



**PENYELESAIAN PERMASALAHAN *OPEN VEHICLE*
ROUTING WITH TIME WINDOWS DENGAN
*ALGORITMA IMPROVED DISCRETE BAT***

SKRIPSI

Oleh

**Livia Ula Mahmudah
NIM 161810101031**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2020**



**PENYELESAIAN PERMASALAHAN *OPEN VEHICLE*
ROUTING WITH TIME WINDOWS DENGAN
*ALGORITMA IMPROVED DISCRETE BAT***

SKRIPSI

Disusun guna memenuhi Tugas Akhir sebagai salah satu persyaratan akademik
pada sarjana (S1) Jurusan Matematika Fakultas Matematika dan Ilmu
Pengetahuan Alam Universitas Jember

Oleh :

**Livia Ula Mahmudah
NIM 161810101031**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2020**

PERSEMBAHAN

Segala puji syukur saya ucapkan kepada Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan skripsi ini dengan lancar. Skripsi ini saya persembahkan untuk orang-orang spesial, yaitu:

1. Cinta pertama saya Mama Siti Musarofah dan Ayah Basuki yang telah memberikan kasih sayang, motivasi, doa, dan pengorbanan yang begitu besar.
2. Adik-adik tersayang Mochammad Inggil Permadi, Wisnu Aji Pamungkas, Muhammad Gifari Syafiullah Al Husayn, Silviana Dwi Saputri yang selalu memberikan semangat, keceriaan, dan kebahagiaan.
3. Semua Guru saya dari SDN Sukorejo 2, SMPN 1 Ngasem, SMAN 1 Pare, serta Dosen Jurusan Matematika FMIPA Universitas Jember yang telah memberikan ilmu yang bermanfaat.
4. Almamater tercinta, Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

MOTTO

Saya tidak peduli seberapa miskinnya seorang manusia, jika dia memiliki keluarga, berarti dia kaya. *)

Ketika seseorang menghina kamu, itu adalah sebuah pujian bahwa selama ini mereka menghabiskan banyak waktu untuk memikirkan kamu, bahkan ketika kamu tidak memikirkan mereka. **)

Pertama, mereka mengabaikan anda. Kemudian, mereka tertawa pada anda. Berikutnya, mereka melawan anda. Lalu, anda menang. ***)

*) Dan Wilcox and Thad Mumford

**) B. J. Habibie

***) Mahatma Gandhi

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Livia Ula Mahmudah

NIM. : 161810101031

Menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Penyelesaian Permasalahan *Open Vehicle Routing With Time Windows* dengan Algoritma *Improved Discrete Bat* “ adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, April 2020

Yang menyatakan,

Livia Ula Mahmudah

NIM 161810101031

SKRIPSI

**PENYELESAIAN PERMASALAHAN *OPEN VEHICLE*
ROUTING WITH TIME WINDOWS DENGAN
ALGORITMA *IMPROVED DISCRETE BAT***

Oleh

Livia Ula Mahmudah

NIM. 161810101031

Pembimbing:

Dosen Pembimbing Utama : Abduh Riski, S.Si., M.Si.

Dosen Pembimbing Anggota : Dr. Agustina Pradjaningsih, S.Si, M.Si.

PENGESAHAN

Skripsi berjudul “Penyelesaian Permasalahan *Open Vehicle Routing With Time Windows* dengan Algoritma *Improved Discrete Bat*” telah diuji dan disahkan pada:

Hari, tanggal :

Tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Jember.

Tim Penguji;

Ketua,

Anggota I,

Abduh Riski, S.Si., M.Si.
NIP. 199004062015041001

Dr. Agustina Pradjaningsih, S.Si., M.Si.
NIP. 197108022000032009

Anggota II

Anggota III

Kosala Dwidja Purnomo, S.Si., M.Si.
NIP. 196908281998021001

Dian Anggraeni, S.Si., M.Si.
NIP. 198202162006042002

Mengesahkan
Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Jember

Drs. Achmad Sjaifullah, M.Sc., Ph.D.
NIP. 195910091986021001

RINGKASAN

PENYELESAIAN PERMASALAHAN *OPEN VEHICLE ROUTING WITH TIME WINDOWS* DENGAN ALGORITMA *IMPROVED DISCRETE BAT*;

Livia Ula Mahmudah; 161810101031; 2020; 63 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Open Vehicle Routing Problem with Time Windows (OVRPTW) adalah permasalahan optimasi penentuan rute dengan keterbatasan kapasitas kendaraan, *time windows*, dengan kendaraan tidak kembali ke depot. Data yang digunakan dalam penelitian ini adalah data primer yang diambil dari CV Medipety Nusantara yaitu perusahaan yang memproduksi obat untuk hewan peliharaan, dan mempunyai 35 pelanggan di beberapa daerah Jawa Timur.

