarak Kandidat-Pelanggan	76
4.3.3.2 Analisis Jarak Depot-Kandidat.....	77
4.3.4 Analisis Biaya	78
BAB 5 KESIMPULAN DAN SARAN.....	82
5.1 Kesimpulan	82
5.2 Saran.....	82
DAFTAR REFERENSI.....	83



DAFTAR TABEL

Tabel 3.1. Daftar Nama Pelanggan Perusahaan	35
Tabel 3.2. Kapasitas Krat untuk Setiap Jenis Sayuran.....	41
Tabel 3.3. Permintaan Selama Tanggal 1-8 Januari 2009.....	42
Tabel 3.4. Permintaan Selama Tanggal 9-16 Januari 2009.....	43
Tabel 3.5. Permintaan Selama Tanggal 17-24 Januari 2009.....	45
Tabel 3.6. Permintaan Selama Tanggal 25-31 Januari 2009.....	46
Tabel 3.7. Jarak Depot ke Kandidat dan Kandidat ke Pelanggan.....	50
Tabel 4.1. Hasil Percobaan Penentuan Parameter	54
Tabel 4.2. Matriks Jarak Data Dummy	59
Tabel 4.3. Permintaan Tiap Pelanggan Data Dummy	59
Tabel 4.4. Parameter yang Digunakan Dalam Proses Validasi.....	60
Tabel 4.5. Contoh Format Spreadsheet Data Jarak Untuk Input Data.....	63
Tabel 4.6. <i>Output</i> Jarak Untuk Masing-masing Kandidat.....	67
Tabel 4.7. Jarak Dari Depot ke Tiap Kandidat.....	68
Tabel 4.8. Biaya Dari Kandidat ke Pelanggan	69
Tabel 4.9. Biaya Dari Depot ke Pelanggan	69
Tabel 4.10. Total Biaya Operasional	69
Tabel 4.11. Konversi Biaya Pembelian Tanah	70
Tabel 4.12. <i>Running Time</i> Program	73
Tabel 4.13. <i>Running Time</i> Untuk Tiap Kombinasi F dan CR.....	74
Tabel 4.14. Total Jarak Harian Untuk <i>Existing System</i> Perusahaan.....	80

DAFTAR GAMBAR

Gambar 1.1. Diagram Keterkaitan Masalah	4
Gambar 1.2. Diagram Alir Metode Penelitian	8
Gambar 2.1. Strategy Dalam Supply Chain Management.....	11
Gambar 2.2. Ilustrasi VRP.....	14
Gambar 2.3. <i>Saving Method</i>	22
Gambar 2.4. Proses Terjadinya Pindah Silang.....	30
Gambar 2.5. Flow Chart Pengerjaan	31
Gambar 2.6. Tahapan Kerja DE.....	32
Gambar 3.1. Letak Depot, Kandidat dan Pelanggan	37
Gambar 4.1. <i>Flowchart</i> Pengerjaan.....	58
Gambar 4.2. Contoh Output Program.....	66
Gambar 4.3. Letak Pelanggan, Depot dan Kandidat	78

DAFTAR LAMPIRAN

Lampiran 1 : Ouput Pengolahan Data
Lampiran 2 : Matriks Jarak Antar Pelanggan
Lampiran 3 : <i>Source Code</i>



BAB 1

PENDAHULUAN

1.1 Latar Belakang Permasalahan

Pada era globalisasi saat ini, persaingan antar perusahaan menjadi semakin ketat. Untuk bertahan pada situasi seperti ini, tiap perusahaan dituntut untuk dapat memuaskan pelanggannya. Namun, pemuasan kebutuhan pelanggan bukanlah sesuatu yang mudah. Banyaknya pilihan yang tersedia membuat pelanggan memiliki kebebasan untuk memilih *supplier* terbaik bagi kebutuhan mereka.

Kondisi seperti ini juga terjadi pada sektor agribisnis. Spiller, Kennerknecht dan Bolten (2006) menyatakan bahwa kepuasan pelanggan pada sektor agribisnis dipengaruhi oleh 4 faktor utama, yaitu pelayanan (keramahan serta ketepatan waktu dan jumlah pengantaran), kesegaran produk, rasa produk, dan harga. Karena itu, setiap perusahaan agribisnis harus memberikan perhatian besar pada empat faktor ini jika ingin menguasai pasar.

Sebuah perusahaan agribisnis di Ciawi yang memproduksi berbagai jenis sayuran juga sedang dihadapkan pada masalah kepuasan pelanggan. Berdasarkan data yang dimiliki perusahaan, sekitar 17% pelanggan sering mengalami pengantaran yang terlambat dan 28% sering mengeluhkan harga yang mereka berikan.

Masalah terlambatnya pengantaran disebabkan oleh jauhnya letak depot dengan pelanggan yang harus dilayani. Faktor jarak ini juga berpengaruh pada tingginya biaya transportasi yang harus mereka keluarkan. Masalah harga yang tinggi sangat dipengaruhi oleh biaya produksi yang dikeluarkan. Ballou (2004) menyatakan bahwa bagian dari biaya produksi yang memberikan kontribusi terbesar adalah biaya distribusi, yaitu sebesar 80%. Pada sistem distribusi sendiri, aktivitas transportasi merupakan aktivitas yang paling berpengaruh. Biaya transportasi merupakan biaya dengan kontribusi yang paling besar, yaitu sekitar 1/3 sampai 2/3 dari total biaya aktivitas distribusi. Jadi, dapat disimpulkan, untuk menyelesaikan masalah kepuasan pelanggan ini, perusahaan harus memperbaiki aktivitas transportasi yang mereka miliki.

Usaha untuk memperbaiki aktivitas transportasi dapat dilakukan dengan beberapa strategi antara lain dengan menggunakan jasa 3PL (*third-party logistics organizations*), membuat rencana rute distribusi yang akan dilalui untuk mengantarkan produk dari depot sampai ke konsumen atau mendirikan *distribution centre* (DC) pada daerah dimana terdapat beberapa titik permintaan. Penggunaan tiap strategi ini harus disesuaikan dengan kebutuhan dan kondisi yang sedang dihadapi oleh suatu perusahaan.

Untuk kasus pada Perusahaan, strategi 3PL tidak mungkin dilakukan karena perusahaan telah memiliki armada untuk mendistribusikan produknya. Jadi, strategi yang paling mungkin adalah strategi mendirikan DC dan perencanaan rute pengantaran. Strategi DC lebih *powerful* daripada strategi perencanaan rute karena dengan strategi ini akan memberikan kesempatan yang lebih luas dalam memperoleh pasar baru, wilayah distribusi yang baru, pelayanan yang lebih baik, dan pengurangan kompleksitas distribusi.

Menurut Chan, Kumar dan Choy (2005), DC adalah sebuah tempat transit barang jadi dari *supplier* atau *assembly lines* sebelum didistribusikan kepada *retailer* atau langsung kepada pelanggan. DC biasanya didirikan untuk melayani kebutuhan beberapa titik permintaan. Kegiatan yang dilakukan dalam DC antara lain pengepakan, pengendalian kualitas, dan penjadwalan pengiriman kepada pelanggan.

Jumlah DC yang akan didirikan pada kasus ini hanya satu buah. Hal ini dikarenakan jarak antar konsumen yang relatif cukup dekat satu sama lain. Penggunaan lebih dari satu DC dikhawatirkan tidak akan menyelesaikan akar permasalahan yang muncul.

Chu dan Lai (2004) menyatakan bahwa aspek lokasi harus menjadi fokus perhatian utama dalam pendirian DC. Pemilihan lokasi yang kurang tepat akan berakibat strategi ini berkurang manfaatnya. Sejauh ini, perusahaan telah memiliki tiga lokasi kandidat, yaitu Cibubur, Bintaro dan Pulo Gadung.

Lokasi yang paling baik adalah lokasi dengan biaya terkecil. Perhitungan biaya akan dilakukan untuk jangka waktu satu bulan. Faktor biaya yang diperhitungkan disini adalah biaya operasional dan biaya pembelian tanah. Biaya operasional adalah biaya pengantaran barang dari DC ke tiap pelanggan dan biaya

pengantaran barang dari depot ke DC. Biaya ini diasumsikan bersifat linear dengan jarak yang harus ditempuh.

Biaya pembelian lahan dapat dengan mudah diketahui, namun biaya operasional agak sulit diperkirakan mengingat DC belum dibuat. Karena itu, untuk menentukan biaya operasional pada tiap kandidat, kita perlu melakukan perhitungan mengenai kemungkinan jarak yang harus ditempuh dari tiap kandidat dalam melayani pelanggan. Jarak tempuh ini diperoleh dari hasil penjumlahan jarak dari depot-kandidat dan jarak dari kandidat ke pelanggan.

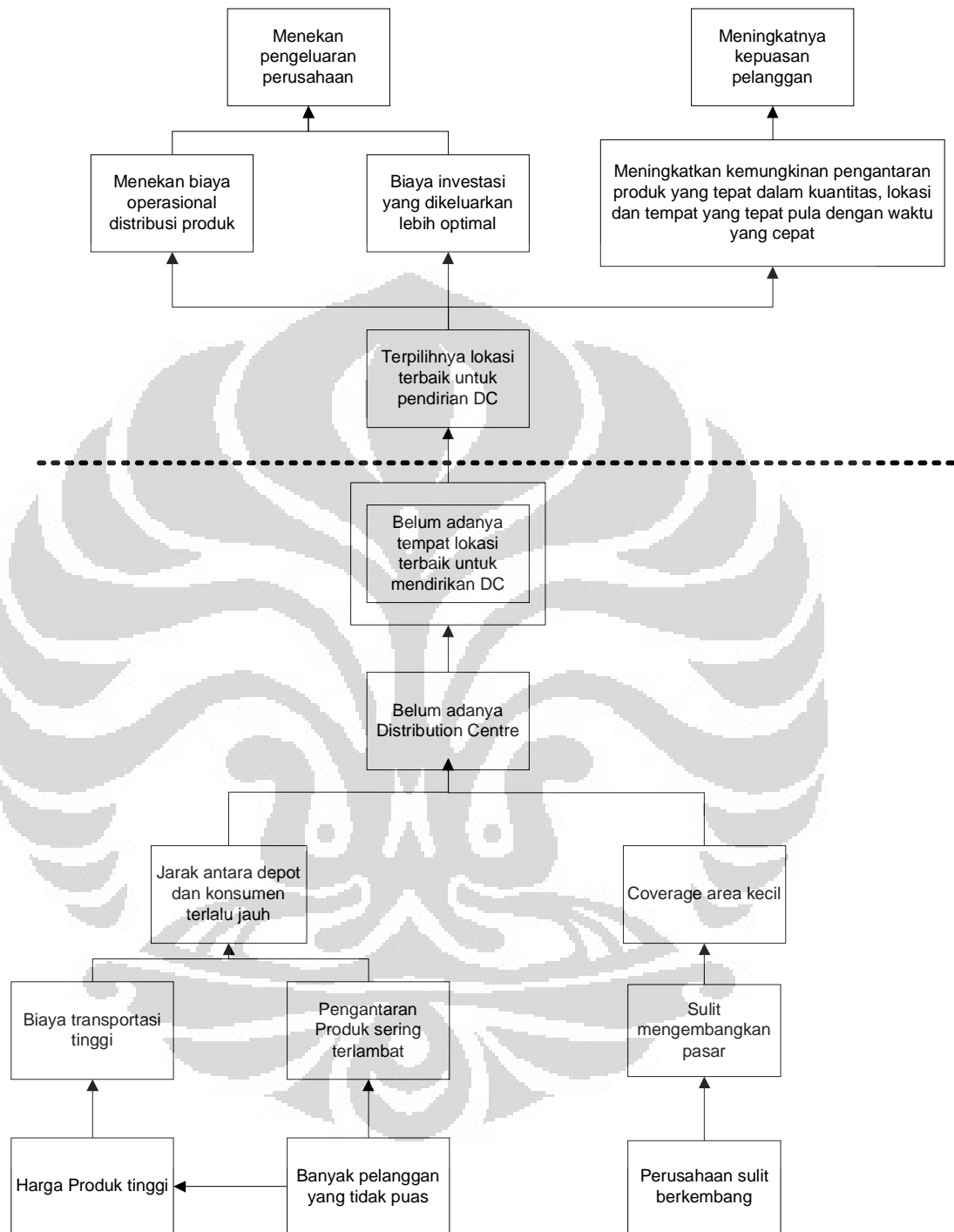
Jarak depot-kandidat dapat diketahui dengan bantuan peta, sedangkan jarak kandidat-pelanggan dapat diketahui dengan pendekatan *Vehicle Routing Problem* (VRP) algoritma *Differential Evolution* (DE). Secara sederhana, VRP merupakan permasalahan yang meliputi konstruksi rute-rute dari sejumlah kendaraan yang dimulai dari suatu depot utama menuju ke lokasi sejumlah konsumen dengan jumlah permintaan tertentu (Poot, Kant dan Wagelmans, 2002). Sementara itu, algoritma DE merupakan algoritma optimasi global yang paling baru dan menurut penelitian, mampu menghasilkan solusi yang paling baik dibandingkan algoritma lainnya (Okdem dan Karaboga, 2004).

1.2 Diagram Keterkaitan masalah

Berdasarkan latar belakang masalah di atas, maka dapat dibuat diagram keterkaitan masalah yang menampilkan permasalahan secara visual dan sistematis. Diagram keterkaitan masalah dari dilakukannya penelitian ini adalah seperti yang ditunjukkan oleh Gambar 1.1.

1.3 Perumusan Permasalahan

Pokok permasalahan yang menjadi fokus dalam penelitian ini adalah perlunya mencari lokasi yang paling optimum untuk pendirian satu buah DC dari 3 buah kandidat lokasi yang ada.



Gambar 1.1. Diagram Keterkaitan Masalah

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah memperoleh suatu usulan mengenai lokasi DC yang paling optimal dari tiga buah kandidat lokasi yang ada melalui analisa terhadap kemungkinan biaya di tiap kandidat.

1.5 Ruang Lingkup Permasalahan

Ruang lingkup dari penelitian ini digunakan agar masalah yang diteliti lebih dapat terarah dan terfokus sehingga penelitian dapat dilakukan sesuai dengan apa yang direncanakan. Ruang lingkup penelitian ini adalah sebagai berikut:

- DC akan dibuat hanya untuk melayani kebutuhan pelanggan di daerah Jakarta, Depok dan Tangerang karena pelanggan diluar daerah ini sangat sedikit jumlahnya
- perhitungan jarak dalam estimasi biaya operasional dilakukan untuk data permintaan selama bulan Januari 2009 karena data tersebut adalah data terbaru yang didokumentasikan perusahaan
- variabel kemacetan tidak diikutsertakan dalam perhitungan jarak karena minimnya waktu penelitian untuk melakukan eksplorasi terhadap titik-titik kemacetan di Jakarta dan sekitarnya
- biaya pembuatan DC tidak dimasukkan ke dalam perhitungan karena besar biaya ini akan sama untuk tiap kandidat lokasi sehingga tidak akan mempengaruhi hasil perhitungan
- pengolahan data dilakukan dengan pengembangan program komputer khusus dengan menggunakan perangkat lunak MATLAB karena keunggulannya dalam mengimplementasikan operasi-operasi dalam algoritma DE yang banyak menggunakan fungsi matrix.

1.6 Metodologi Penelitian

Secara umum, metodologi penelitian ini dibagi dalam 5 bagian, sebagaimana terlihat pada gambar 1.2.

1. Penentuan topik penelitian.

Topik penelitian didapatkan melalui diskusi dengan pihak perusahaan dan dosen pembimbing. Adapun topik penelitian ini adalah mencari lokasi yang paling optimum untuk pendirian satu buah DC dari 3 buah kandidat lokasi yang ada. Pada bagian ini, ditentukan pula hasil akhir yang ingin dicapai dan batasan masalah yang akan diteliti sehingga penelitian lebih terfokus dan berjalan sesuai dengan rencana.

2. Penentuan landasan teori.

Dalam tahap ini, ditentukan landasan teori yang berhubungan dengan topic sebagai dasar dalam pelaksanaan penelitian. Adapun landasan teori yang terkait adalah *facility location problem*, *vehicle routing problem*, *differential evolution algorithm* dan *annual cash flow analysis*.

3. Pengumpulan data.

Pada tahap ini, penulis akan menentukan kebutuhan data. Pertama-tama, penulis mengidentifikasi data yang diperlukan untuk menyelesaikan masalah VRP dengan algoritma DE. Data yang bersifat primer adalah data lokasi kandidat, data jarak antar pelanggan, data jarak antara tiap kandidat dengan tiap pelanggan dan data jarak antara perusahaan dengan tiap kandidat. Data yang bersifat sekunder adalah data permintaan, kapasitas armada, biaya pengantaran per km untuk setiap armada, *time windows*, dan biaya pembelian tanah .

4. Pengolahan data dan analisa.

Setelah semua data yang diperlukan diperoleh, langkah selanjutnya adalah mencari kemungkinan jarak yang harus ditempuh dari tiap kandidat dalam melayani seluruh pelanggan. Untuk mengerjakan hal ini, akan dibuat sebuah program dengan bantuan MATLAB. Hasil program yang berupa data jarak ini selanjutnya diubah menjadi data biaya dengan mempertimbangkan biaya pengantaran per km untuk setiap armada, sehingga pada akhirnya kita dapat mengetahui perkiraan biaya operasional tiap kandidat. Selanjutnya, biaya pembelian lahan akan dikonversikan menjadi biaya bulanan. Biaya bulanan pembelian lahan ditambah proyeksi biaya operasional akan menghasilkan total biaya operasional. Tempat dengan biaya terkecil adalah tempat yang akan dipilih untuk pendirian DC.

5. Penarikan Kesimpulan.

Setelah semua tahap dilakukan dengan baik, penulis akan membuat kesimpulan akhir berdasarkan analisa yang telah dibuat.

1.7 Sistematika Penulisan

Penulisan tugas akhir ini dibuat dalam lima bab yang memberikan gambaran sistematis sejak awal penelitian hingga tercapainya tujuan penelitian.

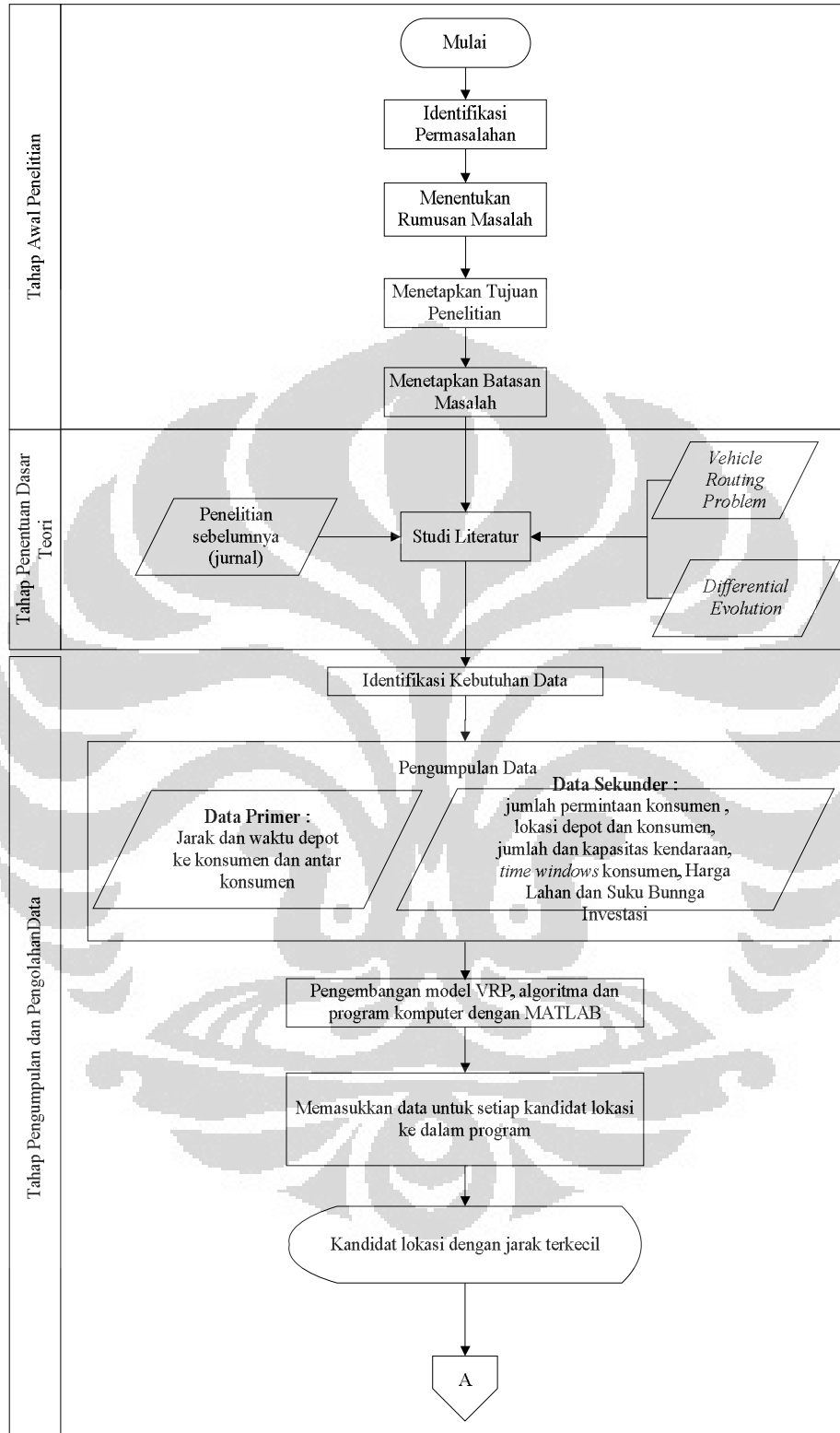
Bab pertama merupakan bab pendahuluan sebagai pengantar untuk menjelaskan isi penelitian secara garis besar. Dalam bab ini terdapat uraian mengenai latar belakang masalah, keterkaitan antar masalah, perumusan masalah, tujuan dan ruang lingkup penelitian, metodologi penelitian, serta sistematika penulisan.

Bab kedua berisi tentang landasan teori yang digunakan dalam penelitian. Secara umum, ada empat subjek yang akan dibahas pada bab ini, yaitu mengenai *Facility Location Problem (FLP)*, VRP, algoritma DE dan *annual cash flow analysis*. Pembahasan FLP dan VRP dimulai dari definisi umum dan modelnya, teknik pencarian solusi melalui algoritma eksak, heuristik, dan metaheuristik. Kemudian dilakukan dijelaskan pembahasan yang lebih mendalam mengenai metode DE sebagai salah satu pendekatan metaheuristik.

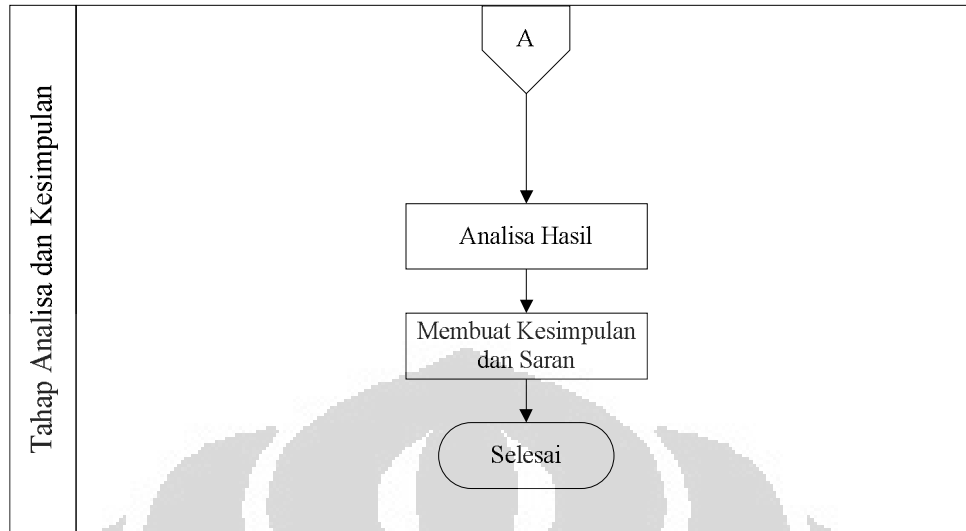
Bab ketiga menjelaskan mengenai pengumpulan data yang dibutuhkan untuk melaksanakan penelitian. Pengumpulan data dilakukan dengan cara observasi, wawancara, dan pengumpulan dokumen di Perusahaan.

Bab keempat berisi tentang pengolahan data dan analisisnya. Pada bab ini, semua data yang telah diperoleh akan diolah menjadi data biaya. Pengolahan data sendiri terbagi menjadi dua tahap, yaitu pengolahan data biaya operasional dan pengolahan data biaya pembelian lahan.

