

**PENENTUAN RUTE DISTRIBUSI DENGAN
ALGORITMA TABU SEARCH
UNTUK VRP DENGAN TIME WINDOWS
(STUDI KASUS DI PT.X)**

TESIS

Oleh :
Clarissa Asteria
0606004325

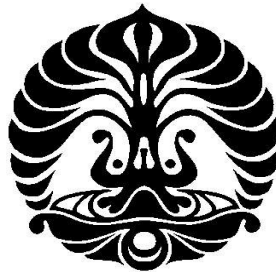


**PROGRAM STUDI TEKNIK INDUSTRI
PROGRAM PASCA SARJANA BIDANG ILMU TEKNIK
UNIVERSITAS INDONESIA
2008**

Universitas Indonesia

**PENENTUAN RUTE DISTRIBUSI DENGAN
ALGORITMA TABU SEARCH
UNTUK VRP DENGAN TIME WINDOWS
(STUDI KASUS DI PT.X)**

Oleh :
Clarissa Asteria
0606004325



**Tesis ini diajukan sebagai
salah satu syarat untuk memperoleh gelar
MAGISTER TEKNIK**

**PROGRAM STUDI TEKNIK INDUSTRI
PROGRAM PASCA SARJANA BIDANG ILMU TEKNIK
UNIVERSITAS INDONESIA
2008**

Universitas Indonesia

PERNYATAAN KEASLIAN TESIS

Saya menyatakan dengan sesungguhnya bahwa tesis saya dengan judul :

**PENENTUAN RUTE DISTRIBUSI DENGAN
ALGORITMA TABU SEARCH
UNTUK VRP DENGAN TIME WINDOWS
(STUDI KASUS DI PT.X)**

Yang dibuat untuk melengkapi sebagian persyaratan menjadi Magister Teknik pada program studi Teknik Industri, Program Pasca Sarjana Bidang Ilmu Teknik, Fakultas Teknik Universitas Indonesia, sejauh saya ketahui bukan merupakan tiruan atau duplikasi dari tesis yang sudah dipublikasikan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau instansi manapun, kecuali di bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, Juni 2008

Clarissa Asteria
NPM 0606004325

LEMBAR PENGESAHAN

Tesis ini diajukan oleh :
 Nama : Clarissa Asteria
 NPM : 0606004325
 Program Studi : Teknik industri
 Judul Tesis : Penentuan Rute Distribusi Dengan Algoritma Tabu Search Untuk VRP Dengan Time Windows (Studi Kasus Di PT.X)

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Pascasarjana Fakultas Teknik Universitas Indonesia

DEWAN PENGUJI

Pembimbing tesis 1

Pembimbing tesis 2

Ir Amar Rachman, MEIM
Msi

Ir Fauzia Dianawati,

Penguji 1

Penguji 2

Penguji 3

Ir Yadrifil, MSc

Ir Erlinda Muslim, MEE

Ir Isti Surjandari MT.,MA.,Ph.D

Depok, Juli 2008

RIWAYAT HIDUP PENULIS

Nama : Clarissa Asteria
 Tempat, Tanggal Lahir : Jakarta, 27 September 1982
 Alamat : Jl Kembang Indah v G5/29
 Puri Indah
 Pendidikan :

a.	SD	:	SDK XI BPK PENABUR (1989-1995)
b.	SMP	:	SMPK VII BPK PENABUR(1995-1998)
c.	SMU	:	SMAK IPEKA (1998-2001)
d.	S-1	:	Departemen Teknik Industri Fakultas Teknik Unika ATMA JAYA, (2001-2005)
e.	S-2	:	Departemen Teknik Industri Fakultas Teknik Universitas Indonesia, Depok (2006-sekarang)

UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yesus Kristus, atas segala berkat dan rahmat-Nya, tesis ini dapat selesai tepat pada waktunya. Tesis ini disusun dalam rangka melengkapi salah satu persyaratan untuk menyelesaikan Program Pendidikan Pasca sarjana Teknik Industri, FTUI.

Penulis menyadari bahwa penulisan tesis ini tidak akan selesai tanpa bantuan dari banyak pihak. Oleh karena itu, pada kesempatan ini, penulis ingin menyampaikan terima kasih kepada :

1. Bapak Amar Rachman dan Ibu Fauzia Dianawati, selaku dosen pembimbing yang telah banyak memberikan dorongan, bantuan, waktu, tenaga dan pikiran dalam membimbing penulis.
2. Papa, mama dan keluarga atas segala perhatian, dukungan dan doa yang telah diberikan
3. Harvian Satya Dharma atas segala perhatian, dukungan dan doa yang telah diberikan.
4. Thomas Bachri atas bantuan dan waktu yang diluangkan untuk penulis.
5. Seluruh teman-teman Pasca sarjana FTUI angkatan 2006
6. Dan seluruh pihak yang telah membantu sehingga tesis ini dapat terselesaikan.

Penulis menyadari bahwa tesis ini masih jauh dari sempurna dan memiliki keterbatasan. Oleh karena itu, segala saran dan kritik yang membangun sangat diharapkan demi kesempurnaan tesis ini. Akhir kata, penulis berharap tesis ini dapat bermanfaat bagi pihak yang berkepentingan.

Depok, Juni 2008

Penulis

UNIVERSITAS INDONESIA
PROGRAM PASCA SARJANA
PROGRAM STUDI TEKNIK INDUSTRI
Tesis, 7 Juli 2008

Clarissa Asteria

Penentuan Rute distribusi dengan algoritma tabu search untuk VRP dengan time windows (Studi kasus di PT.X)

ABSTRAK

Peranan transportasi dan distribusi sangat penting bagi perusahaan karena melalui kegiatan ini perusahaan dapat memenuhi kebutuhan konsumen. Kontribusi biaya transportasi dalam sistem distribusi perusahaan mencapai 1/3 hingga 2/3 dari total biaya distribusi. Karena itu, efisiensi sangat diperlukan untuk mengurangi biaya operasional distribusi perusahaan. Salah satu solusi untuk melakukan efisiensi tersebut adalah dengan menyusun jadwal pengiriman dan rute pengiriman yang lebih optimal sehingga dapat meminimalkan waktu, jarak dan biaya.

Vehicle Routing Problem (VRP) dikenal sebagai konsep yang dapat digunakan untuk mendapatkan rute terbaik bagi kendaraan. Tujuan dari VRP adalah mengatur urutan rute pemberhentian dalam pengiriman sehingga menghasilkan jarak tempuh total seminimal mungkin. Ada 3 jenis penyelesaian VRP; solusi eksak, heuristik dan metaheuristik. *Tabu Search* (TS) adalah salah satu solusi metaheuristik yang berbasis pada pencarian lokal.

Masalah ini juga yang dihadapi PT.X, sebuah perusahaan manufaktur yang memproduksi berbagai jenis cat. Selama ini pertimbangan perusahaan dalam mengatur rute hanyalah keterbatasan kapasitas, berat barang yang diangkut dan lokasi, tidak mempertimbangkan apakah jarak tempuh rute tersebut sudah minimal atau belum sehingga biaya bahan bakar yang dikeluarkan juga belum tentu minimal.

Guna menyelesaikan masalah pengiriman tersebut, maka dikembangkan model penyelesaian VRP dengan metode *Tabu Search* (TS). Penerapan metode TS memerlukan adanya solusi awal. Dalam penelitian ini, metode *sweep* yang digunakan untuk membuat solusi awal. Selanjutnya solusi awal tersebut dioptimalkan dengan menggunakan algoritma TS yang disusun dalam suatu program menggunakan bahasa pemrograman *Visual Basic*. Selanjutnya, dilakukan pengolahan data dengan TS dengan menggunakan data pengiriman satu hari di PT.X. Hasil pengolahan data dengan TS menghasilkan penurunan jarak sebesar 77.7 km atau 18%, waktu tempuh 1.8 jam atau 17.1% dan biaya sebesar 181.087,66 atau 18.5%.

Keyword : *vehicle routing problem, tabu search, cost efficiency*

**UNIVERSITY OF INDONESIA
GRADUATE PROGRAM IN ENGINEERING
INDUSTRIAL ENGINEERING PROGRAM
Thesis, 7 July 2008**

Clarissa Asteria

Distribution Routes Setting with Tabu Search Algorithm for VRP with Time Windows (Case study at PT.X)

ABSTRACT

Transportation and distribution are two important activities for company because by doing these activities, producer will be able to fulfill customer's requirements. The transportation cost contribution on distribution system in company typically range between one-third and two-third of total logistic costs. Therefore, efficiency is needed to decrease company's operational cost on distribution. One of solutions for doing such efficiency is by setting optimal distribution schedule and delivering routes so time, distance and cost can be minimized.

Vehicle Routing Problem (VRP) is known as a concept that can be used to seek the best routes for vehicle. The objective of VRP is to set delivering routes order so that its total distance can be minimized. Basically, there are 3 kinds of VRP solution; exact solution, heuristics and metaheuristics. Tabu Search (TS) is one of metaheuristics that possess local search basic.

PT.X, a manufacture company producing various kind of paints, has the routing problems. So far, the company's consideration in setting routes is merely based on capacity limitation, weight of products carried and location. The distance of route is never been assessed whether it has minimal distance so the delivery cost has been minimized optimally or not.

In order to solve the delivery problems, VRP was developed with Tabu Search (TS) method. Implementation of TS method needs initial solution. Sweep method was used to create initial solution. The initial solution was then optimized using TS algorithm in a visual basic programe. The data produced was then processed with TS using one day delivery data in PT.X. The result is 77,7 km or 18% efficiency in distance, 1.8 hours or 17.1% in time and 181.087,66 or 18,5% in cost.

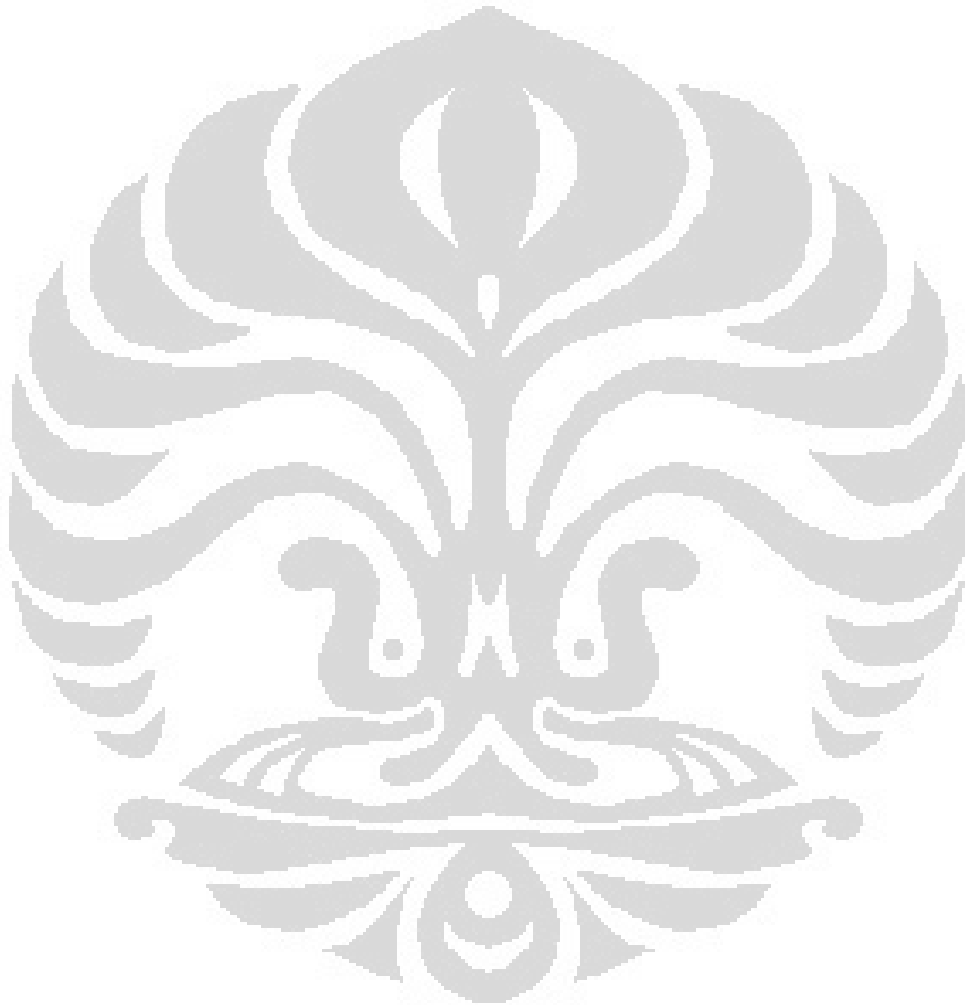
Keyword : vehicle routing problem, tabu search, cost efficiency

DAFTAR ISI

	halaman
HALAMAN JUDUL	ii
PERNYATAAN KEASLIAN SKRIPSI.....	iii
LEMBAR PENGESAHAN.....	iv
RIWAYAT HIDUP PENULIS.....	v
UCAPAN TERIMA KASIH.....	vi
ABSTRAK.....	vii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xiv
1.PENDAHULUAN.....	1
1.1 Latar Belakang Permasalahan	1
1.2 Diagram Keterkaitan Masalah	3
1.3 Rumusan Permasalahan.....	4
1.4 Tujuan Penelitian	4
1.5 Pembatasan masalah.....	4
1.6 Metodologi Penelitian	5
1.7 Sistematika Penulisan	6
2.LANDASAN TEORI	8
2.1 <i>Vehicle Routing Problem</i>	8
2.2 <i>Vehicle Routing and Scheduling</i>	10
2.2.1 <i>Methods for Routing and Scheduling</i>	11
2.3 <i>Penyelesaian Vehicle Routing Problems</i>	11
2.3.1 Solusi Eksak	11
2.3.2 Heuristik	11
2.3.2.1 <i>Sweep method</i>	11
2.3.3 <i>Metaheuristik</i>	12
2.3.4 <i>Tabu Search Meta-heuristik</i>	12
2.3.4.1 <i>Pengertian umum</i>	12

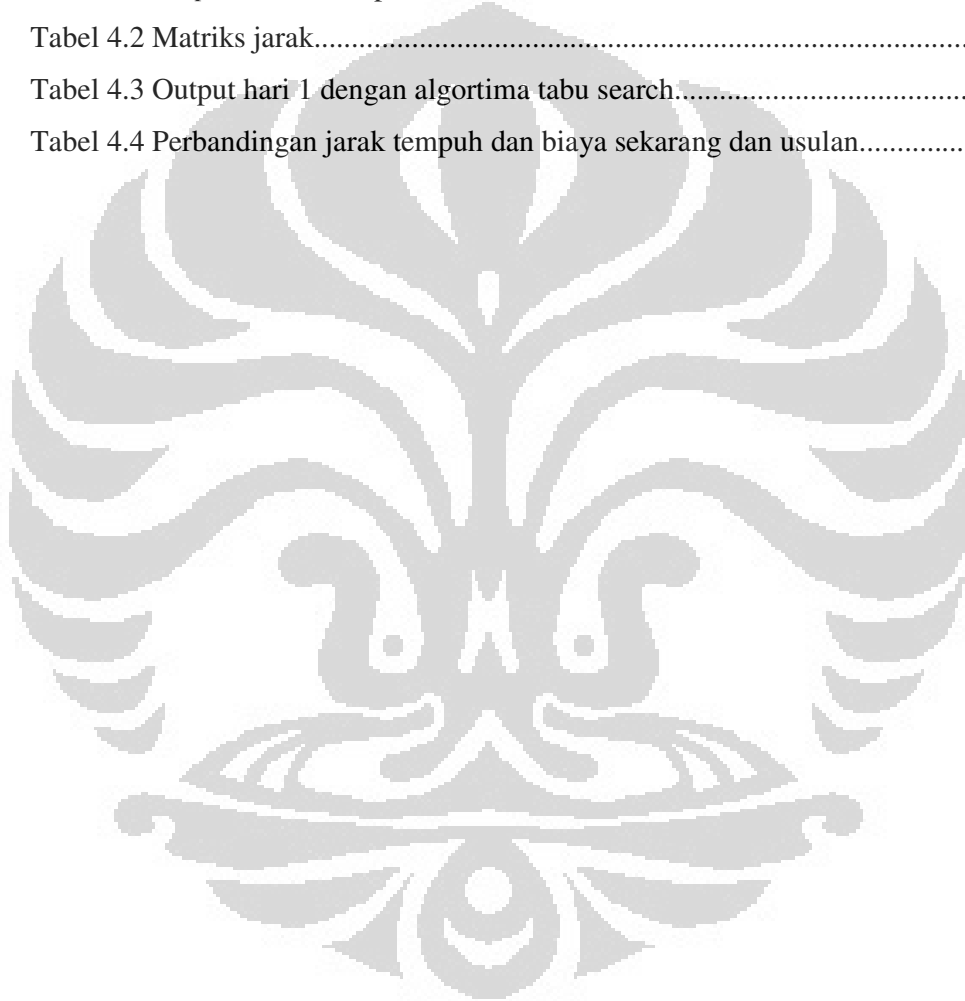
2.3.4.2 Penggunaan memori	14
2.3.4.3 Intensifikasi dan diversifikasi	14
2.3.4.4 <i>Tabu search</i> pada <i>VRP</i>	15
2.3.4.4.1 Solusi awal	16
2.3.4.4.2 Mekanisme pembentukan solusi tetangga	16
2.3.4.4.3 Data Management Structure.....	17
2.3.4.4.4 Komponen <i>Tabu search</i>	18
2.3.4.4.5 Prosedur umum TS-MFVRP.....	19
3. PENGUMPULAN DATA.....	22
3.1 Profil Perusahaan	22
3.2 Pengumpulan Data	22
3.2.1 Armada pengiriman.....	22
3.2.2 Biaya pengiriman	23
3.2.3 Konsumen,Lokasi dan rute yang dilalui Truk menuju konsumen.....	23
3.2.4 Depot dan Informasi lainnya.....	24
3.2.5 Waktu	35
4. PENGOLAHAN DATA DAN ANALISIS	26
4.1 Rute distribusi.....	26
4.1.1.1 Pengerjaan solusi Awal.....	26
4.1.1.2 Langkah pengerjaan.....	26
4.1.1.3 Output.....	27
4.1.2 Pengolahan Solusi Awal dengan algoritma <i>tabu</i> <i>search</i>	28
4.1.2.1 Program visual basic menggunakan algoritma <i>tabu</i> <i>search</i>	28
4.1.2.2 Verifikasi dan validasi program.....	30
4.1.2.3 Tahap pengerjaan algoritma.....	31
4.1.2.4 Output.....	32
4.2 Analisis.....	32
4.2.1 Analisis penjadwalan pengiriman.....	32
4.2.2 Analisis rute distribusi.....	33

4.2.2.1 Analisis hasil perbaikan sistem pengiriman.....	33
4.2.2.1.1 Analisis jarak tempuh,waktu dan biaya pengiriman....	33
4.2.2.1.2 Analisis metode.....	33
5. KESIMPULAN DAN SARAN.....	35
DAFTAR PUSTAKA	36