Penelitian ini bertujuan untuk meminimumkan biaya transportasi pendistribusian barang menggunakan Algoritma *Improved Discrete Bat*. Penyelesaian dilakukan dengan 2 cara yaitu penyelesaian secara manual dan penyelesaian menggunakan program. Data yang digunakan pada penyelesaian manual yaitu 5 pelanggan daerah Jember dan 3 pelanggan daerah Lumajang. Sedangkan data yang digunakan pada penyelesaian menggunakan program dibedakan menjadi 3 kelompok data yaitu data pelanggan Jember, data pelanggan Jember dan Banyuwangi, dan data semua pelanggan.

Berdasarkan hasil yang telah dilakukan dapat disimpulkan bahwa pada data semua pelanggan diperoleh solusi terbaik dengan total jarak tempuh 621.3 km dengan total biaya Rp1.778.105. Penyelesaian permasalahan OVRPTW menggunakan algoritma IDBA dikatakan optimal karena dapat membantu perusahaan untuk menghemat biaya transportasi sebesar Rp277.105 atau sekitar 13,48% pada data semua pelanggan.

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat dan kuasa-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “Penyelesaian Permasalahan *Open Vehicle Routing With Time Windows* dengan Algoritma *Improved Discrete Bat*”. Penulisan tugas akhir ini disusun guna memenuhi salah satu syarat untuk menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusunan skripsi ini tidak lepas dari bantuan dan dukungan berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terima kasih kepada;

1. Bapak Abduh Riski, S.Si., M.Si. selaku dosen pembimbing utama dan Ibu Dr. Agustina Pradjaningsih, S.Si., M.Si.
2. Bapak Kosala Dwidja Purnomo, S.Si., M.Si. selaku dosen penguji I dan Ibu Dian Anggraeni, S.Si., M.Si. selaku dosen penguji II;
3. Seluruh Dosen dan Karyawan Jurusan Matematika FMIPA Universitas Jember;
4. Keluarga tercinta saya Mama Siti Musarofah, Ayah Basuki, Emak, Uti, Kakung, serta Adik-adik yang selalu memberikan kasih sayang, motivasi, doa, dan semangat;
5. Teman-teman Misdirection'16 yang telah memberikan semangat, bantuan, dan kenangan selama di perkuliahan;
6. Teman-teman GJ SQUAD (Ghea, Rif'a, Putsong, Giki, Sofil, Farah) yang selalu menjadi pengingat di jalan yang benar;
7. Teman-teman kos Ima dan Dini yang selalu menjadi alarm dan teman tidur selama di Jember;
8. Jelita Ahmaria, Shinta Octavianty, Dila Ayu, dan Ananda Koespita teman tersayang yang telah menjadi tempat curhat ternyaman;
9. Partner Pelaksana Bidang Musik Bayu Nalendra Utama alias Mas Korek yang telah menjadi partner yang selalu ada dan partner bertukar fikir dalam segala hal;

10. Ahmad Faisal Rosidi yang selalu memberi semangat dan perhatian tanpa bosan;
11. Keluarga Besar UKMS TITIK yang telah mewarnai masa perkuliahan menjadi sangat indah dan memberi ilmu yang bermanfaat dalam bidang seni maupun organisasi;
12. Serta semua pihak yang tidak dapat disebutkan satu persatu yang telah membantu dalam penyelesaian skripsi ini.

Penulis menyadari bahwa penelitian ini masih jauh dari sempurna. Oleh karena itu penulis mengharap kritik dan saran demi kesempurnaan skripsi ini. Semoga skripsi ini dapat bermanfaat untuk kita semua.