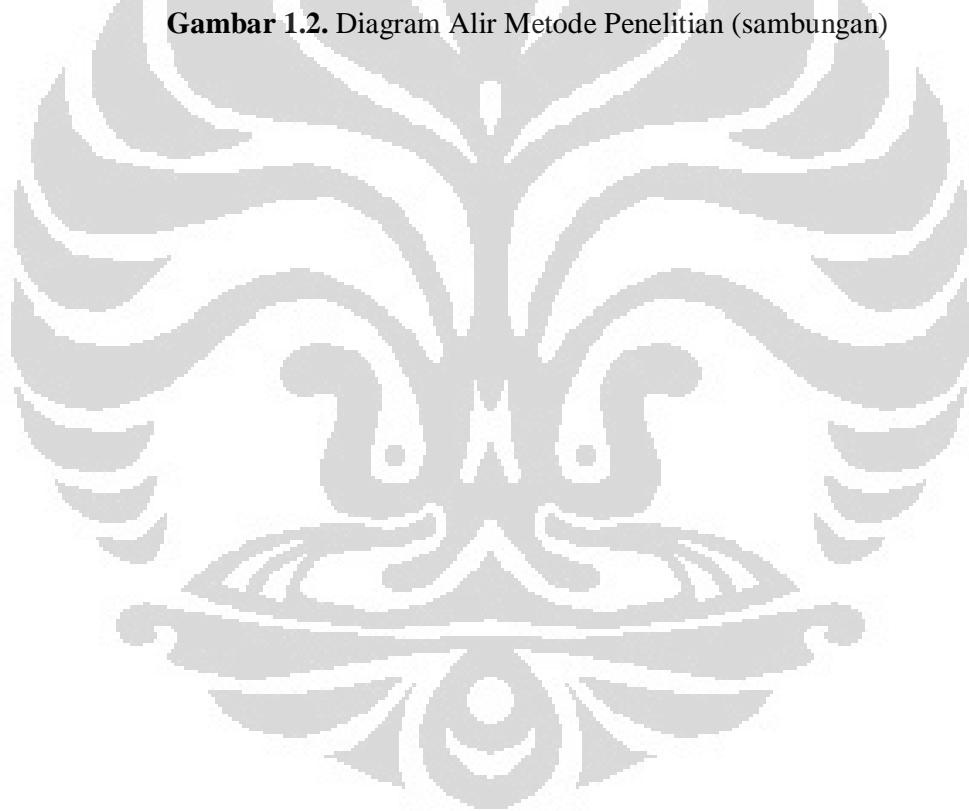
Bab kelima merupakan bab terakhir yang berisi kesimpulan dari pembahasan yang telah dilakukan dari bab-bab sebelumnya.



Gambar 1.2. Diagram Alir Metode Penelitian



Gambar 1.2. Diagram Alir Metode Penelitian (sambungan)



BAB 2

TINJAUAN PUSTAKA

Secara umum, ada empat subjek yang akan dibahas pada bab ini, yaitu mengenai *Facility Location Problem* (FLP), VRP, algoritma DE dan *annual cash flow analysis*. Pembahasan FLP akan dimulai dari definisi umum, jenis dan teknik pencarian solusinya. Pembahasan VRP dimulai dari definisi umum, model matematis, teknik pencarian solusi melalui algoritma eksak, heuristik, dan metaheuristik. Kemudian dilakukan pembahasan yang lebih mendalam mengenai metode DE sebagai salah satu pendekatan metaheuristik. Terakhir, akan dijelaskan metode *annual cash flow analysis* sebagai salah satu cara untuk mengevaluasi keputusan.

2.1 Facility Location Problem

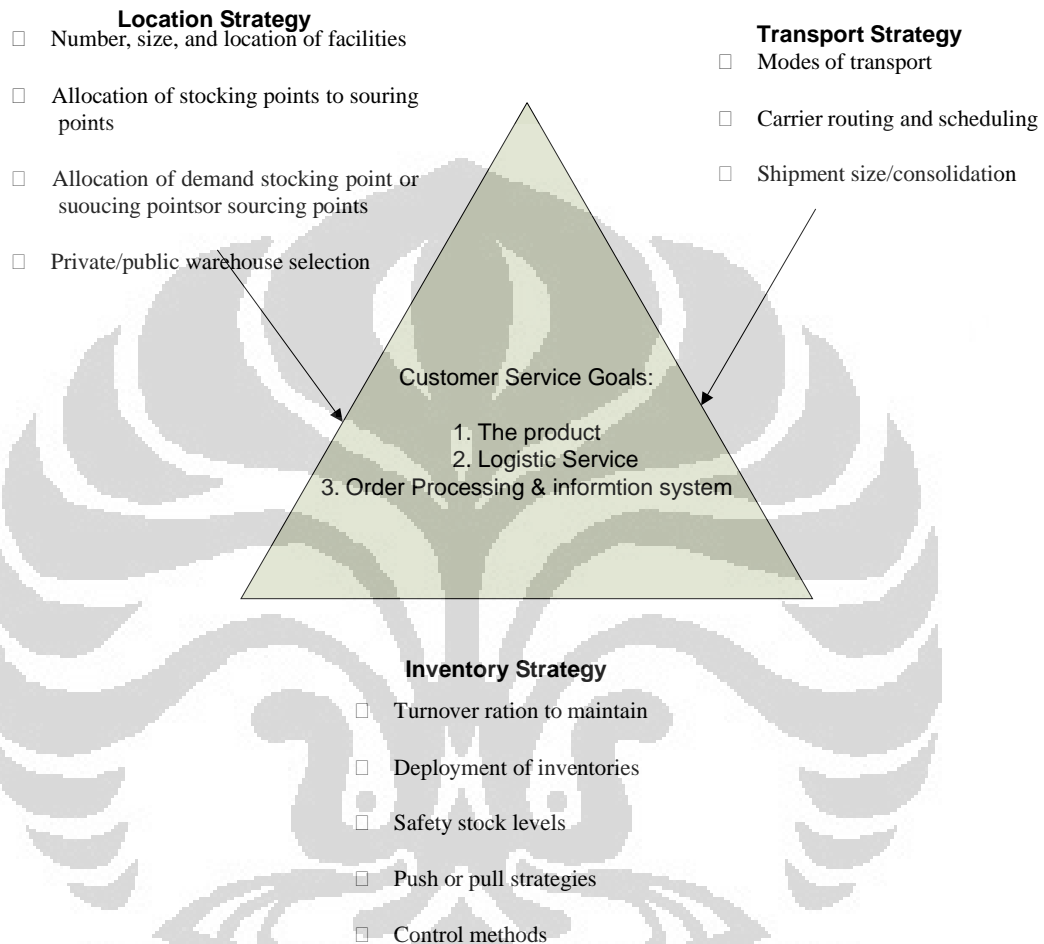
2.1.1 Definisi dan Karakteristik

Charnsethikul dan Singhtaun (2006) mendefinisikan Facility location Problem (FLP) sebagai masalah pencarian lokasi optimal untuk sebuah fasilitas yang memberikan biaya terkecil atau keuntungan terbesar. FLP juga meliputi masalah penentuan jumlah ideal suatu fasilitas yang ingin didirikan. Strategi penentuan lokasi merupakan salah satu aktivitas kunci dalam *supply chain management*, selain strategi transportasi dan strategi inventory.

Pencarian lokasi suatu fasilitas biasanya dipengaruhi oleh sebuah faktor yang dominan. Pada penentuan lokasi untuk suatu fasilitas organisasi *manufacturing*, seperti pabrik atau gudang, faktor biaya menjadi hal yang paling dominan. Lokasi terbaik pada kasus ini adalah lokasi yang memberikan biaya paling minimal untuk pengantaran bahan mentah dari *supplier* ke pabrik dan dalam pengantaran produk jadi dari pabrik ke konsumen.

Pada penentuan lokasi untuk suatu fasilitas organisasi jasa, seperti retail atau *service center*, faktor keuntungan menjadi hal yang paling dominan. Lokasi terbaik pada kasus ini adalah lokasi yang memberikan keuntungan terbesar. Untuk fasilitas publik seperti sekolah, terminal dan rumah sakit, faktor yang paling

dominan adalah kemudahan untuk dijangkau. Lokasi terbaik dalam hal ini adalah tempat yang paling mudah diakses oleh masyarakat.



Gambar 2.1. Strategy Dalam Supply Chain Management

(Sumber: Ronald H Ballou, 2004)

FLP biasanya muncul saat suatu perusahaan akan merancang sistem logistiknya. Selain itu, FLP juga seringkali muncul saat perusahaan mengeluarkan produk baru atau mengganti bahan dasar atau proses produksi untuk salah satu produknya. Ketika menghadapi FLP, perusahaan dihadapkan pada 2 pilihan, yaitu membangun sendiri atau menyewanya dari pihak lain. Keputusan ini akan sangat berpengaruh pada besar investasi dan resiko yang mungkin akan dihadapi. Keuntungan dari membangun sendiri suatu fasilitas adalah biaya operasional yang relatif lebih kecil daripada menyewanya. Namun, resiko yang dihadapi juga lebih

Universitas Indonesia

besar. Bila ternyata lokasi tersebut tidak ditempatkan secara optimal, perusahaan akan kesulitan untuk memindahkan fasilitas tersebut dalam waktu dekat.

Menurut Ballou (2004), FLP memiliki banyak variasi, tergantung pada faktor pengklasifikasiannya. Faktor pengklasifikasian yang umum digunakan adalah jumlah fasilitas yang akan dibangun. Berdasarkan faktor ini, ada dua jenis variasi FLP. Yaitu *single facility location problem* (bila fasilitas yang akan didirikan berjumlah satu buah) dan *multiple facility location problem* (bila fasilitas yang akan didirikan berjumlah lebih dari satu). Kriteria lain yang sering menjadi dasar pengklasifikasian antara lain *time horizon* (*single-period problem* dan *multi-period problem*), *facility typology* (*single-type location problem* dan *multi-type location problem*), *material flow* (*single commodity* dan *multi-commodity*).

2.1.2 Metode Penyelesaian

Menurut Dwiningsih (2002), ada empat metode yang dapat digunakan untuk menganalisis untuk menentukan lokasi bagi perusahaan untuk melakukan kegiatan operasionalnya.

1. Pemingkatan Lokasi.

Metode ini mementingkan adanya obyektifitas dalam proses mengenali biaya yang sulit untuk dievaluasi. Faktor yang dipertimbangkan faktor baik yang kualitatif maupun kuantitatif dianalisis dengan cara mengkuantifisir semua faktor. Metode ini bisa diterapkan untuk faktor-faktor yang secara umum digunakan untuk memilih lokasi, maupun faktor-faktor yang dipertimbangkan untuk memilih Negara, wilayah, tempat bagi pemilihan lokasi untuk perusahaan global.

2. *Break Even Analysis*.

Metode ini merupakan sebuah analisis biaya-volume produksi untuk membuat perbandingan ekonomis alternatif lokasi. Data yang diperlukan adalah biaya baik biaya tetap maupun biaya variabel, sedangkan analisisnya dapat dilakukan secara matematis maupun grafis. Akan tetapi pendekatan grafis memiliki kelebihan karena memberikan rentang jumlah volume dimana lokasi dapat dipilih.

3. *Center of Gravity*.

Metode ini merupakan sebuah teknik matematis yang digunakan untuk menemukan lokasi yang paling baik untuk suatu titik distribusi tunggal yang melayani beberapa toko atau daerah. Metode ini memperhitungkan jarak lokasi pasar, jumlah barang yang dikirim dan biaya pengiriman.

4. Model Transportasi.

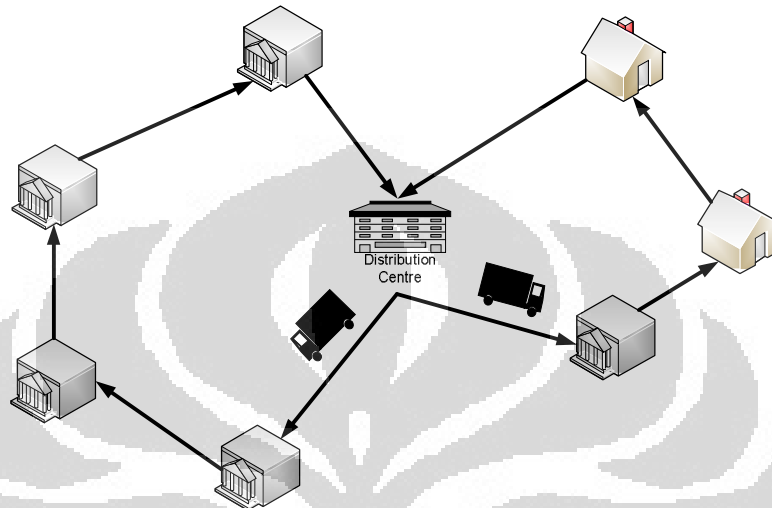
Merupakan sebuah teknik untuk menyelesaikan masalah sebagai bagian dari pemrograman linear. Tujuan model transportasi adalah menetapkan pola pengiriman terbaik dari beberapa titik *supplier* ke beberapa titik permintaan pabrik (tujuan) sedemikian rupa sehingga meminimalkan biaya produksi dan transportasi total. Model ini sering disebut sebagai *vehicle routing problem*. Model inilah yang akan digunakan dalam penelitian ini.

2.2 Vehicle Routing Problem

Dalam sistem distribusi, transportasi merupakan salah satu aktivitas yang mempengaruhi kepuasan pelanggan dan biaya distribusi. Besar biaya transportasi dalam suatu perusahaan biasanya mencapai 1/3-2/3 dari total biaya distribusi (Ballou, 2004). Kontribusi biaya yang cukup besar ini menyebabkan pentingnya usaha untuk meningkatkan efisiensi aktivitas transportasi melalui penggunaan sumber daya secara maksimal. Salah satu cara yang dapat dilakukan adalah memaksimalkan penggunaan kendaraan (*vehicle*) untuk mengantarkan barang dari depot kepada konsumen dengan mencari jalur terbaik yang harus dilewati kendaraan tersebut sehingga waktu dan jarak tempuhnya menjadi minimal. Namun, dalam menentukan jalur terbaik ini, ada beberapa kendala yang harus dihadapi, antara lain kuantitas permintaan pengiriman atau pengambilan pada tiap titik, kapasitas kendaraan yang digunakan, kendala waktu dan lain-lain. Jalur yang dihasilkan harus dapat memenuhi semua kendala tersebut dengan jarak, waktu dan biaya yang optimal.

Masalah pencarian jalur terbaik ini disebut dengan *vehicle routing problem* (VRP). VRP pertama kali diperkenalkan oleh Dantzig dan Ramser pada tahun 1959. VRP dapat didefinisikan sebagai suatu pencarian solusi yang meliputi penentuan sejumlah rute, dimana masing-masing rute dilalui oleh satu kendaraan

yang berawal dan berakhir di depot asalnya, sehingga kebutuhan/permintaan semua pelanggan terpenuhi dengan tetap memenuhi kendala operasional yang ada, juga dengan meminimalkan biaya transportasi global (Toth dan Vigo, 2002).



Gambar 2.2. Ilustrasi VRP

Toth dan Vigo (2002) juga menyebutkan beberapa hal yang merupakan karakteristik utama VRP, yaitu:

- jaringan jalan, biasanya direpresentasikan dalam sebuah *graph* (diagram) yang terdiri dari *arc* (lengkung atau bagian-bagian jalan) dan *vertex* (titik lokasi konsumen dan depot). Tiap lengkung diasosiasikan dengan biaya (jarak) dan waktu perjalanan (tergantung jenis kendaraan, kondisi/karakteristik jalan, dan periode pelintasan)
- konsumen, ditandai dengan *vertex* (titik) dan biasanya memiliki beberapa hal yang perlu diperhatikan seperti jumlah permintaan barang (untuk dikirim ataupun diambil) dan jenis barang dapat berbeda-beda; periode pelayanan tertentu (*time windows*), dimana di luar rentang waktu tersebut konsumen tidak dapat menerima pengiriman maupun pengambilan; Waktu yang dibutuhkan untuk menurunkan atau memuat barang (*loading/unloading time*) pada lokasi konsumen, biasanya tergantung dari jenis kendaraan; Pengelompokan (*subset*) kendaraan yang tersedia untuk melayani konsumen (sehubungan dengan keterbatasan akses atau persyaratan pemuatan dan

penurunan barang); Prioritas atau pinalti sehubungan dengan kemampuan kendaraan untuk melayani permintaan

- depot, juga ditandai dengan suatu titik, merupakan ujung awal dan akhir dari suatu rute kendaraan. Tiap depot memiliki sejumlah kendaraan dengan jenis dan kapasitas tertentu yang dapat digunakan untuk mendistribusikan produk
- kendaraan/armada angkut, memiliki beberapa hal yang perlu diperhatikan seperti depot asal kendaraan, dan kemungkinan untuk mengakhiri rutenya di depot lain; Kapasitas (berat, volume atau jumlah palet yang dapat diangkut); Kemungkinan untuk dipisah menjadi beberapa kompartemen untuk mengangkut barang dengan jenis yang berbeda-beda; Alat yang tersedia untuk operasi (pemuatan atau penurunan barang); Pengelompokan (*subset*) lintasan/lengkung dari diagram jaringan jalan; Biaya yang berhubungan dengan penggunaan kendaraan tersebut (unit per jarak, unit per waktu, unit per rute, dan lainnya)
- pengemudi, memiliki kendala seperti jam kerja harian, jumlah dan jam istirahat, durasi maksimum perjalanan, serta lembur yang biasanya juga dikenakan pada kendaraan yang digunakan.

Masih menurut Toth dan Vigo (2002), dalam membuat konstruksi rute, terdapat beberapa kendala yang harus dipenuhi, seperti jenis barang yang diangkut, kualitas dari pelayanan, juga karakteristik konsumen dan kendaraan. Beberapa kendala operasional yang sering ditemui misalnya:

- pada tiap rute, besar muatan yang diangkut oleh kendaraan tidak boleh melebihi kapasitas kendaraan tersebut
- konsumen yang dilayani dalam sebuah rute dapat hanya merupakan pengiriman atau pengambilan, atau mungkin keduanya
- konsumen hanya boleh dilayani sekali oleh sebuah kendaraan
- konsumen mungkin hanya dapat dilayani dalam rentang waktu tertentu (*time windows*) dan jam kerja dari pengemudi kendaraan yang melayaninya
- waktu kendaraan untuk mengirim barang sampai kembali ke depot tidak boleh melebihi waktu operasional depot
- kendala prioritas juga mungkin akan timbul ketika suatu konsumen harus dilayani sebelum konsumen lain. Kendala seperti ini biasanya terdapat pada

kasus *pickup and delivery* (pengambilan dan pengiriman dalam satu rute) atau *VRP with backhauls* dimana pengambilan baru dapat dilakukan setelah semua pengiriman selesai dikarenakan kesulitan dalam mengatur peletakan muatan.

Selain itu, secara umum, ada empat tujuan dari VRP, yaitu:

- meminimalkan biaya transportasi global, terkait dengan jarak dan biaya tetap yang berhubungan dengan kendaraan
- meminimalkan jumlah kendaraan (atau pengemudi) yang dibutuhkan untuk melayani semua konsumen
- menyeimbangkan rute, untuk waktu perjalanan dan muatan kendaraan
- meminimumkan penalti akibat pelayanan yang kurang memuaskan terhadap konsumen, seperti ketidaksanggupan melayani konsumen secara penuh ataupun keterlambatan pengiriman.

Toth dan Vigo (2002) juga menyebutkan bahwa dalam penggunaan VRP untuk dunia nyata, banyak faktor sampingan yang muncul. Faktor-faktor tersebut berpengaruh pada munculnya variasi dari VRP, antara lain:

- *Capacitated VRP (CVRP)*, merupakan kelas VRP yang paling sederhana dan yang paling banyak dipelajari dimana kendala yang ada hanya berupa kapasitas kendaraan yang terbatas
- *Distance Constrained VRP (DCVRP)*, merupakan VRP dengan kendala batasan panjang rute
- *VRP with time windows (VRPTW)*, yaitu kasus VRP dimana setiap konsumen memiliki batasan rentang waktu pelayanan
- *VRP with Pick up and Delivery (VRPPD)*, merupakan VRP dengan pelayanan campuran, yaitu pengiriman dan pengambilan barang dalam satu rute
- *VRP with Backhauls (VRPB)*, dimana pengambilan baru dapat dilakukan setelah semua pengiriman selesai
- *Multiple Depot VRP (MDVRP)*, dimana vendor memiliki lebih dari satu depot untuk melakukan pengiriman kepada konsumen
- *Stochastic VRP (SVRP)*, dimana beberapa nilai seperti jumlah pelanggan, permintaannya, waktu pelayanan dan waktu perjalanan bersifat acak
- *Periodic VRP (PVRP)*, dimana pengiriman harus dilakukan dalam beberapa hari.

2.2.1 Vehicle Routing Problem with *Time Windows*

Braysy dan Gendreau (2002) mendefinisikan *vehicle routing problem with time windows* (VRPTW) sebagai perluasan dari normal VRP yang paling sering ditemukan dalam pengambilan keputusan mengenai distribusi barang dan jasa. Setiap kendaraan yang bertugas pada VRP jenis ini hanya dapat keluar dari depot pada jam kerja depot dan melayani konsumen pada jangka waktu tertentu yang ditentukan oleh pihak konsumen. Tiap kendaraan juga harus kembali lagi ke depot sebelum jam kerja depot berakhir. Tujuan dari VRPTW adalah menentukan sejumlah rute untuk melayani seluruh konsumen dengan biaya terkecil (dalam hal ini yang dimaksud dengan biaya adalah jarak tempuh) tanpa melanggar batasan kapasitas dan waktu tempuh kendaraan serta batasan waktu yang diberikan oleh pihak pelanggan. Jumlah rute yang ditentukan tidak boleh melebihi jumlah kendaraan yang ada.

2.2.1.1 Model Matematika VRPTW

Dalam penelitian mengenai VRPTW, Tan, Lee dan Zhu (1999) menyatakan bahwa VRPTW didefinisikan oleh sejumlah kendaraan yang identik (V), sebuah titik khusus yang disebut depot, sejumlah pelanggan yang harus dilayani (C), dan sebuah rute yang menghubungkan depot dengan sejumlah pelanggan. Asumsikan bahwa terdapat K kendaraan sehingga $V = \{0, 1, 2, \dots, K-1\}$ dan terdapat $N+1$ pelanggan sehingga $C = \{0, 1, 2, \dots, N\}$. Depot disebut dengan pelanggan ke-0.

Sebuah rute akan dimulai dari depot kemudian menuju sejumlah pelanggan pada rute tersebut dan akan berakhir kembali di depot. Jumlah rute yang tercipta pada VRPTW sama dengan jumlah kendaraan yang ada, yaitu K . Jadi, bila diilustrasikan akan ada K garis yang keluar dari depot dan K garis yang masuk ke depot. Setiap rute memiliki biaya sebesar c_{ij} (jarak antara titik i ke titik j) dan waktu tempuh t_i (waktu antara titik i ke titik j). Setiap pelanggan dalam sebuah rute hanya dapat dilayani sebanyak 1 kali oleh 1 kendaraan. Karena tiap kendaraan memiliki kapasitas maksimum sebesar q_k dan masing-masing pelanggan memiliki permintaan yang berbeda-beda, maka q_k harus lebih besar atau sama dengan jumlah permintaan seluruh pelanggan yang berada pada suatu

rute. Disamping itu, setiap pelanggan harus dilayani dalam interval waktu tertentu. Kendaraan yang tiba sebelum interval waktu harus menunggu, sedangkan kendaraan yang datang setelah interval waktu akan mendapatkan penalti. Setiap kendaraan harus menyelesaikan tugasnya selama jam kerja depot. Untuk menyelesaikan masalah VRPTW, kita harus menentukan sejumlah rute (sekaligus urutan pengantaran di tiap rute) untuk melayani seluruh konsumen dengan biaya terkecil tanpa melanggar batasan-batasan yang disebutkan diatas.

Ada dua jenis variabel keputusan dalam VRPTW. Variabel keputusan pertama adalah x_{ijk} ($i, j=0, 1, 2, \dots, N; k=0, 1, 2, \dots, k : i \neq j$). Variabel ini akan bernilai 1 bila kendaraan k berangkat dari titik i menuju j dan akan bernilai 0 untuk keadaan sebaliknya. Variabel keputusan kedua adalah t_i . Variabel ini menunjukkan waktu sebuah kendaraan saat mulai melayani pada titik i .