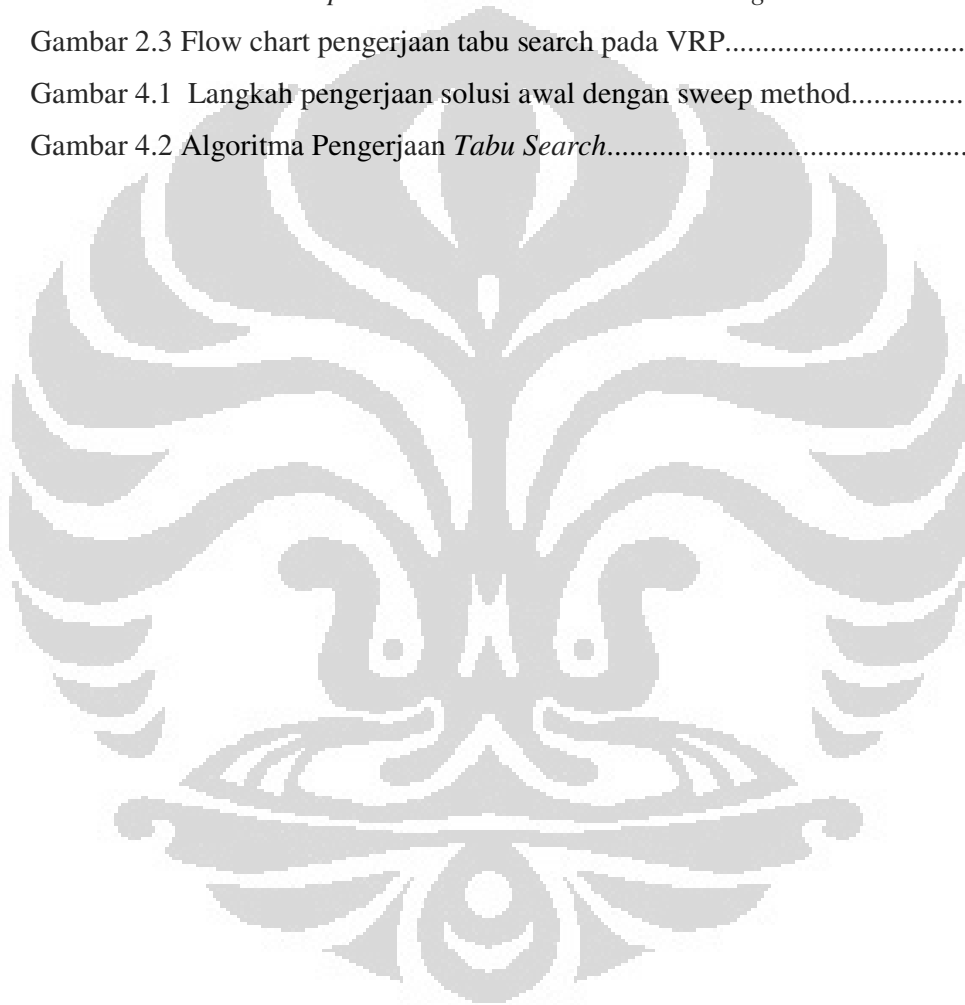
DAFTAR TABEL

Nomor Tabel	halaman
Tabel 3.1 Lokasi konsumen.....	23
Tabel 3.2 Permintaan 1 hari pelanggan	24
Tabel 3.3 Pengiriman 1 hari PT.X.....	24
Tabel 4.1 <i>Output</i> rute hari 1 pada solusi awal.....	28
Tabel 4.2 Matriks jarak.....	30
Tabel 4.3 Output hari 1 dengan algoritma tabu search.....	32
Tabel 4.4 Perbandingan jarak tempuh dan biaya sekarang dan usulan.....	33



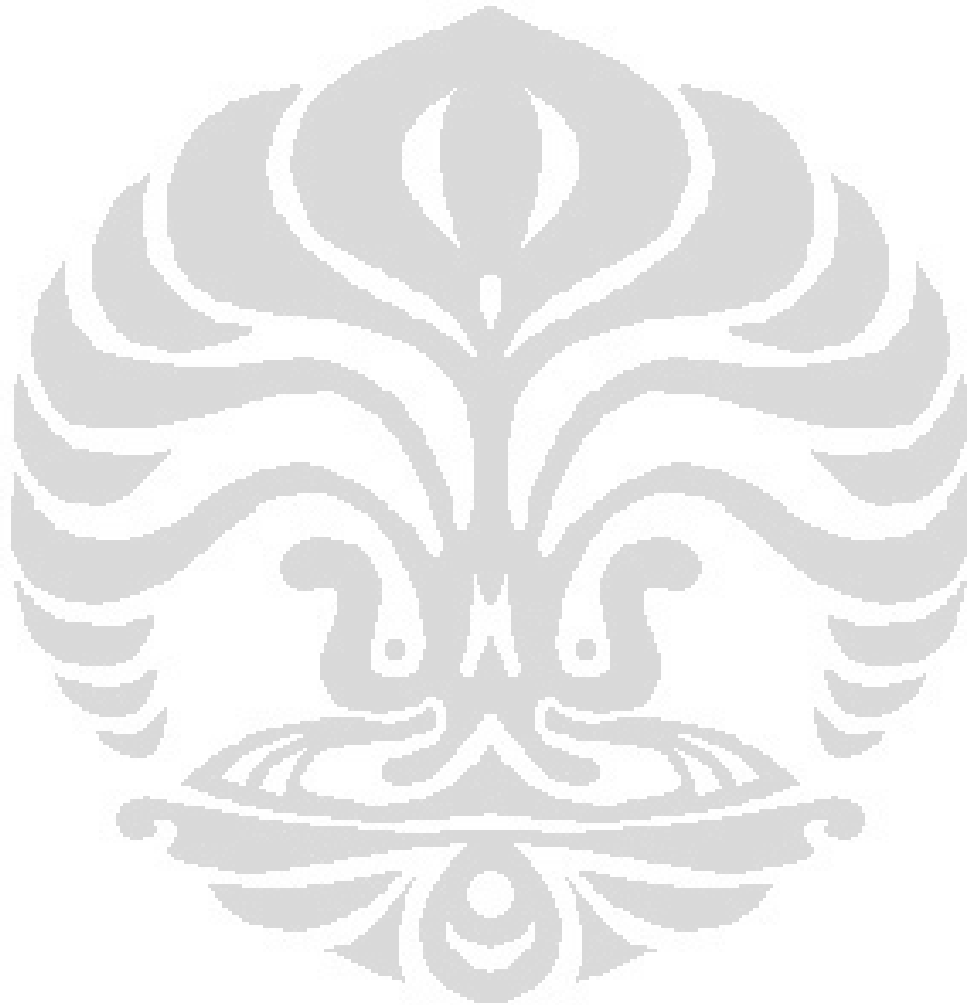
DAFTAR GAMBAR

Nomor Gambar	halaman
Gambar 1.1 Diagram Keterkaitan Permasalahan.....	3
Gambar 1.2 Metodologi Penelitian.....	5
Gambar 2.1 <i>Last trial move pada 1-interchange mechanism</i>	17
Gambar 2.2 <i>Insert move pada 2-consecutive node interchange mechanism</i>	17
Gambar 2.3 Flow chart pengerjaan tabu search pada VRP.....	21
Gambar 4.1 Langkah pengerjaan solusi awal dengan sweep method.....	27
Gambar 4.2 Algoritma Pengerjaan <i>Tabu Search</i>	29



DAFTAR LAMPIRAN

Lampiran 1 : Matriks jarak depot-konsumen.....	38
Lampiran 2 : Matriks waktu depot-konsumen.....	39
Lampiran 3 : Perhitungan kecepatan rata-rata.....	40
Lampiran 4 : <i>Source Code Visual Basic Algoritma Tabu Search</i>	41



1.PENDAHULUAN

I.1 LATAR BELAKANG PERMASALAHAN

Dalam dunia industri, sistem pengiriman dan penjadwalan memiliki peranan penting dalam suatu perusahaan. Manajemen logistik sendiri memiliki tujuan akhir yaitu mengantarkan produk ke konsumen tepat waktu dengan cara yang efektif dan efisien. Perusahaan mulai menyadari bahwa logistik mempunyai pengaruh yang signifikan terhadap biaya dan keputusan mengenai logistik akan menghasilkan level pelayanan kepada konsumen yang berbeda-beda. Tujuan akhir manajemen logistik adalah mendapatkan sejumlah barang atau jasa yang tepat pada tempat dan waktu yang tepat, serta kondisi yang diinginkan dengan memberikan kontribusi terbesar bagi perusahaan¹.

Untuk mencapai tujuan akhir manajemen logistik, diperlukanlah suatu sistem distribusi produk yang :

- Memastikan bahwa produk yang tersedia pada waktu dan jumlah yang tepat sesuai permintaan konsumen
- Memiliki kualitas yang terjamin
- Memperhatikan tingkat keselamatan dalam pendistribusiannya.

Suatu perusahaan harus dapat mengoptimalkan sistem distribusinya agar dapat bersaing dengan perusahaan sejenis lainnya. Salah satu caranya adalah dengan pengoptimalan transportasi.

Salah satu permasalahan dalam transportasi adalah *Vehicle Routing Problems* (VRP) yaitu merancang m set rute kendaraan dengan biaya terkecil dimana tiap kendaraan berawal dan berakhir di depot, setiap konsumen hanya dilayani sekali oleh sebuah kendaraan, serta total permintaan yang dibawa tidak melebihi kapasitas kendaraan. Transportasi ini memberikan kontribusi biaya 1/3 sampai 2/3 dari total biaya distribusi¹. Melihat kontribusi biaya sebesar itu, maka sangat penting adanya usaha dalam rangka meningkatkan efisiensi transportasi melalui optimalisasi penggunaan peralatan dan personil transportasi. Beberapa

¹ Ronald H. Ballou, 2004

metode yang digunakan untuk menyelesaikan VRP antara lain adalah dengan pendekatan eksak, heuristik dan metaheuristik. Dibandingkan dengan heuristik klasik, metaheuristik menunjukkan pencarian solusi yang lebih teliti. Penelitian dalam metaheuristik ini lebih menunjukkan perkembangan yang hebat dalam dekade terakhir dan telah menghasilkan heuristic VRP yang lebih efektif dan fleksible². *Tabu search* (TS) merupakan metode terbaik yang dapat diimplementasikan pada VRP dibanding metaheuristik yang lain seperti *simulated annealing*, *genetic search*, *ant system* dan *neural network*.

Vehicle Routing Problem with time windows (VRPTW) merupakan perluasan dari VRP konvensional dimana terdapat pengaruh batasan kapasitas dan masing-masing pelanggan i berhubungan dengan suatu interval waktu (a_i, b_i), yang disebut *time window*. Waktu tersebut terdiri dari saat kendaraan meninggalkan depot, waktu perjalanan dan waktu pelayanan untuk masing-masing pelanggan. Waktu pelayanan untuk masing-masing pelanggan harus dimulai dalam kurun waktu *time window*, dan kendaraan harus berhenti pada lokasi pelanggan untuk waktu s_i (waktu pelayanan pelanggan). Jika kendaraan datang lebih awal pada pelanggan i , kendaraan tersebut secara umum menunggu sampai waktu a_i , sampai waktu pelayanan dimulai³.

Secara umum masalah yang dihadapi perusahaan dalam melakukan pengiriman barang antara lain kuantitas permintaan pengiriman yang berbeda-beda untuk setiap titik, keterbatasan kapasitas, batasan waktu pengiriman untuk suatu titik, lokasi pelanggan, permintaan yang berfluktuatif, kapan sebaiknya pengiriman dan pengambilan itu dilakukan dan bagaimana membuat suatu rute kendaraan untuk mencapai daerah tertentu dengan menggunakan bahan bakar yang efisien. Masalah ini juga yang dihadapi PT.X, sebuah perusahaan manufaktur yang memproduksi berbagai jenis cat. Selama ini pertimbangan perusahaan dalam mengatur rute hanyalah keterbatasan kapasitas, berat barang yang diangkut dan lokasi, tidak pernah mempertimbangkan apakah jarak tempuh rute tersebut sudah minimal atau belum sehingga biaya bahan bakar yang dikeluarkan juga belum tentu minimal. Penjadwalan pengiriman di PT.X,

² Gendreau M, Laporte G and Potvin J-Y, "Metaheuristic for the capacitated VRP, 2002

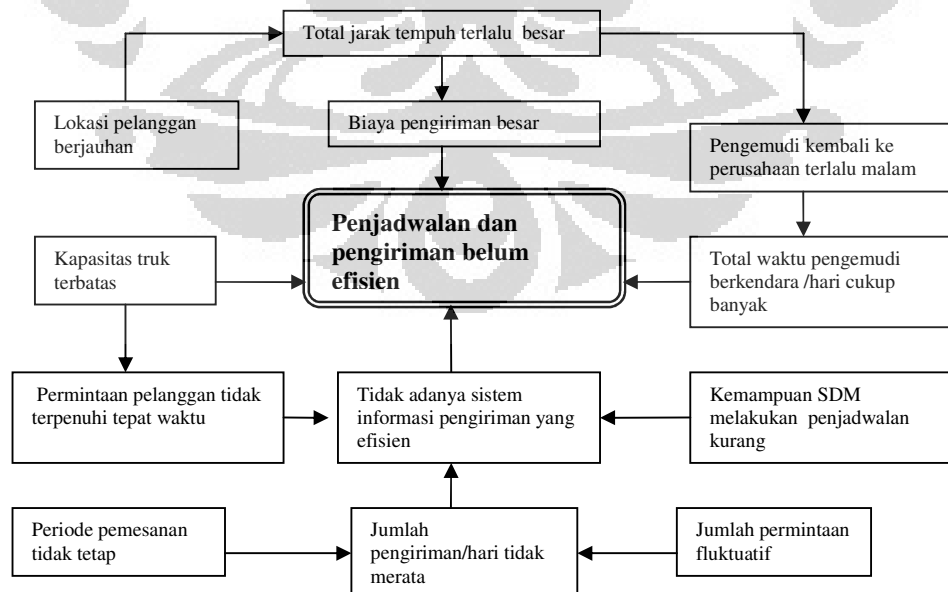
³ Daniele Vigo, 2002

dilakukan berdasarkan kapan pesanan dari pelanggan datang. Karena pelanggan memesan dalam jangka waktu dan jumlah yang tidak tetap maka keadaan yang terjadi adalah ada kalanya kapasitas kendaraan tidak mencukupi dan ada kalanya kendaraan tersebut dalam keadaan tidak terutilisasi secara penuh.

Selain itu, waktu untuk memuat barang sebelum dikirim juga relatif cukup lama karena permintaan konsumen yang fluktuatif, kadang permintaan konsumen sedikit sehingga barang harus dikumpulkan dulu sampai jumlah yang mencukupi untuk dikirim. Akibatnya perusahaan masih menemui kesulitan dalam mengirim barang pada konsumen tepat waktu.

Semua masalah tersebut harus dicari solusi yang optimal baik dari segi penggunaan kendaraan, waktu pengiriman, ketersediaan mobil dan pengemudi, efisiensi penggunaan bahan bakar dalam memenuhi semua pengiriman dan pada akhirnya diperlukan juga suatu sistem terintegrasi yang dapat mengatur sistem pengiriman perusahaan tersebut. Semua faktor-faktor yang mempengaruhi kendala pengiriman saling terkait satu sama lain sehingga dibutuhkan analisa dan perhitungan secara keseluruhan dalam proses penjadwalan dan pengiriman sehingga tujuan akhirnya memenuhi kepuasan pelanggan dan memberikan pelayanan yang terbaik dan optimal.

I.2 DIAGRAM KETERKAITAN MASALAH



Gambar 1.1 Diagram keterkaitan masalah

I.3 PERUMUSAN PERMASALAHAN

Permasalahan yang dijadikan fokus pada penelitian ini adalah bagaimana membuat suatu sistem informasi penjadwalan dan pengiriman untuk menentukan jadwal dan rute distribusi yang dapat meningkatkan efisiensi penjadwalan dan pengiriman cat di PT.X

I.4 TUJUAN PENELITIAN

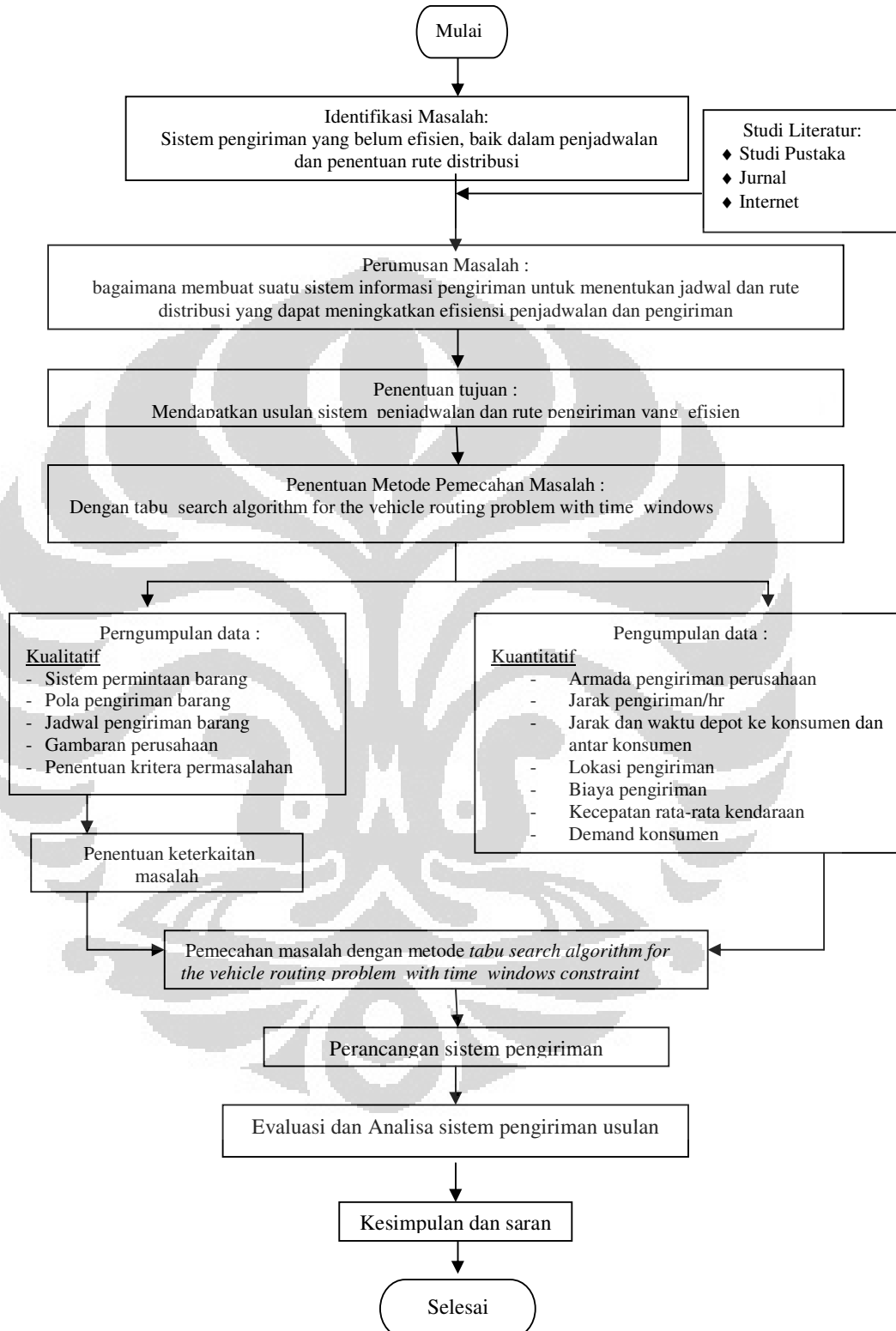
Tujuan yang ingin dicapai dalam penelitian ini adalah memperoleh sistem penjadwalan dan rute distribusi yang optimal untuk mengatasi kendala-kendala yang terjadi, serta dapat meningkatkan efisiensi penjadwalan dan pengiriman cat di PT.X dengan menggunakan algoritma *Tabu Search for the vehicle routing problem with time windows*.

I.5 PEMBATASAN MASALAH

Batasan masalah dari penelitian ini digunakan agar masalah yang diteliti lebih terarah dan terfokus sehingga penelitian dapat dilakukan sesuai dengan apa yang direncanakan dan memberikan hasil yang optimal. Batasan masalah yang digunakan pada penelitian ini adalah sebagai berikut :

- Penelitian dilakukan di gudang PT.X yang berlokasi di daerah Sunter
- Kondisi truk dianggap sama untuk semua armada dan jumlahnya tetap untuk satu periode

I.6 METODOLOGI PENELITIAN



Gambar 1.2 Metodologi penelitian

1.7 SISTEMATIKA PENULISAN

Sistematika yang dipergunakan dalam penulisan penelitian ini mengikuti aturan standar baku penulisan tesis. Penulisan tesis ini dibuat dalam lima bab yang memberikan gambaran sistematis sejak awal penelitian hingga tercapainya tujuan penelitian.