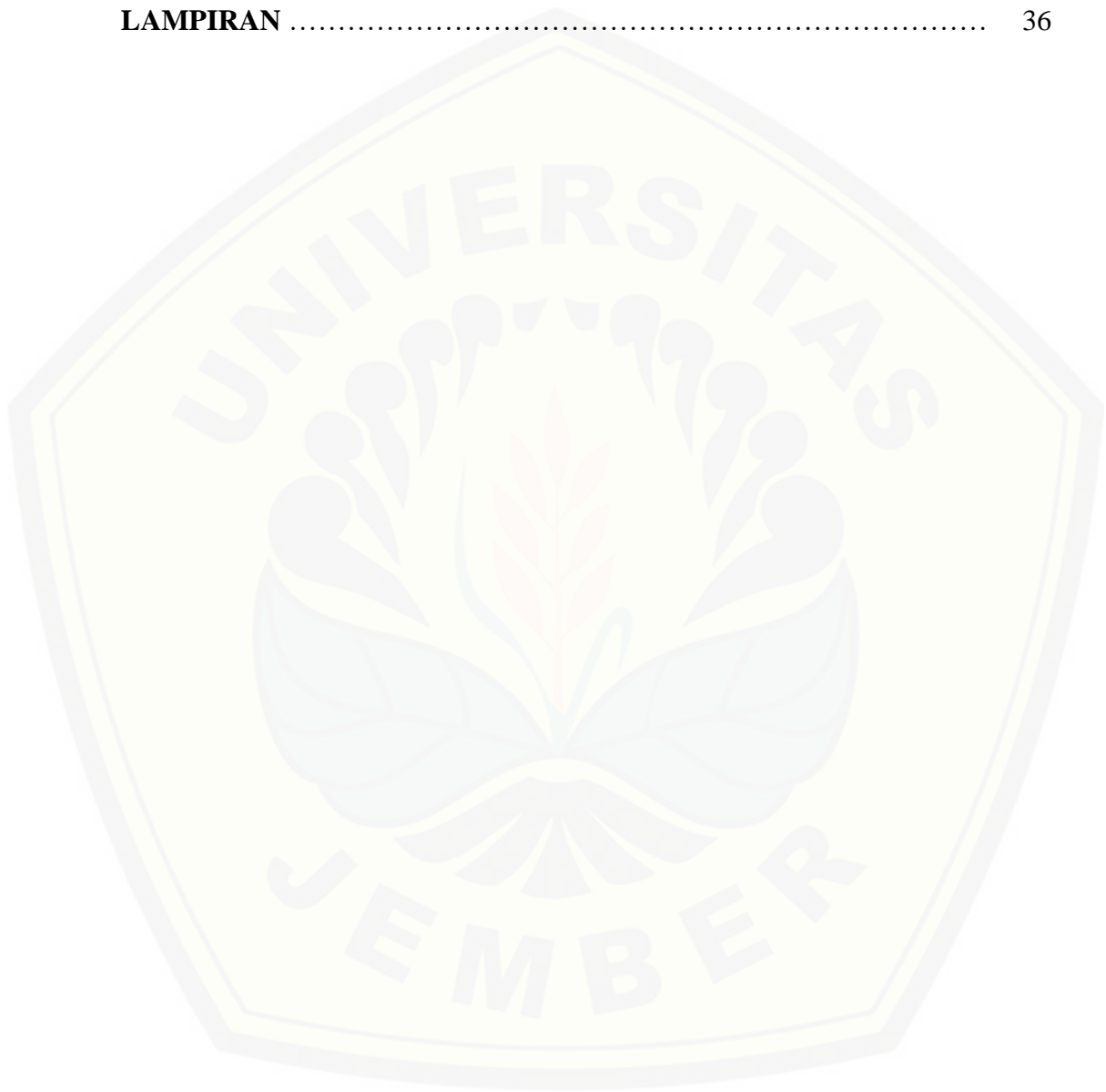
Jember, April 2020

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAN	iv
HALAMAN PEMBIMBINGAN	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
BAB 1. PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
BAB 2. TINJAUAN PUSTAKA	
2.1 <i>Vehicle Routing Problem (VRP)</i>	4
2.2 <i>Open Vehicle Routing Problem Time Window (OVRPTW)</i> ..	7
2.3 <i>Bat Algorithm (BA)</i>	8
2.4 <i>Improved Discrete Bat Algorithm (IDBA)</i>	9
BAB 3. METODE PENELITIAN	
3.1 Data Penelitian	11
3.2 Langkah Penelitian	11
BAB 4. HASIL DAN PEMBAHASAN	
4.1 Hasil	15
4.1.1 Penyelesaian Manual	15
4.1.2 Penyelesaian Menggunakan Program	27
4.2 Pembahasan	31