Model matematik untuk masalah ini adalah sebagai berikut:

Tujuan:

$$\text{Min} \sum_{i=0}^N \sum_{j=0}^N \sum_{k=0}^{K-1} c_{ij} x_{ijk} \quad (2.1)$$

Kendala:

$$\sum_{k=0}^{K-1} \sum_{j=1}^N x_{ijk} = K \text{ for } i = 0 \quad (2.2)$$

$$\sum_{k=0}^{K-1} \sum_{j=0, j \neq i}^N x_{ijk} = 1 \text{ for } i = 1, 2, \dots, N \quad (2.3)$$

$$\sum_{i=0}^N x_{ihk} - \sum_{j=0}^N x_{hjk} = 0 \quad \forall h \in [1, N]; k \in [0, K-1] \quad (2.4)$$

$$\sum_{i=0}^N m_i \sum_{j=0, j \neq i}^N x_{ijk} \leq q_k \quad \forall k \in [0, K-1] \quad (2.5)$$

$$\sum_{i=0}^N \sum_{j=0, j \neq i}^N x_{ijk} (t_{ij} + f_i + w_i) \leq r_k \quad \forall k \in [0, K-1] \quad (2.6)$$

$$t_0 = 0 \quad (2.7)$$

$$t_i + x_{ijk}(t_{ij} + f_i + w_i) \leq t_j \quad i, j \in [1, N]; i \neq j; k \in [0, K-1] \quad (2.8)$$

$$e_i \leq t_i \leq l_i \quad k \in [0, K-1] \quad (2.9)$$

Keterangan:

K = Jumlah kendaraan

N = Jumlah pelanggan

C_i = pelanggan ke- i

C_0 = depot

C_{ij} = jarak dari titik i ke titik j

t_{ij} = waktu tempuh dari titik i ke titik j

m_i = permintaan di titik i

q_k = kapasitas kendaraan k

t_i = waktu sampai di titik i

e_i = *earliest arrival time* di titik i

f_i = *service time* di titik i

w_i = waktu tunggu di titik i

r_k = waktu kerja maksimum kendaraan k

Kendala 1 untuk memastikan bahwa ada sebanyak K rute yang keluar dari depot. Kendala 2 dan 3 memastikan bahwa sebuah pelanggan hanya dilayani oleh sebuah kendaraan. Kendala 4 memastikan bahwa muatan yang dibawa tidak melebihi kapasitas maksimum kendaraan. Kendala 5 merupakan kendala batasan *travel time* kendaraan (*time windows* depot). Kendala 6 – kendala 8 memastikan bahwa pengantaran dilakukan dalam *time windows* pelanggan.

2.2.2 Metode Penyelesaian

Menurut Toth dan Vigo (2002), VRP dapat diselesaikan dengan menggunakan 2 jenis pendekatan, yaitu eksak dan heuristik. Kemudian, secara umum, kelas penyelesaian heuristik dalam VRP dapat dibagi menjadi dua, yaitu heuristik klasik dan heuristik modern (metaheuristik).

2.2.2.1 Pendekatan Eksak

Pada solusi eksak, dilakukan pendekatan dengan menghitung setiap solusi yang mungkin sampai satu terbaik dapat diperoleh. Terdapat beberapa algoritma eksak utama penyelesaian VRP, yaitu:

- *branch and bound*

- *branch and cut*
- *set covering based*.

Secara umum penggunaan metode eksak untuk penyelesaian VRP akan menghabiskan waktu yang lama. Hal tersebut dikarenakan VRP termasuk dalam permasalahan NP-hard (*Non Polynomial-hard*) dimana kompleksitas penyelesaian permasalahan akan meningkat secara eksponensial dengan semakin besarnya permasalahan.

2.2.2.2 Heuristik Klasik

Prosedur konstruksi dan perbaikan solusi masalah VRP yang umum digunakan saat ini berasal dari kelas heuristik klasik. Secara umum, terdapat tiga metode heuristik klasik VRP, yaitu:

1. Heuristik konstruktif (*constructive heuristics*), yaitu penyusunan solusi yang memungkinkan dengan memperhatikan biaya solusi tersebut, tanpa dilakukan fase perbaikan.
2. Heuristik dua fase (*two-phase heuristics*), dimana penyelesaian masalah dibagi menjadi dua bagian, yaitu pengelompokan (*clustering*) rute dan pengkonstruksian rute-rute tersebut. Urutan pengerjaan dapat berupa *Route First - Cluster Second*, dimana pengelompokan dilakukan setelah rute dibuat, atau *Cluster First - Route Second*, dimana rute dibuat setelah pengelompokan dilakukan.
3. Metode perbaikan (*improvement method*), yaitu metode yang bertujuan untuk memperbaiki solusi yang mungkin dengan melakukan pertukaran titik dalam satu rute atau antar-rute.

Dalam metode heuristik konstruktif, terdapat dua teknik penyelesaian masalah VRP, yaitu dengan penyatuan dua rute berdasarkan kriteria penghematan dan dengan penugasan secara bertahap dari masing-masing titik ke dalam rute berdasarkan biaya penugasan (*insertion cost*). Untuk teknik yang pertama, biasa digunakan algoritma penghematan (*saving algorithm*). Kemudian, untuk teknik yang kedua, *sequential Insertion heuristics*, dapat digunakan algoritma Mole & Jameson (1976) dan algoritma Christofides, Mingozzi & Toth (1979).

Metode heuristik dua fase dapat dibagi menjadi dua kelompok (Toth dan Vigo, 2002), yaitu:

- *Cluster First – Route Second* (pengelompokan kemudian pembuatan rute), terdiri dari Metode pengelompokan elementer, yaitu dengan melakukan suatu pengelompokan dan kemudian menentukan rute pada tiap kelompok. Contoh algoritmanya adalah algoritma penyapuan (*Sweep Algorithm*), algoritma Fisher & Jaikumar (1981), dan algoritma Bramel & Simichi-Levi (1995); *Truncated Branch-and-Bound* yang merupakan penyederhanaan dari algoritma eksak *Branch-and-Bound*; Algoritma kelopak bunga (*Petal Algorithm*) yang menghasilkan sekeluarga besar dari kelompok-kelompok yang saling bertumpuk (*overlapping cluster*) dan rute di dalamnya, kemudian menyeleksi sejumlah rute yang memungkinkan.
- *Route First – Cluster Second* (pembuatan rute kemudian pengelompokan). Dalam fase pertama metode ini, dibuat sebuah tur raksasa berdasarkan TSP (*Traveling salesman Problem*) yang tidak melibatkan kendala-kendala sampingan. Pada fase kedua dilakukan pemecahan tur tersebut menjadi beberapa kelompok rute yang memungkinkan.

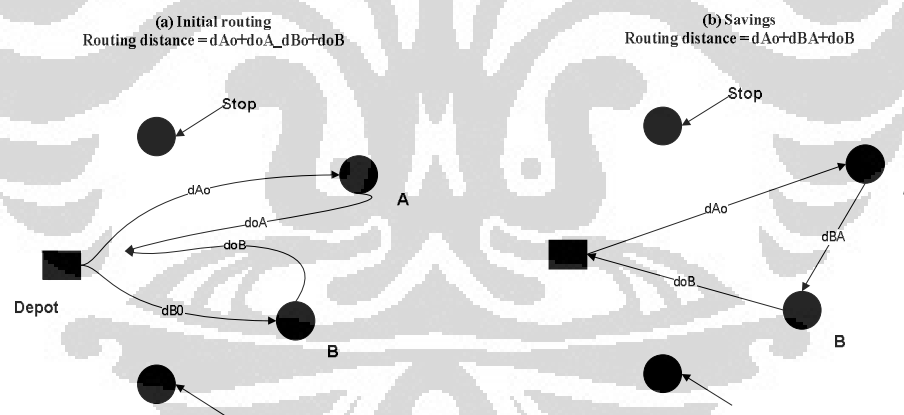
Diantara sekian banyak algoritma heuristik diatas, ada dua algoritma yang paling banyak dikenal, yaitu *Sweep method* dan *Saving method*. *Sweep method* merupakan salah satu metode dalam penyelesaian VRP dengan pengelompokan elementer. Menurut Ballou dan Agarwal (1998), keakuratan metode ini adalah pada cara pembuatan jalur rutenya. Prosesnya terdiri dari dua tahap, pertama titik pemberhentian ditentukan untuk kendaraan yang ada. Tahap kedua adalah menentukan urutan titik pemberhentian pada rute. Karena melibatkan dua tahapan proses maka total waktu dalam suatu rute dan batasan waktu tidak dapat ditangani dengan baik oleh metode ini.

Metode ini termasuk di dalam jenis metode *cluster* atau pengelompokan, yang mana pengelompokan awal dilakukan dengan menggabungkan perhentian-perhentian yang setiap kelompok mengakomodasi volume masing-masing perhentian. Volume total perhentian dari satu rute mungkin akan melebihi kapasitas kendaraan. Jika terjadi hal tersebut maka beberapa perhentian

dipindahkan ke kendaraan yang kapasitasnya belum penuh. Relokasi seperti ini dilakukan dengan menggunakan metode transportasi *linear programming*.

Saving method ditemukan oleh Clarke dan Wright pada tahun 1964. Konsep dasar dari metode ini adalah melakukan penghematan jarak tempuh dengan melakukan penyatuan dua rute. Rute yang dipilih untuk digabung adalah yang memberikan penghematan (*saving*) terbesar dari jarak total yang ditempuh untuk semua rute. Penggabungan rute ini dilakukan terus sehingga tidak ada lagi penghematan yang dapat dilakukan atau sampai kapasitas kendaraan tidak dapat ditambah lagi.

Menurut Ballou (2004), keunggulan dari metode ini adalah hambatan-hambatan dalam praktek dapat diakomodir, misalnya pengambilan dan pengiriman dalam satu rute, adanya waktu khusus untuk pengiriman, dan tipe kendaraan yang beragam. Namun metode ini juga memiliki kelemahan, yaitu waktu perhitungan cenderung meningkat secara geometri seiring dengan bertambahnya jumlah rute atau perhentian.



Gambar 2.3. *Saving Method*

(Sumber : Ronald H Ballou, Business Logistic Management 5th Edition, 2004)

2.2.2.3 Metaheuristik

Toth dan Vigo (2002) mendefinisikan heuristik modern atau metaheuristik sebagai prosedur pencarian solusi umum untuk melakukan eksplorasi yang lebih dalam pada daerah yang menjanjikan dari ruang solusi yang ada. Perbedaannya dengan heuristik klasik adalah diperbolehkannya perusakan solusi atau penurunan

fungsi tujuan. Kualitas solusi yang dihasilkan dari metode ini jauh lebih baik daripada heuristik klasik. Beberapa contoh metaheuristik adalah *simulated annealing*, *deterministic annealing*, *genetic algorithm*, *neural network*, *ant colony sistem*, dan *tabu search*.

Prinsip dasar algoritma metaheuristik adalah pencarian lokal dan pencarian populasi. Dalam metode pencarian lokal, eksplorasi yang intensif dilakukan terhadap ruang solusi dengan berpindah dari satu solusi ke solusi tetangga lainnya yang potensial dalam satu lingkungan (*neighbourhood*).

Beberapa algoritma metaheuristik yang sering dipakai untuk menyelesaikan masalah VRP adalah sebagai berikut:

1. *Simulated Annealing* (SA).

Ide dasar SA terbentuk dari proses pengolahan logam. *Annealing* dalam proses pengolahan logam berarti proses membuat bentuk cair berangsur-angsur menjadi bentuk yang lebih padat seiring penurunan temperatur. SA biasanya digunakan untuk penyelesaian masalah yang mana perubahan keadaan dari suatu kondisi ke kondisi lainnya membutuhkan ruang yang sangat luas.

2. *Genetic Algorithm* (GA).

Algoritma GA dimodelkan berdasarkan proses alami, yaitu model seleksi alam oleh Darwin. Individu yang dapat bertahan dari proses seleksi adalah individu yang kualitasnya sangat sesuai dengan lingkungannya. Teknik pencarian dilakukan terhadap sejumlah populasi solusi. Tiap populasi diperoleh dari populasi sebelumnya dengan mengkombinasikan elemen-elemen terbaiknya dan membuang elemen yang buruk untuk menghasilkan populasi turunan.

3. *Tabu Search* (TS).

TS merupakan metode optimasi yang menggunakan *short term memory* untuk menjaga agar proses pencarian tidak terjebak pada nilai optimal lokal. Metode ini menggunakan *tabu list* untuk menyimpan sekumpulan solusi yang baru saja dievaluasi. Selama proses optimasi, pada setiap iterasi, solusi yang akan dievaluasi akan dicocokkan terlebih dahulu dengan isi *tabu list* untuk melihat apakah solusi tersebut sudah ada pada *tabu list*. Apabila sudah ada, maka solusi tersebut tidak akan dievaluasi lagi. Keadaan ini terus berulang sampai tidak ditemukan lagi solusi yang tidak terdapat pada *tabu list*. Solusi baru

dipilih jika solusi tersebut merupakan anggota bagian himpunan solusi tetangga dengan fungsi tujuan paling optimal jika dibandingkan dengan solusi-solusi lainnya dalam himpunan solusi tetangga tersebut.

4. *Ant Colony Sistem.*

Pada *Ant colony sistem* sejumlah solusi baru dibuat pada setiap iterasi menggunakan sebagian informasi yang didapat dari iterasi-iterasi sebelumnya.

Selain algoritma-algoritma diatas, ada satu algoritma baru yang juga dapat digunakan untuk menyelesaikan masalah VRP. Algoritma tersebut adalah *Differential Evolution*.

2.3 Algoritma *Differential Evolution*

2.3.1 Sejarah

Onwubolu dan Davendra (2008) menyatakan bahwa algoritma DE pertama kali dikembangkan oleh Storn dan Price pada tahun 1995. DE pertama kali mulai dikembangkan ketika Price mencoba memecahkan permasalahan polinomial Chebyshev yang diajukan oleh Storn. Dalam mencoba memecahkan permasalahan tersebut, Price terinspirasi untuk menggunakan selisih dari vektor dalam mencari suatu solusi penyelesaian, hingga setelah melalui diskusi yang panjang dan simulasi dengan menggunakan program komputer yang dilakukan oleh Storn dan Price, dikembangkanlah *Genetic Annealing* yang merupakan cikal bakal dari DE itu sendiri. DE merupakan algoritma yang masuk kedalam kelompok optimasi yang masuk ke dalam sub-kelompok algoritma evolusioner (EA). Sama seperti EA yang lainnya seperti *Genetic Algorithm (GA)*, *Evolution Strategy*, *Learning Classifier Sistem*, dan lain-lain, DE memiliki konsep yang terinspirasi dari teori evolusi biologi, dimana di dalamnya terdapat reproduksi, mutasi, rekombinasi, dan seleksi.

DE pertama kali dijelaskan oleh Price dan Storn di ICSI *technical report* pada tahun 1995. Satu tahun kemudian, DE sukses didemonstrasikan di *First International Contest on Evolutionary Optimization* yang diadakan bersamaan dengan *International Conference on Evolutionary Computation* yang diadakan oleh IEEE (*Institute of Electrical and Electronics Engineers*) dan berhasil memenangkan tempat ketiga. Terinspirasi dari hasil tersebut, Price dan Storn

menulis sebuah artikel untuk jurnal Dr. Dobbs (“*Differential Evolution: A Simple Evolution Strategy for Fast Optimization*”) yang diterbitkan pada April 1997 dan selanjutnya mereka menerbitkan artikel lagi untuk *Journal of Global Optimization* (“*Differential Evolution: A Simple and Efficient Heuristic for Global Optimization over Continuous Space*”).

DE telah sukses diterapkan di berbagai bidang, baik teknik maupun sains, beberapa contoh diantaranya adalah :

- desain *filter* digital
- pengambilan keputusan untuk produksi bahan bakar alkohol
- proses fermentasi untuk *lot size* tertentu
- perpaduan multi sensor
- optimasi dinamis untuk reaksi polimer yang terus-menerus
- optimasi pertukaran panas
- perencanaan persediaan.

2.3.2 Definisi

Price, Storn dan Lampinen (2005), menyatakan bahwa DE merupakan algoritma optimasi global yang didasarkan pada prinsip evolusi. DE menggunakan vektor-vektor yang merepresentasikan kandidat-kandidat penyelesaian dimana teknik pencariannya dilakukan sekaligus atas sejumlah solusi yang disebut dengan populasi. Populasi awal (generasi ke nol) dibentuk dengan membangkitkan bilangan acak, sedangkan populasi berikutnya merupakan hasil evolusi dari vektor-vektor yang telah melalui tahap reproduksi, mutasi, rekombinasi, dan seleksi. Setiap individu didefinisikan sebagai vektor berdimensi-D dimana vektor-vektor tersebut dilambangkan sebagai $x_{i,g}$ yang merupakan anggota populasi pada generasi ke-g. Populasi dinotasikan sebagai P_x yang terdiri atas vektor-vektor tersebut yang berdimensi N_p dimana N_p merupakan ukuran populasi. Oleh karena itu, Populasi dan vektor yang menjadi calon-calon penyelesaian dapat dilambangkan ke dalam bentuk umum seperti berikut:

$$P_{x,g} = (x_{i,g}), \quad i=0,1,\dots,N_p-1, \quad g=0,1,\dots,g_{\max} \quad (2.10)$$

$$x_{i,g} = (x_{j,i,g}), \quad j=0,1,\dots,D-1 \quad (2.11)$$

Pada setiap generasi, tiap individu calon penyelesaian akan melewati proses evaluasi dimana individu-individu tersebut akan membentuk vektor target dan dihitung fungsi objektifnya (atau seringkali disebut sebagai *fitness function*). Selain itu, individu-individu tersebut akan dilakukan proses mutasi dan pindah silang (*crossover*) agar dapat membentuk vektor *trial* yang digunakan untuk membentuk populasi anak (populasi pada generasi selanjutnya). Populasi generasi selanjutnya akan dibentuk dengan cara membandingkan fungsi objektif dari vektor induk dan anak (vektor *trial*) dimana individu dengan nilai fungsi objektif yang terbaik akan lolos ke generasi selanjutnya. Proses tersebut akan terus diulang hingga kriteria terminasi terpenuhi.

DE memiliki beberapa keunggulan dibandingkan dengan metode optimasi klasik, diantaranya adalah:

- memiliki populasi yang berisikan calon-calon penyelesaian
- merupakan metode non-deterministik yang menghasilkan solusi-solusi yang berbeda meskipun model awalnya tidak dirubah, karena bekerja dengan menggunakan *random sampling*
- menggunakan elemen-elemen dari solusi-solusi yang telah ada untuk menciptakan solusi baru dengan cirri-ciri yang diwariskan dari elemen-elemen induknya.

2.3.3 Tahapan Penyelesaian

Menurut Price, Storn dan Lampinen (2005), tahap-tahap dalam algoritma DE meliputi inisialisasi, mutasi, pindah silang dan penyeleksian. Pada akhir tahap inisialisasi dan pindah silang (tahap yang menghasilkan sebuah populasi) akan dilakukan evaluasi terhadap tiap individu dalam populasi dengan menggunakan fungsi objektif. Untuk lebih jelasnya, langkah-langkah penyelesaian algoritma DE adalah sebagai berikut:

1. Inisialisasi

Tahapan inisialisasi merupakan penetapan parameter kontrol dan populasi awal (generasi ke-0). Tujuan penetapan parameter kontrol adalah untuk menemukan solusi yang dapat diterima melalui sejumlah evaluasi fungsi dan nantinya akan berdampak pada performa DE. DE memiliki parameter kontrol

yang sedikit, dimana hal ini merupakan salah satu keunggulan DE dibandingkan algoritma optimasi lainnya. Parameter kontrol pada DE diantaranya adalah ukuran populasi, parameter kontrol mutasi, dan parameter kontrol pindah silang.

Parameter Ukuran populasi (N_p) digunakan untuk menentukan jumlah solusi awal yang akan dibentuk. Solusi-solusi awal ini akan tergabung dalam sebuah matriks yang disebut populasi awal. Nilai NP didapatkan dari perkalian suatu bilangan n (umumnya 10) dengan suatu dimensi. Dimensi yang digunakan akan berbeda-beda tergantung dengan masalah yang akan diselesaikan. Misalnya, untuk kasus penjadwalan *flow shop*, yang menjadi dimensi adalah jumlah pekerjaan, sedangkan untuk kasus VRP, yang menjadi dimensi adalah jumlah pelanggan yang akan dilayani. Jadi, untuk suatu kasus VRP dengan 20 pelanggan didalamnya, kita akan mempunyai solusi awal sejumlah $NP = n \times 20$. Umumnya nilai NP akan bernilai konstan selama proses DE berjalan, namun bila proses mengalami hambatan maka nilai NP dapat dinaikkan dengan cara menaikkan nilai bilangan n .

Parameter kontrol mutasi (F) merupakan parameter kontrol bernilai bilangan asli positif yang berfungsi dalam mengendalikan tingkat evolusi dari populasi pada proses mutasi. Nilai efektif dari F umumnya berada pada kisaran $[0.4-1]$. Jika F lebih dari 1 maka perpindahan vektor target akan besar, bahkan bisa melewati batas fungsi objektif. Jika F kurang dari 0.4 maka perpindahan vektor target akan sangat kecil bahkan terkesan tidak mengalami perpindahan sehingga dapat mengakibatkan kondisi *stuck*. Selain itu, nilai F yang lebih kecil dari 0.4 juga tidak efektif karena akan membawa vektor mutasi yang mendekati vektor target. Bila proses DE mengalami hambatan maka cara yang dapat dilakukan untuk menyelesaikannya antara lain dengan menaikkan nilai F .

Parameter kontrol pindah silang (Cr) berperan sebagai *fine tuning element* (elemen penentuan) pada saat operasi pindah silang. Nilai dari Cr ini berkisar pada antara $[0-1]$. Nilai CR yang tinggi, misal $CR=1$, mempercepat terjadinya konvergensi. Menurut analisis yang dilakukan, hal tersebut dikarenakan peluang munculnya bilangan acak yang lebih kecil atau sama nilainya dari CR akan semakin besar. Dengan demikian sebagian besar nilai dimensi individu *trial*

berasal dari nilai dimensi individu mutan, dimana individu mutan itu merupakan hasil mutasi tiga buah vektor.

Akibat dari hal itu adalah individu-individu atau vektor-vektor di dalam populasi akan lebih cepat bergerak saling bertemu di suatu titik, namun bukan titik yang optimum. Tidak mencapainya titik optimum karena kurangnya peluang nilai dimensi individu *trial* terisi dari nilai dimensi individu target, padahal individu target itu merupakan individu yang dianggap baik nilainya ketika dievaluasi berdasarkan fungsi tujuan. Salah satu hal yang membedakan algoritma DE dengan algoritma genetik adalah bahwa pada algoritma genetik proses pindah silang mendahului proses mutasi.

Selain penentuan parameter kontrol, tahap inisialisasi juga bertujuan untuk membentuk populasi awal. Populasi awal merupakan matriks yang berisikan sejumlah solusi awal untuk menyelesaikan suatu masalah. Jumlah solusi awal ditentukan oleh parameter ukuran populasi yang telah ditetapkan sebelumnya.

Representasi solusi awal berbeda-beda untuk tiap permasalahan. Untuk masalah penjadwalan flow shop, solusi awal akan merepresentasikan urutan pengerjaan suatu pekerjaan, sedangkan untuk masalah VRP, solusi awal akan menunjukkan urutan konsumen yang harus dilayani pada suatu rute. Solusi awal dapat diperoleh melalui suatu metode heuristik ataupun dengan menggunakan bilangan acak.

2. Evaluasi

Setelah populasi awal didapatkan, maka selanjutnya masing-masing solusi awal pada populasi awal (selanjutnya disebut dengan individu awal) akan dievaluasi sesuai dengan fungsi objektifnya. Individu awal dengan nilai objektif terbaik akan dijadikan individu target.

3. Mutasi

Setelah melakukan inisialisasi, proses selanjutnya adalah proses mutasi. Mutasi merupakan proses untuk membentuk vektor mutasi yang diperoleh dari mengalikan selisih dari dua individu pada populasi awal yang dipilih secara acak dengan parameter kontrol mutasi (F) lalu dijumlahkan dengan individu yang ketiga yang juga dipilih secara acak. Rumus dari proses mutasi ini adalah sebagai berikut:

$$X_{c'} = X_c + F(X_a - X_b) \quad (2.12)$$

Hasil dari proses mutasi adalah terbentuknya populasi individu mutan. Populasi individu mutan dan populasi awal akan dijadikan input untuk proses pindah silang.