Dalam melakukan penelitian, pertama kali adalah menceritakan latar belakang pemilihan topik penelitian tesis ini. Selanjutnya membuat diagram keterkaitan masalah untuk melihat akar permasalahan sebagai suatu sistem yang memiliki sub sistem yang saling berhubungan. Diagram Keterkaitan Masalah di sini menjelaskan perlunya suatu sistem informasi pengiriman untuk menentukan jadwal dan rute distribusi yang dapat meningkatkan efisiensi penjadwalan dan pengiriman. Setelah menjelaskan latar belakang permasalahan, selanjutnya menguraikan tujuan penelitian dan manfaat yang diperoleh melalui studi penelitian dengan batasan-batasan penelitian agar fokus pada penelitian. Kemudian menjelaskan Metodologi Penelitian dan Sistematika Penulisan Laporan yang menggambarkan langkah – langkah proses penelitian agar mudah bagi pembaca dalam memahami hasil penelitian ini.

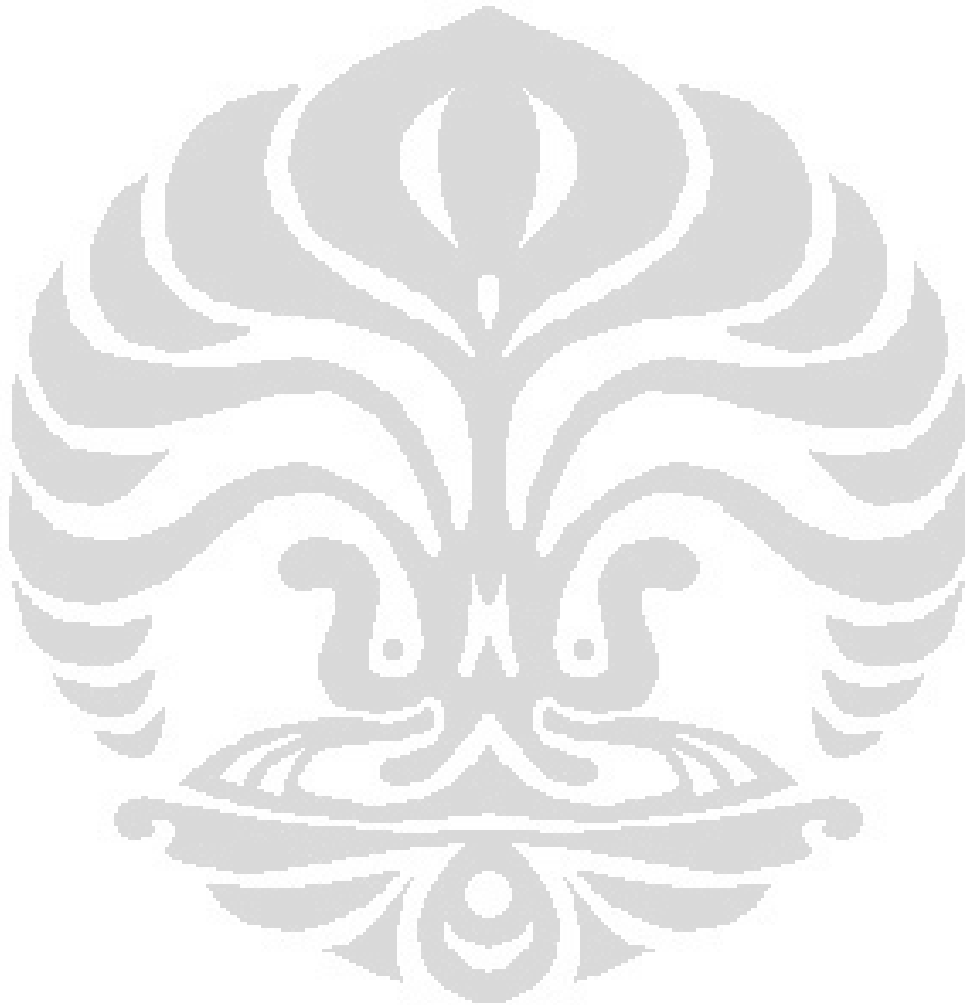
Bab II merupakan landasan teori yang digunakan dalam penelitian meliputi permasalahan penjadwalan pengiriman dan penyusunan rute distribusi. Secara umum pembahasan difokuskan pada VRP mulai dari definisi umum dan modelnya, teknik pencarian solusi melalui algoritma eksak heuristik dan metaheuristik. Dalam hal ini pembahasan mendalam diutamakan mengenai metode *tabu search* sebagai salah satu metaheuristik.

Proses pengumpulan dan pengolahan data dilakukan pada bab III. Pada bab ini berisi tentang keadaan penjadwalan dan pengiriman saat ini serta karakteristik data yang dibutuhkan baik data untuk penyusunan jadwal pengiriman maupun data untuk dijadikan parameter pencapaian tujuan.

Bab IV adalah pengolahan data dan analisis. Sistem pemesanan dan pengiriman diperoleh dengan melakukan penghitungan terhadap jumlah permintaan pelanggan, kapasitas mobil tangki, serta rata-rata konsumsi oleh pelanggan per hari. Setelah berhasil disusun jadwal pengiriman maka data tersebut akan dimasukkan dalam *software visual basic* dengan metode *tabu search*

guna memperoleh rute distribusi yang optimal menghasilkan jarak tempuh total terpendek. Analisa dilakukan untuk membandingkan proses pengiriman yang diterapkan sekarang dengan proses hasil penelitian.

Bab terakhir yaitu bab kesimpulan yang menyajikan beberapa hal penting sebagai kesimpulan atas seluruh penelitian yang dilakukan.



2.LANDASAN TEORI

2.1. VEHICLE ROUTING PROBLEMS

Logistik mempunyai pengaruh yang signifikan terhadap biaya dan keputusan suatu perusahaan, logistik juga berpengaruh untuk menghasilkan level pelayanan kepada konsumen yang berbeda-beda. Tujuan akhir manajemen logistik adalah mendapatkan sejumlah barang atau jasa yang tepat pada tempat dan waktu yang tepat, serta kondisi yang diinginkan dengan memberikan kontribusi terbesar bagi perusahaan¹.

Untuk mencapai tujuan akhir manajemen logistik, diperlukanlah suatu sistem distribusi produk yang :

- Memastikan bahwa produk yang tersedia pada waktu dan jumlah yang tepat sesuai permintaan konsumen
- Memiliki kualitas yang terjamin
- Memperhatikan tingkat keselamatan dalam pendistribusiannya.

Suatu perusahaan harus dapat mengoptimalkan sistem distribusinya agar dapat bersaing dengan perusahaan sejenis lainnya. Salah satu caranya adalah dengan pengoptimalan transportasi. Salah satu permasalahan dalam transportasi adalah *Vehicle Routing Problems* (VRP) yaitu merancang m set rute kendaraan dengan biaya rendah dimana tiap kendaraan berawal dan berakhir di depot, setiap konsumen hanya dilayani sekali oleh sebuah kendaraan, serta total permintaan yang dibawa tidak melebihi kapasitas kendaraan. Transportasi ini memberikan kontribusi biaya 1/3 sampai 2/3 dari total biaya distribusi¹.

Vehicle routing problems (VRP), pertama kali dikenalkan oleh Dantzig dan Ramser pada tahun 1959. VRP ini memegang peranan penting pada manajemen distribusi dan telah menjadi salah satu permasalahan dalam optimalisasi kombinasi yang dipelajari secara luas. VRP merupakan manajemen distribusi barang yang memperhatikan pelayanan, periode waktu tertentu, sekelompok konsumen dengan sejumlah kendaraan yang berlokasi pada satu atau lebih depot yang dijalankan oleh sekelompok pengemudi, menggunakan *road*

network yang sesuai. Solusi dari sebuah VRP yaitu menentukan sejumlah rute, yang masing-masing dilayani oleh suatu kendaraan yang berasal dan berakhir pada depotnya, sehingga kebutuhan pelanggan terpenuhi, semua permasalahan operasional terselesaikan dan biaya transportasi secara umum diminimalkan.

Karakteristik konsumen dalam VRP:

- Menempatkan *road graph* dimana konsumen berada
- Adanya *demand* dalam berbagai tipe dan harus diantarkan ke tempat konsumen
- Terdapat periode waktu (*time window*) dimana konsumen dapat dilayani
- Waktu yang dibutuhkan untuk mengantarkan barang ke lokasi konsumen (*loading time*), hal tersebut dapat berhubungan dengan jenis kendaraan
- Sekelompok kendaraan tersedia digunakan untuk melayani konsumen

Terdapat empat tujuan umum VRP⁴, 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
- Meminimalkan penalti akibat *service* yang kurang memuaskan dari konsumen

Menurut Toth dan Vigo (2002) ditemukan variasi permasalahan utama VRP yaitu:

- Setiap kendaraan memiliki kapasitas yang terbatas (*Capacitated VRP-CVRP*)
- Setiap konsumen harus dikirim barang dalam waktu tertentu (*VRP with time windows-VRPTW*)
- Vendor menggunakan banyak depot untuk mengirim konsumen (*Multiple Depot VRP – MDVRP*)
- Konsumen dapat mengembalikan barang-barang kembali ke depot (*VRP with pick up and delivering – VRPPD*)

⁴ Toth and Vigo, 2002

- Konsumen dilayani dengan menggunakan kendaraan yang berbeda-beda (*Split Delivery VRP – SDVRP*)
- Beberapa besaran (seperti jumlah konsumen, jumlah permintaan, waktu melayani dan waktu perjalanan)
- Pengiriman dilakukan dalam periode waktu tertentu (*Periodic VRP-PVRP*)

2.2 VEHICLE ROUTING AND SCHEDULING

Vehicle routing and scheduling merupakan perluasan dari *vehicle routing problem*. Beberapa batasan yang realistis yang termasuk didalamnya adalah sebagai berikut¹ :

1. Dalam setiap titik pemberhentian, ada sejumlah volume yang diambil dan dikirim.
2. Beragam kendaraan kemungkinan digunakan, disebabkan karena beragam batasan kapasitas pengangkutan.
3. Maksimum total waktu kerja operator kendaraan untuk melakukan pengiriman sebelum periode istirahat selama kurang lebih 8 jam.
4. Titik pemberhentian (konsumen) hanya memperbolehkan pengiriman dan/atau pengambilan produk pada waktu tertentu (disebut : *time windows*).
5. Pengambilan hanya boleh dilakukan setelah dilakukan pengiriman.
6. Operator kendaraan diperbolehkan istirahat atau makan siang pada waktu tertentu.

Beberapa batasan diatas menambah kompleksitas masalah *routing* ini dan mempersulit kita dalam pemilihan solusi yang paling optimal. Solusi yang paling optimal dapat diperoleh dengan cara menerapkan beberapa panduan untuk menghasilkan *routing* dan *scheduling* yang baik atau beberapa prosedur *logical heuristic* dengan pertimbangan kendaraan memulai perjalanan dari pabrik (depot), menuju ke beberapa titik pemberhentian (stop) untuk melakukan pengiriman, dan kembali ke pabrik (depot) pada hari yang sama.

2.2.1. Methods for Routing and Scheduling

Permasalahan untuk mendapatkan hasil solusi yang optimal dari pemecahan VRP (*Vehicle Routing Problems*) menjadi bertambah jika terdapat penambahan kendala (*constraint*) pada kasus yang harus diselesaikan. Kendala-kendala tersebut antara lain batasan waktu (*time window*), jenis kendaraan angkut yang berbeda-beda kapasitas angkutnya, total waktu maksimum operator kendaraan melakukan pengiriman, hambatan-hambatan yang di perjalanan, waktu istirahat operator kendaraan ketika melakukan pengiriman dan lain sebagainya. Dari banyak pendekatan untuk memecahkan masalah VRP terdapat dua metode yang paling umum digunakan yaitu *sweep method* dan *savings method*. Kedua metode tersebut merupakan tehnik pemecahan VRP secara *heuristic*.

2.3 PENYELESAIAN VEHICLE ROUTING PROBLEMS

Pada dasarnya , terdapat 3 macam penyelesaian VRP :

2.3.1. Solusi eksak

Pada solusi eksak dilakukan pendekatan dengan menghitung setiap solusi yang mungkin sampai satu terbaik dapat diperoleh. *Branch and bound* dan *branch and cut* merupakan contoh dari penyelesaian eksak.

2.3.2. Heuristik

Metode Heuristik memberikan suatu cara untuk menyelesaikan permasalahan optimasi yang lebih sulit dan dengan kualitas dan waktu penyelesaian yang lebih cepat daripada solusi eksak. Contoh metode heuristik antara lain: Saving Based,

Matching based, Multiroute improvement *heuristic*, dll.

2.3.2.1. Sweep Method

Sweep method adalah metode yang sederhana dalam perhitungannya, bahkan untuk memecahkan masalah dengan ukuran yang cukup besar. Keakuratan metode ini rata-rata kesalahan perhitungannya adalah sebesar 10 persen⁵

⁵ Ronald h Ballou dan Yogesh K Agarwal.” A Performance Comparison of Several Popular Algorithms for Vehicle Routing and Scheduling”. *Journal of Business Logistics* 9, no.1.1998: 51-65

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 pengelompokkan, yang mana pengelompokkan awal dilakukan dengan menggabungkan perhentian-perhentian yang setiap kelompok mengakomodasi volume masing-masing perhentian. Volume total perhentian dari satu *cluster* mungkin akan melebihi kapasitas kendaraan karenanya beberapa perhentian dipindahkan ke kendaraan yang kapasitasnya belum penuh. Relokasi seperti ini dilakukan dengan menggunakan metode transportasi *linear programming*.

Yang menarik dari metode ini adalah perhentian dikelompokkan berdasarkan kedekatan dan logikanya akan menghasilkan jarak total yang rendah. Ketika volume *cluster* melebihi kapasitas kendaraan realokasi perhentian ke *cluster* lain dilakukan untuk mendapatkan keseimbangan optimum di antara *cluster*. Karena pengelompokkan terpisah dari pengurutan (*sequencing*), kendala waktu tidak dapat diselesaikan menggunakan metode ini.

2.3.3. METAHEURISTIK

Metaheuristik, adalah suatu metode untuk melakukan eksplorasi yang lebih dalam pada daerah yang menjanjikan dari ruang solusi yang ada. Kualitas solusi yang dihasilkan dari metode ini jauh lebih baik daripada yang didapat heuristik klasik. Contoh metaheuristik adalah *genetic algorithm*, *simulated annealing*, *tabu search*, *ant colony system* dsb.

2.3.4. Tabu Search Meta-Heuristic

2.3.4.1. Pengertian Umum

Kata tabu atau *taboo* berasal dari bahasa Tongan yaitu salah satu bahasa Polynesia yang digunakan oleh penduduk pribumi dari pulau Tonga untuk mengungkapkan sesuatu yang tidak boleh disentuh karena merupakan sesuatu

yang keramat⁶. Menurut kamus Webster juga berarti “ sebuah larangan yang diturunkan secara sosial sebagai mekanisme protektif ” atau sesuatu yang dilarang sebab mengandung resiko. Resiko yang dihindari dalam hal ini adalah hal yang kontra produktif.

Lebih rinci lagi, tabu search berdasarkan premis yang bersifat *problem solving* atau memecahkan masalah, untuk dikualifikasikan cerdas, harus menyertakan *adaptive memory* dan *responsive exploration*. Fitur *adaptive memory* dan *responsive exploration* dalam TS membuat implementasi prosedur yang dapat melakukan pencarian berbagai solusi secara ekonomis dan efektif. Karena pilihan-pilihan lokal dipandu dengan informasi yang dikumpulkan selama pencarian tabu search sangat berbeda dibandingkan dengan pola tanpa memori (*memoriless*) yang sangat bergantung pada proses semi acak yang mengimplementasikan sebuah bentuk sampling. Contoh dari metode tanpa memori adalah *heuristic greedy*, dan pendekatan *annealing* dan *genetic* terinspirasi oleh metafor fisika dan biologi. *Adaptive memory* juga berbeda dengan desain memori yang kaku pada algoritma *branch and bound*.

Upaya untuk melakukan eksplorasi responsif dalam *tabu search*, baik itu implementasi *deterministic* atau *probabilistic*, berasal dari pemahaman bahwa suatu pilihan strategi yang buruk dapat menghasilkan informasi yang lebih banyak dibandingkan suatu pilihan acak yang baik. Dalam suatu sistem yang menggunakan memori sebuah pilihan buruk yang berdasarkan strategi dapat memberikan petunjuk yang bermanfaat tentang bagaimana strategi tersebut dapat diubah menjadi lebih baik.

Responsive exploration mengintegrasikan prinsip-prinsip dasar dari *intelligence search*, seperti memanfaatkan fitur solusi yang baik saat menjelajahi area baru yang menjanjikan. Tabu search memperhatikan pencarian cara baru yang lebih efektif dalam memperoleh keuntungan dari mekanisme yang berhubungan dengan *adaptive memory* dan *responsive exploration*. Pengembangan pola baru dan kombinasi-kombinasi strategi membuat tabu suatu area yang luas untuk penelitian dan studi empiris.

⁶ Glover, Fred and Manuel Laguna, 1997, *Tabu search*

2.3.4.2. *Penggunaan Memori*

Struktur memori dlm tabu search beroperasi atas referensi empat dimensi utama yaitu referensi *frequency*, *quality*, dan *influence*. Dimensi *quality* mengacu pada kemampuan untuk membedakan kelebihan dari solusi-solusi yang dikunjungi selama pencarian. Pada konteks tersebut, memori dapat digunakan untuk mengidentifikasi elemen-elemen yang umum tentang solusi yang baik atau tentang jalan yang membawa kepada solusi tersebut. Pada prakteknya, *quality* menjadi landasan untuk pembelajaran berbasis intensif, dimana penghargaan diberikan untuk meningkatkan tindakan yang menghasilkan solusi yang baik, dan penalti diberikan untuk menghindari tindakan-tindakan yang menyebabkan solusi yang buruk. Fleksibilitas dari struktur memori tersebut memungkinkan pencarian untuk diarahkan dalam suatu lingkungan *multi objektif*, dimana kebaikan dari suatu arah pencarian tertentu dapat ditentukan oleh lebih dari satu fungsi. Konsep *quality* dari tabu search lebih luas dibandingkan dengan metode optimisasi standard.

Memori yang digunakan dalam tabu search bersifat eksplisit dan juga atributif. Memori eksplisit merekam seluruh solusi, terutama terdiri dari solusi penting yang dikunjungi selama pencarian. Suatu perluasan dari memori ini merekam solusi penting yang sangat atraktif namun merupakan solusi tetangga yang belum tereksplorasi.

Sebagai alternatif, tabu search menggunakan memori atributif untuk tujuan sebagai panduan. Jenis memori ini merekam informasi tentang atribut-atribut solusi yang mengalami perubahan dalam proses perpindahan dari satu solusi ke solusi yang lain. Sebagai contoh, dalam suatu grafik atau jaringan, atribut dapat terdiri dari nodes atau arah yang ditambahkan, dihilangkan atau direposisi dengan mekanisme perpindahan. Dalam penjadwalan produksi, daftar digunakan sebagai atribut untuk mencegah atau mendorong metode untuk mengikuti arah pencarian tertentu

2.3.4.3. *Intensifikasi dan Diversifikasi*

Dua komponen yang sangat penting dari tabu search adalah strategi intensifikasi dan diversifikasi. Strategi intensifikasi berdasarkan modifikasi

aturan-aturan pilihan untuk memacu kombinasi pergerakan dan fitur-fitur solusi yang terbukti baik. Ini juga berarti mulainya pencarian daerah yang menarik secara lebih menyeluruh. Karena solusi elit harus dicatat untuk mencari solusi-solusi tetangga, memori eksplisit sangat berhubungan dengan implementasi dari strategi intensifikasi. Perbedaan utama antara intensifikasi dan diversifikasi adalah bahwa selama masa intensifikasi pencarian difokuskan pada pemeriksaan untuk solusi elit.