BAB 5. KESIMPULAN DAN SARAN	
5.1 Kesimpulan	34
5.2 Saran	34
DAFTAR PUSTAKA	35
LAMPIRAN	36

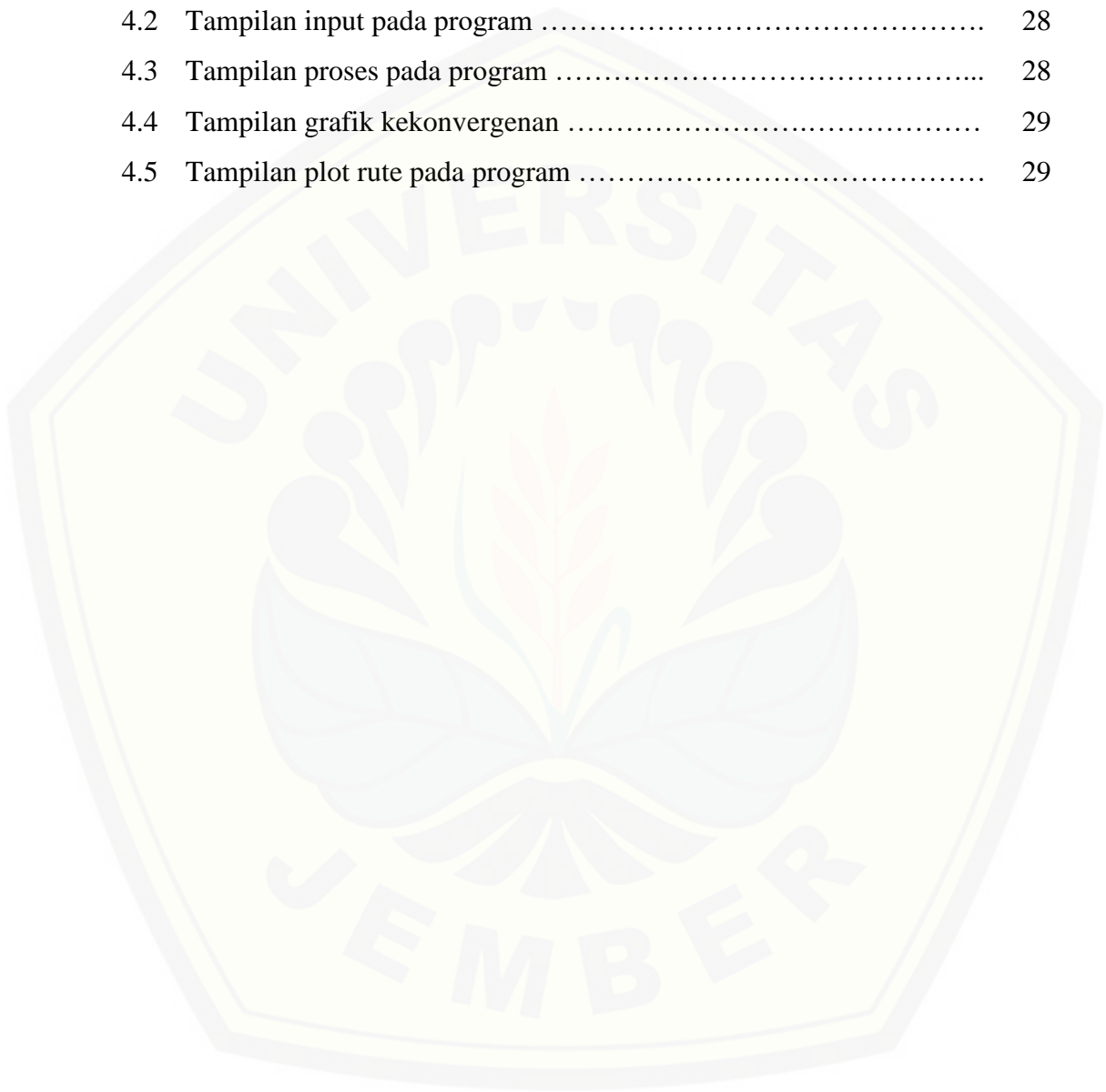


DAFTAR TABEL

	Halaman
4.1 Tabel data depot dan delapan pelanggan	16
4.2 Tabel jarak pelanggan (km)	17
4.3 Tabel waktu perjalanan (jam)	17
4.4 Tabel hasil inisialisasi parameter	18
4.5 Tabel nilai <i>time windows</i>	19
4.6 Tabel hasil nilai <i>random X</i>	19
4.7 Tabel hasil pembentukan rute dan fungsi tujuan	22
4.8 Tabel nilai <i>random v_i</i>	23
4.9 Tabel hasil pembentukan solusi baru	25
4.10 Tabel nilai <i>pulse rate (r_i)</i> dan <i>loudness (A_i)</i>	26
4.11 Tabel hasil <i>running</i> program data pelanggan Jember	30
4.12 Tabel hasil <i>running</i> program data pelanggan Jember dan Banyuwangi	30
4.13 Tabel hasil <i>running</i> program data semua pelanggan	31

DAFTAR GAMBAR

	Halaman
3.1 Skema metode penelitian	13
4.1 Tampilan awal program	27
4.2 Tampilan input pada program	28
4.3 Tampilan proses pada program	28
4.4 Tampilan grafik kekonvergenan	29
4.5 Tampilan plot rute pada program	29



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Pembangunan ekonomi merupakan salah satu aspek penting yang perlu diperhatikan di suatu negara. Dalam hal ini pelaku ekonomi memiliki peran penting terhadap pembangunan ekonomi. Pelaku ekonomi merupakan pihak yang melakukan kegiatan ekonomi berupa produksi, distribusi, dan konsumsi. Dalam kegiatan produksi, logistik memiliki pengaruh yang cukup besar. Logistik adalah pengadaan, transportasi, manajemen persediaan, dan aktifitas pergudangan dalam menyediakan alat untuk memenuhi kebutuhan pelanggan baik internal maupun eksternal.

Logistik yang efisien dapat meningkatkan produktivitas suatu perusahaan. Hal ini dapat dilakukan salah satunya yaitu dengan sistem pendistribusian yang baik. Distribusi merupakan kegiatan pengiriman barang dari produsen kepada konsumen. Sistem distribusi barang menyangkut penyaluran dan penyebaran barang-barang melalui berbagai