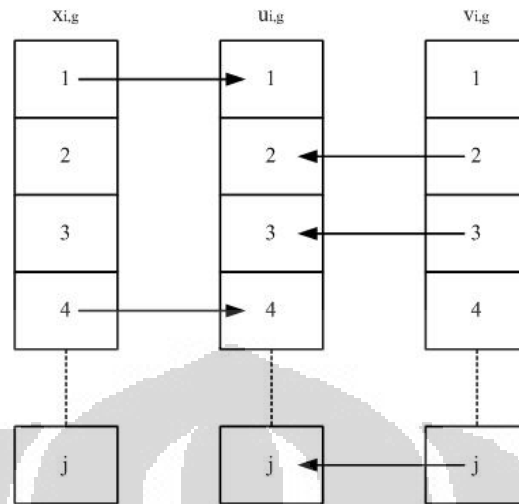
4. Pindah Silang

Untuk melengkapi proses mutasi, DE juga menggunakan proses pindah silang (*crossover*), atau terkadang disebut sebagai rekombinasi diskrit (*discrete recombination*). Menurut Price, Kenneth, Storn dan Lampinen (2005), pindah silang merupakan proses yang bertujuan untuk memperkaya keanekaragaman gen dalam populasi yang akan memasuki generasi yang berikutnya dengan menyilangkan gen yang dimiliki oleh populasi individu mutan dengan populasi awal sehingga membentuk populasi individu *trial*. Proses pindah silang ini melibatkan parameter kontrol pindah silang (*Cr*). Penentuan ini dilakukan dengan cara membandingkan nilai *Cr* tersebut dengan bilangan yang dibangkitkan secara acak. Jika nilai *Cr* lebih besar dari bilangan acak, maka gen dari individu mutasi akan lolos untuk memasuki vektor *trial*, sedangkan jika nilai *Cr* lebih kecil atau sama dengan bilangan acak, maka gen dari individu awal yang akan lolos memasuki vektor *trial*.

Bila individu awal dinotasikan dengan $x_{j,i,g}$ dan individu mutan dinotasikan dengan $v_{j,i,g}$ maka formula umum dari proses pindah silang untuk menghasilkan individu *trial* $u_{i,g}$ adalah sebagai berikut:

$$\mathbf{u}_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} & \text{if } (\text{rand}_j(0,1) \leq Cr \text{ or } j = j_{rand}) \\ x_{j,i,g} & \text{if } (\text{rand}_j(0,1) > Cr \text{ or } j = j_{rand}) \end{cases} \quad (2.13)$$

Setelah diperoleh populasi dari individu *trial*, maka individu *trial* itu akan dievaluasi nilai objektifnya sebagaimana evaluasi yang dilakukan terhadap individu target dimana nilai ini digunakan pada proses selanjutnya, yaitu proses seleksi.



Gambar 2.4. Proses Terjadinya Pindah Silang

(Sumber :Ownobolu dan Davendra (2006))

5. Penyeleksian

Tahapan ini merupakan tahapan dimana terjadi pemilihan antara individu target dan individu *trial* yang akan lolos untuk masuk ke generasi yang selanjutnya. Penyeleksian dilakukan dengan cara membandingkan nilai yang merupakan hasil dari evaluasi nilai objektif pada individu target dan individu *trial*. Individu yang akan lolos ke generasi selanjutnya adalah individu yang memiliki nilai evaluasi yang terbaik seperti yang ditunjukkan oleh bentuk umum di bawah ini:

(2.14)

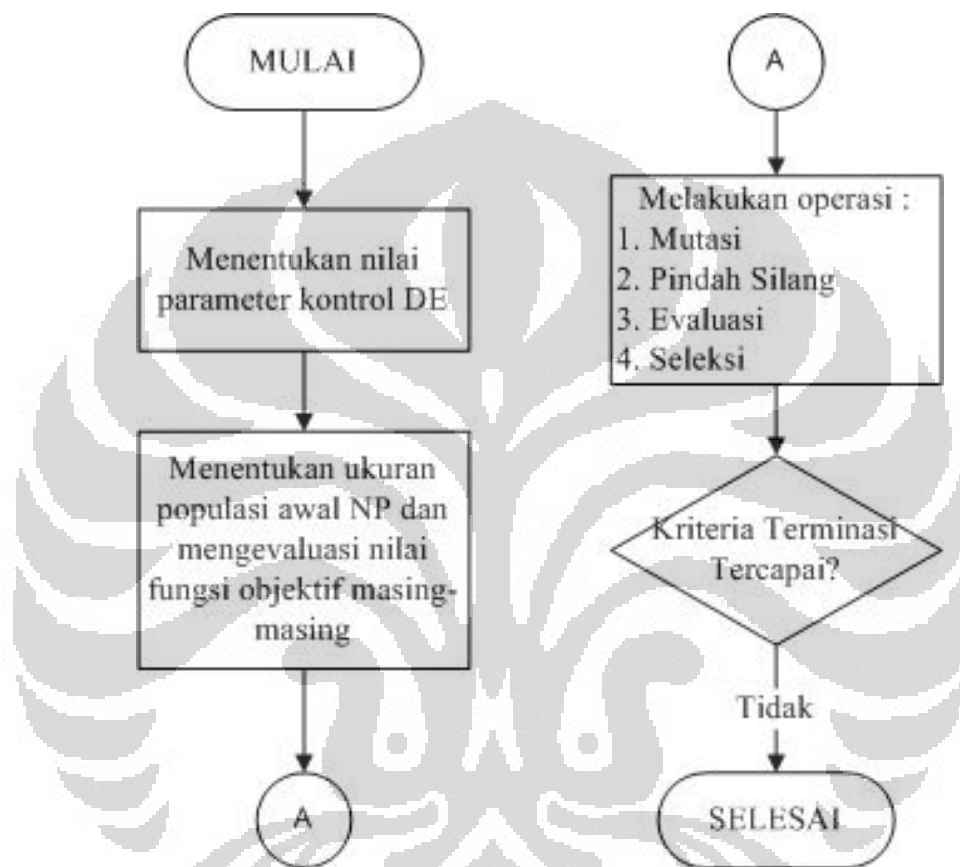
6. Terminasi

Terminasi merupakan keadaan dimana proses pencarian solusi optimal berhenti. Terminasi terjadi ketika proses pencarian solusi optimal telah mencapai kriteria terminasi. Namun, bila kriteria terminasi belum terpenuhi, maka akan dibentuk lagi generasi baru dengan mengulangi langkah-langkah sebelumnya dari awal. Umumnya kriteria terminasi adalah sebagai berikut:

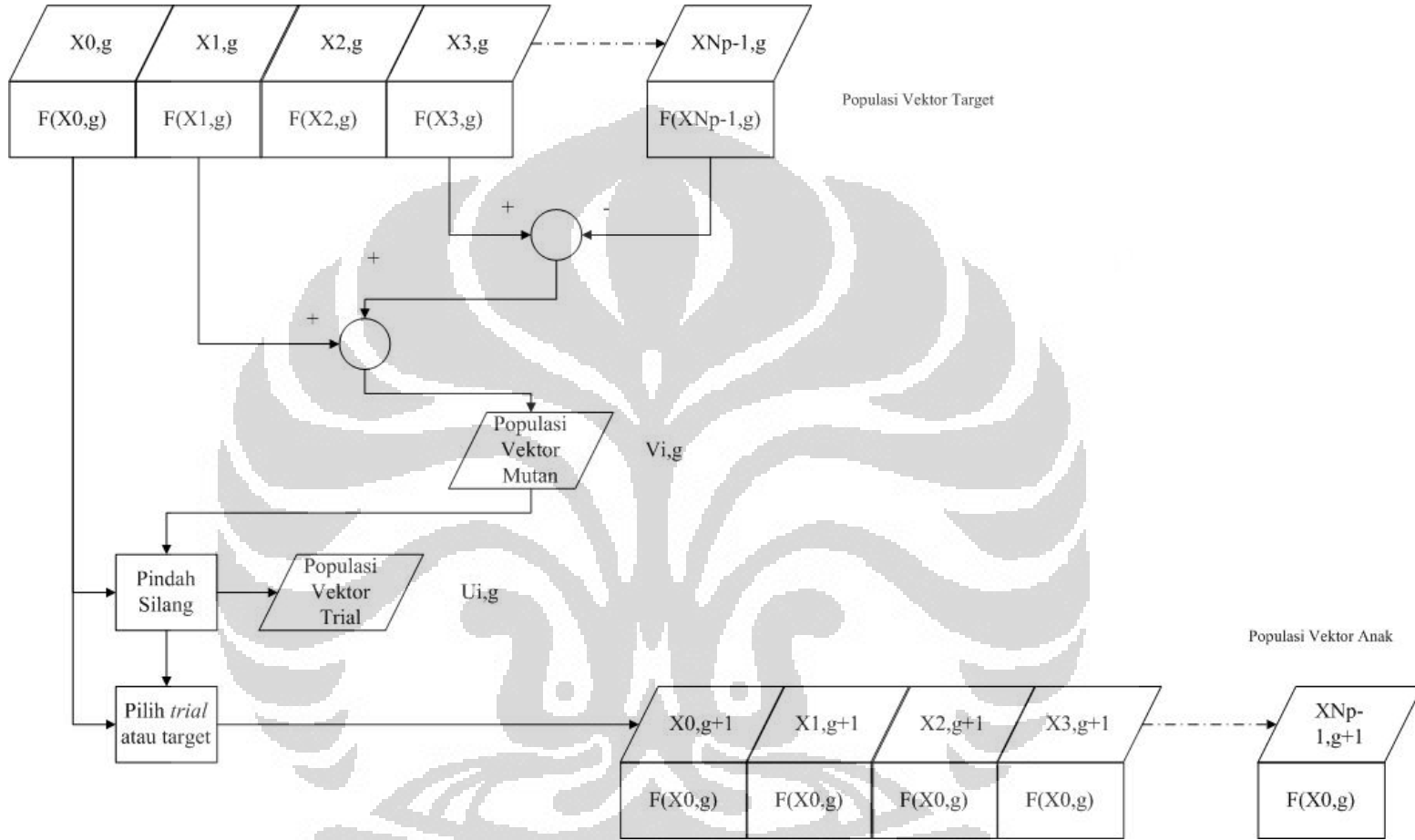
- Jumlah iterasi maksimum
- Waktu komputasi maksimum

- Mencapai keadaan konvergen (nilai dari fungsi objektif yang optimal tidak lagi berubah)

Secara lebih detail, tahapan proses pengerjaan DE ditunjukkan pada gambar berikut:



Gambar 2.5. Flow Chart Pengerjaan



Gambar 2.6. Tahapan Kerja DE

2.4 Annual Cash Flow Analysis

Newnan (1999) menyebutkan bahwa Annual cash flow analysis merupakan salah satu teknik evaluasi biaya terhadap beberapa alternative untuk menghasilkan data yang akurat dalam proses pengambilan keputusan. Dengan teknik ini, suatu biaya yang dikeluarkan pada suatu waktu dapat dirubah menjadi biaya equivalen untuk jangka waktu tertentu (*Equivalent Uniform Annual Cost*). Misalnya, bila kita membeli sebuah mesin saat ini dan mengharapkan mesin itu dapat bekerja selama 20 tahun, maka kita dapat mengkonversi biaya pembelian mesin tersebut menjadi biaya tahunan yang equivalen untuk jangka waktu 20 tahun. Rumus dasar untuk menghitungnya adalah sebagai berikut:

$$EUAC = P(A/P, i, n) \quad (2.15)$$

Dengan P adalah biaya pembelian suatu peralatan, i adalah interest rate dan n adalah jangka waktu pemakaian fasilitas tersebut (*expected life time*). Dalam beberapa kasus, umur pakai fasilitas tidak dapat diperkirakan. Contohnya adalah tanah. Tanah diasumsikan memiliki umur pakai yang tidak terbatas (dapat dipakai selamanya). Untuk menghitung umur fasilitas yang tidak dapat diperkirakan umur pakainya, rumus yang dapat digunakan adalah:

$$EUAC = P I \quad (2.16)$$

BAB 3

PENGUMPULAN DATA

Bab ini menjelaskan mengenai pengumpulan data yang dibutuhkan untuk melaksanakan penelitian. Pengumpulan data dilakukan dengan cara observasi, wawancara, dan pengumpulan dokumen di Perusahaan. Data yang diperlukan dibagi menjadi dua, yaitu data untuk menghitung biaya operasional bulanan dan data untuk menghitung biaya pembelian tanah bulanan.

3.1 Data Untuk Menghitung Biaya Operasional

Untuk mengetahui lokasi paling optimal untuk mendirikan DC, kita perlu menghitung biaya yang perlu dikeluarkan di tiap kandidat lokasi pendirian. Biaya ini meliputi biaya operasional dan biaya pembelian tanah. Biaya operasional adalah biaya yang perlu dikeluarkan untuk mendistribusikan barang sampai ke tangan konsumen, meliputi biaya dari depot ke DC dan dari DC ke konsumen. Biaya ini linear dengan jarak. Karena itu, untuk mengetahui biaya ini, kita perlu mengetahui jarak tempuh untuk masing-masing kandidat dalam melayani semua konsumen. Untuk melakukan ini, data yang diperlukan adalah data mengenai depot, kandidat lokasi, armada pengiriman, biaya pengiriman, konsumen, permintaan, waktu dan jarak.

3.1.1 Depot

Depot merupakan titik awal dimana pengiriman produk dimulai. Pada kasus ini, depot terletak di perusahaan agribisnis itu sendiri, yaitu di daerah Ciawi, Bogor.

Saat ini, depot mulai bekerja sejak pukul 02.00 sampai dengan pukul 12.00. Sayuran yang datang sejak malam hari akan mulai diproses pada pukul 02.00 dan mulai diantarkan ke konsumen pada pukul 04.00. Namun, bila DC sudah berhasil dibuat, jam kerja depot akan dimulai dari pukul 00.00-12.00. Setiap sayuran yang masuk ke depot, baik itu dari perkebunan sendiri ataupun dari mitra tani akan diproses sejak pukul 00.00 dan kemudian diangkut menuju DC pada pukul 02.00.

3.1.2 Kandidat Lokasi Pendirian DC

Saat ini, Perusahaan memiliki 3 buah kandidat lokasi untuk pendirian DC. Lokasi ini adalah Cibubur, Bintaro dan Pulo Gadung. Luas area dan luas bangunan DC akan sama, dimanapun letaknya. Jadi, faktor yang membedakan ketiga kandidat ini satu sama lain adalah harga tanah dan letaknya terhadap konsumen dan depot (dinilai dengan jarak). Untuk selanjutnya, lokasi Cibubur, Bintaro dan Pulo Gadung masing-masing akan disebut Kandidat 1, Kandidat 2 dan Kandidat 3.

3.1.3 Konsumen

Jumlah total konsumen yang dimiliki Perusahaan adalah 84 pelanggan, yang tersebar di daerah Jabodetabek, Bandung, Semarang dan Yogyakarta. Dari jumlah ini, hanya 64 pelanggan yang akan dilayani kebutuhannya oleh DC. Pelanggan ini merupakan pelanggan yang terletak di daerah Jabodetabek. Pelanggan untuk daerah Bogor, Bandung, Semarang dan Yogyakarta akan dilayani langsung dari depot, karena itu dalam penelitian ini pelanggan tersebut tidak akan diikutsertakan dalam perhitungan.

Setiap pelanggan memiliki kode nama tersendiri yang digunakan untuk memudahkan pengidentifikasian untuk beberapa kepentingan. Berikut ini adalah nama pelanggan (titik kirim) beserta dengan kode namanya.

Tabel 3.1. Daftar Nama Pelanggan Perusahaan

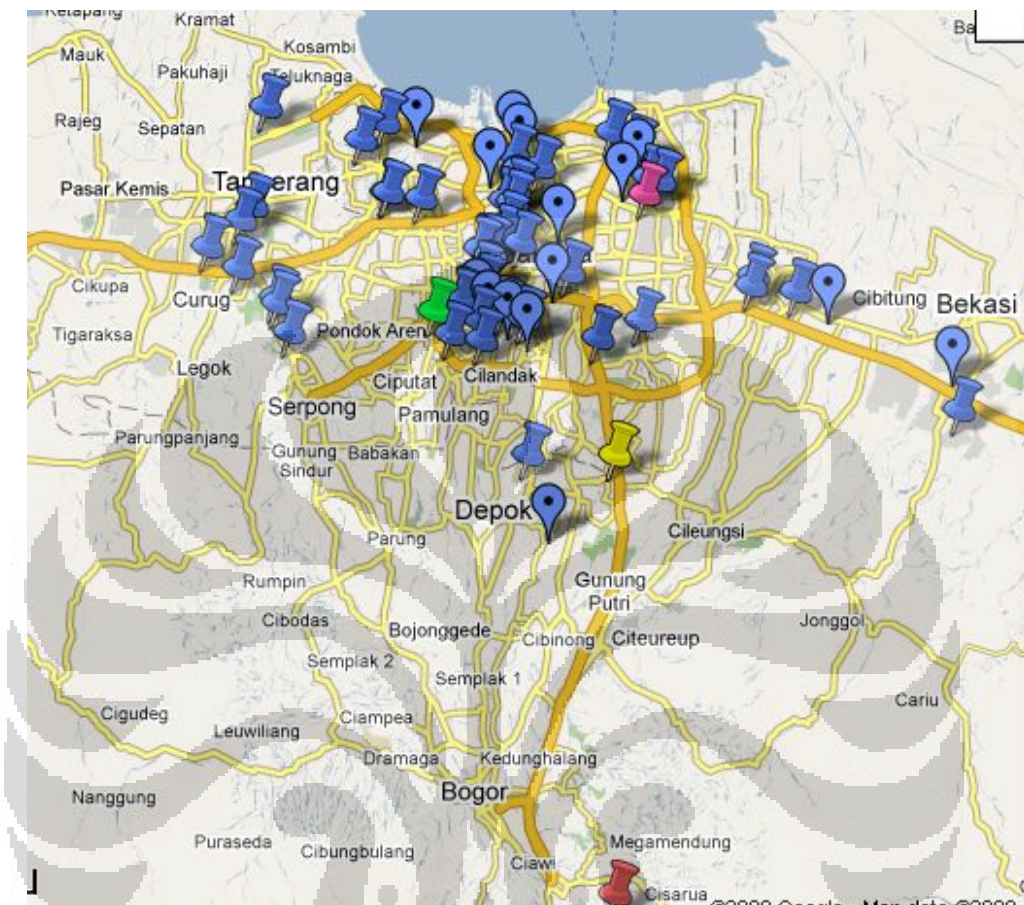
Cstmr	Keterangan	Lokasi	Cstmr	Keterangan	Lokasi
CRA	Carrefour Ambassador	Jakarta	MDMK	Mc. Donald Makasar	Jakarta
CRB	Carrefour Bumi Serpong D	Tangerang	MDR	Mc. Donald Palembang	Jakarta
CRE	Carrefour Cikokol	Tangerang	MDSD	Mc. Donald Samarinda	Jakarta
CRH	Carrefour Mt Haryono	Jakarta	MPB	Matahari Cibubur	Jakarta
CRL	Carrefour Lebak Bulus	Jakarta	MPC	Matahari Cilandak	Jakarta
CRN	Carrefour Tm Plm	Tangerang	MPG	Matahari Serpong	Jakarta

Tabel 3.1. Daftar Nama Pelanggan Perusahaan (sambungan)

Cstmr	Keterangan	Lokasi	Cstmr	Keterangan	Lokasi
DMA	Diamond Artha Gading	Jakarta	MPL	Matahari Klp Gading	Jakarta
DMF	Diamond Fatmawati	Jakarta	MPM	Matahari Hpy Pejaten	Jakarta
DOG	Domino Pizza Kl Gading	Jakarta	MPN	Matahari Daan Mogot	Jakarta
DOK	Domino Pizza	Jakarta	MPO	Matahari Depok	Depok
DOP	Domino Pizza PI	Jakarta	MPQ	Matahari Hpy Glodog	Jakarta
FMG	Farmers market Kpl Gdg	Jakarta	MPR	Matahari Hyper Cikarang	Bekasi
FMS	Farmers market Serpong	Tangerang	MPS	Matahari Fatmawati Karawaci	Jakarta
GNL	Grand Lucky	Jakarta	MPU	Matahari Hypermart- Puri	Tangerang
GRC	Pt Burger King - Cilandak	Jakarta	MPX	Matahari Hyper Jacc	Jakarta
GRG	Pt Burger King - Grand	Jakarta	PRN	Purantara	Tangerang
GRK	Pt Burger King - Senayan	Jakarta	PY	Papaya Fresh G	Jakarta
GRL	Pt Burger King- Klp Gading	Jakarta	PYB	Papaya Bali	Jakarta
GRM	Pt Sari Burger King - Thamrim	Jakarta	PYS	Papaya Surabaya	Jakarta
GRP	Pt Burger King -PI	Jakarta	PZC	Sari Pizza City Walk	Jakarta
GRS	Pt Burger King- Semanggi	Jakarta	PZG	Sari Pizza Grand	Jakarta
HHB	Hari2 Bekasi	Bekasi	PZK	Sari Pizza -Kemang	Jakarta
HHC	HARI2 Cyber Park	Bekasi	PZO	Sari Pizza -Marzano	Jakarta
HHK	Hari2 Kalideres	Jakarta	PZS	Sari Pizza Senayan City	Jakarta
JPG	Pt Jaddi Pastrisindo	Jakarta	RCD	Ranch Market Darmawangsa	Jakarta
MBP	Mos Burger Plaza	Jakarta	RCJ	Ranch Market Pejaten	Jakarta
MD	Mc. Donald Jakarta	Jakarta	RCM	Ranch Market 99	Jakarta
MDB	Mc. Donald Batam	Jakarta	RCP	Ranch Market PI	Jakarta
MDBC	Mc. Donald Batam Formosa	Jakarta	SMP	Pt San Miguel	Depok
MDMK	Mc. Donald Makasar	Jakarta	SRK	Sari Kuring	Bekasi


(Sumber : Perusahaan)

Ilustrasi letak depot, Konsumen dan masing-masing kandidat pendirian DC dapat dilihat pada gambar dibawah ini:



Gambar 3.1. Letak Depot, Kandidat dan Pelanggan

Keterangan:

-  adalah depot,
-  adalah pelanggan,
-  adalah kandidat lokasi Bintaro,
-  adalah kandidat lokasi Cibubur dan
-  adalah kandidat Lokasi Pulo Gadung

3.1.4 Armada Pengiriman

Ketika DC sudah dibuat, armada yang digunakan dalam proses pendistribusian produk ada dua jenis. Jenis pertama merupakan kendaraan berukuran 4000 Kg yang berjumlah dua buah. Kendaraan ini akan digunakan

untuk mendistribusikan produk dari depot ke DC. Kendaraan kedua berukuran 2000 Kg yang jumlahnya 6 buah. Kendaraan inilah yang akan digunakan untuk mengantarkan produk dari DC ke tiap pelanggan. Kedua jenis kendaraan ini dilengkapi dengan pendingin untuk menjaga kesegaran sayuran yang diantarkan.

Berikut ini adalah data-data mengenai armada angkut yang digunakan:

Untuk kendaraan 2000 Kg

- Kapasitas muatan : 75 krat
- Jumlah ban : 4 buah ban
- Bahan Bakar : solar
- Ratio bahan bakar : 1 : 7

Untuk kendaraan 4000 kg

- Kapasitas muatan : 200 krat
- Jumlah ban : 4 buah ban
- Bahan Bakar : solar
- Ratio bahan bakar : 1 : 4

Kapasitas muatan kendaraan dinyatakan dalam satuan krat karena pengiriman produk sayuran dilakukan dengan menggunakan wadah berupa krat. Krat-krat tersebut memiliki ukuran dimensi yang sama. Kapasitas kendaraan tidak dapat dinyatakan dalam satuan berat ataupun satuan volume dikarenakan komoditas yang diangkut (sayuran) memiliki dimensi, volume, dan berat yang berbeda untuk setiap jenisnya.

3.1.5 Biaya Pengiriman Barang

Besar biaya pengiriman barang bergantung pada jenis kendaraan yang digunakannya. Karena itu, besar biaya untuk aktivitas pengantaran barang dari depot-DC akan berbeda dengan aktivitas pengantaran barang dari DC-pelanggan.

Biaya pengiriman diperoleh dengan melakukan perhitungan terhadap jumlah dan harga bahan bakar, biaya ban, biaya pemeliharaan dan biaya supir beserta pendampingnya untuk masing-masing kendaraan. Besarnya diketahui

melalui wawancara dengan pihak sopir perusahaan. Berikut ini adalah rincian biaya pengiriman per kilometer per kendaraan.