Strategi intensifikasi membutuhkan cara untuk mengidentifikasi suatu set solusi elit sebagai dasar untuk menggabungkan atribut-atribut yang baik menjadi solusi yang baru. Keanggotaan dalam suatu set elit sering ditentukan dengan menentukan ambang batas yang dihubungkan dengan nilai fungsi objektif dari solusi terbaik yang ditemukan selama pencarian.

2.3.4.4. *Tabu Search pada VRP*

Tabu search adalah salah satu metode yang tergabung dalam satu kelas yang disebut *meta-heuristic*⁷. Metode *Tabu search* ini terbukti sukses dalam memecahkan permasalahan kombinatorial terkait dengan masalah optimasi. Dasar dari TS *meta-heuristic* adalah dengan menggunakan strategi pengawalan yang agresif untuk memotong prosedur pencarian lokal untuk membawa keluar eksplorasi dari himpunan solusi dalam rangka menghindari keterjebakan dalam *local optima*. Ketika *local optima* ditemui, strategi agresif bergerak ke solusi terbaik di setiap tetangga walaupun mungkin akan mengakibatkan penurunan dalam nilai tujuan. Untuk menghindari pencarian ke tempat yang baru saja diperoleh, TS menggunakan struktur memori untuk menyimpan atribut dari solusi yang diterima yang baru saja ditemui dalam *tabu list*. Atribut yang disimpan dalam *tabu list* disebut *tabu-active*, dan solusi-solusi yang memiliki elemen *tabu active* dikatakan sebagai *tabu*. Sebuah atribut tetap *tabu active* selama durasi tt , dikenal sebagai *tabu tenure* sebelum ini dibuat tidak *tabu active*. Algoritma TS melanjutkan pencariannya sampai iterasi tertentu sebelum ini diakhiri.

⁷ Osman IH (1995), An introduction to meta-heuristic. In: Lawrence M and Wilson C (eds). *Operational Research Society Tutorial*. Operational Research Society : Birmingham, pp 92-122

TS meta-*heuristic* membutuhkan :

- Solusi awal
- Mekanisme pembentukan solusi tetangga
- Data management structure
- Set komponen untuk algoritma TS

2.3.4.4.1. *Solusi Awal*

Solusi awal yang digunakan untuk algoritma TS adalah solusi dari kelas yang lebih rendah dalam hal ini dapat melalui pendekatan eksak ataupun metode heuristik. Prosedur VRP digunakan untuk mendapatkan solusi awal secara cepat dan selanjutnya diperbaiki menggunakan algoritma TS. Untuk setiap kendaraan tipe t , beberapa solusi dihasilkan. Kemudian dipilih solusi awal terbaik dengan mempertimbangkan semua jenis kendaraan yang ada. Kendaraan yang berbeda-beda diatur untuk sekelompok rute terbaik dalam solusi awal untuk menurunkan biaya tetap total dan biaya variabel.

2.3.4.4.2. *Mekanisme Pembentukan Solusi Tetangga*

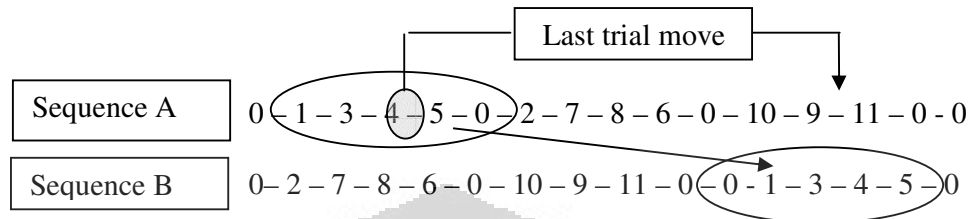
Anggap $S = \{R_1, \dots, R_p, \dots, R_q, \dots, R_v\}$ adalah merupakan solusi MFVRP dimana v adalah jumlah total kendaraan dari seluruh jenis. Mekanisme pembentukan solusi tetangga menentukan suatu set operator yang dapat diaplikasikan pada S untuk menghasilkan *move* ke solusi S' yang lain sebagai tetangga S , $N(S)$. Untuk implementasinya diadopsi mekanisme λ -interchange oleh Osman untuk masalah *routing* dan *grouping*. Misalkan sepasang rute (R_p, R_q) dalam S , λ -interchange yang dapat digunakan adalah:

- 1-interchange mechanism

Proses pada mekanisme ini ada dua yaitu proses pindah (*shift*) dan proses tukar (*exchange*). Proses pindah berdasarkan operator (1,0) dan (0,1), sedangkan proses tukar berdasarkan operator (1,1). Operator pindah (1,0) memindahkan satu konsumen dari rute R_p ke R_q sedangkan operator pindah (0,1) memindahkan satu konsumen dari rute R_q ke R_p . Operator

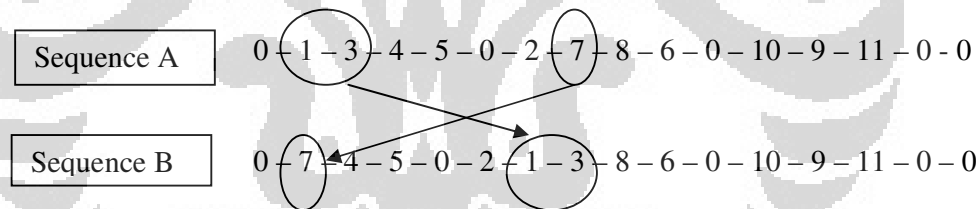
(1,1) menukar masing-masing satu konsumen dari rute R_p ke R_q secara serentak.

Gambar 2.1. *last trial move* pada *1-interchange mechanism*



- 2-Consecutive node interchange mechanism

Mekanisme ini menggunakan semua operator pada *1-interchange mechanism* ditambah operator pindah (2,0) dan (0,2) dan operator tukar (2,1), (1,2) dan (2,2). Cara kerja operator-operator tersebut sama dengan pada *1-interchange mechanism* hanya saja jumlah konsumen yang dipindahkan atau dipertukarkan berbeda. Pada operator tambahan tadi yang berpindah atau dipertukarkan sebanyak dua konsumen, yang mana dua konsumen tersebut berurutan atau tidak berurutan.



Gambar 2.2. *insert move* pada *2-Consecutive node interchange mechanism*

2.3.4.4.3. *Data Management Structure*

Untuk meningkatkan kecepatan heuristik dikembangkan sebuah struktur management data untuk merekam jarak masing-masing *route delivery*, permintaan, dan kendaraan yang dialokasikan. Ketika sebuah solusi tetangga dibentuk dengan satu *move* tetangga, hanya dua dari rute yang terlibat yang diitung ulang.

2.3.4.4.4. *Komponen Tabu Search*

- *Tabu list*

Tabu list adalah memori jangka pendek yang digunakan untuk menyimpan beberapa atribut dari *move* yang sedang dilakukan untuk menentukan status tabu di *move* selanjutnya.

- *Tabu restriction*

Tabu restriction adalah kriteria untuk menentukan status *move* yang tabu. Ada beberapa arahan yang dapat digunakan untuk membuat *tabu restriction* ini. Misalnya untuk mekanisme 2-consecutive node interchange, *move* dikatakan tabu jika konsumen i dan j dan konsumen l dan s kembali ke rute semulanya.

- *Aspiration criteria*

Aspiration criteria mengesampingkan status tabu dari sebuah *move* yang merupakan tabu-active dan membuat *move* tersebut diizinkan jika *move* tersebut menghasilkan solusi terbaik baru.

- *Stopping rule*

Merupakan aturan atau kriteria untuk menghentikan seluruh proses *tabu search*.

- Skema *tabu tenure*

Tabu tenure adalah durasi suatu atribut dikatakan *tabu active*, setelah melewati nilai *tabu tenure* ini maka atribut tersebut tidak lagi *tabu active*. Skema yang digunakan untuk melakukan kontrol *tabu tenure* adalah :

- Skema *Fixed TS* (F-tabu)

Merupakan cara yang paling pertama dikenal dan digunakan yaitu menerapkan nilai tt selama proses.

- Skema *Robust TS* (Rb-tabu)

Menggunakan nilai tt secara acak pada kisaran tertentu. Selama pencarian nilai tt secara periodik berubah setelah melakukan sebanyak m iterasi. Nilai m pun diambil secara acak.

○ Skema *Periodic TS* (P-tabu)

Menerapkan nilai tt yang berubah secara periodic dari nilai yang kecil, sedang hingga besar. Perubahan tersebut dilakukan setelah melakukan iterasi sebanyak m iterasi.

○ Skema *Reversed deterministic TS* (Rd-tabu)

Merupakan skema baru yang diusulkan dengan mekanisme mengubah dan membalikkan nilai tt selama pencarian. Skema ini diawali dari penggunaan beberapa nilai tt yang tetap yang sebelumnya telah ditentukan, dimana $tt = n/p$ dan nilai p diambil dari kisaran 2 hingga 7. Skema Rd-tabu secara dinamis mengubah nilai tt selama proses pencarian. Nilai tt diambil dari perhitungan dengan membagi jumlah konsumen (n) dengan suatu nilai p , setelah iterasi sejumlah m , yaitu $tt = n/p$ dan $m = T_{itr}/n \times p$, T_{itr} adalah jumlah total iterasi yang harus diselesaikan selama pencarian. Pada awal pencarian, nilai p adalah 9 lalu diturunkan satu setelah melakukan iterasi sejumlah m . Setelah mencapai nilai 1, lalu nilai p diulang lagi dari 9, proses berlanjut hingga pencarian dihentikan. Dengan cara ini kita dapat melakukan kontrol terhadap proses diversifikasi dengan memberikan nilai p yang kecil, dan melakukan kontrol terhadap proses intensifikasi dengan memberikan nilai p yang besar.

2.3.4.4.5. *Prosedur umum TS- MFVRP*

1. Menentukan solusi awal

- Solusi awal diperoleh dari hasil pengolahan data dengan algoritma *saving based*.
- Tentukan $S_{best} = S$, $C_{itr} = 0$ (*current iteration counter*)

2. Inisialisasi *Tabu search*

- Tentukan skema tabu tenure serta nilai untuk tiap parameternya.
- Tentukan jumlah total iterasi T_{itr} , dan $B_{itr} = 0$ (*best iteration counter*)
- Matrik tabu list diset nol

3. Lakukan iterasi

- Lakukan *move* untuk membuat solusi tetangga, dan pilih solusi S' terbaik yang diizinkan dari daftar kandidat.
- Tentukan solusi saat ini (*current solution*) S menjadi S' , $C_{itr} = C_{itr} + 1$
- Lakukan heuristik *2-opt* untuk memperbaiki urutan dalam rute tersebut

4. Perbaharui skema *tabu search*

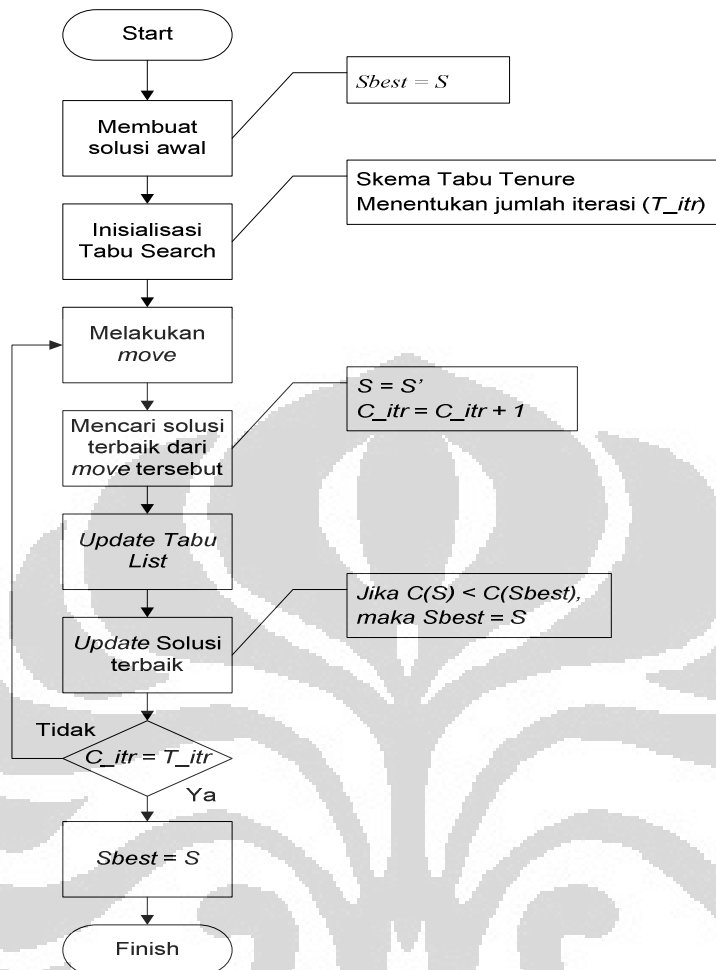
- Perbaharui daftar dalam *tabu list*
- Perbaharui komponen skema *tabu tenure* jika diperlukan

5. Perbaharui solusi baru

- Jika $C(S) < C(S_{best})$, maka tentukan $S_{best} = S$ dan $B_{itr} = C_{itr}$

6. Penghentian

- Jika $C_{itr} = T_{itr}$ maka , pencarian dihentikan, laporkan S_{best} dan B_{itr} , jika tidak kembali ke langkah 3



Gambar.2.3. Flowchart pengerjaan *Tabu Search* pada VRP

3.PENGUMPULAN DATA

3.1 PROFIL PERUSAHAAN

PT.X merupakan perusahaan cat internasional yang didirikan pada tahun 1976 dan tersebar di 35 negara termasuk di Indonesia. PT. X merupakan perusahaan cat asing dengan kantor induknya berada di London, Inggris. Di Indonesia kantor pusatnya terletak di Plaza DM, Sudirman yang merupakan kantor *marketing* dan *finance* dengan jumlah karyawan ± 100 orang, sedangkan untuk gudang dan bagian pengiriman ke toko-toko di seluruh jabodetabek, terletak di Sunter Agung Timur dengan jumlah pegawai ± 85 orang. Produk PT.X antara lain adalah cat tembok luar, dalam, many, thinner,dll.

3.2 PENGUMPULAN DATA

Adapun data yang harus dikumpulkan adalah sebagai berikut :

3.2.1 Armada Pengiriman

Jumlah Truk	:	9 Buah		
Kapasitas muatan Truk	:	7 truk: kapasitas 2000 kg		
		2 truk: kapasitas 500 kg		
Jumlah ban	:	Tiap truk memakai 4 ban		
Bahan Bakar	:	Solar		
Ratio bahan bakar	:	Truk kapasitas 2000 kg : 1:5		
		500 kg : 1: 10		
Harga Solar	:	Rp. 4300,-		
Biaya Perawatan Truk:				
Ganti Oli	:	Setiap 5000 km	:	Rp 450.000,-
Filter Oli	:	Setiap 10000 km	:	Rp 400.000,-
Service Besar	:	Setiap 100000 km	:	Rp 4500.000,-

3.2.2 Biaya Pengiriman

Biaya pengiriman diperoleh dengan memperhitungkan jumlah bahan bakar yang digunakan, biaya ban truk, biaya pemeliharaan dan biaya supir. Berikut ini adalah rincian biaya pengiriman per kilometer (km):

- Biaya bahan bakar

Kebutuhan bahan bakar per km untuk truk kapasitas 2000 kg adalah 0.2L/km sedangkan untuk kapasitas 500 kg adalah 0.1 L/km

Harga bahan bakar solar per liter adalah Rp 4300,-

Jadi biaya bahan bakar untuk truk kapasitas 2000 kg = 0.2 L/km x Rp 4300 = Rp 860, sedangkan untuk kapasitas 500 kg = 0.1L/km x Rp 4300 = Rp 430

- Biaya ban truk

1 ban truk kapasitas 2000 kg dan 500 kg maksimal dapat digunakan untuk 20.000 km. Harga 1 ban adalah Rp 853.000,-

Biaya ban/ km untuk 1 truk = $4x(Rp 853.000/20.000 km) = Rp 170.6/ km$

- Biaya supir = Rp 1000/km

- Biaya perawatan = Rp 300/km

Jadi total biaya pengiriman untuk truk kapasitas 2000 kg adalah Rp 2330.6/km dan truk kapasitas 500 kg adalah Rp 1900.6/km

3.2.3 Konsumen, Lokasi dan Rute yang Dilalui Truk Menuju Konsumen

Pada tabel berikut adalah konsumen dan lokasi yang dikunjungi oleh truk

Tabel 3.1 Lokasi Konsumen

Cust ID	Customer name	Address	City
1	BERKAH JAYA , TOKO	JL.CIBENING BINTARA JAYA	BEKASI
2	SANTOSA , TOKO	JL.JATIWARINGIN I/9B	BEKASI
3	MEGA INDAH	JL.TRIDAYA INDAH I NO.28	BEKASI
4	YASMINE JAYA. PD	JL. WIJAYA KUSUMA RAYA NO.R6	BOGOR
5	WIRA SAKTI	JL. RAYA TLAJUNG UDIK KM.2.3	BOGOR
6	JAYA AGUNG	JL. BOJONGSARI LAMA NO.34	BOGOR
7	BUMI MAKMUR	JL. CINANGKA RAYA NO. 138	DEPOK
8	JAYA BARU	JL.RAYA KEBON JERUK 14	JAK- BAR
9	SINAR BANGUNAN	JL.DURI RAYA NO.20	JAK- BAR
10	PUTRA BUANA	JL.KEMBANGAN RAYA NO.1	JAK- BAR
11	KALI MAS	JL.KALILIO 25	JAK-PUS
12	MUTIARA , TOKO	JL.GUNUNG SAHARI XI NO.30	JAK-PUS

13	RODA SELATAN	JL.BENDUNGAN HILIR	JAK-PUS
14	MADJU	JL.PANGLIMA POLIM RAYA	JAK- SEL
15	HIDUP BARU	JL.PANGLIMA POLIM RAYA	JAK- SEL
16	CATUR TIRTA MURNI ,PT.	JL.BAMBU KUNING A1/287 A	JAK- SEL
17	ANUGRAH JAYA	JL.RAYA PULO GEBANG 101	JAK- TIM
18	INTI ,PD.	JL. UTAN KAYU NO.54-61	JAK- TIM
19	SURYA JAYA	DANAU AGUNG UTARA BLOK G	JAK-UT
20	BUDI JAYA	JL. BOULEVARD QA 1 NO.19	JAK-UT
21	BANGUN JAYA I	JL.DANAU AGUNG II NO.13	JAK-UT
22	SINAR JAYA	JL.MERDEKA NO.120/250 KARAWACI	TANGERANG
23	SUMBER WARNA	RUKO VILLA MELATI MAS	TANGERANG
24	HARAPAN MAKMUR	JL. RAYA TAMAN ADIYASA KM 2	TANGERANG

Dari data alamat konsumen dan rute yang dilalui oleh truk pengiriman, maka dicarilah jarak depot-konsumen dan konsumen-konsumen sehingga dihasilkan matrik jarak dan matriks waktu yang disajikan pada Lampiran. Berikut adalah data permintaan pelanggan dan data pengiriman PT.X dalam satu hari pengiriman.

Tabel 3.2 Permintaan 1 hari pelanggan PT.X

Konsumen	Permintaan
Putra buana (PB)	1000
Sinar bangunan (SB)	750
Madju (M)	500
Santosa toko (SANT.TK)	1200
wira sakti (WS)	200
Sinar Jaya (SNR JY)	300
Kalimas (KM)	400

Tabel 3.3 Pengiriman 1 hari PT.X

trip	titik 1	titik 2	titik 3	demand (kg)	truk	jarak (km)	waktu(jam)	biaya
1	PB	SB		1750	B 2345 AE	78.5	2	182952.1
2	KM	M		900	B 2772 AA	114.7	2.8	267319.8
3	SANT.TK	SNR JY	WS	1700	B 3993 AF	226.9	5.7	528813.1
					Total	420.1	10.5	979085.1

3.2.4 Depot dan Informasi Lainnya

Depot mulai buka pada pukul 8.00 WIB sampai dengan pukul 17.00 WIB, yang terletak di Sunter, Danau Agung Timur-Jakarta Utara.

Man Power /Truk : 2 orang (1 sopir, 1 kernet)

Hari Kerja : Senin – Jumat
 Jam Kerja : Senin – jumat : Jam 8.00 – 17.00

3.2.5 Waktu

Time windows

Time windows adalah waktu tertentu untuk menerima pengiriman. Waktu ini didefinisikan sebagai waktu awal dan waktu akhir pelayanan di pelanggan tersebut. Pada umumnya time windows pelanggan PT.X adalah toko-toko bangunan yang dimulai dari jam 8.00 sampai 17.00.

Waktu Tempuh

Kecepatan mobil rata-rata adalah 40 km/jam (data lihat lampiran), dengan mempertimbangkan jarak dan waktu aktual dari data primer, serta karakteristik dari jalan yang dilalui.

Cara menghitung waktu tempuh = jarak/kecepatan

Misal : Jarak tempuh = 95.5 km
 Waktu tempuh = $95.5 / 40 = 2.38$ Jam

Waktu Loading

Loading waktunya 2 jam dilakukan dari jam 8⁰⁰ sampai 10⁰⁰

Waktu Istirahat

Istirahat untuk makan siang dan sholat adalah 1 jam. Waktunya disesuaikan oleh masing-masing sopir dan kernet.

Waktu Unloading

Waktu unloading terdiri dari 2 unsur yaitu waktu penurunan barang di toko dan pengurusan administrasi. Waktu penurunan untuk tiap dus = 1 menit/dus

Jadi waktu unloading 15 menit/toko

Waktu Delivery

Waktu delivery ke tiap konsumen setiap harinya : jam 10. 00 – 14.00

4.PENGOLAHAN DATA DAN ANALISIS

4.1. RUTE DISTRIBUSI

Rute distribusi diperoleh dengan menggunakan metode *sweep* yang termasuk dalam *cluster methods*. Pengerjaannya dilakukan secara manual dengan melakukan pengelompokkan pelanggan ke zona lokasi tertentu. Selanjutnya ditentukan beberapa titik pemberhentian yaitu pengiriman ke beberapa pelanggan sampai kapasitas truk yang mengangkut sejumlah pengiriman ke titik-titik yang bersangkutan terpenuhi semaksimal mungkin. Dari rute distribusi yang dihasilkan dengan metode ini, nantinya didapatkan jumlah trip, jarak yang ditempuh truk serta biaya pengirimannya. Rute distribusi ini merupakan solusi awal untuk menyelesaikan permasalahan VRP ini dengan menggunakan metode *tabu search*. Setelah dilakukan pengolahan data terhadap solusi awal dengan algoritma *tabu search* selanjutnya solusi yang didapat dengan algoritma *tabu search* ini juga dibandingkan dengan keadaan yang diterapkan di PT.X sekarang ini.

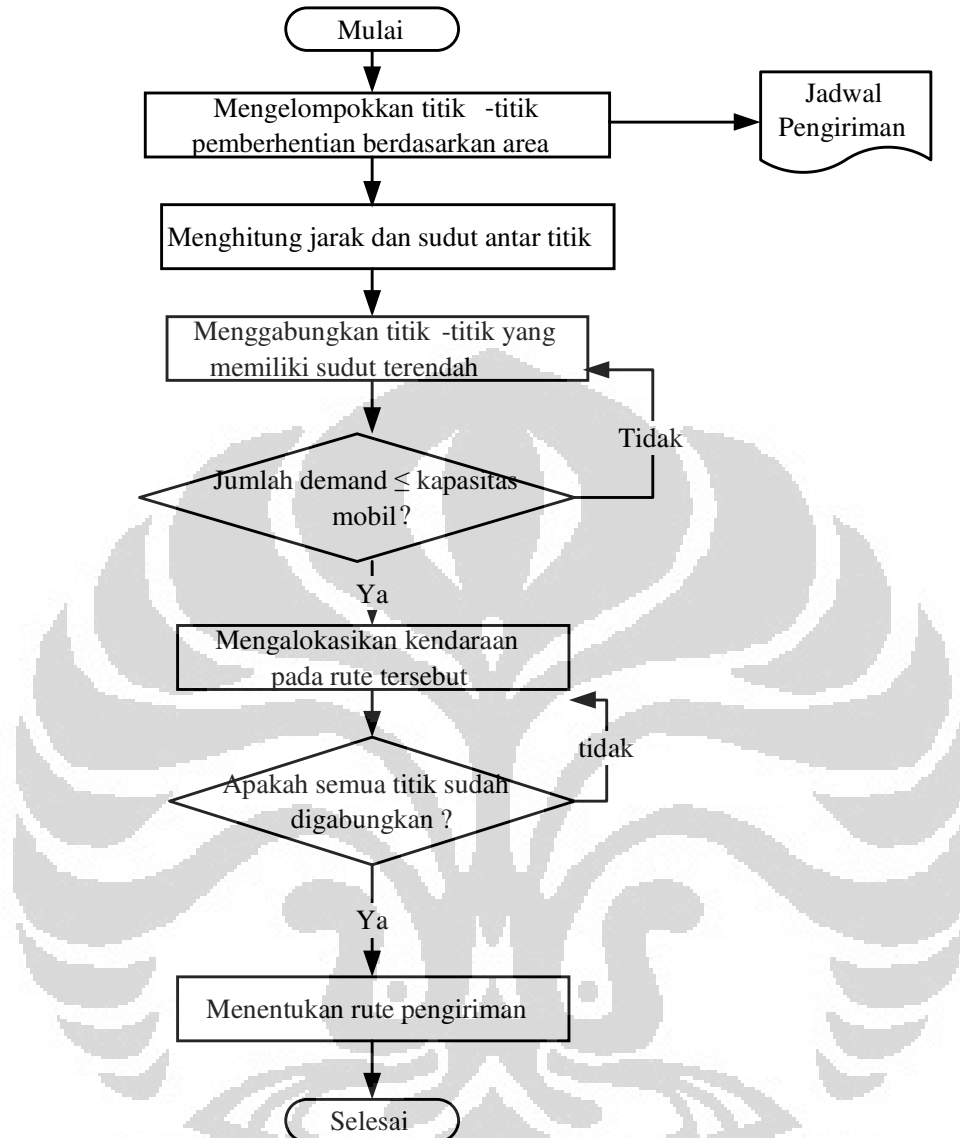
4.1.1. Pengerjaan Solusi Awal

4.1.1.1. *Input*

Input data yang diperlukan untuk pengolahan data awal dengan metode *sweep* ini adalah jadwal pengiriman, lokasi depot dan pelanggan, jarak depot ke tiap pelanggan, waktu dari depot ke pelanggan, kapasitas dan jumlah truk, serta *time windows* pelanggan.

4.1.1.2. *Langkah Pengerjaan*

Berikut ini adalah diagram alir langkah pengerjaan solusi awal dengan *sweep method*. Langkah pengerjaan solusi awal ini bertujuan untuk memperoleh rute pengiriman awal berdasarkan jadwal pengiriman yang sebelumnya sudah disusun.



Gambar 4.1. Langkah pengerjaan solusi awal dengan *sweep method*

4.1.1.3. Output

Hasil dari tahap pengerjaan awal ini berupa rute distribusi awal yang menjadi solusi awal bagi tahap pengerjaan selanjutnya yaitu menggunakan algoritma *tabu search*. Rute distribusi tersebut berupa trip-trip yang ada dalam setiap harinya. Dari trip yang ada tersebut didapatkan total jarak tempuh truk serta biaya pengirimannya. Tabel berikut adalah output rute pengiriman hari-1 pada solusi awal.

Tabel 4.1. Ouput rute hari-1 pada solusi awal

Trip	titik 1	titik 2	titik 3	demand (kg)	kendaraan	jarak (km)	waktu (jam)	biaya
1	PB	KM		1400	B 4959 AH	105	2.6	244713
2	SANT.TK	SB		1950	B 4532 AC	107.3	2.7	250073.4
3	M	WS	SNR JY	1000	B 3993 AF	257	6.5	598964.2
Total						469.3	11.8	1093751

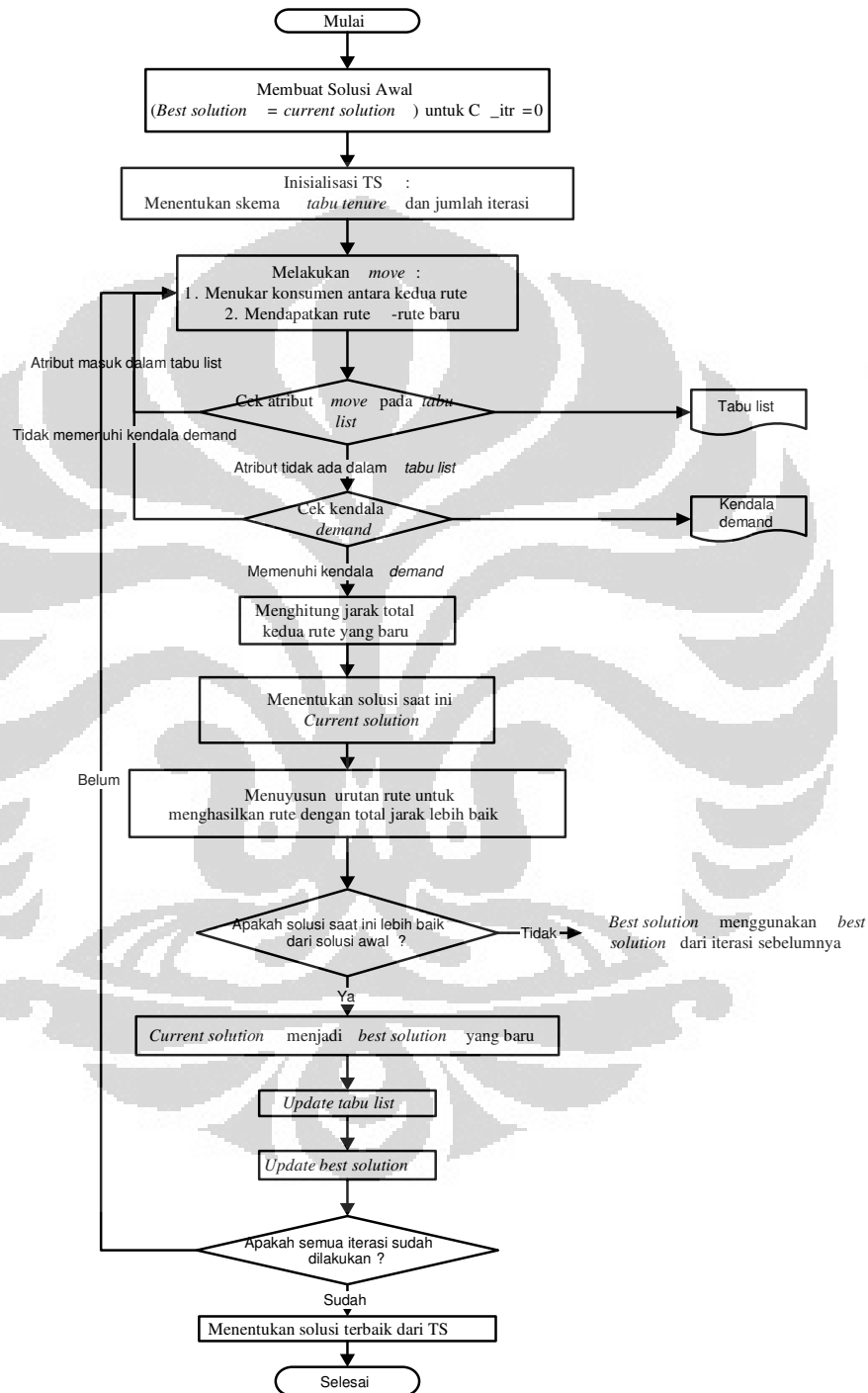
Angka–angka yang ada dalam urutan masing-masing trip adalah customer ID dari tiap konsumen yang bersangkutan. Output rute di atas menghasilkan jarak 469.3 km. Nilai ini akan lebih kecil bila melalui pengolahan data dengan algoritma *tabu search*

4.1.2. Pengolahan solusi awal dengan algoritma tabu search

4.1.2.1. Program Visual Basic Menggunakan Algoritma Tabu Search

Untuk melakukan pengolahan data lebih lanjut terhadap solusi awal yang sudah didapat, dibuatlah program dengan menggunakan software *visual basic* dengan menerapkan algoritma *tabu search*. Data yang diperlukan untuk membuat program ini antara lain adalah data jumlah permintaan tiap pengiriman untuk tiap pelanggan, matrik jarak antara depot ke pelanggan dan jarak antar pelanggan. Data tersebut dimasukkan dalam bentuk *database*. Pada setiap proses pengerjaan atau *run* program, data yang perlu dimasukkan adalah data rute pengiriman solusi awal yang akan dioptimalkan. Untuk melakukan satu kali *run program*, data rute pengiriman yang dibutuhkan adalah rute pengiriman per hari, sehingga data yang dimasukkan akan berbeda untuk mengolah data pada masing-masing hari yang berbeda pula. Selanjutnya data rute pengiriman yang dimasukkan pada proses *run* program ini akan diolah sesuai dengan tahap algoritma *tabu search* yang secara skematis terlihat pada gambar 4.2. Prosedur pemilihan atribut perpindahan artinya konsumen mana saja yang dipindah akan dilakukan oleh program secara acak. Agar program dapat mengolah data rute pengiriman yang telah dimasukkan, perlu juga ditentukan mengenai jumlah iterasi yang harus dilakukan oleh program dan besar *tabu tenure* atau panjang *tabu list* yang mana menggambarkan berapa panjang iterasi suatu atribut *move* dikatakan tabu atau tidak boleh dilakukan. Jumlah iterasi dan panjang *tabu tenure* harus ditentukan setelah memasukkan data rute pengiriman karena program tidak dapat menentukannya sendiri. Kedua faktor ini juga sangat mempengaruhi waktu proses *run* program dan output yang

dihasilkan. Adapun output dari pengolahan data ini adalah urutan rute pengiriman yang baru setiap hari yang berbeda dengan urutan rute solusi awal. Urutan rute baru ini akan menghasilkan jarak tempuh dan biaya pengiriman yang minimal.



Gambar 4.2. Algoritma tabu search

4.1.2.2. Verifikasi dan Validasi Program

Sebelum menggunakan program untuk mengolah data solusi awal, perlu dilakukan verifikasi dan validasi terhadap program Tujuannya untuk memverifikasi apakah program sudah berjalan sesuai aturan dan membandingkan hasil pengerjaan program dengan pengerjaan secara manual.

Data yang digunakan adalah data dua rute dengan dua kendaraan, Trip pertama adalah 17-7-8 dan trip kedua adalah 10-19-12. Tabel di bawah adalah matriks jarak antar konsumen dan depot.

Tabel 4.2. Matriks jarak (km)

jarak	depot	17	7	8	10	19	12
depot	0	41.5	35.8	29.3	30.2	18.5	26.3
17	41.5	0	49.4	59.4	67.4	38.3	47.7
7	35.8	49.4	0	59.3	63.9	59.5	62.4
8	29.3	59.4	59.3	0	16.0	47.4	28.7
10	30.2	67.4	63.9	16.0	0	47.4	51.5
19	18.5	38.3	59.5	47.4	47.4	0	31.6
12	26.3	47.7	62.4	28.7	51.5	31.6	0

Verifikasi adalah tahap mengolah data dengan me-run program, dan validasi adalah tahap membandingkan hasil run program dan pengerjaan manual. Berikut adalah langkah pengerjaan dan hasil secara manual:

- Menentukan solusi awal yaitu total jarak trip Anugrah jaya- bumi makmur- jaya baru (AJ-BM-JB) dan putra buana-surya jaya-mutiara toko (PB-SJ-MT) adalah 315 km
- Melakukan beberapa iterasi
 - Iterasi 1: pertukaran MT dan BM
Hasil: AJ-MT-JB dan PB-SJ-BM total jarak 313.9 km
 - Iterasi 2 : pertukaran AJ dan MT
Hasil : JB-MT-SJ dan PB-AJ-BM total jarak 290.9 km
 - Iterasi 3: pertukaran AJ dan MT
Hasil : JB-AJ-SJ dan PB-MT-BM total jarak 286.1 km
- Solusi hasil beberapa iterasi menunjukkan hasil yang lebih baik dari solusi awal
- Solusi terbaik adalah 286.1 km

Hasil ini sama dengan hasil run program, sehingga program dikatakan telah terverifikasi.

4.1.2.3. Tahap Pengerjaan Algoritma

Pada tahap awal, program akan meminta input berupa rute pengiriman hari apa yang akan diselesaikan. Setelah rute pengiriman dimasukkan, program akan mengakses *database* dan mengambil data yang sesuai dengan apa yang dimasukkan sebagai input. Data jarak yang dari rute pengiriman yang dimasukkan diperoleh dari perhitungan jarak antara depot ke konsumen dan jarak antar depot. Jarak dari solusi awal ini dijadikan sebagai solusi terbaik saat ini yang nantinya akan diganti jika ditemukan jarak yang lebih pendek.

Selanjutnya dilakukan tahapan inisialisasi yaitu menentukan jumlah iterasi dan penggunaan *tabu tenure*. Jumlah iterasi yang digunakan berkisar antara 100 hingga 300 iterasi, bergantung pada jumlah konsumen yang diproses. Sedangkan dalam penelitian ini, skema *tabu tenure* yang digunakan adalah *fix tabu tenure*, artinya selama iterasi maksimal *tabu tenure* adalah tetap.

Kemudian program akan memilih secara beraturan dua rute untuk dilakukan kombinasi antara konsumen di dua rute tersebut. Kombinasi ini melibatkan proses yang dinamakan sebagai *move* antara konsumen kedua rute. Dalam hal ini diasumsikan bahwa satu *move* sama dengan satu iterasi. Di setiap iterasi, dilakukan pengecekan apakah atribut *move* yang digunakan masuk dalam *tabu list* atau tidak. Jika ada, maka *move* tersebut tidak boleh melanjutkan proses selanjutnya, sedangkan jika *move* yang digunakan tidak terdapat dalam daftar tabu, maka solusi yang dihasilkan harus dicek mengenai kapasitas permintaannya, apakah melebihi kapasitas truk atau tidak. Jika tidak memenuhi, maka *move* tersebut tidak dapat melanjutkan ke proses selanjutnya, namun jika memenuhi, maka solusi tersebut menjadi solusi yang dipilih.

Selanjutnya adalah mengatur ulang urutan konsumen dalam rute tersebut untuk mendapatkan hasil yang lebih optimal.

Jika solusi yang dipilih tersebut memiliki jarak yang lebih baik daripada solusi terbaik pada iterasi saat ini, maka solusi tersebut menjadi solusi terbaik yang baru dan akan menjadi solusi saat ini yang akan diproses pada iterasi-iterasi selanjutnya. Atribut *move* yang menghasilkan solusi terbaik yang baru tersebut

direkam dalam *tabu list* sehingga dalam beberapa iterasi selanjutnya, atribut *move* tersebut dilarang untuk dilakukan.

4.1.2.4. Output

Output hasil pengolahan data dengan algoritma *tabu search* adalah urutan konsumen baru pada masing-masing rute dengan total jarak tempuh yang lebih optimal daripada total jarak tempuh solusi awal. Tabel di bawah adalah output rute pengiriman hari-1 dengan algoritma *tabu search*.