Untuk kendaraan 2000 kg

- Biaya bahan bakar

Ratio bahan bakar kendaraan jenis ini adalah 1:7 (1 liter untuk 7 km). Jadi, kebutuhan bahan bakar adalah 0.15 L/km. Harga bahan bakar solar per liter adalah Rp. 4500. Jadi, biaya bahan bakar per km untuk kendaraan ini = $0.15 \times 4500 = \text{Rp. } 675$.

- Biaya pemeliharaan

Perusahaan melakukan service berkala setiap 4 bulan. Biaya satu kali service adalah Rp. 7.000.000. Biaya ini sudah termasuk ganti oli, filter oli, minyak rem, service alat pendingin dll. Rata-rata selama 1 bulan, satu kendaraan akan menempuh jarak 4500 km, maka untuk 4 bulan akan menempuh jarak 18000. Jadi, biaya pemeliharaan/km = $7000000/18000 = 388.88$ /km

- Biaya Ban

Satu ban kendaraan kapasitas 2000 kg maksimal dapat digunakan untuk 20.000 km. Harga 1 ban untuk kendaraan 2000 kg saat ini adalah Rp. 870.000. Jadi, biaya ban/km untuk 1 kendaraan = $4 \times (\text{Rp. } 880.000/20.000) = \text{Rp. } 176/\text{km}$

- Biaya supir = Rp. 75.000/trip

Untuk kendaraan 4000 kg

- Biaya bahan bakar

Ratio bahan bakar kendaraan jenis ini adalah 1:4 (1 liter untuk 4 km). Jadi, kebutuhan bahan bakar adalah 0.25 L/km. Harga bahan bakar solar per liter adalah Rp. 4500. Jadi, biaya bahan bakar per km untuk kendaraan ini = $0.25 \times 4500 = \text{Rp. } 1125$

- Biaya pemeliharaan

Biaya pemeliharaan sama dengan kendaraan 2000 kg yaitu 388.88/km

- Biaya Ban
Satu ban kendaraan kapasitas 4000 kg maksimal dapat digunakan untuk 40.000 km. Harga 1 ban untuk kendaraan 4000 kg saat ini adalah Rp. 1.500.000. Jadi, biaya ban/km untuk 1 kendaraan = $4 \times (\text{Rp. } 1.500.000 / 40.000) = \text{Rp. } 150/\text{km}$
- Biaya supir dan pendamping = Rp. 75.000/km

3.1.6 Permintaan

Seperti yang telah dijelaskan sebelumnya, jumlah permintaan konsumen yang harus dipenuhi berbeda-beda setiap harinya. Konsumen yang melakukan pemesanan juga berbeda-beda untuk setiap harinya. Kemudian, permintaan yang dilakukan konsumen akan berbeda-beda untuk setiap konsumen, baik jumlah permintaannya maupaun jumlah jenis sayurannya.

Satuan yang biasa digunakan untuk jumlah permintaan ini adalah satuan berat (kg). Namun dalam melakukan *loading* barang, pihak perusahaan bisanya menggunakan satuan krat. Untuk itu, perlu dilakukan pengkonversian satuan permintaan ini ke dalam satuan krat. Jumlah kilogram sayuran yang dapat dimuat ke dalam sebuah krat berbeda-beda untuk setiap jenis sayuran tersebut. Data konversi untuk setiap sayuran dapat dilihat pada tabel 3.2.

Tabel 3.2. Kapasitas Krat untuk Setiap Jenis Sayuran

			Kirim	Beli				Kirim	Beli
1	ASG	Asparagus	10	10	31	LBC	Labu Acar	15	15
2	BBY	Bawang bombay	15	15	32	NAS	Nasubhi	10	15
3	BCM	Buncis mini	10	10	33	LBS	Labu Siam	15	15
4	BCT	Buncis Taiwan	15	15	34	LLS	Lolorosa	5	6
5	BIT	Bit	10	15	35	LTD	letuce head	5	7
6	BRC	Brocoli	6	8	36	LTM	letuce head	6	8
7	BSL	Basil	3	3	37	NDV	Endive	5	6
8	BWP	Bawang putih	12	15	38	OKA	Okra	8	10
9	BWR	Bawang merah	12	15	39	PCB	Pakhcoy baby	5	7
10	CBR	Cabe Merah	10	10	40	PCH	Pakhcoy hijau	5	7
11	CLD	Coliander	5	6	41	PCP	Pakhcoy putih	6	8
12	CYS	Caisin	5	7	42	PPH	Paprika hijau	8	10
13	DBW	Bawang Daun besar	7	8	43	PPK	Paprika kuning	8	10
14	DKN	Daikon	15	15	44	PPO	Paprika orange	8	10
15	EDA	Edamame	15	15	45	PPR	Paprika merah	8	10
16	HRN	Horinso	5	6	46	PRL	Petersely	5	6
17	JGA	Jagung Manis	10	15	47	SDT	Seledri stik	5	8
18	JGC	Jagung Acar	10	12	48	SLI	Seledri lokal	7	8
19	JGL	Jagung Kulit	10	15	49	SLR	Selada merah	5	7
20	KBG	Kol Green Coronet	8	10	50	SLT	Selada kriting	5	7
21	KBP	Kol Bulat Putih	8	10	51	SST	Shishito	7	8
22	KGP	Kol Gepeng	8	10	52	SWP	Sawi Putih	10	12
23	KKL	Kembang kol	6	8	53	TMC	Tomat chery	15	15
24	KLB	Kailan Baby	5	6	54	TMT	Tomat Recento	15	15
25	KLN	Kailan Besar	5	7	55	TMW B	Tomat Tw	15	15
26	KLR	Kol Merah	8	10	56	TNC	Timun acar	15	15
27	KPB	Kol Putih Baby	7	10	57	TNJ	Timun jepang	15	15
28	KPI	Kapri	8	10	58	TNM	Timun mini	15	15
29	KPW	Kapri Manis	8	10	59	WRL	Wortel	15	15
30	KRB	Kol Merah Baby	7	10	60	ZKN	Zhukini hijau	15	15

(Sumber : Perusahaan)

Setelah dilakukan konversi satuan kg ke krat untuk setiap jenis sayuran, maka dapat diperoleh jumlah permintaan harian setiap konsumen dalam satuan krat. Berikut ini adalah data jumlah permintaan konsumen selama 31 hari pada bulan Januari 2009.

Tabel 3.3. Permintaan Selama Tanggal 1-8 Januari 2009

Tanggal															
1		2		3		4		5		6		7		8	
Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demar
GNL	6	CRE	15	CRB	11	CRH	7	CRL	10	CRH	9	CRB	4	CRA	11
MPK	24	DMA	9	CRE	10	CRL	15	DOP	1	CRL	14	CRH	5	CRB	7
DMF	10	DMF	8	CRH	14	DMA	9	GRC	2	DMA	6	CRL	28	CRH	13
PZC	2	DOG	1	CRL	30	DMF	10	GRG	2	DMF	8	DMA	7	CRL	21
DOP	2	GRC	4	CRN	4	DOG	1	GRK	6	DOG	2	DMF	7	CRP	11
MBP	8	GRK	13	CRP	13	FMS	4	GRL	3	FMG	16	FMG	22	DMA	7
GRC	3	GRL	4	DMA	13	GRC	1	GRM	1	FMS	4	FMS	12	DMF	9
GRS	2	GRM	7	DMF	10	GRG	6	GRP	4	GRC	4	GNL	7	DOK	1
GRM	5	GRP	2	DOK	1	GRL	4	GRS	2	GRG	2	GRC	1	FMG	8
DMA	8	GRS	2	DOP	1	GRS	3	JPG	13	GRM	4	GRG	2	FMS	14
CRE	16	HHB	6	FMG	32	MBP	5	MBP	7	GRS	1	GRK	6	GRC	1
HHK	19	HHK	6	FMS	6	MPC	10	MD	69	JPG	16	GRL	3	GRG	2
GRL	4	JPG	36	GRC	7	MPL	7	PRN	17	MD	30	GRM	3	GRL	4
CRP	8	MBP	8	GRG	5	MPR	19	PZK	1	MDB	4	GRS	1	GRM	4
CRH	12	MD	31	GRL	5	MPS	8			MDBC	1	JPG	31	GRP	4
MPC	14	MDB	5	GRM	9	SMP	35			MDMK	4	MBP	4	GRS	2
MPN	9	MDMK	3	GRP	3					MDR	4	MD	28	JPG	19
JPG	17	MPC	10	GRS	2					MDS	5	MPK	23	MBP	3
FMG	18	MPK	21	JPG	23					MPB	16	MPL	5	MD	54
FMS	15	PRN	18	MBP	4					MPC	9	MPO	7	MPC	11
RCD	5	PZK	1	MPC	11					MPG	8	MPQ	15	MPK	21

Tabel 3.3. Permintaan Selama Tanggal 1-8 Januari 2009 (sambungan)

Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
				MPN	6					MPL	5	PRN	12	PRN	2
				MPQ	7					MPN	8	PY	3	PZK	2
				MPR	14					MPR	15	PYB	1	RCM	4
				MPS	10					MPS	5	PYS	2		
				RCJ	2					PZC	2	PZK	4		
				RCM	2					PZK	2	PZO	1		
										PZO	1	RCJ	3		
										PZS	4	SRK	65		

(Sumber : Perusahaan)

Tabel 3.4. Permintaan Selama Tanggal 9-16 Januari 2009

Tanggal															
9		10		11		12		13		14		15		16	
Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
CRB	3	CRB	11	CRE	15	CRL	11	CRB	7	CRE	11	CRA	9	CRA	3
CRH	15	CRE	17	CRL	14	DOK	1	CRE	16	CRH	9	CRB	6	CRE	12
CRL	25	CRH	6	DMA	11	DOP	2	CRH	14	CRL	20	CRH	13	CRH	6
CRN	4	CRL	36	DMF	11	GRC	2	CRN	7	CRN	4	CRL	14	CRL	11
CRQ	2	DMA	11	FMG	21	GRG	2	DMA	10	DMA	8	DMA	9	CRP	11
DMA	10	DMF	13	FMS	10	GRK	6	DMF	11	DMF	9	DMF	9	DMA	10
DMF	8	DOK	2	GNL	2	GRM	2	DOG	1	DOG	1	FMG	34	DMF	8
DOP	2	FMG	19	GRC	4	GRP	2	FMG	16	DOP	2	GRC	1	DOK	1

Tabel 3.4. Permintaan Selama Tanggal 9-16 Januari 2009 (sambungan)

Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
GRK	6	GRG	4	GRS	3	MD	52	GRM	4	GRG	2	GRP	2	GRG	5
GRM	7	GRL	5	MBP	6	MPB	7	GRP	2	GRK	5	GRS	1	GRK	10
GRP	4	GRM	5	MD	1	PRN	10	GRS	2	GRL	2	JPG	13	GRL	4
GRS	3	GRP	4	MPC	9	PZK	1	JPG	15	GRM	4	MBP	2	GRM	5
HHB	8	GRS	1	MPK	12			MBP	4	GRP	3	MD	49	GRP	2
HHC	7	JPG	31	MPL	6			MD	14	GRS	1	MPB	6	GRS	2
HHD	9	MBP	5	MPN	7			MDB	4	JPG	22	MPC	11	HHB	19
JPG	33	MD	4	MPO	10			MDBC	1	MBP	3	MPK	22	HHC	25
MBP	4	MPB	12	MPS	3			MDMK	4	MD	22	MPL	6	HHK	20
MD	16	MPC	8					MDR	2	MPC	7	MPN	13	JPG	25
MDB	5	MPK	14					MDS	4	MPH	6	MPO	11	MBP	4
MDBC	2	MPL	7					MPC	7	MPK	17	MPS	6	MD	34
MDMK	3	MPR	24					MPH	38	MPL	6			MDB	5
MDR	3	PZC	2					MPK	17	MPN	7			MDBC	1
MPC	10	PZG	2					MPL	5	MPO	41			MDMK	3
MPH	4	RCD	7					MPR	15	MPS	8			MDR	2
MPK	18	RCJ	3					MPS	8	MPU	30			MPC	7
MPL	6							PZC	1	MPX	5			MPK	23
MPN	6							PZG	1	PRN	9			MPN	11
MPO	6							RCM	6	PYB	1			MPO	68
MPQ	23									PYS	1			MPQ	15
MPS	4									PZG	1			MPS	12
PRN	16									PZO	1			PRN	22

(Sumber : Perusahaan)

Tabel 3.5. Permintaan Selama Tanggal 17-24 Januari 2009

Tanggal															
17		18		19		20		21		22		23		24	
Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
CRC	5	CRA	8	DOK	2	CRL	20	CRA	14	CRE	14	CRA	7	CRB	7
CRE	9	CRB	3	DOP	2	DMA	9	CRB	6	CRL	16	CRB	5	CRE	14
CRH	10	CRE	7	GRC	4	DMF	8	CRE	7	DMA	8	CRE	13	CRH	13
DMA	10	DMA	10	GRK	14	FMG	15	CRN	4	DMF	9	CRL	13	CRL	24
DMF	11	DMF	9	GRM	5	FMS	19	DMA	8	FMG	34	CRQ	2	DMA	18
DOP	1	FMS	6	GRP	3	GNL	4	DMF	7	FMS	16	DMA	11	DMF	14
FMG	13	GRC	5	GRS	2	GRC	6	DOG	1	GNL	2	DMF	8	DOP	2
FMS	8	GRG	5	JPG	20	GRL	4	DOP	2	GRC	3	DOG	2	FMG	31
GNL	4	GRK	10	MBP	5	GRM	4	FMG	23	GRG	3	DOK	2	GNL	3
GRC	6	GRL	5	MD	48	GRP	2	FMS	19	GRM	4	FMG	27	GRC	4
GRG	7	GRM	5	PRN	22	GRS	2	GRC	2	GRP	2	FMS	13	GRG	6
GRK	4	GRP	6	PZG	2	JPG	18	GRG	3	GRS	2	GRG	6	GRL	5
GRM	6	GRS	4	PZK	2	MBP	2	GRK	4	JPG	28	GRK	9	GRM	6
GRP	4	MBP	3			MD	31	GRM	3	MBP	6	GRL	2	GRP	3
GRS	3	MPC	15			MDB	4	GRP	2	MD	42	GRM	6	GRS	1
JPG	28	MPK	16			MDBC	1	GRS	1	MPC	10	GRP	5	JPG	28
MBP	6	MPL	5			MDMK	4	JPG	15	MPK	16	GRS	3	MBP	6
MD	3	MPN	4			MDR	3	MBP	4	MPL	6	HHB	13	MD	6
MPC	9	MPS	5			MPC	11	MD	24	MPN	11	HHC	8	MPC	6
MPK	16	PY	43			MPH	7	MPC	7	MPR	17	HHK	26	MPG	6
MPL	5					MPK	19	MPK	21	MPS	4	JPG	22	MPK	16

Tabel 3.5. Permintaan Selama Tanggal 17-24 Januari 2009 (sambungan)

Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
MPS	20					MPR	16	PRN	15	PZS	1	MDB	5	MPO	8
MPU	10					MPS	15	PY	4	RCM	5	MDBC	2	MPR	9
RCD	2					PRN	2	PYS	2			MDMK	3	MPU	9
RCM	5					PZC	3	RCJ	3			MDR	3	PRN	18
SRK	19					PZK	4	SRK	46			MDSD	7	PZC	1
												MPB	10	PZG	2
												MPC	8	PZK	1
												MPH	5	PZO	1
												MPK	14	RCD	4
												MPL	5	SMP	30
												PRN	17		
												PZO	1		

(Sumber : Perusahaan)

Tabel 3.6. Permintaan Selama Tanggal 25-31 Januari 2009

Tanggal													
25		26		27		28		29		30		31	
Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
CRB	7	CRL	12	DMF	11	CRB	5	CRB	5	CRC	4	CRE	20
CRE	11	GRC	3	DOK	2	CRE	17	CRE	11	CRE	13	CRH	11
CRH	15	GRG	4	DOP	2	CRH	9	CRL	22	CRH	11	CRL	17
CRL	16	GRK	4	GRC	4	CRL	28	DMA	8	CRL	13	DMA	12

Tabel 3.6. Permintaan Selama Tanggal 25-31 Januari 2009 (sambungan)

Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
DMA	18	GRS	2	GRP	2	CRP	8	DOG	1	DMA	10	DOK	1
DMF	12	JPG	23	GRS	2	DMA	7	DOK	2	DMF	8	DOP	2
FMG	33	MBP	7	JPG	21	DMF	8	DOP	1	FMG	24	FMG	20
FMS	7	MD	31	MBP	6	DOG	1	FMG	7	FMS	7	FMS	12
GNL	2	MPN	7	MD	29	FMG	4	GRC	2	GNL	2	GNL	2
GRC	4			MDB	4	FMS	6	GRL	3	GRC	4	GRG	6
GRG	3			MDBC	1	GNL	4	GRP	2	GRG	3	GRL	5
GRL	5			MDMK	4	GRC	4	GRS	2	GRK	10	GRM	6
GRM	4			MPB	12	GRK	5	JPG	18	GRL	3	GRP	2
GRP	4			MPC	10	GRL	4	MBP	11	GRM	3	GRS	1
GRS	2			MPG	53	GRM	4	MD	45	GRP	1	JPG	28
MBP	8			MPL	6	GRP	1	MPB	1	GRS	2	MBP	2
MPC	7			MPS	7	GRS	1	MPC	7	HHB	19	MD	3
MPK	19			PZS	2	JPG	11	MPG	84	HHC	18	MPB	16
MPL	5					MD	25	MPK	25	HHK	23	MPC	9
MPS	7					MPC	11	MPL	6	JPG	22	MPK	18
SRK	45					MPH	3	MPN	2	MBP	2	MPL	5
						MPK	17	MPO	10	MD	31	MPN	7
						MPL	7	MPR	11	MDB	5	MPS	7
						MPN	8	MPS	4	MDBC	1	PZK	2
						MPR	11	PRN	5	MDMK	2	PZO	1
						MPS	6	SRK	35	MDR	3	RCJ	1
						MPU	11			MDS	5	SMP	49

Tabel 3.6. Permintaan Selama Tanggal 25-31 Januari 2009 (sambungan)

Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand	Cstmr	Demand
						PYB	2			MPK	17		
						PYS	2			MPL	5		
						PZC	2			MPO	8		
						PZK	5			MPS	6		
						RCD	2			MPU	10		
						RCJ	4			PRN	8		
						SRK	54			RCM	4		
										SMP	44		

(Sumber : Perusahaan)

3.1.7 Waktu dan Kecepatan

Data waktu yang perlu diketahui adalah *time windows*, dan *service time*. *Time windows* adalah waktu dimana konsumen dapat dilayani atau masih dapat menerima kiriman produk. Melalui wawancara dengan pihak perusahaan agribisnis ini, diketahui bahwasanya setiap konsumen dapat menerima kiriman produk mulai dari pukul 05.00. Kemudian, rata-rata konsumen menginginkan produk sudah harus sampai atau terkirim maksimal jam 10.00. Dari hasil wawancara tersebut, maka ditentukan *time windows* untuk setiap konsumen adalah dari pukul 05.00 – 10.00.

Data selanjutnya adalah data *service time* atau waktu pelayanan di setiap lokasi konsumen. *Service time* ini dapat dibagi menjadi dua bagian, yaitu waktu penurunan barang (*unloading*) dan waktu untuk pengurusan administrasi. Data ini juga diperoleh dari hasil wawancara pihak perusahaan agribisnis yang mengatakan bahwa waktu rata-rata untuk melakukan penurunan barang pesanan adalah sekitar 5 menit dan waktu untuk mengurus masalah administrasi sekitar 5 menit, sehingga total *service time* di setiap lokasi konsumen adalah 10 menit.

Data kecepatan adalah kecepatan rata-rata kendaraan dalam melakukan pengantaran barang. Data ini diasumsikan bernilai sama untuk kedua jenis kendaraan yang digunakan dalam proses distribusi produk. Dari hasil wawancara dengan sopir, disimpulkan bahwa kecepatan rata-rata setiap kendaraan adalah 45 km/jam.

3.1.8 Jarak

Data jarak yang dikumpulkan adalah jarak antara depot dengan masing-masing kandidat, jarak masing-masing kandidat ke tiap pelanggan dan data jarak antar pelanggan. Pengambilan data jarak ini dilakukan dengan menggunakan bantuan peta digital dan GPS. Peta digital yang digunakan merupakan aplikasi yang dikeluarkan oleh Google, yaitu *Googlemaps* (www.maps.google.com). Aplikasi peta digital tersebut memiliki *tool* atau alat bantu yang bernama *distance measurement tool*. *Tool* ini dapat digunakan untuk mengukur jarak antara dua titik yang berada di peta. Jarak yang dihasilkan dari *tools* ini relative akurat. Ini

dibuktikan dengan hasil yang sama bila pengukuran jarak dilakukan dengan menggunakan GPS.

Penggunaan GPS digunakan untuk menghitung jarak lokasi beberapa tempat yang tidak terlihat di peta. Pengukuran jarak ini dilakukan dengan bantuan fasilitas *get directions* yang ada pada GPS.

Pengukuran jarak antara dua titik dilakukan dengan mengikuti alur jalan yang ada di peta sehingga data jarak yang diperoleh dapat mendekati jarak aktual yang ditempuh oleh kendaraan. Pemilihan jalan yang menghubungkan dua titik tertentu dilakukan dengan pertimbangan jarak terdekat dan juga kondisi atau karakteristik jalan (tingkat kemacetan). Kemudian, diasumsikan jarak tempuh dari titik A ke titik B sama dengan jarak tempuh dari titik B ke titik A, sehingga matriks jarak yang dihasilkan akan simetris.

Berikut ini merupakan data jarak dari depot menuju kandidat dan dari kandidat ke tiap pelanggan. Matriks jarak antar pelanggan dapat dilihat pada bagian lampiran.

Tabel 3.7. Jarak Depot ke Kandidat dan Kandidat ke Pelanggan

	Kandidat 1	Kandidat 2	Kandidat 3		Kandidat 1	Kandidat 2	Kandidat 3
PT SM	38.10283	61.8976	66.0866	MDBC	26.38807	10.64132	41.4996
CRA	24.34007	15.0363	21.2203	MDMK	26.38807	10.64132	41.4996
CRB	39.22177	20.577	52.5254	MDR	26.38807	10.64132	41.4996
CRC	50.44647	15.0363	41.4638	MDSD	26.38807	10.64132	41.4996
CRE	11.82147	17.5701	23.7514	MPB	1.58717	22.4114	33.0986
CRH	18.15187	16.3761	20.263	MPC	19.20677	5.38676	35.1616
CRL	21.30097	2.6976	29.9058	MPG	41.93037	26.2627	52.4747
CRN	40.25897	30.8336	30.3926	MPH	52.54504	31.34997	43.56237
CRP	37.87617	42.9602	34.992	MPK	50.60346	34.4633	50.3447
CRQ	26.29167	36.2475	23.4068	MPL	26.72465	24.94888	28.83578
DMA	28.74047	26.3364	8.475	MPN	40.82607	22.6849	35.0522
DMF	21.5979	6.68822	37.55273	MPO	11.14286	18.5181	38.5847
DOG	27.47267	25.3903	6.6531	MPQ	27.27317	37.385	14.27626
DOK	20.13557	9.22745	29.4559	MPR	49.90787	55.2669	52.5557
DOP	21.78397	5.73248	30.7843	MPS	42.93037	26.1574	52.9745
FMG	29.77267	27.6903	4.9131	MPU	36.64697	18.9499	28.3767
FMS	48.84613	32.70597	48.58737	MPX	28.17064	16.22047	29.71687
GNL	25.76487	11.95477	28.2104	PRN	50.87647	22.6849	40.0297

Universitas Indonesia

Tabel 3.7. Jarak Depot ke Kandidat dan Kandidat ke Pelanggan (sambungan)

	Kandidat 1	Kandidat 2	Kandidat 3		Kandidat 1	Kandidat 2	Kandidat 3
GRG	26.99997	15.9677	17.391	PYB	24.91817	9.17142	27.9783
GRK	26.67677	11.58658	29.1262	PYS	24.91817	9.17142	27.9783
GRL	30.33267	28.2503	4.3531	PZC	26.20427	14.2541	27.7505
GRM	28.29687	16.8492	16.4823	PZG	26.99997	15.9677	17.391
GRP	21.78397	5.73248	30.7843	PZK	21.01157	10.10345	4.3531
GRS	24.11227	13.2207	20.3719	PZO	26.67677	11.58658	29.1262
HHB	30.10458	40.06041	27.21971	PZS	26.67237	11.32294	28.7994
HHC	26.49667	32.4775	24.276	RCD	24.37697	9.4912	57.5009
HHK	47.29567	24.2763	32.5003	RCJ	18.67647	10.1204	34.1406
JPG	30.33267	28.2503	4.3531	RCM	33.34427	14.8319	32.8953
MBP	35.27787	11.58658	29.1262	RCP	24.66827	5.94563	40.3128
MD	35.27787	33.1955	0.5921	SMP	18.26643	25.64167	45.70827
MDB	26.38807	10.64132	41.4996	SRK	50.22477	55.3088	47.3406

3.2 Data Untuk Menghitung Biaya Pembelian Tanah Bulanan

3.2.1 Biaya Pembelian Tanah

Biaya pembelian tanah untuk tiap kandidat berbeda antara satu dengan lainnya, walaupun luas area yang akan dibeli sama. Dari hasil wawancara dengan pihak perusahaan dan survey lapangan, harga tanah untuk kandidat Cibubur, Bintaro dan Pulo Gadung adalah Rp. 3.360.000.000, Rp 4.704.000.000 dan Rp. Rp. 4.778.060.000.