Tabel 4.3 Output hari 1 dengan algoritma tabu search

Trip	titik 1	titik 2	titik 3	demand (kg)	truk	jarak(km)	waktu (jam)	biaya
1	M	SNR JY		800	B 4959 AH	127.3	3.1	296685.4
2	SB	PB		1750	B 4532 AC	78.5	2	182952.1
3	KM	SANT.TK	WS	1800	B 3993 AF	136.6	3.6	318360
					Total	342.4	8.7	797997.4

Dari hasil pengolahan dengan algoritma *tabu search* dan solusi awal dengan *sweep method*, maka dapat dilihat adanya penurunan jarak dan biaya yang lebih kecil menggunakan algoritma *tabu search*, pengolahan solusi awal menghasilkan jarak

469.3 km dan biaya 1.093.751 sedangkan dengan *tabu search* dihasilkan jarak 342.4 dan biaya 797.997,4, maka terjadi penurunan jarak tempuh sebesar 126.9 km atau 27% dan penurunan biaya pengiriman sebesar 295.753,6 atau 27%.

4.2. ANALISIS

Analisis adalah tahap membandingkan sistem yang diterapkan sekarang oleh PT.X dengan sistem baru yang diusulkan.

4.2.1. Analisis Penjadwalan Pengiriman

Dengan sistem pengiriman baru yang diusulkan, dapat disusun suatu jadwal pengiriman yang dapat menghasilkan jarak dan waktu pengiriman yang lebih singkat dibandingkan dengan sistem pengiriman sekarang, dimana penentuan rute dilakukan secara manual. Penjadwalan pengiriman dengan sistem baru ini juga lebih fleksibel karena dapat selalu diproses sesuai jadwal permintaan konsumen. Prosesnya adalah memasukkan data konsumen yang memesan, maka akan didapat rute pengiriman. Keuntungan lainnya adalah, jumlah truk, supir, waktu dan tujuan pengiriman per hari dapat diketahui dengan jelas dan prosesnya lebih singkat.

4.2.2. Analisis Rute Distribusi

Rute distribusi yang diterapkan sekarang ditentukan secara manual dan berdasarkan pengalaman dari si pengemudi, sehingga jarak tempuh, biaya pengiriman dan waktu pengiriman menjadi kurang optimal sehingga dapat menyebabkan pesanan sampai tidak tepat waktu ke konsumen, supir seringkali terlambat untuk kembali ke depot. Dengan adanya sistem usulan ini penentuan rute pengiriman lebih terkontrol dan optimal dengan waktu pengolahan yang lebih singkat, hal ini dapat dilihat dari hasil perbandingan jarak dan biaya antara sistem sekarang dengan sistem usulan.

4.2.2.1. Analisis Hasil Perbaikan Sistem Pengiriman

Analisis hasil perbaikan dilakukan dengan membandingkan total jarak tempuh, waktu dan biaya pengiriman antara sistem lama dengan sistem baru yang diusulkan.

4.2.2.1.1 Analisis Jarak Tempuh, Waktu dan Biaya Pengiriman

Jarak tempuh pada sistem sekarang adalah 420.1 km, dengan waktu tempuh 10.5 jam dan biaya pengiriman Rp 979.085,06 sedangkan sistem usulan dengan algoritma tabu search adalah 342.4 km, waktu tempuh 8.7 jam dan biaya pengiriman Rp 797.997,4. Hal ini menurunkan jarak sebesar 77.7 km atau 18%, waktu tempuh 1.8 jam atau 17.1% dan biaya sebesar Rp 181.087,66 atau 18.5%. Perbaikan ini masih dapat ditingkatkan dengan lebih banyak melakukan eksperimen dalam penggunaan jumlah iterasi. Tabel berikut menunjukkan perbandingan jarak tempuh truk, waktu pengiriman dan biaya pengiriman antara sistem sekarang dengan sistem usulan.

Tabel 4.4 Perbandingan jarak tempuh dan biaya sistem sekarang dan usulan

	Sistem sekarang	sistem usulan
Jarak(km)	420.1	342.4
Waktu(jam)	10.5	8.7
Biaya(Rp)	979.085,06	797.997,4

4.2.2.1.2. Analisis Metode

Hasil penerapan metode *tabu search* dalam penyelesaian masalah VRP bergantung pada pemilihan *tabu tenure* atau seberapa panjang iterasi untuk suatu *move* beratribut tabu atau tidak boleh digunakan. *Tabu tenure* juga menunjukkan

berapa panjang *tabu list*. Dalam penelitian ini digunakan *skema fixed tabu tenure* karena sederhana dan memudahkan dalam pengolahan data. *Skema fixed tabu tenure* berarti bahwa tingkat lama atribut tabu yang tetap sepanjang algoritma *tabu search* digunakan. Mengenai waktu proses pengolahan data, hal ini tergantung pada jumlah iterasi dan panjang *tabu list (tabu tenure)* yang digunakan. Dalam hal ini juga perlu dilakukan eksperimentasi mencari waktu proses sesingkat mungkin dengan hasil seoptimal mungkin dengan cara mengkombinasikan jumlah iterasi dan *tabu tenure* yang sesuai. Waktu yang dibutuhkan untuk mengolah data pada penelitian ini dengan menggunakan algoritma *tabu search* dan perangkat lunak *visual basic* rata-rata adalah lima detik. Waktu proses ini masih dapat dipersingkat dengan cara memperbaiki bahasa pemrograman yang digunakan. Dalam proses pengolahan data mungkin juga terjadi perbedaan hasil untuk setiap proses *run* program walaupun rute input, jumlah iterasi dan *tabu tenure* sama. Hal ini disebabkan karena penggunaan prosedur acak atau random untuk menentukan atribut *move* atau ID konsumen yang akan ditukar pada setiap kali iterasi.

5. KESIMPULAN DAN SARAN

Penelitian ini menghasilkan suatu rute pengiriman cat yang berbeda dengan dengan sistem pengiriman sekarang yang diterapkan oleh perusahaan. Berdasarkan jadwal dan rute pengiriman yang diperoleh melalui pengolahan data dengan menggunakan algoritma *tabu search*, diperoleh kesimpulan sebagai berikut:

- Pada sistem pengiriman usulan didapat bahwa pada sistem ini model yang digunakan adalah model pengiriman yang fleksibel dan sesuai dengan pola permintaan konsumen PT.X yang fluktuatif, sehingga pengolahan data dapat dilakukan dalam berbagai keadaan dan situasi, dengan waktu yang lebih singkat. Proses pengolahan data juga lebih mudah karena hanya memanggil data konsumen yang akan memesan maka akan keluar rute pengiriman.

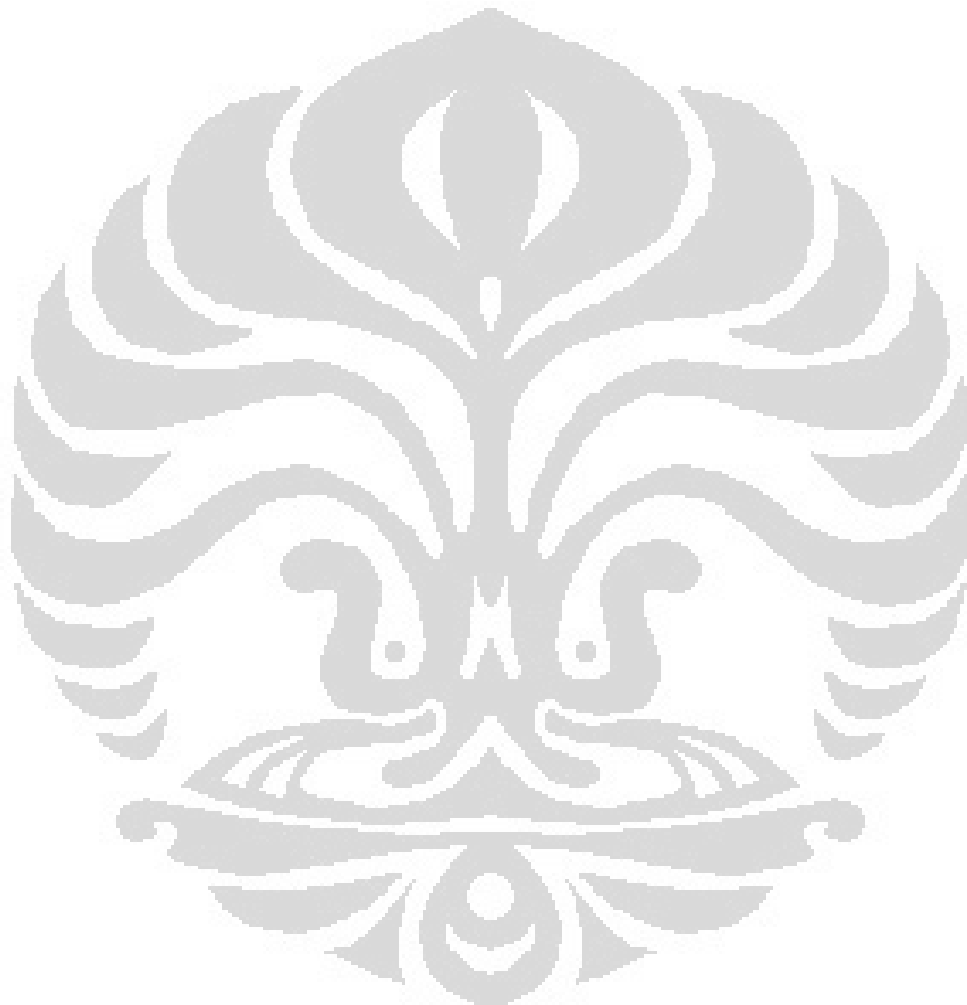
Berikut adalah saran untuk penelitian selanjutnya:

Untuk penelitian yang akan datang, akan lebih baik bila sistem pengiriman yang akan diusulkan, juga terintegrasi dengan sistem lain, seperti sistem keuangan, sistem pemesanan, sehingga akan lebih memudahkan bagi perusahaan. Juga peningkatan penggunaan metode atau bahasa pemrograman yang lebih maju dari yang sekarang.

DAFTAR PUSTAKA

- Ballou, Ronald H., 1992, *Business Logistics Management*, 4th ed., Prentice-Hall, Inc. New Jersey.
- Ballou, Ronald H dan Yogesh K Agarwal , 1998, “ A Performance Comparison of Several Popular Algorithms for Vehicle Routing and Scheduling”, *Journal of Business Logistic* 9, no.1.
- Braysy, Olli dan Michel Gendreau, 2001, “Tabu Search Heuristics for Vehicle Routing Problem with Time Windows”, *Journal of SINTEF*, hal.5
- Chase, Richard B., Nicholas J. Aquilano dan F. Robert Jacobs, 2001, *Operations Management for Competitive Advantage* 9th ed., New York: McGraw-Hill Co.
- Cordeau, Jean Francois dan Gilbert Laporte, 2002, “Tabu Search Heuristics for The Static Multi Vehicle Dial –a-Ride Problem”, *Transportation Research*, bag. B 37, hal. 579-594
- Diaz, Berbane Dorronsoro, 2002, What is VRP?, [www.http//.neo.lcc.uma.es](http://neo.lcc.uma.es)
- Glover, Fred dan Manual Laguna, 1997, Tabu Search, www.geocities.com/francorbusetti/laguna.pdf, (last updated January 6, 2002)
- Lee, Tzong-Ru dan Ji-Hwa Ueng., 1999, “A Study of Vehicle Routing Problem with Load-Balancing”, *International Journal of Physical Distribution and Logistics management*, vol. 29, no. 10, hal.646-658
- Liu, F. H. dan S Y. Shen, 1999, “The Fleet Size and Mix Vehicle Routing Problem with Time Windows”, *Journal of Operational Research Society*, hal. 721-732
- Rizolli, AE dkk, 2004, “Ant Colony Optimisation for Vehicle Routing Problems: from Theory to Applications”, Switzerland.
- Shiho, Sodekoda, 2004, “Vehicle Route Planning System Based on Combinatorial Optimization Techniques”, *Engineering review*, vol. 37, no.1.
- Taha, Hamdy A., 1997, *Operation Research An Introduction* 6th ed., Prentice-Hall, Inc. New Jersey

Toth, Paolo dan Daniel Vigo, 2001, *The Vehicle Routing Problem*, Philadelphia :
Society for Industrial and Applied Mathematics



APPENDICES

Data kecepatan

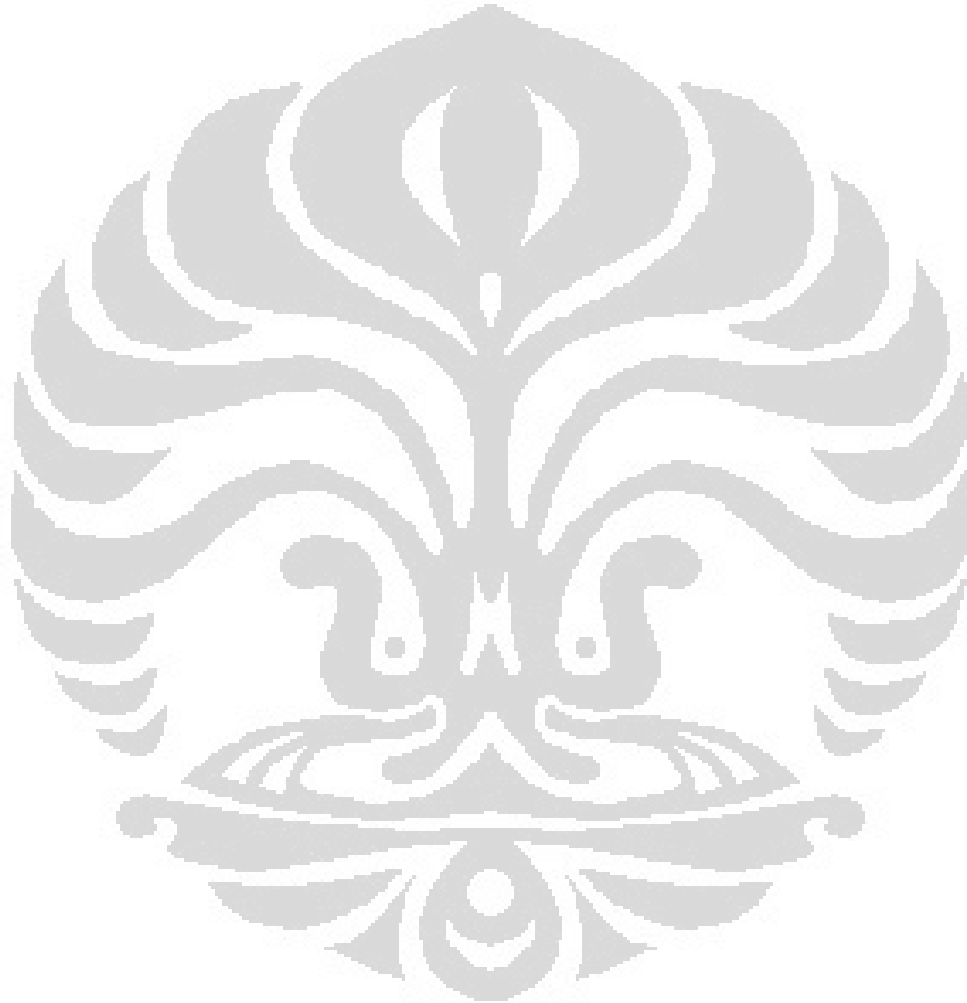
Dari	Ke	berangkat	tiba	Km berangkat	Km tiba	Jarak	waktu(jam)	kecepatan
DEPOT	JAYA BARU	10.00	10.50	141227	141257	29.3	0.83	35.3012
JAYA BARU	PUTRA BUANA	10.20	10.45	141257	141273	16	0.42	38.09524
PUTRA BUANA	ANUGRAH JAYA	11.00	12.15	141273	141340	67.4	1.25	53.92
ANUGRAH JAYA	HIDUP BARU	13.05	14.10	141340	141379	39.4	1.083	36.38042
HIDUP BARU	MUTIARA ,TOKO	14.25	15.25	141379	141431	52.4	1	52.4
MUTIARA ,TOKO	SURYA JAYA	15.40	16.35	141341	141462	31.6	0.916	34.49782
SURYA JAYA	DEPOT	16.35	17.10	141462	141480	18.5	0.583	31.73242
Total						254.6	6.082	282.3271
Kecepatan rata-rata								40.33244

Data konsumen

Cust ID	Customer name	Address	City
1	BERKAH JAYA ,TOKO	JL.CIBENING BINTARA JAYA	BEKASI
2	SANTOSA ,TOKO	JL.JATIWARINGIN I/9B	BEKASI
3	MEGA INDAH	JL.TRIDAYA INDAH I NO.28	BEKASI
4	YASMINE JAYA. PD	JL. WIJAYA KUSUMA RAYA NO.R6	BOGOR
5	WIRA SAKTI	JL. RAYA TLAJUNG UDIK KM.2.3	BOGOR
6	JAYA AGUNG	JL. BOJONGSARI LAMA NO.34	BOGOR
7	BUMI MAKMUR	JL. CINANGKA RAYA NO. 138	DEPOK
8	JAYA BARU	JL.RAYA KEBON JERUK 14	JAK- BAR
9	SINAR BANGUNAN	JL.DURI RAYA NO.20	JAK- BAR
10	PUTRA BUANA	JL.KEMBANGAN RAYA NO.1	JAK- BAR
11	KALI MAS	JL.KALILIO 25	JAK-PUS
12	MUTIARA ,TOKO	JL.GUNUNG SAHARI XI NO.30	JAK-PUS
13	RODA SELATAN	JL.BENDUNGAN HILIR	JAK-PUS
14	MADJU	JL.PANGLIMA POLIM RAYA	JAK- SEL
15	HIDUP BARU	JL.PANGLIMA POLIM RAYA	JAK- SEL
16	CATUR TIRTA MURNI ,PT.	JL.BAMBU KUNING A1/287 A	JAK- SEL
17	ANUGRAH JAYA	JL.RAYA PULO GEBANG 101	JAK- TIM
18	INTI ,PD.	JL. UTAN KAYU NO.54-61	JAK- TIM
19	SURYA JAYA	DANAU AGUNG UTARA BLOK G	JAK-UT
20	BUDI JAYA	JL. BOULEVARD QA 1 NO.19	JAK-UT
21	BANGUN JAYA I	JL.DANAU AGUNG II NO.13	JAK-UT
22	SINAR JAYA	JL.MERDEKA NO.120/250 KARAWACI	TANGERANG
23	SUMBER WARNA	RUKO VILLA MELATI MAS	TANGERANG
24	HARAPAN MAKMUR	JL. RAYA TAMAN ADIYASA KM 2	TANGERANG

Data pengiriman

trip	titik 1	titik 2	titik 3	demand (kg)	truk	jarak (km)	waktu(jam)	biaya
1	PB	SB		1750	B 2345 AE	78.5	2	182952.1
2	KM	M		900	B 2772 AA	114.7	2.8	267319.8
3	SANT.TK	SNR JY	WS	1700	B 3993 AF	226.9	5.7	528813.1
					Total	420.1	10.5	979085.1



Bahasa program

```
Option Compare Database
```

```
Option Explicit
```

```
'Dim theArray(1, 1) As String
Dim theArray(50, 50) As String
Dim arrTotJarak() As Single
Dim arrTotWaktu() As Single
```

```
Private Sub cmd_FinalSolution_Click()
cmd_ViewFinal_Click
End Sub
```

```
Private Sub cmdGenerateTabu_Click()
Dim RouteID As Long
```

```
RouteID = findRoute(TransDate.Value)
```

```
If RouteID > 0 Then
PrepareRouteTemp (RouteID)
```

```
GenerateTabu (TransDate.Value)
```

```
cmd_ViewFinal_Click
End If
```

```
End Sub
```

```
Private Sub cmdGenRoute_Click()
```

```
Dim hsl As Integer
```

```
hsl = vbYes
```

```
If findRoute(TransDate.Value) > 0 Then
hsl = MsgBox("Rute sudah didefinisikan. Apakah rute tersebut akan dihapus?",
vbYesNo Or vbQuestion)
```

```
End If
```

```
If hsl = vbYes Then
```

```
'Generate the route
```

```
GenerateRoute (TransDate.Value)
```

```
'Me.Requery
```

```
Me.Refresh
```

```
cmdViewRoute2_Click
```

```
End If
```

```
End Sub
```

```
Private Sub GenerateRoute(trDate As Date)
```

On Error GoTo err_GenerateRoute

```
Dim rsOrder As New ADODB.Recordset
Dim rsMasterRoute As New ADODB.Recordset
Dim rsVehicle As New ADODB.Recordset
Dim sqlstr As String
```

```
Dim RouteID As Long
Dim KapasitasKG As Long
Dim FromCustID As String
Dim RouteOrder As Long
Dim DistanceFromLast As Single
Dim WaktuJamFromLast As Single
```

```
'Delete the existing Route in Master route
DoCmd.SetWarnings (False)
DoCmd.RunSQL ("delete RouteDate from [tr Master Route] where
RouteDate=cdate('"' & trDate & "'")")
```

```
'Open Order list
sqlstr = "select * from vw_order where transdate=cdate('"' & trDate & "'")"
rsOrder.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly
```

```
'Open Vehicle list
sqlstr = "select * from vw_Vehicle"
rsVehicle.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly
```

```
'Open MasterRoute
sqlstr = "select * from [tr Master Route]"
rsMasterRoute.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockPessimistic
```

```
'Buat Route baru
rsMasterRoute.AddNew
RouteID = rsMasterRoute!RouteID
rsMasterRoute!Routedate = trDate
rsMasterRoute.Update
```

```
'Mulai dari Order dengan sudut terkecil
If Not (rsOrder.EOF) Then
rsOrder.MoveFirst
```

```
'Mulai dari vehicle dengan kapasitas terberat
```

```

If Not (rsVehicle.EOF) Then
  rsVehicle.MoveFirst
  KapasitasKG = rsVehicle!CapacityKG
  'kendaraan Mulai jalan dari depot
  FromCustID = "0"
  'the route order for this vehicle starts from 0
  RouteOrder = 1

  While Not (rsOrder.EOF) And Not (rsVehicle.EOF)
    'jika kendaraan sudah tidak muat
    If KapasitasKG < rsOrder!OrderKG Then
      'Kendaraan yang sekarang diarahkan ke Depot
      DistanceFromLast = FindJarak(FromCustID, "0")
      WaktuJamFromLast = FindWaktuJam(FromCustID, "0")
      BuatRuteCustID RouteID, rsVehicle!PoliceNo, RouteOrder, "0", 0,
      DistanceFromLast, WaktuJamFromLast

      'Ganti kendaraan baru
      If Not (rsVehicle.EOF) Then
        rsVehicle.MoveNext

        KapasitasKG = rsVehicle!CapacityKG
        'kendaraan Mulai jalan dari depot
        FromCustID = "0"
        'the route order for this vehicle starts from 0
        RouteOrder = 1

      End If
    End If

    'Jika kendaraan masih dapat dimuati
    If KapasitasKG >= rsOrder!OrderKG Then
      'Lewati rute tsb
      DistanceFromLast = FindJarak(FromCustID, rsOrder!custID)
      WaktuJamFromLast = FindWaktuJam(FromCustID, rsOrder!custID)
      BuatRuteCustID RouteID, rsVehicle!PoliceNo, RouteOrder, rsOrder!custID,
      rsOrder!OrderKG, DistanceFromLast, WaktuJamFromLast

      KapasitasKG = KapasitasKG - rsOrder!OrderKG
      'sekarang vehicle sudah ada di custID
      FromCustID = rsOrder!custID
      RouteOrder = RouteOrder + 1
    End If
  rsOrder.MoveNext

  'Jika order sudah habis tapi masih ada kapasitas -> pulang ke Depot
  If rsOrder.EOF And (FromCustID <> "0") Then

```



```

        DistanceFromLast = FindJarak(FromCustID, "0")
        WaktuJamFromLast = FindWaktuJam(FromCustID, "0")
        BuatRuteCustID RouteID, rsVehicle!PoliceNo, RouteOrder, "0", 0,
DistanceFromLast, WaktuJamFromLast
    End If
Wend

'Not (rsVehicle.EOF)
End If

'Not (rsOrder.EOF)
End If

rsOrder.Close
rsVehicle.Close
rsMasterRoute.Close

exit_GenerateRoute:
Set rsOrder = Nothing
Set rsVehicle = Nothing
Set rsMasterRoute = Nothing

DoCmd.SetWarnings (True)
Exit Sub

err_GenerateRoute:
MsgBox Err.Description
Resume exit_GenerateRoute
End Sub

Private Function FindJarak(Cust1 As String, Cust2 As String) As Single
On Error GoTo err_FindJarak

Dim rsJarakWaktu As New ADODB.Recordset
Dim sqlstr As String

'Open Jarak Waktu
sqlstr = "select * from vw_Customer_jarak where FromCustID=""" & Cust1 &
"""" and ToCustID=""" & Cust2 & """"
rsJarakWaktu.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly

FindJarak = -1
If Not (rsJarakWaktu.EOF) Then
    FindJarak = rsJarakWaktu!DistanceKM
End If

rsJarakWaktu.Close

```

```

exit_findJarak:
    Set rsJarakWaktu = Nothing
    Exit Function
err_FindJarak:
    MsgBox Err.Description
    Resume exit_findJarak
End Function
Private Function FindWaktuJam(Cust1 As String, Cust2 As String) As Single
On Error GoTo err_FindWaktuJam

Dim rsJarakWaktu As New ADODB.Recordset
Dim sqlstr As String

'Open Jarak Waktu
sqlstr = "select * from vw_Customer_jarak where FromCustID=""" & Cust1 &
"""" and ToCustID=""" & Cust2 & """"
rsJarakWaktu.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly

FindWaktuJam = -1
If Not (rsJarakWaktu.EOF) Then
    FindWaktuJam = rsJarakWaktu!WaktuJam
End If

rsJarakWaktu.Close

exit_FindWaktuJam:
    Set rsJarakWaktu = Nothing
    Exit Function
err_FindWaktuJam:
    MsgBox Err.Description
    Resume exit_FindWaktuJam
End Function

Private Sub BuatRuteCustID(RouteID As Long, PoliceNo As String, RouteOrder
As Long, custID As String, OrderKG As Long, DistanceFromLast As Single,
WaktuJamFromLast As Single)
On Error GoTo err_BuatRuteCustID
Dim rsDetailRoute As New ADODB.Recordset
Dim sqlstr As String

'Open DetailRoute
sqlstr = "select * from [tr Detail Route]"
rsDetailRoute.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockPessimistic

rsDetailRoute.AddNew
rsDetailRoute!RouteID = RouteID

```

```

rsDetailRoute!PoliceNo = PoliceNo
rsDetailRoute!RouteOrder = RouteOrder
rsDetailRoute!custID = custID
rsDetailRoute!OrderKG = OrderKG
rsDetailRoute!DistanceFromLast = DistanceFromLast
rsDetailRoute!WaktuJamFromLast = WaktuJamFromLast
rsDetailRoute.Update

```

```
rsDetailRoute.Close
```

```

exit_err_BuatRuteCustID:
    Set rsDetailRoute = Nothing
    Exit Sub

```

```

err_BuatRuteCustID:
    MsgBox Err.Description
    Resume exit_err_BuatRuteCustID
End Sub

```

```
Private Sub Command17_Click()
```

```
End Sub
```

```
Private Sub Form_Current()
```

```

    If TransDate.Value <> vbNull Then
    If findRoute(TransDate.Value) > 0 Then
        cmdViewRoute2.Enabled = True
        cmd_TabuSetting.Enabled = True
        cmdGenerateTabu.Enabled = True
    Else
        cmdViewRoute2.Enabled = False
        cmd_TabuSetting.Enabled = False
        cmdGenerateTabu.Enabled = False
    End If
    End If
End Sub
Private Function findRoute(trDate As Date) As Long
On Error GoTo err_findRoute

```

```
Dim rs1 As New ADODB.Recordset
```

```
Dim sqlstr As String
```

```
Dim numrec As Long
```

```
numrec = 0
```

```
sqlstr = "select RouteID from [tr Master Route] where RouteDate=cdate('"' & trDate & "'")"
```

```
'sqlstr = "select * from [tr Master Route]"
```

```
rs1.Open sqlstr, CurrentProject.Connection, adOpenForwardOnly,
adLockReadOnly
```

```
If Not (rs1.EOF) Then
```

```
    rs1.MoveFirst
```

```
    numrec = rs1!RouteID
```

```
    ' While Not (rs1.EOF)
```

```
    '     numrec = numrec + 1
```

```
    '     rs1.MoveNext
```

```
    ' Wend
```

```
End If
```

```
findRoute = numrec
```

```
rs1.Close
```

```
exit_findRoute:
```

```
    Set rs1 = Nothing
```

```
    Exit Function
```

```
err_findRoute:
```

```
    MsgBox Err.Description
```

```
    Resume exit_findRoute
```

```
End Function
```

```
Private Sub test()
```

```
On Error GoTo Err_Command20_Click
```

```
Dim addhub
```

```
Dim rs1 As New ADODB.Recordset
```

```
Dim RS2 As New ADODB.Recordset
```

```
Dim RS3 As New ADODB.Recordset
```

```
Dim rsbarang As New ADODB.Recordset
```

```
Dim qtytable1
```

```
addhub = "SELECT Header_Gdg_Brg.Kode_Toko_Gdg,
```

```
Detail_Gdg_Brg.Kode_Barang, Detail_Gdg_Brg.Quantity FROM
```

```
Header_Gdg_Brg INNER JOIN Detail_Gdg_Brg ON
```

```
Header_Gdg_Brg.Kode_Gdg_Brg = Detail_Gdg_Brg.Kode_Gdg_Brg Where
```

```
Kode_Toko_Gdg=" & kodegdg & ""
```

```
rs1.Open addhub, CurrentProject.Connection, adOpenDynamic,
```

```
adLockOptimistic
```

```
RS2.Open "select * from Detail_Brg_Masuk Where No_Brg_Masuk=" &
```

```
nomasuk & "", CurrentProject.Connection, adOpenDynamic, adLockOptimistic
```

```
RS3.Open "select * from table1 where No_Brg_Masuk=" & nomasuk & "",
```

```
CurrentProject.Connection, adOpenDynamic, adLockOptimistic
```

```

rsbarang.Open "Master_Brg", CurrentProject.Connection, adOpenDynamic,
adLockOptimistic
'rs1 = detilToko gidang
'rs2 = Table1
If Not (rs1.EOF Or RS3.EOF) Then
While Not RS2.EOF
    rsbarang.MoveFirst
    rsbarang.Find "Kode_Brg=" & RS2!Kode_Brg & ""
    If Not rsbarang.EOF Then

        kodeBrg = rsbarang!Kode_Brg
        End If
        RS3.MoveFirst
        RS3.Find "Kode_brg=" & kodeBrg & ""

        rs1.MoveFirst
        rs1.Find "Kode_barang=" & kodeBrg & ""
        If Not (rs1.EOF Or RS3.EOF) Then
            Qty = Val(RS2!QtyDtg)
            qtytable1 = Val(RS3!QtyDtg)
            'Jika di click ubah berkali-kali qty di Detail_Gdg_brg bertambah terus
            'jadi rumusnya diganti Qty(gdg_brg) = Qty(gdg_brg) - Qty (brg_masuk) +
            qty(table1)
            rs1!Quantity = Val(rs1!Quantity) - Val(Qty) + qtytable1
            RS2!QtyDtg = qtytable1

        End If
        rs1.Update
        RS2.MoveNext
    Wend
End If
Set rs1 = Nothing
Set RS2 = Nothing
Set RS3 = Nothing
No_Brg_Masuk.SetFocus
Exit_Command20_Click:
    Exit Sub

Err_Command20_Click:
    MsgBox Err.Description
    Resume Exit_Command20_Click
Set rs1 = Nothing
Set RS2 = Nothing
Set RS3 = Nothing
End Sub
Private Sub cmdViewRoute2_Click()
On Error GoTo Err_cmdViewRoute2_Click

```

```

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Master Route"

'stLinkCriteria = "[RouteDate]=" & "#" & Me![TransDate] & "#"
stLinkCriteria = "[RouteDate]=" & "cdate(\"" & Me![TransDate] & "\")"
DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmdViewRoute2_Click:
Exit Sub

Err_cmdViewRoute2_Click:
MsgBox Err.Description
Resume Exit_cmdViewRoute2_Click

End Sub

Private Sub Form_Load()
Dim a() As String
Dim b() As String

ReDim a(3)
a(0) = "1"
a(1) = "1"
a(2) = "1"
a(3) = "1"

b = a
End Sub

Private Sub TransDate_LostFocus()
DoCmd.Save
End Sub

Private Sub cmd_TabuSetting_Click()
On Error GoTo Err_cmd_TabuSetting_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "Tabu Setting"
DoCmd.OpenForm stDocName, acFormDS, , stLinkCriteria, acFormEdit,
acWindowNormal

Exit_cmd_TabuSetting_Click:
Exit Sub

```

Err_cmd_TabuSetting_Click:

MsgBox Err.Description

Resume Exit_cmd_TabuSetting_Click

End Sub

Private Sub GenerateTabu(trDate As Date)

On Error GoTo err_GenerateTabu

Dim vwRouteJmlNode As New ADODB.Recordset

Dim sqlstr As String

'Tabu Setting

Dim JmlIteration As Long

Dim TabuTenure As Long

Dim JumlahRute As Long

Dim JmlMunculTabu As Long

'counter

Dim i, j As Long

Dim Iterasi As Long

Dim batas As Long

'customer yg akan ditukar

Dim ruteA As String

Dim ruteB As String

Dim Sinyal As Integer

Dim random1 As Long

Dim random2 As Long

Dim jmlNodeA As Long

Dim jmlNodeB As Long

Dim PoliceNoA As String

Dim PoliceNoB As String

Dim OrderKGA As Long

Dim OrderKGB As Long

Dim RouteID As Long

'Kapasitas

Dim kapasitasA As Long

Dim kapasitasB As Long

Dim KapTrukA As Long

Dim KapTrukB As Long

'Hasil Rute A setelah iterasi

Dim newKapasitasA As Long

Dim newDistanceA As Single

Dim newWaktuJamA As Single

Dim oldDistanceA As Single

Dim oldWaktuJamA As Single

AmbilTabuSetting JmlIteration, TabuTenure

JumlahRute = AmbilJmlRuteHari(trDate)

'Mulai Tabu Algorithm

batas = 0

Iterasi = 0

'Ambil data2 Route

sqlstr = "select * from [vw_route_Jumlah_node_temp] where
RouteDate=cdate('' & trDate & ''")"

vwRouteJmlNode.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockOptimistic

If Not (vwRouteJmlNode.EOF) Then

RouteID = vwRouteJmlNode!RouteID

PrepareFinalRoute RouteID

While Iterasi < JmlIteration

For i = 1 To JumlahRute - 1

' For i = 1 To 1

If Iterasi < JmlIteration Then

ruteA = "0"

ruteB = "0"

'kapasitasA = 20000

'kapasitasB = 20000

' For j = i + 1 To JumlahRute

For j = i + 1 To JumlahRute

'menuju record rute A

vwRouteJmlNode.MoveFirst

If i > 0 Then

vwRouteJmlNode.Move i - 1

End If

'Ambil jumlah node dalam Rute A

jmlNodeA = vwRouteJmlNode!JumlahNode

kapasitasA = vwRouteJmlNode!SumOfOrderKG

KapTrukA = vwRouteJmlNode!CapacityKG

PoliceNoA = vwRouteJmlNode!PoliceNo

oldDistanceA = vwRouteJmlNode!SumOfDistanceFromLast

oldWaktuJamA = vwRouteJmlNode!SumOfWaktuJamFromLast


```

'menuju record rute B
vwRouteJmlNode.MoveFirst
If j > 0 Then
  vwRouteJmlNode.Move j - 1
End If

'Ambil jumlah node dalam Rute B
jmlNodeB = vwRouteJmlNode!JumlahNode
kapasitasB = vwRouteJmlNode!SumOfOrderKG
KapTrukB = vwRouteJmlNode!CapacityKG
PoliceNoB = vwRouteJmlNode!PoliceNo

Sinyal = 1
While Sinyal = 1
  batas = 0
  Do
    batas = batas + 1

    Randomize
    'Ambil random dari 1 sampai JmlNode - 1
    ' node terakhir adalah Depot

    random1 = Int((jmlNodeA - 1) * Rnd + 1)

    'random1 = 3

    AmbilCustRute trDate, PoliceNoA, random1, ruteA, OrderKGA

    Randomize
    random2 = Int((jmlNodeB - 1) * Rnd + 1)
    'random2 = 1
    AmbilCustRute trDate, PoliceNoB, random2, ruteB, OrderKGB

    'Cek jumlah kemunculan dalam tabu list
    CekTabuList ruteA, ruteB, JmlMunculTabu

    If JmlMunculTabu = 0 Then
      Sinyal = 0
    Else
      If JmlMunculTabu = TabuTenure Then
        'Tabu
        Sinyal = 1
      Else
        Sinyal = 0
      End If
    End If
  End If
End If

```

```

'Do
  'Hitung dulu apakah hasil pertukarang masing2 route masih tetap di dalam
  kapasitas angkut
  Loop Until (kapasitasA - OrderKGA + OrderKGB <= KapTrukA) And
  (kapasitasB - OrderKGB + OrderKGA <= KapTrukB)
  'Loop While (kapasitasA > KapTrukA) Or (kapasitasB > KapTrukB) And
  (batas < 12)

  'while sinyal=1
  Wend

  'Jika sinyal = 0 -> masukkan tabu List
  TambahTabuList ruteA, ruteB

  Iterasi = Iterasi + 1

  SimulasiPindahNode trDate, PoliceNoA, 0, random1, PoliceNoB, random2,
  ruteB, OrderKGB, newKapasitasA, newDistanceA, newWaktuJamA, False

  'Jika Jarak yang baru lebih kecil dan kendala kapasitas terpenuhi maka
  If (newKapasitasA <= KapTrukA) And (newDistanceA < oldDistanceA) And
  (newWaktuJamA < oldWaktuJamA) Then
  'Tulis hasil simulasi ke dalam tabel
  SimulasiPindahNode trDate, PoliceNoA, 0, random1, PoliceNoB, random2,
  ruteB, OrderKGB, newKapasitasA, newDistanceA, newWaktuJamA, True

  'Optimasi Rute
  OptimasiRute RouteID, PoliceNoA

  vwRouteJmlNode.Requery

  'Pindahkan ke Rute Final
  CopyRuteToFinal RouteID, Iterasi
  End If

  'Hapus tabu list
  DoCmd.SetWarnings (False)
  DoCmd.RunSQL ("delete * from tabuList")
  DoCmd.SetWarnings (True)

  'next j
  Next

  'if iterasi < JmlIteration
  End If

  'next i
  Next

```

```

'while iterasi < jmlIteration
Wend

'vw RouteJmlNode
End If

exit_GenerateTabu:
  Set vwRouteJmlNode = Nothing
  Exit Sub
err_GenerateTabu:
  MsgBox Err.Description
  Resume exit_GenerateTabu
End Sub
Private Sub AmbilTabuSetting(ByRef iJmlIteration As Long, ByRef iTabuTenure
As Long)
On Error GoTo err_AmbilTabuSetting
Dim sqlstr As String
Dim TabuSetting As New ADODB.Recordset

'Ambil Tabu Setting dari tabel
sqlstr = "select * from [Tabu Setting]"
TabuSetting.Open sqlstr, CurrentProject.Connection, adOpenForwardOnly,
adLockReadOnly

If Not (TabuSetting.EOF) Then
  TabuSetting.MoveFirst
  While Not (TabuSetting.EOF)

    Select Case TabuSetting!ID
      Case 1: iJmlIteration = (TabuSetting!Value)
      Case 2: iTabuTenure = Val(TabuSetting!Value)
    End Select

    TabuSetting.MoveNext
  Wend
End If
TabuSetting.Close

exit_AmbilTabuSetting:
  Set TabuSetting = Nothing
  Exit Sub
err_AmbilTabuSetting:
  MsgBox Err.Description
  Resume exit_AmbilTabuSetting
End Sub