3.2.2 Suku Bunga Kredit Investasi

Suku bunga pinjaman bank merupakan besar persentase bunga yang harus dibayarkan perusahaan kepada bank, bila perusahaan meminjam uang kepada bank untuk kepentingan investasinya. Suku bunga ini akan digunakan dalam perhitungan *equivalent monthly cost* untuk biaya pembelian lahan. Berdasarkan data yang diperoleh dari www.kompas.com untuk tanggal 4 Juni 2009, rata-rata nilai suku bunga kredit investasi di tiap bank adalah sebesar 14.05%.

BAB 4

PENGOLAHAN DATA DAN ANALISA

Pada bab ini, semua data yang telah diperoleh akan diolah menjadi data biaya. Pengolahan data sendiri terbagi menjadi dua tahap, yaitu pengolahan data biaya operasional dan pengolahan data biaya pembelian lahan. Selanjutnya akan dilakukan analisa terhadap hasil pengolahan data. Analisa akan meliputi analisa metode perhitungan jarak, analisa program dan analisa biaya.

4.1 Pengolahan Data

Seperti telah dijelaskan dalam pendahuluan, tempat terbaik untuk mendirikan DC adalah tempat dengan biaya terkecil, dalam konteks jumlah biaya operasional dan biaya investasi lahan. Perhitungan biaya ini akan dilakukan untuk bulan Januari 2009. Biaya investasi lahan dapat dengan mudah diketahui, sedangkan untuk mengetahui biaya operasional, terlebih dahulu harus diketahui kemungkinan jarak total dari tiap kandidat untuk melayani pelanggan (jarak depot-kandidat dan jarak kandidat-depot).

Karena itu, pengolahan data untuk menyelesaikan penelitian ini dilakukan dalam dua tahap. Pertama, pengolahan data permintaan selama bulan Januari 2009 untuk mengetahui jarak tiap kandidat-konsumen (Jarak depot-kandidat sudah diketahui dengan bantuan *Googlemaps*). Hasil perhitungan jarak selanjutnya akan digunakan untuk menghitung biaya operasional. Tahap kedua ditujukan untuk menentukan nilai ekuivalen bulanan untuk biaya pembelian lahan di masing-masing kandidat.

4.1.1 Perhitungan Jarak Antara Tiap Kandidat ke Konsumen

Perhitungan jarak dari tiap kandidat ke konsumen dapat dilakukan dengan pendekatan VRP algoritma DE. Penyelesaian algoritma ini sendiri dilakukan dengan menggunakan bahasa pemrograman MATLAB R2008a.

4.1.1.1 Langkah Pembuatan Program MATLAB

Program MATLAB dibuat sesuai dengan urutan proses dalam algoritma *Differential Evolution*. Output yang diharapkan dari program ini adalah berupa jumlah rute, urutan pengantaran dalam tiap rute dan jarak total. Langkah pengerjaannya adalah sebagai berikut:

1. Penetapan Parameter Kontrol

Ada tiga parameter kontrol yang perlu ditetapkan nilainya, yaitu ukuran populasi, parameter kontrol mutasi dan parameter kontrol pindah silang. Parameter ukuran populasi merupakan parameter untuk menentukan jumlah solusi awal. Pada masalah ini, ukuran populasi ditetapkan sebesar 2 kali jumlah konsumen. Hal ini dilakukan untuk mempercepat waktu program dalam menentukan populasi awal.

Nilai parameter mutasi (F) dan pindah silang (CR) dalam proses pembuatan program, didapatkan dari jurnal utama yang digunakan untuk penelitian ini, yaitu Erbao, Mingyong dan Kai (2008). Nilai F dan CR sendiri masing-masing sebesar 0.4 dan 0.5. Setelah program berhasil dibuat, dilakukan percobaan untuk memastikan kedua nilai ini memang nilai yang paling baik.

Percobaan dilakukan untuk nilai efektif F, yaitu antara 0.4-1 dan nilai efektif CR, yaitu 0-1. Percobaan ini dilakukan untuk data permintaan pada tanggal 26 januari 2009 karena pada tanggal ini jumlah pelanggan sangat sedikit sehingga dapat menghemat waktu percobaan. Bila hasil percobaan menunjukkan bahwa nilai F=0.4 dan CR=0.5 memang hasil terbaik, maka proses pengolahan data akan terus menggunakan parameter ini. Namun, bila ternyata terdapat nilai lain yang lebih baik, maka nilai hasil percobaan tersebut yang akan digunakan untuk mengolah data. Hasil percobaan ini dapat dilihat pada tabel 4.1.

Dari tabel 4.1 terlihat bahwa parameter F=0.4 dan CR=0.5 memberikan hasil yang paling baik, walaupun tidak terlalu signifikan. Kombinasi dari dua parameter ini sebenarnya menghasilkan *computation time* yang cukup lama, jika dibandingkan dengan penggunaan CR yang lebih besar. Namun, karena penelitian ini tidak memiliki batasan waktu, maka yang lebih dipertimbangkan

adalah hasil yang diperoleh. Jadi, parameter $F=0.4$ dan $CR=0.5$ akan digunakan seterusnya dalam pengolahan data.

Selain 3 parameter kontrol diatas, pada tahap ini juga ditentukan kriteria terminasi program. Kriteria terminasi dapat ditentukan berdasarkan jumlah iterasi maksimum ataupun waktu proses. Pada kasus ini, kriteria yang digunakan adalah jumlah iterasi maksimum. Program akan terminasi dengan sendirinya bila telah melakukan 1000 iterasi. Asumsinya adalah dengan melakukan iterasi sebanyak 1000 kali, program akan dapat mencari solusi yang benar-benar paling baik. Namun, pemilihan kriteria terminasi seperti ini tentu saja membuat waktu *running* program menjadi lebih lama.

Tabel 4.1. Hasil Percobaan Penentuan Parameter

Parameter	Computation Time	Hasil	Parameter	Computat	Hasil		
F=0.4	CR=0.5	11.0333	90.4245	F=0.8	CR=0.5	14.52185	98.4782
	CR=0.6	11.9822	95.9993		CR=0.6	12.5778	97.0923
	CR=0.7	8.998228	97.9665		CR=0.7	7.039893	98.6702
	CR=0.8	7.245561	97.9993		CR=0.8	5.446932	103.7799
	CR=0.9	5.963244	98.4241		CR=0.9	4.136981	114.988
	CR=1	4.266739	111.4854		CR=1	4.48679	119.6309
F=0.5	CR=0.5	12.194342	95.4364	F=0.9	CR=0.5	11.1342	95.4364
	CR=0.6	10.6338	99.7116		CR=0.6	10.6391	99.7116
	CR=0.7	9.041027	104.3389		CR=0.7	9.0117	104.3389
	CR=0.8	7.836255	126.6937		CR=0.8	7.8893	126.6937
	CR=0.9	6.456608	133.1137		CR=0.9	6.476608	133.1137
	CR=1	4.97551	136.2691		CR=1	4.67551	136.2691
F=0.6	CR=0.5	12.121645	107.2653	F=1	CR=0.5	12.15982	98.4364
	CR=0.6	11.5778	103.7823		CR=0.6	12.66009	109.1316
	CR=0.7	9.039893	98.3886		CR=0.7	9.3117	110.3489
	CR=0.8	7.446932	110.3398		CR=0.8	6.22567	126.6937
	CR=0.9	5.146981	112.8854		CR=0.9	6.476608	129.1407
	CR=1	4.48679	115.8894		CR=1	4.67551	132.7691
F=0.7	CR=0.5	12.128874	108.8282				
	CR=0.6	11.00987	97.0923				
	CR=0.7	9.893076	113.6702				
	CR=0.8	7.644002	99.8799				
	CR=0.9	5.12982	114.5427				
	CR=1	4.411828	119.6309				

2. Menentukan Populasi Awal

Langkah selanjutnya adalah membentuk populasi awal. Populasi awal merupakan sebuah matriks yang berisikan sejumlah individu awal. Dalam kasus VRP, setiap individu awal akan merepresentasikan jumlah rute untuk melayani seluruh konsumen dan urutan pengantaran di setiap rute tersebut.

Populasi awal dapat dicari secara acak atau dengan menggunakan pendekatan heuristik. Untuk menyederhanakan program, maka populasi awal pada masalah ini didapatkan secara acak. Setiap individu awal dicari dengan rumus:

$$\text{Individu awal} = \text{batas_bawah} + (\text{batas_atas} - \text{batas_bawah}) \times \text{bil acak} \quad (4.1)$$

Pada proses ini bilangan acak yang digunakan besarnya antara 0-1. Batas bawah dan batas atas masing-masing ditetapkan sebesar -1 dan 1, sesuai dengan yang ditetapkan pada jurnal Tasgetiren et al (2004).

Jumlah individu awal tergantung pada parameter kontrol ukuran populasi. Pada kasus ini, parameter ukuran populasi = 2 x jumlah pelanggan. Artinya, bila pada tanggal 1 Januari 2009 terdapat 17 pelanggan, maka jumlah individu awalnya adalah 34. Karena populasi terdiri dari 34 individu dan tiap individu terdiri dari 17 dimensi, maka populasi awal pada tanggal tersebut merupakan matriks berukuran 34 x 17.

Populasi awal masih berupa bilangan acak yang besarnya antara -1 sampai 1, sehingga populasi awal ini belum merepresentasikan urutan pengantaran ke tiap pelanggan. Dengan demikian, perlu dilakukan proses konversi sehingga urutan pengantaran menjadi jelas. Misalnya, suatu individu awal memiliki dimensi sebagai berikut:

0.0145	0.0014	-0.0016	-0.0029	-0.1135	0.0082	-0.0285	0.0048	0.1143
8	5	4	3	1	7	2	6	9

Pada individu awal diatas, angka -0.01135 merupakan bilangan terkecil, maka dia akan dikonversikan menjadi bilangan 1. Angka -0.0285 merupakan bilangan kedua terkecil, maka dikonversikan menjadi bilangan 2. Angka-angka sisanya juga mengalami perlakuan sama, sehingga setelah dilakukan konversi, hasilnya adalah sebagai berikut:

Maksud dari angka-angka ini adalah pengantaran akan dilakukan untuk pelanggan nomor 8 lalu pelanggan nomor 5 lalu pelanggan nomor 4 dan seterusnya.

3. Menentukan Fungsi objektif

Fungsi objektif yang digunakan adalah fungsi objektif untuk permasalahan VRP dengan *time windows*, yaitu meminimumkan jarak yang harus ditempuh untuk melayani seluruh konsumen melalui penetapan sejumlah rute dan urutan pengantaran dalam rute tersebut tanpa melanggar batasan-batasan yang telah ditetapkan.

Batasan-batasan ini adalah :

- pada tiap rute, besar muatan yang diangkut oleh kendaraan tidak boleh melebihi kapasitas kendaraan tersebut
- konsumen yang dilayani dalam sebuah rute dapat hanya merupakan pengiriman atau pengambilan, atau mungkin keduanya
- jumlah rute tidak boleh lebih banyak daripada jumlah kendaraan
- konsumen mungkin hanya dapat dilayani dalam rentang waktu tertentu (*time windows*)
- waktu kendaraan untuk mengirim barang sampai kembali ke depot tidak boleh melebihi waktu operasional depot

Model matematika untuk fungsi objektif ini telah dijelaskan pada bab II.

4. Evaluasi

Setiap individu awal selanjutnya akan dievaluasi dengan menggunakan fungsi objektif diatas. Tujuannya adalah mengetahui individu mana yang memberikan jarak terkecil tanpa melanggar batasan yang telah ditetapkan. Individu dengan jarak terkecil selanjutnya akan disebut dengan individu target.

5. Proses Mutasi

Proses mutasi bertujuan untuk membuat individu baru yang disebut individu mutan. Secara teori, individu ini merupakan individu awal yang mengalami perubahan nilai pada dimensinya. Karena itu, untuk membuatnya, kita harus mengambil secara *random* 3 individu dari populasi awal. Misalkan individu tersebut adalah individu a, b dan c. Maka rumus untuk membentuk individu mutan adalah:

$$\text{Individu mutan} = a + (b-c) * 0.4 \quad (4.2)$$

Angka 0.4 merupakan parameter mutasi (F) yang telah ditentukan pada langkah nomor 1.

6. Pindah Silang

Proses pindah silang bertujuan untuk membentuk individu baru yang disebut individu *trial*. Parameter atau nilai dimensi individu *trial* ini sebagian berasal dari parameter individu target dan sebagian lagi berasal dari individu mutan, dengan mempertimbangkan operator pindah silang (CR) dan bilangan acak.

Jika bilangan acak r (antara 0 sampai 1) yang dihasilkan lebih kecil atau sama nilainya dengan CR maka yang berpeluang menjadi nilai dimensi ke- i individu *trial* adalah nilai dimensi ke- i individu mutan. Bila hal sebaliknya yang terjadi, maka nilai dimensi ke- i individu *trial* adalah nilai dimensi ke- i individu awal.

7. Evaluasi

Setiap individu pada populasi *trial* selanjutnya juga akan dievaluasi seperti yang dilakukan pada individu pada populasi awal. Individu dengan nilai jarak paling kecil selanjutnya akan masuk ke dalam proses seleksi.

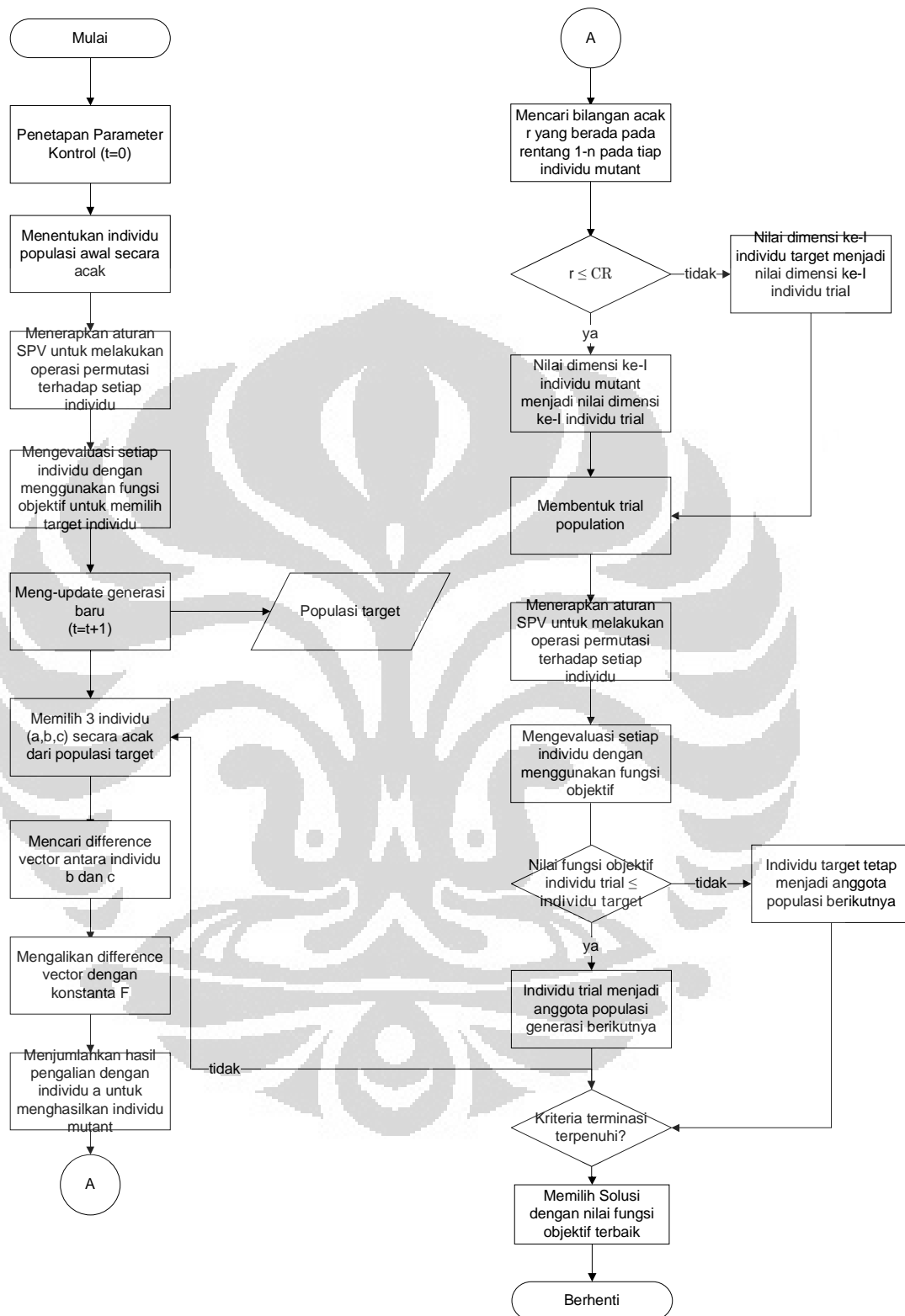
8. Proses Penyeleksian

Proses ini untuk menentukan individu manakah yang layak menjadi anggota generasi berikutnya. Proses penyeleksian dilakukan dengan membandingkan nilai fungsi objektif individu target dengan individu *trial*. Dalam kasus ini, individu dengan jarak terkecil yang akan terpilih.

9. Terminasi

Proses 2-7 merupakan proses untuk 1 kali iterasi. Proses ini akan berulang terus sampai dengan jumlah iterasi mencapai angka 1000. Program selanjutnya akan memilih individu dengan jarak terkecil dari 1000 iterasi yang telah dilakukan.

Source code pembuatan program dapat dilihat pada bagian lampiran, sedangkan langkah pembuatan program secara jelas dapat dilihat pada *flowchart* pada gambar 4.1.



Gambar 4.1. Flowchart Pengerjaan

4.1.1.2 Verifikasi dan Validasi Program

Sebelum program digunakan untuk menganalisa kemungkinan total jarak yang harus ditempuh dari masing-masing kandidat dalam mendistribusikan produknya, terlebih dahulu harus dilakukan verifikasi dan validasi program. Tahap verifikasi merupakan tahap melihat kesesuaian antara model program yang didapat dengan konsep model yang kita buat/inginkan. Bila program dapat berjalan sesuai dengan keinginan, maka program tersebut telah terverifikasi. Salah satu indikator yang dapat dilihat untuk membuktikannya adalah ketika dilakukan perubahan pada nilai-nilai parameter, output yang dihasilkan juga akan berubah.

Setelah program terverifikasi, maka selanjutnya perlu dilakukan validasi dengan tujuan untuk dapat memastikan bahwa program menghasilkan output yang benar. Indikatornya adalah ketika program dihadapkan pada suatu masalah, hasil perhitungannya bernilai sama dengan perhitungan manual. Validasi dapat dilakukan dengan menggunakan data *dummy*. Data *dummy* ini memiliki 5 pelanggan. Matriks jarak antar pelanggan dan antara DC dengan pelanggan dapat dilihat pada tabel berikut ini:

Tabel 4.2. Matriks Jarak Data Dummy

	0	1	2	3	4	5
0	0	4	6	9	7	2
1	4	0	2	5	3	6
2	6	2	0	7	5	4
3	9	5	7	0	2	5
4	7	3	5	2	0	10
5	2	6	4	5	10	0

Tabel 4.3. Permintaan Tiap Pelanggan Data Dummy

Konsumen	Jumlah Pengiriman (krat)
1	20
2	22
3	45
4	28
5	45

Kapasitas kendaraan yang digunakan adalah 75 krat. Data *service time*, kecepatan kendaraan dan *time windows* sama untuk kasus Perusahaan ini. Kemudian konfigurasi parameter yang digunakan untuk verifikasi program ini adalah sebagai berikut.

Tabel 4.4. Parameter yang Digunakan Dalam Proses Validasi

NP	1*jumlah Pelanggan
F	0.4
CR	0.5
Kriteria Terminasi	1000 iterasi

Dari data *dummy* tadi, dilakukan *run* program dengan menggunakan konfigurasi diatas. Hasil *run* program menunjukkan solusi rute urutan perjalanan **0-1-4-2-0-5-0-3-0** dan jarak total perjalanan adalah 22 km. Selanjutnya, perlu dilakukan perhitungan manual untuk membuktikan hasil ini benar. Iterasi perhitungan manual yang dilakukan adalah sebagai berikut..

1. Pembentukan Solusi Awal

Solusi awal dibentuk dengan menggunakan sistem *random*. Jumlah rute acak yang harus dibentuk ada 5 buah (sesuai dengan parameter NP). Kelima solusi awal ini adalah 0-5-1-4-2-3, 0-1-4-5-2-3, 0-2-3-5-4-1, 0-4-1-3-5-2 dan 0-3-2-4-1-5.

Selanjutnya, solusi awal ini dipecah berdasarkan kapasitas maksimum kendaraan dan batasan waktu. Diperoleh urutan rute 0-5-1-0-4-2-0-3-0, 0-1-4-0-5-2-0-3-0, 0-2-3-0-5-0-4-1-0, 0-4-1-0-3-0-5-2 dan 0-3-2-0-4-1-0-5-0. Jarak masing-masing solusi ini adalah 48, 26, 40, 44 dan 38. Dari data ini terlihat bahwa solusi awal terbaik adalah 0-1-4-0-5-2-0-3-0.

2. Mutasi

Untuk proses mutasi, dilakukan pengambilan 2 solusi secara *random* sebanyak 5 kali. Pada setiap dilakukan satu kali pengambilan, individu mutan akan dibentuk dengan rumus:

$$\text{Individu mutan} = a + (b-c) * 0.4$$

Individu b dan c adalah solusi yang diambil acak, sedangkan individu a merupakan solusi awal terbaik (individu target). Misalnya, bila solusi yang diambil adalah solusi 1 dan 3, maka hasil dari $a+(b-c)*0.4$ adalah 2.2-3.2-4.6-

1.2-3.8. Nilai ini kemudian dibulatkan menjadi 2-3-5-1-4. Urutan inilah yang disebut dengan individu mutan. Dengan cara yang sama diperoleh individu mutan yang lain, yaitu 1-4-5-3-2, 2-1-3-5-4, 3-4-2-5-1 dan 4-2-3-1-5.

3. Pindah Silang

Nilai individu target dengan individu mutan akan dipertukarkan dengan menggunakan perbandingan antara nilai CR dan sebuah bilangan acak sehingga menghasilkan sebuah solusi baru (solusi *trial*). Pencarian bilangan acak digunakan dengan bantuan MS. Excel.

Hasil pindah silang ini adalah:

- individu target dan mutan 1, hasilnya 1-3-5-2-4
- individu target dan mutan 2, hasilnya 1-4-5-2-3
- individu target dan mutan 3, hasilnya 3-4-5-2-1
- individu target dan mutan 4, hasilnya 1-4-2-5-3
- individu target dan mutan 5, hasilnya 4-2-5-1-3

Selanjutnya, solusi *trial* ini dipecah berdasarkan kapasitas maksimum kendaraan dan batasan waktu. Diperoleh urutan rute 0-1-3-0-5-2-0-4-0, 0-1-4-0-5-2-0-3-0, 0-3-4-0-5-0-2-1-0, 0-1-4-2-0-5-0-3-0 dan 0-4-2-0-5-1-0-3-0. Jarak masing-masing solusi ini adalah 44, 26, 34, 22 dan 48 km.

4. Seleksi

Selanjutnya, jarak tiap solusi *trial* dibandingkan dengan jarak individu target. Ternyata, solusi *trial* nomor 4 memiliki jarak yang lebih baik daripada individu target. Solusi ini memiliki urutan pengantaran **0-1-4-2-0-5-0-3-0**. Jadi, hasil perhitungan manual menghasilkan urutan pengantaran **0-1-4-2-0-5-0-3-0** dengan jarak 22 km.

Dapat dilihat bahwa hasil perhitungan manual diatas sama dengan hasil dari *run* program. Dengan demikian program telah tervalidasi.