```

```

Private Function AmbilJmlRuteHari(itrDate As Date) As Long
On Error GoTo err_AmbilJmlRuteHari
Dim vwRouteJmlRute As New ADODB.Recordset
Dim sqlstr As String
Dim JumlahRute As Long

'Ambil jumlah rute dalam 1 hari
sqlstr = "select * from [vw_Route_Jumlah_rute_temp] where
RouteDate=cdate('"' & itrDate & "'")"
vwRouteJmlRute.Open sqlstr, CurrentProject.Connection, adOpenForwardOnly,
adLockReadOnly

If Not (vwRouteJmlRute.EOF) Then
    vwRouteJmlRute.MoveFirst
    JumlahRute = vwRouteJmlRute!JumlahRute
Else
    JumlahRute = 0
End If
vwRouteJmlRute.Close

AmbilJmlRuteHari = JumlahRute

exit_AmbilJmlRuteHari:
Set vwRouteJmlRute = Nothing
Exit Function
err_AmbilJmlRuteHari:
MsgBox Err.Description
Resume exit_AmbilJmlRuteHari
End Function

Private Sub AmbilCustRute(itrDate As Date, iPoliceNo As String, iRouteOrder
As Long, ByRef oCustID As String, ByRef oOrderKG As Long)
On Error GoTo err_AmbilCustRute

Dim vwRoute As New ADODB.Recordset
Dim sqlstr As String

sqlstr = "select CustID, OrderKG from vw_route_temp where
RouteDate=cdate('"' & itrDate & "'") and PoliceNo='"' & iPoliceNo & "' and
RouteOrder=" & iRouteOrder
vwRoute.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly

oCustID = ""
oOrderKG = 0

If Not (vwRoute.EOF) Then
    oCustID = vwRoute!custID
    oOrderKG = vwRoute!OrderKG

```

```

End If
vwRoute.Close

exit_AmbilCustRute:
  Set vwRoute = Nothing
  Exit Sub
err_AmbilCustRute:
  MsgBox Err.Description
  Resume exit_AmbilCustRute
End Sub
Private Sub CekTabuList(iDari As String, iKe As String, ByRef oJumlah As
Long)
On Error GoTo err_CekTabuList
Dim sqlstr As String
Dim TabuList As New ADODB.Recordset
Dim jumlah As Long

jumlah = 0
'Ambil Tabu Setting dari tabel
sqlstr = "select Jumlah from vw_TabuList where Dari=" & iDari & " and
Ke=" & iKe & """"
TabuList.Open sqlstr, CurrentProject.Connection, adOpenForwardOnly,
adLockReadOnly

If Not (TabuList.EOF) Then
  TabuList.MoveFirst
  jumlah = TabuList!jumlah
End If
TabuList.Close

oJumlah = jumlah

exit_CekTabuList:
  Set TabuList = Nothing

  Exit Sub
err_CekTabuList:
  MsgBox Err.Description
  Resume exit_CekTabuList
End Sub
Private Sub TambahTabuList(iDari As String, iKe As String)
On Error GoTo err_TambahTabuList
Dim sqlstr As String
Dim TabuList As New ADODB.Recordset
Dim jumlah As Long

CekTabuList iDari, iKe, jumlah

```



```

Private Sub SimulasiPindahNode(itrDate As Date, iPoliceNo As String, iIteration
As Long, iRouteOrderA As Long, iPoliceNoB As String, iRouteOrderB As Long,
iNewCustB As String, iNewOrderKGB As Long, ByRef oTotalKG As Long,
ByRef oTotalDistance As Single, ByRef oTotalWaktu As Single, iWrite As
Boolean)
On Error GoTo err_SimulasiPindahNode
Dim vw_RouteTemp As New ADODB.Recordset
Dim sqlstr As String

'info Cust sebelumnya
Dim PrevCustID As String
Dim newJarak As Single
Dim newWaktu As Single
Dim theCustID As String
Dim theOrderKg As Long

'Untuk Route A
oTotalKG = 0
oTotalDistance = 0
oTotalWaktu = 0

'Untuk Route B
Dim iNewCustA As String
Dim iNewOrderKGA As Long

Dim bolehTulis As Boolean

sqlstr = "select * from vw_route_temp where RouteDate=cdate('' & itrDate &
''') and PoliceNo=''' & iPoliceNo & ''' order by
RouteID,RouteDate,PoliceNo,RouteOrder"
vw_RouteTemp.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly

PrevCustID = "0"
If Not (vw_RouteTemp.EOF) Then
While Not (vw_RouteTemp.EOF)

theCustID = vw_RouteTemp!custID
theOrderKg = vw_RouteTemp!OrderKG

'Jika node ini adalah node yang akan diganti customernya
If vw_RouteTemp!RouteOrder = iRouteOrderA Then
'Jika PrevCustID="" berarti prev=Depot
theCustID = iNewCustB
newJarak = FindJarak(PrevCustID, iNewCustB)
newWaktu = FindWaktuJam(PrevCustID, iNewCustB)

theOrderKg = iNewOrderKGB

```

```

oTotalKG = oTotalKG + iNewOrderKGB
oTotalDistance = oTotalDistance + newJarak
oTotalWaktu = oTotalWaktu + newWaktu

```

```

'Untuk Route B
iNewCustA = vw_RouteTemp!custID
iNewOrderKGA = vw_RouteTemp!OrderKG

```

```

bolehtulis = True

```

```

'informasi customer sebelumnya
PrevCustID = iNewCustB

```

```

Else
'Untuk Node setelah node yang ditukar
If vw_RouteTemp!RouteOrder = iRouteOrderA + 1 Then
newJarak = FindJarak(PrevCustID, vw_RouteTemp!custID)
newWaktu = FindWaktuJam(PrevCustID, vw_RouteTemp!custID)

oTotalDistance = oTotalDistance + newJarak
oTotalWaktu = oTotalWaktu + newWaktu

```

```

bolehtulis = True

```

```

Else
oTotalKG = oTotalKG + vw_RouteTemp!OrderKG
newJarak = vw_RouteTemp!DistanceFromLast
newWaktu = vw_RouteTemp!WaktuJamFromLast

```

```

oTotalDistance = oTotalDistance + newJarak
oTotalWaktu = oTotalWaktu + newWaktu

```

```

bolehtulis = False
End If

```

```

'informasi customer sebelumnya
PrevCustID = vw_RouteTemp!custID

```

```

End If

```

```

'Jika ditulis
If iWrite And bolehtulis Then
'sqlstr = "insert into [tr detail route temp] values (RouteID,
PoliceNo,Iteration,RouteOrder,CustID,OrderKG,DistanceFromLast,WaktuJamFromLast) values (" & _

```



```
'vw_routeTemp!RouteID & ", "" & vw_routeTemp!PoliceNo & """, " &
iteration & ", " & vw_routeTemp!RouteOrder & ", "" & theCustID & """, " &
theOrderKg & ", " & newJarak & ", " & newWaktu
```

```
sqlstr = "update [tr detail route temp] set custID="" & theCustID & "",
OrderKG=" & theOrderKg & ", DistanceFromLast=" & newJarak & ",
WaktuJamFromLast=" & newWaktu & _
" where RouteID=" & vw_RouteTemp!RouteID & " and PoliceNo="" &
vw_RouteTemp!PoliceNo & "" and RouteOrder=" &
vw_RouteTemp!RouteOrder
DoCmd.SetWarnings (False)
DoCmd.RunSQL sqlstr
DoCmd.SetWarnings (True)
End If
```

```
vw_RouteTemp.MoveNext
Wend
End If
vw_RouteTemp.Close
```

'Untuk Customer B saja

If iWrite Then

'Untuk rute B, iteration number sengaja ditambah dengan 1000 untuk
membedakan dari rute B lainnya

```
'sqlstr "delete * from vw_route_temp where RouteDate=cdate("" & trDate &
""") and PoliceNo="" & iPoliceNoB & "" and Iteration=" & iIteration
```

```
DoCmd.RunSQL sqlstr
```

```
sqlstr = "select * from [vw_route_temp] where RouteDate=cdate("" & itrDate &
""") and PoliceNo="" & iPoliceNoB & ""
```

```
vw_RouteTemp.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly
```

```
PrevCustID = "0"
```

```
If Not (vw_RouteTemp.EOF) Then
```

```
While Not (vw_RouteTemp.EOF)
```

```
theCustID = vw_RouteTemp!custID
```

```
theOrderKg = vw_RouteTemp!OrderKG
```

'Jika node ini adalah node yang akan diganti customernya

```
If vw_RouteTemp!RouteOrder = iRouteOrderB Then
```

```
'Jika PrevCustID="" berarti prev=Depot
```

```
theCustID = iNewCustA
```

```
newJarak = FindJarak(PrevCustID, iNewCustA)
```

```
newWaktu = FindWaktuJam(PrevCustID, iNewCustA)
```

```
theOrderKg = iNewOrderKGA
```

```

    bolehtulis = True
    'informasi customer sebelumnya
    PrevCustID = iNewCustA

Else
    'Untuk Node setelah node yang ditukar
    If vw_RouteTemp!RouteOrder = iRouteOrderB + 1 Then
        newJarak = FindJarak(PrevCustID, vw_RouteTemp!custID)
        newWaktu = FindWaktuJam(PrevCustID, vw_RouteTemp!custID)

        bolehtulis = True
    Else
        newJarak = vw_RouteTemp!DistanceFromLast
        newWaktu = vw_RouteTemp!WaktuJamFromLast

        bolehtulis = False
    End If
    'informasi customer sebelumnya
    PrevCustID = vw_RouteTemp!custID

End If

If bolehtulis Then
    'Tulis untuk B
    'sqlstr = "insert into [tr detail route temp] values (RouteID,
    PoliceNo,Iteration,RouteOrder,CustID,OrderKG,DistanceFromLast,WaktuJamFromLast) values (" & _
    'vw_routeTemp!RouteID & "," & vw_routeTemp!PoliceNo & """, " &
    iteration & "," & vw_routeTemp!RouteOrder & """, " & theCustID & """, " &
    theOrderKg & "," & newJarak & "," & newWaktu
    sqlstr = "update [tr detail route temp] set custID="" & theCustID & """,
    OrderKG=" & theOrderKg & ", DistanceFromLast=" & newJarak & ",
    WaktuJamFromLast=" & newWaktu & _
    " where RouteID=" & vw_RouteTemp!RouteID & " and PoliceNo="" &
    vw_RouteTemp!PoliceNo & "" and RouteOrder=" &
    vw_RouteTemp!RouteOrder
    DoCmd.SetWarnings (False)
    DoCmd.RunSQL sqlstr
    DoCmd.SetWarnings (True)
End If
    vw_RouteTemp.MoveNext
Wend
End If
vw_RouteTemp.Close

```

End If

exit_SimulasiPindahNode:

Set vw_RouteTemp = Nothing

DoCmd.SetWarnings (True)

Exit Sub

err_SimulasiPindahNode:

MsgBox Err.Description

Resume exit_SimulasiPindahNode

End Sub

Private Sub OptimasiRute(iRouteID As Long, iPoliceNoA As String)

On Error GoTo err_OptimasiRute

Dim JmlPermutasi As Long

Dim arrCustOccur() As Integer

Dim i As Long

Dim row As Long

Dim JarakTerkecil As Single

Dim idx As Integer

Dim sqlstr As String

Dim OrderKG As Long

Dim JarakFromLast As Single

Dim WaktuFromLast As Single

Dim PrevCustID As String

Dim OldOrderKG As Long

Dim arrCustID() As String

Dim vw_RouteTemp As New ADODB.Recordset

sqlstr = "select * from [tr detail route temp] where RouteID=" & iRouteID & "
and PoliceNo=" & iPoliceNoA & """"

vw_RouteTemp.Open sqlstr, CurrentProject.Connection, adOpenForwardOnly,
adLockReadOnly

If Not (vw_RouteTemp.EOF) Then

vw_RouteTemp.MoveFirst

i = 0

While Not (vw_RouteTemp.EOF)

If vw_RouteTemp!custID <> "0" Then

ReDim Preserve arrCustID(i)

arrCustID(i) = vw_RouteTemp!custID

i = i + 1

End If

vw_RouteTemp.MoveNext

Wend

```

vw_RouteTemp.Close
End If

'nPn = n!
JmlPermutasi = 1
For i = UBound(arrCustID) + 1 To 1 Step -1
    JmlPermutasi = JmlPermutasi * i
Next

'ReDim theArray(JmlPermutasi - 1, UBound(arrCustID))
'theArray(0) = arrCustID

ReDim arrTotJarak(JmlPermutasi - 1)
ReDim arrTotWaktu(JmlPermutasi - 1)

'Array of Customer Occurence-> by default 0
ReDim arrCustOccur(UBound(arrCustID))

IsiArr arrCustID, arrCustOccur, "0", 0, 0, 0, 0

'cari yang totjarak paling kecil
JarakTerkecil = arrTotJarak(0)

For i = 1 To UBound(arrTotJarak)
    If JarakTerkecil > arrTotJarak(i) Then
        idx = i
        JarakTerkecil = arrTotJarak(i)
    End If
Next

PrevCustID = "0"

'tulis ke tabel
For i = 0 To UBound(arrCustID)

    AmbilCustRute2 iRouteID, theArray(idx, i), OldOrderKG
    JarakFromLast = FindJarak(PrevCustID, theArray(idx, i))
    WaktuFromLast = FindWaktuJam(PrevCustID, theArray(idx, i))

    sqlstr = "update [tr detail route temp] set CustID=""" & theArray(idx, i) & """,
    OrderKG=" & OldOrderKG & ", DistanceFromLast=" & JarakFromLast & ",
    WaktuJamFromlast=" & WaktuFromLast & _
    " where RouteID=" & iRouteID & " and PoliceNo=""" & iPoliceNoA & """" and
    RouteOrder=" & i + 1

    DoCmd.SetWarnings (False)
    DoCmd.RunSQL sqlstr
    DoCmd.SetWarnings (True)

```

```

'Pulang ke Depot
If i = UBound(arrCustID) Then
    JarakFromLast = FindJarak(theArray(idx, i), "0")
    WaktuFromLast = FindWaktuJam(theArray(idx, i), "0")

    sqlstr = "update [tr detail route temp] set CustID=""0"", OrderKG=0,
DistanceFromLast=" & JarakFromLast & ", WaktuJamFromlast=" &
WaktuFromLast & _
    " where RouteID=" & iRouteID & " and PoliceNo="" & iPoliceNoA & ""
and RouteOrder=" & i + 2

    DoCmd.SetWarnings (False)
    DoCmd.RunSQL sqlstr
    DoCmd.SetWarnings (True)
End If

PrevCustID = theArray(idx, i)
Next

exit_OptimasiRute:
Set vw_RouteTemp = Nothing
Exit Sub
err_OptimasiRute:
MsgBox Err.Description
Resume exit_OptimasiRute

End Sub
Private Sub IsiArr(iarrCustID() As String, ByRef iarrOccur() As Integer,
iPrevCustID As String, TotJarak As Single, TotWaktu As Single, ByRef row As
Long, idx As Long)
Dim i As Long
Dim j As Long
Dim Jarak As Single
Dim Waktu As Single

If idx = UBound(iarrCustID) Then
For i = 0 To UBound(iarrCustID)
'Cari Cust yang belum terisi
If (iarrOccur(i) = 0) Then
theArray(row, idx) = iarrCustID(i)

arrTotJarak(row) = TotJarak + FindJarak(iPrevCustID, iarrCustID(i))
arrTotWaktu(row) = TotJarak + FindWaktuJam(iPrevCustID, iarrCustID(i))

'Tambahkan ke Depot
arrTotJarak(row) = arrTotJarak(row) + FindJarak(iarrCustID(i), "0")
arrTotWaktu(row) = arrTotWaktu(row) + FindWaktuJam(iarrCustID(i), "0")

```

```

iarrOccur(i) = 0

'theArray(row, j + 1) = "0"

'untuk baris berikutnya
row = row + 1
For j = 0 To UBound(iarrCustID)
    theArray(row, j) = theArray(row - 1, j)
Next

End If
Next
Else
For i = 0 To UBound(iarrCustID)
'Cari Cust yang belum terisi
If (iarrOccur(i) = 0) Then
theArray(row, idx) = iarrCustID(i)

Jarak = FindJarak(iPrevCustID, iarrCustID(i))
Waktu = FindWaktuJam(iPrevCustID, iarrCustID(i))

iarrOccur(i) = 1
IsiArr iarrCustID, iarrOccur, iarrCustID(i), TotJarak + Jarak, TotWaktu +
Waktu, row, idx + 1
iarrOccur(i) = 0
End If
Next
End If

End Sub

'private function GetCustsRoute (itrDate as Date,
Private Sub AmbilCustRute2(iRouteID As Long, iCustID As String, ByRef
oOrderKG As Long)
On Error GoTo err_AmbilCustRute2

Dim vwRoute As New ADODB.Recordset
Dim sqlstr As String

sqlstr = "select CustID,RouteOrder, OrderKG from vw_route where RouteID=" &
iRouteID & " and CustID=""" & iCustID & """"
vwRoute.Open sqlstr, CurrentProject.Connection, adOpenDynamic,
adLockReadOnly

oOrderKG = 0

If Not (vwRoute.EOF) Then

```

```

' oRouteOrder = vwRoute!RouteOrder
  oOrderKG = vwRoute!OrderKG
End If
vwRoute.Close

exit_AmbilCustRute2:
  Set vwRoute = Nothing
  Exit Sub
err_AmbilCustRute2:
  MsgBox Err.Description
  Resume exit_AmbilCustRute2
End Sub
Private Sub CopyRuteToFinal(iRouteID As Long, iIteration As Long)
On Error GoTo err_CopyRuteToFinal
Dim sqlstr As String

sqlstr = "insert into [tr detail route final]
(routeID,PoliceNo,Iteration,RouteOrder,CustID,OrderKG,DistanceFromLast,WaktuJamFromLast) " & _
"SELECT [Tr Detail Route Temp].RouteID, [Tr Detail Route Temp].PoliceNo, "
& iIteration & ", [Tr Detail Route Temp].RouteOrder, [Tr Detail Route
Temp].CustID, [Tr Detail Route Temp].OrderKG, [Tr Detail Route
Temp].DistanceFromLast, [Tr Detail Route Temp].WaktuJamFromLast " & _
"FROM [Tr Detail Route Temp] where RouteID=" & iRouteID

DoCmd.SetWarnings (False)
DoCmd.RunSQL sqlstr
DoCmd.SetWarnings (True)

exit_CopyRuteToFinal:
  Exit Sub
err_CopyRuteToFinal:
  MsgBox Err.Description
  Resume exit_CopyRuteToFinal
End Sub
Private Sub PrepareFinalRoute(iRouteID As Long)
On Error GoTo err_PrepareFinalRoute
Dim sqlstr As String

sqlstr = "delete * from [tr detail route final] where RouteID=" & iRouteID

DoCmd.SetWarnings (False)
DoCmd.RunSQL sqlstr
DoCmd.SetWarnings (True)

exit_PrepareFinalRoute:
  Exit Sub
err_PrepareFinalRoute:

```

```
MsgBox Err.Description
Resume exit_PrepareFinalRoute
End Sub
Private Sub cmd_ViewFinal_Click()
On Error GoTo Err_cmd_ViewFinal_Click

Dim stDocName As String
Dim stLinkCriteria As String

stDocName = "vw_Route_Jumlah_Iter_Final"

'stLinkCriteria = "[RouteDate]=" & "#" & Me![TransDate] & "#"
stLinkCriteria = "[RouteDate]=" & "cdate(" & Me![TransDate] & ")"
DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_cmd_ViewFinal_Click:
Exit Sub

Err_cmd_ViewFinal_Click:
MsgBox Err.Description
Resume Exit_cmd_ViewFinal_Click

End Sub
```