4.1.1.3 Input Data

Pengolahan data akan dilakukan terhadap data permintaan selama bulan Januari 2009. Hal ini dilakukan karena tren jumlah permintaan berulang setiap bulannya, meskipun jumlah konsumen dan permintaannya tidak tepat sama untuk setiap bulannya.

Program akan melakukan perhitungan data secara harian. Artinya, input data untuk tanggal 1 akan berbeda dengan input data untuk tanggal 2. Jadi, untuk mendapatkan data jarak satu kandidat, input ke program dilakukan sebanyak 31 kali (jumlah hari dalam bulan Januari 2009). Selanjutnya, data jarak harian yang dihasilkan oleh program akan diakumulasikan sehingga dihasilkan data jarak selama satu bulan. Akumulasi data jarak inilah yang selanjutnya akan digunakan untuk menentukan biaya operasional tiap kandidat.

Untuk melakukan perhitungan data jarak ini, data yang diperlukan adalah sebagai berikut.

- Data jarak harian tiap kandidat dengan titik-titik pelanggan dan juga jarak antarpelanggan. Untuk memudahkan proses input data, data jarak ini terlebih dahulu dimasukkan ke dalam *spreadsheet*. Contohnya dapat dilihat pada tabel 4.5, yang merupakan contoh bentuk spreadsheet untuk tanggal 5 Januari 2009.
- Data permintaan harian tiap konsumen dan kapasitas kendaraan yang digunakan untuk melakukan pengantaran. Data tersebut digunakan untuk menghitung permintaan yang harus dipenuhi pada tiap rute dan membandingkannya dengan batasan kapasitas kendaraan.
- Data mengenai kecepatan rata-rata kendaraan waktu *unloading* (*service time*) dan batasan waktu (*time windows*). Data tersebut digunakan untuk menghitung waktu tempuh perjalanan tiap rute dan membandingkannya dengan batasan waktu yang ada. Untuk kecepatan rata-rata kendaraan digunakan kecepatan 45 km/ jam. Hal tersebut didasarkan dari hasil wawancara dengan para sopir kendaraan dengan mempertimbangkan karakteristik jalan dan tingkat kemacetan yang terjadi.

Tabel 4.5. Contoh Format Spreadsheet Data Jarak Untuk Input Data

	Kandidat 1	CRL	DOP	GRC	GRG	GRK	GRL	GRM	GRP	GRS	JPG	MBP	MD	PRN	PZK
Kandidat 1	0.000	21.301	21.784	19.207	27.000	26.677	30.333	28.297	21.784	24.112	30.333	35.278	35.278	50.876	21.012
CRL	21.301	0.000	3.035	2.689	13.270	8.889	25.553	14.152	3.035	10.523	25.553	8.889	30.498	19.987	7.406
DOP	21.784	3.035	0.000	5.466	10.700	5.903	26.431	11.622	0.000	7.626	26.431	5.903	31.376	28.924	8.981
GRC	19.207	2.689	5.466	0.000	12.730	8.142	30.809	13.704	5.466	9.864	30.809	8.142	35.754	36.855	5.898
GRG	27.000	13.270	10.700	12.730	0.000	5.813	13.038	1.133	10.700	3.452	13.038	5.813	17.983	29.155	12.213
GRK	26.677	8.889	5.903	8.142	5.813	0.000	24.773	7.593	5.903	2.576	24.773	0.000	29.718	28.960	9.505
GRL	30.333	25.553	26.431	30.809	13.038	24.773	0.000	12.129	26.431	16.019	0.000	24.773	4.945	35.677	25.979
GRM	28.297	14.152	11.622	13.704	1.133	7.593	12.129	0.000	11.622	5.073	12.129	7.593	17.074	25.545	13.808
GRP	21.784	3.035	0.000	5.466	10.700	5.903	26.431	11.622	0.000	33.199	26.431	5.903	31.376	28.924	8.981
GRS	24.112	10.523	7.626	9.864	3.452	2.576	16.019	5.073	33.199	0.000	16.019	2.576	20.964	29.273	10.820
JPG	30.333	25.553	26.431	30.809	13.038	24.773	0.000	12.129	26.431	16.019	0.000	2.169	4.945	35.677	27.978
MBP	35.278	8.889	5.903	8.142	5.813	0.000	24.773	7.593	5.903	2.576	2.169	0.000	29.718	28.960	9.505
MD	35.278	30.498	31.376	35.754	17.983	29.718	4.945	17.074	31.376	20.964	4.945	29.718	0.000	40.622	4.945
PRN	50.876	19.987	28.924	36.855	29.155	28.960	35.677	25.545	28.924	29.273	35.677	28.960	40.622	0.000	12.358
PZK	21.012	7.406	8.981	5.898	12.213	9.505	25.979	13.808	8.981	10.820	27.978	9.505	4.945	12.358	0.000

4.1.2 Perhitungan Data Jarak antara Depot ke Tiap Kandidat

Untuk mengetahui total jarak antara depot ke tiap kandidat selama 1 bulan, kita perlu memahami rencana kegiatan pengantaran barang dari depot ke DC, ketika DC telah selesai dibangun.

Bila DC sudah beroperasi, setiap hari akan ada 2 kendaraan berukuran 5000 Kg yang mengantarkan barang dari depot ke DC. Kedua kendaraan ini akan kembali lagi ke depot setelah pengantaran barang selesai. Artinya, dalam satu hari, jarak yang harus ditempuh untuk mengantarkan barang dari depot ke DC adalah:

$$\text{Jarak 1 hari} = 2 * \text{Jarak depot-DC} \quad (4.3)$$

Jadi, jarak depot-DC selama 1 bulan akan diperoleh dengan:

$$\text{Jarak 1 bulan} = 31 * 2 * \text{jarak depot-DC} \quad (4.4)$$

Jarak depot-DC diperoleh dengan menggunakan bantuan *Googlemaps*, seperti yang ditunjukkan pada tabel 3.4. Dengan pendekatan diatas, kita dapat menghitung jarak antara depot ke tiap kandidat selama bulan Januari 2009. Jarak ini kemudian akan diubah menjadi data biaya juga.

4.1.3 Perhitungan Biaya

Data jarak antara tiap kandidat-pelanggan yang dihasilkan oleh program dan data jarak depot-kandidat akan dikonversikan menjadi biaya operasional dengan mempertimbangkan berbagai faktor biaya yang ada, seperti bahan bakar, ban, perawatan kendaraan, dan biaya supir.

Proses konversi data jarak depot-kandidat dan kandidat-pelanggan menjadi data biaya ini akan dilakukan secara terpisah. Hal ini dikarenakan kendaraan yang digunakan dalam pengantaran barang depot-kandidat berbeda dengan yang digunakan dalam pengantaran kandidat-pelanggan, sehingga faktor biaya antara kedua kendaraan ini pasti berbeda pula.

Aktivitas pengantaran barang dari kandidat ke pelanggan memiliki biaya sebesar Rp 1239.88/km, sedangkan untuk aktivitas pengantaran barang dari depot ke kandidat, biaya yang digunakan adalah sebesar Rp 1663.88.88/km. Biaya supir dan pendamping untuk dua aktivitas ini sama, yaitu Rp.75.000/trip. Hasil

perhitungan biaya ini akan menghasilkan estimasi total biaya operasional per bulan untuk tiap kandidat.

Karena biaya operasional dihitung dalam jangka waktu 1 bulan, maka biaya pembelian tanah juga harus dikonversikan menjadi biaya bulanan untuk mengetahui berapa biaya *equivalent* bulanan untuk pembelian tanah ini. Langkah pertama untuk mengerjakannya adalah dengan mencari *Equivalent Uniform Annual Cost* (EUAC) dari biaya pembelian tanah ini.

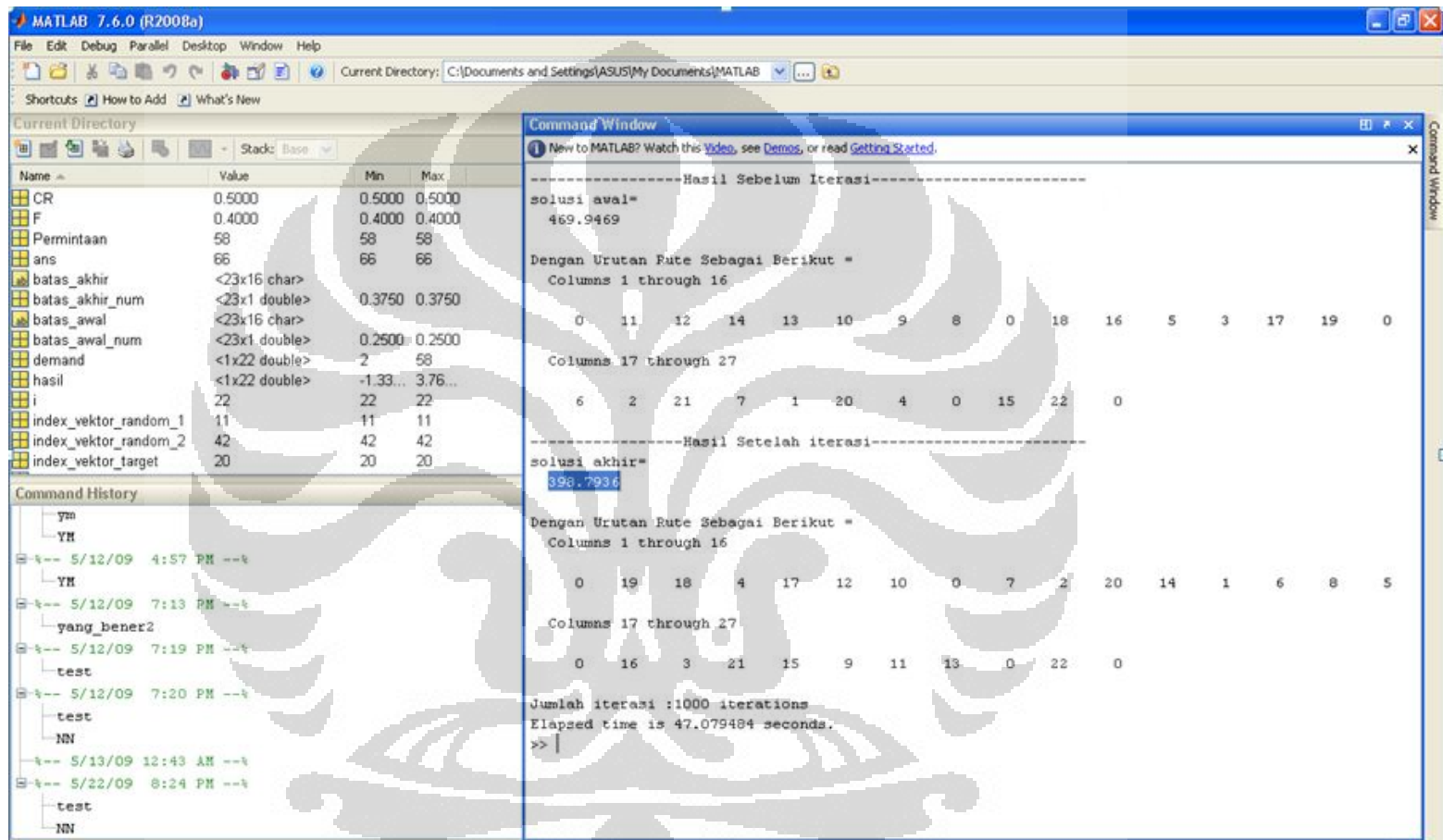
Setelah nilai EUAC diketahui, langkah selanjutnya adalah menjadikan nilai EUAC ini sebagai biaya bulanan. Caranya dengan membagi nilai EUAC ini dengan 12. Hasil dari pembagian ini ditambah dengan total biaya operasional akan menghasilkan biaya total untuk setiap kandidat. Kandidat dengan total biaya paling kecil merupakan kandidat terbaik untuk mendirikan DC.

4.2 Hasil Pengolahan Data

4.2.1 Jarak Masing-masing Kandidat untuk Melayani Pelanggan

Output dari program MATLAB yang telah dibuat adalah berupa rute-rute pengiriman barang dari masing-masing kandidat lokasi ke pelanggan dan juga total jarak yang harus ditempuhnya. Karena data yang diinputkan ke dalam program ini merupakan data harian, maka hasilnya pun masih bersifat harian. Data jarak selama satu bulan didapatkan dengan menjumlahkan semua data harian ini. Contoh output dari program MATLAB dapat dilihat pada gambar 4.2. Hasil pengolahan data secara lengkap dapat dilihat pada bagian lampiran.

Secara ringkas, hasil perhitungan jarak yang dilakukan program dapat dilihat pada tabel 4.6.



Gambar 4.2. Contoh Output Program

Tabel 4.6. *Output* Jarak Untuk Masing-masing Kandidat

Hari	Total Km		
	Kandidat 1	Kandidat 2	Kandidat 3
1	398.7936	338.7069	284.7745
2	384.5686	342.6715	363.6717
3	548.1919	468.6408	521.1407
4	313.9666	279.6036	307.2157
5	212.4972	165.5674	141.4205
6	670.3309	638.4167	677.625
7	627.5338	494.8416	549.1536
8	507.5125	378.3941	397.1186
9	786.9321	641.1272	756.9761
10	457.0729	418.8604	480.2712
11	380.8062	312.2274	388.6604
12	226.4908	205.3736	160.5737
13	616.1744	568.72	664.9873
14	649.2564	552.0209	571.3685
15	343.7457	351.5307	341.6781
16	673.9978	605.2936	617.3976
17	529.7907	497.8858	500.2175
18	329.3318	315.6351	338.8242
19	200.0289	156.6578	150.7233
20	606.3443	456.4207	516.6095
21	518.9955	411.225	472.992
22	454.2825	380.5258	380.0509
23	752.8962	674.9433	747.1861
24	614.7643	563.0772	575.035
25	339.3741	354.6988	430.2746
26	185.5285	128.2234	101.0868
27	374.6132	298.1478	298.1889
28	707.2577	596.1907	673.3627
29	595.4024	525.7583	608.4139
30	769.7099	711.0534	742.4014
31	512.944	461.8936	481.9812
Σ	15289.135	13294.333	14241.381

Keterangan:

Kandidat 1 = Cibubur

Kandidat 2 = Bintaro

Kandidat 3 = Pulo Gadung

4.2.2 Jarak Depot-Kandidat

Data jarak dari depot ke tiap kandidat diperoleh dengan bantuan *Googlemaps*. Dari bab 3, kita tahu bahwa jarak dari depot ke kandidat 1, 2 dan 3 masing-masing adalah 38.10283 Km, 61.8976 Km dan 66.0866 Km. Dengan pendekatan perhitungan yang disebutkan pada sub bab 4.1.2, diperoleh data seperti yang ditampilkan pada tabel 4.7 berikut ini:

Tabel 4.7. Jarak Dari Depot ke Tiap Kandidat

	Jarak dari depot ke-	Jarak tempuh 1 hari	Jarak Tempuh 1 Bulan
Kandidat 1	38.10283	76.20566	2362.37546
Kandidat 2	61.8976	123.7952	3837.6512
Kandidat 3	66.0866	132.1732	4097.3692

Berdasarkan data ini, kandidat 1 (Cibubur) merupakan tempat paling baik karena akan paling dekat dengan depot.

4.2.3 Konversi Biaya

4.2.3.1 Biaya Operasional

Dari tabel 4.6 dan 4.7, kita mendapatkan data total jarak selama 1 bulan untuk kegiatan pengantaran barang dari tiap kandidat ke pelanggan dan dari depot ke tiap kandidat. Kedua jenis data ini selanjutnya akan dikonversikan menjadi data biaya operasional. Karena biaya operasional diasumsikan linear dengan jarak, maka perhitungannya cukup dengan mengalikan data jarak dengan berbagai faktor biaya yang ada. Faktor biaya yang digunakan disini meliputi biaya bahan bakar, biaya ban, biaya supir dan perawatan.

Faktor biaya yang digunakan dalam perhitungan bergantung pada kendaraan yang digunakan. Untuk aktivitas kandidat-pelanggan, kendaraan yang digunakan adalah kendaraan 2000 kg, sedangkan untuk aktivitas depot-kandidat menggunakan kendaraan 4000 kg.

Kendaraan 2000 kg memiliki total biaya Rp 1239.88/km dan biaya supir Rp 75000/trip. Jarak kandidat 1, 2 dan 3 untuk aktivitas pengantaran barang menuju konsumen besarnya masing-masing 15289,135 km, 13294.,333 km dan 14241,381 km. Sedangkan total jumlah trip selama 1 bulan untuk masing-masing

kandidat adalah 132, 133 dan 137. Jadi, besar biaya untuk masing-masing kandidat adalah Rp. 28.856.693,20 Rp. 26.458.377,72 dan Rp. 27.932.603,72.

Kendaraan 4000 kg memiliki total biaya Rp 1663.88/km dan biaya supir Rp 75000/trip. Jarak kandidat 1, 2 dan 3 untuk aktivitas pengantaran barang dari depot ke kandidat besarnya masing-masing 2362,37546 km, 3837,6512 km dan 4097,3692 km. Jadi, besar biaya untuk masing-masing kandidat adalah Rp. 4.080.709,28, Rp. 6.535.391,08 dan Rp.6.967.530,66.

Secara ringkas, hasil konversi biaya untuk masing-masing kandidat dapat dilihat pada tabel 4.8 sampai dengan 4.10 berikut ini:

Tabel 4.8. Biaya Dari Kandidat ke Pelanggan

	Jarak tempuh 1 bulan	Jumlah trip	Total Biaya
Kandidat 1	15289.1354	138	Rp29,081,693.20
Kandidat 2	13294.3331	133	Rp26,458,377.72
Kandidat 3	14241.3812	137	Rp27,932,603.72

Tabel 4.9. Biaya Dari Depot ke Pelanggan

	Jarak Tempuh 1 Bulan	Jumlah Trip	Total Biaya
Kandidat 1	2362.37546	60	Rp4,080,709.28
Kandidat 2	3837.6512	60	Rp6,535,391.08
Kandidat 3	4097.3692	60	Rp6,967,530.66

Tabel 4.10. Total Biaya Operasional

	Biaya Kandidat ke Pelanggan	Biaya Depot ke Kandidat	Total Biaya Operasional
Kandidat 1	Rp28,856,693.20	Rp4,080,709.28	Rp33,162,402
Kandidat 2	Rp26,458,377.72	Rp6,535,391.08	Rp32,993,769
Kandidat 3	Rp27,932,603.72	Rp6,967,530.66	Rp34,900,134

4.2.3.2 Biaya Pembelian Tanah

Konversi biaya pembelian lahan menjadi biaya bulanan dapat dilihat pada tabel 4.8 berikut ini. Konversi dilakukan dengan pendekatan seperti pada sub bab 4.1.3.

Tabel 4.11. Konversi Biaya Pembelian Tanah

	Biaya Pembelian Lahan	Biaya perbulan
Kandidat 1	Rp3,360,000,000.00	Rp39,228,000.00
Kandidat 2	Rp4,704,000,000.00	Rp54,919,200.00
Kandidat 3	Rp4,778,059,928.90	Rp55,783,849.67

4.3 Analisis

4.3.1 Analisis Metode Perhitungan Jarak

Perhitungan jarak antara depot ke tiap kandidat tidak memerlukan metode khusus, cukup dengan menggunakan bantuan *Googlemaps*. Hal ini dikarenakan nantinya depot hanya perlu melayani sebuah DC. Artinya, depot tidak perlu memikirkan alokasi kendaraan dan urutan pengantaran seperti yang diperlukan untuk menghitung jarak dari tiap kandidat ke masing-masing konsumen.

Selain dengan pendekatan VRP, perhitungan jarak antara tiap kandidat dan konsumen sebenarnya dapat dilakukan dengan pendekatan program integer. Dalam program integer, pengantaran barang diasumsikan diantarkan dengan menggunakan 1 kendaraan untuk 1 pelanggan. Artinya, bila pada satu hari terdapat 25 pelanggan, perusahaan harus memiliki 25 kendaraan untuk melayani seluruhnya. Pendekatan ini tidak mungkin diterapkan pada kasus di Perusahaan, dimana perusahaan hanya memiliki 6 buah kendaraan yang dapat digunakan untuk melayani seluruh konsumennya. Jadi, pendekatan yang paling baik adalah dengan menggunakan pendekatan VRP. Penggunaan VRP juga sangat cocok untuk kasus ini, karena perusahaan juga menggunakan *routing sistem* dalam mendistribusikan produknya.

4.3.2 Analisis Program

4.3.2.1 Analisis Penggunaan Algoritma DE

Pada permasalahan ini, penyelesaian masalah VRP dilakukan dengan menggunakan program MATLAB yang berbasis pada Algoritma *differential evolution*. Algoritma DE merupakan salah satu jenis algoritma evolusioner (EA) yang menggunakan konsep evolusi biologi yang dikemukakan oleh Darwin sebagai cara bekerjanya, dimana pada algoritma jenis ini individu yang kuat akan bertahan dan maju ke generasi selanjutnya dan individu yang lemah akan musnah. Kuat atau lemahnya individu ditentukan berdasarkan hasil evaluasi individu tersebut terhadap fungsi objektifnya, yang dalam kasus ini adalah meminimumkan total jarak yang harus ditempuh dalam melayani seluruh konsumen.

Metode DE ini merupakan metode optimasi global. Berbeda dengan algoritma seperti *Tabu search* dan SA yang merupakan metode optimasi lokal. Metode optimasi global akan menghasilkan hasil yang lebih baik daripada metode optimasi lokal. Prinsip dasar DE hampir sama dengan algoritma genetik (karena algoritma DE merupakan turunan dari algoritma genetik), hanya yang membedakan pada algoritma genetik proses pindah mendahului proses mutasi.

Populasi awal untuk metode ini dapat dibuat dengan dua cara, yaitu dengan menggunakan metode heuristik (*sweep, saving, nearest neighbor, dll*) atau dengan menggunakan bilangan acak. Untuk program ini, penulis memilih menggunakan bilangan acak dengan tujuan mempercepat terjadinya konvergensi. Individu pada populasi awal yang memiliki nilai fungsi objektif terbaik akan bertahan, dan selanjutnya disebut dengan individu target.

Pada proses seleksi, akan terjadi kompetisi antara individu target dan individu *trial* (individu hasil perkawinan antara individu target dengan individu awal yang mengalami mutasi). Ketika nilai hasil evaluasi dari individu anak lebih baik dibandingkan nilai individu target, maka individu anak akan menjadi individu target baru untuk generasi yang berikutnya dan menggantikan peranan individu target pada generasi itu. Hal inilah yang menyebabkan mengapa dengan algoritma DE cepat terjadi konvergensi, karena populasi baru tiap generasi selalu menghasilkan nilai fungsi objektif yang lebih baik dari populasi sebelumnya.

DE relatif lebih tangguh jika dibandingkan dengan algoritma evolusioner lainnya dan dapat memproduksi hasil yang sama secara konsisten untuk banyak percobaan jika telah mencapai keadaan konvergen. Selain itu, karena strukturnya yang cukup sederhana dan penggunaan parameter kontrol yang lebih sedikit, DE juga memiliki waktu yang relatif lebih singkat dalam mencari solusi optimal jika dibandingkan dengan teknik algoritma yang lainnya.

Salah satu kelemahan dari metode ini adalah solusi akhir (berupa rute terbaik dan jarak tempuhnya) yang ditawarkan akan berbeda jika dilakukan *re-run* program, walaupun hasil solusi yang ditawarkan tersebut tidak akan jauh berbeda dengan yang ada sebelumnya. Hal ini disebabkan oleh pembangkitan bilangan acak yang dilakukan terhadap pembentukan populasi target, populasi mutan dan saat proses pindah silang.

4.3.2.2 Analisis Jalannya Program

Secara umum, program dapat berjalan dengan baik. Ini dibuktikan dengan solusi akhir yang ditawarkan selalu lebih baik dari solusi awal. Selain itu, Hasil perhitungan jarak secara manual untuk rute-rute yang dihasilkan oleh program memberikan hasil yang sama dengan kalkulasi oleh program.

Kelemahan dari program ini adalah waktu *running* yang relative lambat. Seperti telah dijelaskan sebelumnya, *running time* program sangat dipengaruhi oleh beberapa hal, seperti parameter F, CR, NP dan kriteria terminasi. Selain itu, jumlah pelanggan juga akan mempengaruhi *running time*. Untuk lebih jelasnya dapat dilihat pada tabel berikut ini:

Tabel 4.12. *Running Time* Program

Tanggal	Jumlah Pelanggan	Rata-rata Waktu <i>Running</i> Program (detik)	Tanggal	Jumlah Pelanggan	Rata-rata Waktu <i>Running</i> Program (detik)
1	22	47	17	28	70
2	21	47	18	20	47
3	29	70	19	13	20
4	16	25	20	28	67
5	14	22	21	28	67
6	32	97	22	25	51
7	31	97	23	36	119
8	26	64	24	33	98
9	35	115	25	22	47
10	28	67	26	10	9
11	20	43	27	19	39
12	15	24	28	37	119
13	37	119	30	38	121
14	23	49	31	29	73
15	36	117			

Dari tabel terlihat bahwa untuk jumlah pelanggan 10-20, waktu yang dibutuhkan sekitar 30-40 detik, jumlah pelanggan 20-30 membutuhkan waktu 60-90 detik dan jumlah pelanggan diatas 30 mencapai lebih dari 1 menit. Program akan berjalan semakin lambat bila pelanggan makin banyak. Hal ini disebabkan program akan memerlukan waktu yang lebih lama untuk mencari kombinasi yang paling optimal. Hal ini sesuai dengan pemikiran Toth dan Vigo (2002) yang menyatakan bahwa permasalahan VRP merupakan permasalahan *nonpolynomial hard (NP-hard)*, dimana semakin besarnya permasalahan (semakin banyak jumlah konsumen), maka usaha perhitungan untuk menyelesaikan masalah akan semakin besar. Usaha perhitungan inilah yang akan berbanding lurus dengan *running time* program.

Waktu *running* yang lama ini juga dipengaruhi oleh faktor kriteria terminasi program. Penulis menetapkan program baru akan berhenti bila telah

melakukan 1000 iterasi. Hal ini tentu saja memakan waktu lebih lama dibandingkan dengan bila program hanya perlu melakukan 10 iterasi. Namun, penerapan jumlah iterasi yang sangat banyak ini bertujuan agar program benar-benar mendapatkan hasil yang optimal. Untuk menurunkan *running time* program, cara yang dapat dilakukan adalah menurunkan jumlah iterasi ataupun menetapkan *running time* maksimum sebagai kriteria terminasi. Resiko bila hal ini dilakukan adalah hasil output program yang tidak terlalu baik.

Faktor lain yang berpengaruh terhadap *running time* adalah parameter pindah silang (CR). Untuk membuktikannya, penulis mencoba melakukan kembali perhitungan untuk data tanggal 26 Januari 2009, namun dengan nilai F dan CR yang berbeda-beda. Hasilnya dapat dilihat pada tabel 4.13.

Tabel 4.13. *Running Time* Untuk Tiap Kombinasi F dan CR

Parameter	Computation Time	Hasil	Parameter	Computat	Hasil	
F=0.4	CR=0.5	11.0333	F=0.8	CR=0.5	14.52185	98.4782
	CR=0.6	11.9822		CR=0.6	12.5778	97.0923
	CR=0.7	8.998228		CR=0.7	7.039893	98.6702
	CR=0.8	7.245561		CR=0.8	5.446932	103.7799
	CR=0.9	5.963244		CR=0.9	4.136981	114.988
	CR=1	4.266739		CR=1	4.48679	119.6309
F=0.5	CR=0.5	12.194342	F=0.9	CR=0.5	11.1342	95.4364
	CR=0.6	10.6338		CR=0.6	10.6391	99.7116
	CR=0.7	9.041027		CR=0.7	9.0117	104.3389
	CR=0.8	7.836255		CR=0.8	7.8893	126.6937
	CR=0.9	6.456608		CR=0.9	6.476608	133.1137
	CR=1	4.97551		CR=1	4.67551	136.2691
F=0.6	CR=0.5	12.121645	F=1	CR=0.5	12.15982	98.4364
	CR=0.6	11.5778		CR=0.6	12.66009	109.1316
	CR=0.7	9.039893		CR=0.7	9.3117	110.3489
	CR=0.8	7.446932		CR=0.8	6.22567	126.6937
	CR=0.9	5.146981		CR=0.9	6.476608	129.1407
	CR=1	4.48679		CR=1	4.67551	132.7691
F=0.7	CR=0.5	12.128874				
	CR=0.6	11.00987				
	CR=0.7	9.893076				
	CR=0.8	7.644002				
	CR=0.9	5.12982				
	CR=1	4.411828				

Dari tabel terlihat bahwa makin besar CR, waktu *running time* program menjadi lebih singkat. Hal tersebut dikarenakan nilai CR yang besar akan membuat peluang munculnya bilangan acak yang lebih kecil atau sama nilainya dari CR semakin besar. Namun, nilai CR yang makin besar memiliki konsekuensi solusi yang dihasilkan tidak terlalu baik. Hal ini dikarenakan konvergensi terjadi secara prematur (terlalu cepat). Nilai F sendiri tidak berpengaruh pada *running time* program. Berapapun nilai F ternyata tidak memberikan perbedaan berarti pada *running time* program. Karena itu, dalam pembuatan suatu program, perlu ditentukan mana yang akan dijadikan prioritas, *running time* ataukah solusi yang dihasilkan. Dalam kasus ini, yang dijadikan prioritas adalah solusi yang dihasilkan. Jadi, nilai CR yang ditetapkan adalah yang memberikan solusi terbaik, bukan *running time* terbaik.

Selain masalah *running time*, ada beberapa kekurangan lain yang ada pada proses perhitungan jarak dengan menggunakan program ini. Kekurangan pertama adalah masalah data jarak yang digunakan. Data jarak antartitik yang digunakan dalam program merupakan jarak sesuai dengan alur jalan dengan menggunakan bantuan peta digital *Googlemaps* dan GPS. Penggunaan kedua *tools* ini sebenarnya lebih baik daripada kita hanya menggunakan asumsi garis lurus antara dua titik. Pada kenyataannya hasil keluaran *Googlemaps* dan GPS mungkin tidak akan terlalu akurat. Cara lain yang dapat digunakan untuk menghilangkan kekurangan ini adalah dengan menghitung keadaan aktual (jalur aktual yang dilalui) berdasarkan odometer kendaraan, yaitu dengan menghitung nilai odometer sebelum berangkat dan nilai odometer setelah sampai ditujuan. Selisih antar nilai akhir dan nilai awal merupakan jarak yang ditempuh kendaraan. Perhitungan cara ini merupakan cara perhitungan yang paling akurat. Namun, kelemahan dari cara ini adalah prosesnya akan memakan waktu sangat lama. Karena itu, penulis tetap menggunakan metode pengukuran dengan *Googlemaps* dan GPS.

Kekurangan kedua adalah data waktu. Pada program ini, data waktu tempuh dicari dengan menggunakan hasil bagi antara jarak dengan kecepatan rata-rata. Untuk mendapatkan data yang lebih akurat, waktu tempuh dapat dihitung dengan menggunakan bantuan *stopwatch* selama proses pengantaran barang berlangsung. Namun, hal ini tentu saja juga akan memakan waktu yang lama.

Kekurangan ketiga adalah program dibuat hanya untuk satu jenis kendaraan dan satu jenis *time windows* pelanggan. Bila ada dua atau lebih jenis kendaraan dan *time windows* pelanggan berubah-ubah, program ini tidak akan dapat berjalan. Artinya, program ini hanya bisa digunakan untuk kasus di perusahaan agribisnis ini. Bila ingin digunakan untuk kasus lain, program ini perlu ditinjau ulang *source code*-nya.

Selain untuk perhitungan data jarak di tiap kandidat, program ini juga dapat digunakan dalam mencari rute distribusi optimal dari tiap kandidat ke tiap pelanggan. Artinya, saat DC sudah dibuat nanti, program ini tetap dapat digunakan oleh perusahaan untuk mencari rute pengantaran yang paling optimal dari DC dalam melayani pelanggannya.

4.3.3 Analisis Hasil

4.3.3.1 Analisis Jarak Kandidat-Pelanggan

Dari tabel 4.1 terlihat bahwa jarak harian kandidat nomor 2 (Bintaro) selalu lebih kecil dibandingkan kandidat yang lain, kecuali pada tanggal 5, 15 dan 25 dimana jarak terbaik dimiliki kandidat nomor 3 (Pulo Gadung). Hasil ini tentu akan berimbas pada data jarak selama 1 bulan yang merupakan akumulasi dari data harian, dimana Bintaro memiliki jarak terkecil (13294.333 km), diikuti Pulo Gadung (14241.381 km) dan Cibubur (15289.135 km).

Perhitungan ini menunjukkan bahwa Bintaro merupakan tempat paling efektif untuk mendistribusikan barang kepada konsumen. Perbedaan jarak tempuh yang mencapai 1000 km (dengan pulo gadung) dan 2000 km (dengan Cibubur) makin membuktikan hal ini. Kedekatan jarak DC dengan pelanggan memberikan keuntungan berupa biaya transportasi yang lebih murah dan makin rendahnya kemungkinan terjadinya keterlambatan pengantaran. Semua hal ini pada akhirnya akan meningkatkan kepuasan pelanggan dan keuntungan perusahaan.

Jarak terkecil yang didapatkan Bintaro disebabkan oleh letaknya yang berada dekat dengan Jakarta selatan, dimana 60% pelanggan Perusahaan berdomisili di sini. Kedekatan letak ini tentu saja akan mempermudah pengantaran barang. Cibubur memiliki jarak terjauh karena letaknya yang begitu

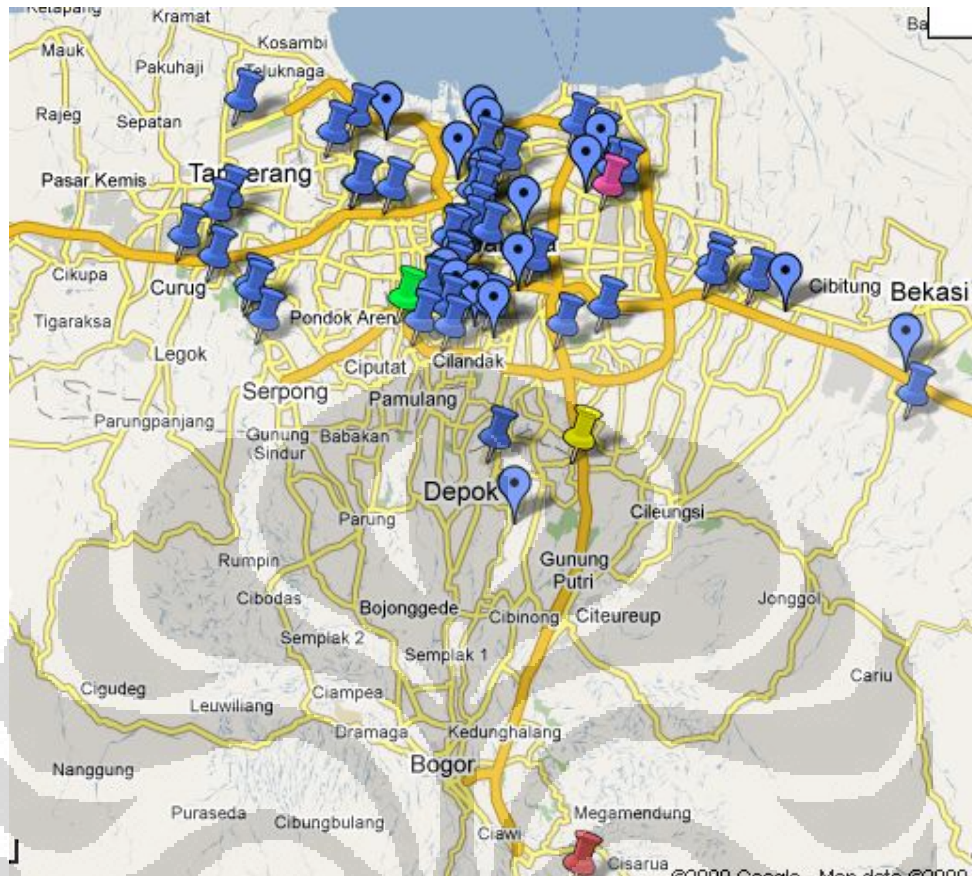
jauh dari kumpulan konsumen. Hasil ini sesuai dengan gambaran pada peta, seperti yang ditunjukkan gambar 4.3 dibawah ini.

Data ini belum dapat dijadikan sebagai alat pengambilan keputusan. Hal ini karena proses pengambilan keputusan masih harus mempertimbangkan faktor jarak dari depot ke tiap kandidat dan biaya pembelian lahan.

4.3.3.2 Analisis Jarak Depot-Kandidat

Dari tabel 4.2 terlihat bahwa, depot memiliki jarak terdekat dengan Cibubur (38.10283 km). Jarak dari depot ke Bintaro dan Pulo Gadung hampir sama yaitu masing-masing sebesar 61.89 km dan 66.08 km. Letak DC yang makin dekat dengan depot tidak memberikan keuntungan apapun selain biaya transportasi yang lebih murah. Karena walaupun barang dapat lebih cepat sampai ke DC (jarak rendah mengakibatkan waktu pengantaran barang menuju DC menjadi makin cepat), proses pengantaran barang baru dapat dilakukan pada masa *time windows* pelanggan. Bila pengantaran dilakukan sebelum *time windows* dimulai, maka kendaraan tetap akan menunggu sampai *time windows* dimulai. Letak DC yang makin dekat dengan depot juga mungkin akan menimbulkan kerugian berupa makin tingginya kemungkinan terjadinya keterlambatan pengantaran. Hal ini karena letak DC dan kumpulan pelanggan yang relatif jauh.

Aktivitas pengantaran barang dari depot ke kandidat merupakan aktivitas yang paling menentukan total biaya operasional di tiap kandidat. Hal ini dikarenakan aktivitas ini merupakan aktivitas yang memerlukan biaya paling besar. Jadi, tempat dengan jarak terdekat dengan depot merupakan tempat yang paling berpotensi memiliki biaya operasional terkecil dibandingkan dengan dua tempat lainnya.



Gambar 4.3. Letak Pelanggan, Depot dan Kandidat

Keterangan:

-  adalah depot,
-  adalah pelanggan,
-  adalah kandidat lokasi Bintaro,
-  adalah kandidat lokasi Cibubur dan
-  adalah kandidat Lokasi Pulo Gadung

4.3.4 Analisis Biaya

Ada dua jenis biaya yang digunakan dalam proses penentuan lokasi terbaik untuk DC, yaitu biaya operasional dan biaya pembelian tanah. Biaya pembuatan bangunan tidak digunakan karena dimanapun DC ini akan dibangun, luas bangunannya akan sama. Artinya, biaya pembuatan bangunan tidak memberikan dampak apa-apa terhadap hasil perhitungan.

Biaya operasional merupakan biaya yang dibutuhkan untuk mendistribusikan produk sampai ke tangan pelanggan. Pada kasus ini, cara perhitungannya adalah dengan menambahkan biaya pengantaran barang dari depot ke DC dengan biaya distribusi dari DC ke pelanggan. Biaya operasional dihitung dengan asumsi biaya ini akan linear dengan jarak. Makin besar jarak, makin besar pula biayanya. Perhitungan ini dilakukan karena sulitnya menentukan biaya operasional *real* yang dimiliki oleh pihak perusahaan. Selain itu, aktivitas distribusi produk dari DC ke pelanggan masih dalam tahap rencana. Artinya, pihak perusahaan belum memiliki data ini. Jadi, cara terbaik untuk mengukur biaya operasional adalah dengan mengasumsikannya linear dengan jarak.

Besarnya biaya operasional sangat dipengaruhi oleh biaya/km untuk tiap aktivitas. Proses pengantaran barang dari depot ke DC membutuhkan biaya per km yang lebih besar daripada pengantaran dari DC ke pelanggan karena menggunakan kendaraan yang lebih besar. Perbandingan biaya/km untuk kedua aktivitas ini adalah 1,67:1.

Dari tabel 4.9 kita dapatkan hasil bahwa biaya terkecil untuk pengantaran produk dari depot ke DC dimiliki oleh Cibubur (Rp. 4.080.709,28). Nilai ini tidak terlalu signifikan karena perbedaannya dengan dua kandidat lainnya hanya sekita dua juta rupiah saja.

Untuk pengantaran dari DC ke pelanggan, biaya terkecil dimiliki oleh Bintaro (Rp. 26.458.377,72). Nilai ini juga tidak terlalu signifikan karena perbedaannya dengan dua kandidat lainnya hanya sekita tiga juta rupiah saja.

Ketika nilai kedua biaya ini ditambahkan, maka didapatkan bahwa, dari segi biaya operasional, tempat terbaik pendirian DC adalah Bintaro (Rp. 32.993.769). Namun, selisih biaya operasional pada kandidat ini dengan biaya operasional pada kandidat Cibubur tidaklah terlalu besar, hanya sekitar Rp 168.000.

Besar biaya ini sesuai dengan besar jarak di masing-masing kandidat. Dari sisi total jarak (jarak depot-DC dan DC-pelanggan), kandidat Bintaro jauh lebih baik daripada Cibubur. Total jarak Bintaro adalah 17.131,984 km, sedangkan cibubur 17.651,511 km.

Untuk biaya pembelian lahan, biaya terkecil kembali dimiliki oleh Cibubur (Rp.3.920.000.000) dengan perbedaan sekitar 800 juta rupiah dengan kandidat lainnya. Setelah dikonversi dengan menggunakan *interest rate* sebesar 14.05%, biaya bulanan pembelian lahan di Cibubur adalah Rp. 45.766.000. Nilai ini memiliki perbedaan sekitar 15 juta rupiah dengan biaya bulanan lahan di dua kandidat lokasi lainnya. Harga tanah di Cibubur jauh lebih rendah kemungkinan disebabkan letaknya yang masih jauh dari keramaian dan pusat kota. Jadi dapat disimpulkan bahwa dari segi biaya bulanan pembelian lahan, kandidat Cibubur adalah tempat yang paling menguntungkan.

Untuk lebih memperjelas masalah biaya ini, penulis mencoba membandingkan total biaya untuk pendirian DC di Cibubur dengan biaya yang dihabiskan dalam sistem distribusi yang saat ini digunakan Perusahaan. Saat ini, Perusahaan menggunakan sistem distribusi langsung. Dari hasil wawancara dengan divisi distribusi Perusahaan, didapatkan rute-rute pengantaran yang telah dilakukan untuk bulan Januari 2009. Rute-rute ini kemudian dicari jarak tempuhnya. Jarak tempuh *existing system* Perusahaan dapat dilihat pada tabel berikut ini:

Tabel 4.14. Total Jarak Harian Untuk *Existing System* Perusahaan

Hari	Jarak	Hari	Jarak
1	767.21844	17	813.25154
2	739.85169	18	784.2428
3	1054.6381	19	1117.9164
4	604.02415	20	640.2656
5	408.8124	21	433.34115
6	1289.615	22	1366.9919
7	1207.2799	23	1279.7167
8	976.37712	24	1034.9597
9	1513.9381	25	1604.7744
10	879.33898	26	932.09932
11	732.61341	27	776.57021
12	435.73397	28	461.87801
13	1185.4261	29	1256.5517
14	1249.0709	30	1324.0152
15	661.31462	31	700.9935
16	1296.6696	Total	29413.979

Dari tabel terlihat bahwa total jarak sistem lama adalah 29.413,979 km. Secara keseluruhan, nilai ini masih lebih besar daripada total jarak tempuh tiap kandidat lokasi. Artinya, dimana pun DC didirikan, perusahaan akan dapat menghemat jarak tempuhnya. Bila jarak tempuh dapat dihemat, maka waktu pengantaran pun akan lebih cepat sehingga keterlambatan pengantaran dapat dihindari.

Bila dibandingkan dengan sistem lama ini, kandidat lokasi Cibubur dengan total jarak tempuh 17.651,511 km akan memberikan penghematan jarak sebesar 39.90%. Hal ini menunjukkan bahwa dari sisi jarak tempuh, sistem baru dengan DC ditempatkan di Cibubur jauh lebih baik daripada sistem lama.

Bila ditinjau dari segi biaya yang harus dikeluarkan, sistem lama akan menghasilkam biaya sebesar Rp. 57.759.201,55. Bila dibandingkan dengan sistem baru, nilai ini jauh lebih kecil, dimana sistem baru membutuhkan biaya Rp. 78.928.402,48. Artinya setiap bulan perusahaan harus mengeluarkan biaya lebih besar 41.7% apabila rencana pendirian DC jadi direalisasikan.

Sistem baru memberikan biaya yang lebih besar dikarenakan sistem baru perlu mengeluarkan biaya pembelian tanah selain biaya operasional. Namun, dengan sistem baru, perusahaan dapat mengharapkan pemasukan lebih banyak karena *coverage area* perusahaan saat ini menjadi semakin luas. Perusahaan berpeluang mendapatkan pelanggan yang lebih bayak tanpa perlu takut lagi akan terjadi keterlambatan pengiriman.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Sesuai dengan tujuan penelitian, yaitu mencari lokasi pendirian DC yang paling optimal dari tiga buah kandidat lokasi yang ada, maka didapatkan dua buah kesimpulan. Pertama, bila ditinjau dari sisi biaya operasional, maka tempat terbaik untuk mendirikan DC adalah di Bintaro. Hanya saja, karena perhitungan biaya operasional tidak memperhitungkan faktor kemacetan dan biaya-biaya lain, maka ada kemungkinan tempat terbaik tidak pada lokasi ini. Kedua, bila ditinjau dari sisi biaya pembelian lahan, tempat terbaik pendirian DC adalah di Cibubur

5.2 Saran

Untuk penelitian selanjutnya, diharapkan ada beberapa perubahan yang harus dilakukan demi tercapainya hasil penelitian yang lebih baik. Saran yang penulis dapat berikan adalah:

1. Sebaiknya masalah biaya tenaga kerja di tiap kandidat diperhitungkan, karena biaya tenaga kerja di satu tempat bisa jadi akan berbeda dengan tempat lainnya
2. Dalam perhitungan jarak sebaiknya masalah kemacetan diperhitungkan

DAFTAR REFERENSI

- Ballou, R.H. (2004). *Business logistics management* (5th ed). New Jersey: Prentice-Hall Inc.
- Ballou, R.H., & Agarwal, Y.K. (1998). A performance comparison of several popular algorithms for vehicle routing and scheduling. *Journal of Business Logistics*, 9, 51 - 65.
- Blank, L., & Parquin, A. (2005). *Engineering Economy* (6th ed). New York: McGraw-Hill.
- Braysy, O., & Gendreau, M. (2002). Tabu search heuristics for the vehicle routing problem with *time windows*. *Journal of SINTEF Applied Mathematics*, 211-237.
- Chan, F.T.S., Kumar, N., & Choy, K.L. (2005). Decision making approach for the distribution centre location problem in a supply chain network using the fuzzy-based hierarchycal concept. *ProQuest Science Journals*, 221, 725-740.
- Charnsethikul, P., & Singhtaun, C. (2006). Single facility location problem under uncertain emerging of demand. *Department of Industrial Engineering, Faculty of Engineering, Kasetsart University*, 1-14.
- Chu, T.C., & Lai, M.T. (2004). Selecting distribution centre location using an improved fuzzy MCDM approach. *International Journal Manufacturing Technology*, 26, 293-299.
- Dwiningsih, N. (2002). *Supply chain management dan E-Commerce*. Jakarta: Stekpi.
- Erbao, C., Mingyong, L., & Kai, N. (2008, July). *A differential evolution & genetic algorithm for vehicle routing problem with simultaneous delivery and pick-up and time windows*. Paper presented at 17th World Congress of The International Federation of Automatic Control, Seoul.

- Homberger, J., & Gehring, H. (2004). A Two-Phase Hybrid Metaheuristik for The Vehicle Routing Problem with *Time Windows*. *European Journal of Operation Research*, hal. 220 - 238.
- Karaboga, D. & Okdem, S. (2004). A simple and global optimization algorithm for engineering problems: Differential Evolution algorithm. *Turkey Journal Engineering*, 12, 1-8.
- Onwubulo, C.G., & Davendra, D. (2008). *Differential evolution: A handbook for global-permutation based combinatorial optimization*. California: Springer.
- Poot, A., Kant, G., Wagelmans, A.P.M. (2002). A Saving Based Method for Real Life Vehicle Routing Problem. *Journal of The Operational Research Society*, hal. 57 – 68.
- Price, K.V., Storn, M.R., & Lampinen, J.A. (2005). *Differential evolution: a practical approach to global optimization*. California: Springer.
- Spiller, A., Bolten, J., & Kennerknecht, R. (2006, June). *Customer satisfaction and loyalty as success faktors in organic food retailing*. Paper presented at the 16th Annual World Forum and Symposium “Agribusiness, Food, Health, and Nutrition”, IAMA Conference, Buenos Aires.
- Tan, K.C., Lee, L.H., Zhu, K.Q. (1999). Heuristik methods for vehicle routing problem with time windows. *National University of Singapore*, 1-34.
- Toth, P., & Vigo, D. (2002). *The Vehicle Routing Problem*. Philadelphia: Society for Industrial and Applied Mathematics.