saluran dari produsen ke konsumen. Permasalahan sistem distribusi bermacam-macam, diantaranya terbatas oleh waktu dan kapasitas. Apabila tidak dipikirkan dengan baik, hal ini dapat menyebabkan perusahaan mengeluarkan biaya transportasi yang banyak. Oleh karena itu diperlukan suatu strategi dalam pendistribusian barang produksi. Strategi yang dilakukan dengan cara merencanakan rute perjalanan distribusi untuk meminimalisir biaya transportasi. Permasalahan seperti ini terdapat pada perusahaan yang tidak memiliki kendaraan untuk pendistribusian barang produksi, sehingga untuk memenuhi permintaan pelanggan perusahaan menggunakan kendaraan sewa. Rute perjalanan yang digunakan kendaraan sewa adalah berakhir pada pelanggan terakhir dan tidak kembali ke depot.

Vehicle Routing Problem (VRP) adalah permasalahan optimasi penentuan rute dengan keterbatasan kapasitas kendaraan. Tujuan VRP adalah untuk meminimumkan jarak tempuh perjalanan dan biaya total perjalanan tanpa melebihi kapasitas kendaraan. Semua permintaan pelanggan harus terpenuhi dan setiap pelanggan dilayani oleh satu kendaraan.

Open Vehicle Routing Problem with Time Windows (OVRPTW) adalah salah satu cabang dari VRP. Permasalahan OVRPTW diberi tambahan *time windows*, dan kendaraan tidak kembali ke depot. *Time windows* berupa waktu tempuh perjalanan dan waktu pelayanan.

Penelitian tentang VRPTW pernah dilakukan sebelumnya oleh beberapa peneliti menggunakan Algoritma metaheuristic diantaranya yaitu Yansyah (2016) membahas tentang Permasalahan pengangkutan sampah VRPTW menggunakan dua algoritma yaitu algoritma *Cheapest Insertion Heuristic* (CIH) dan *Cheapest Insertion Heuristic with Savings Method*. Taqwiya (2017) yang membahas tentang aplikasi Algoritma Genetika pada kasus VRPTW. Suyono (2018) membahas tentang Penyelesaian *Biobjective Vehicle Routing Problem With Time Windows* Dengan Algoritma Genetika dan Algoritma *Cat Swarm Optimization*.

Selain algoritma di atas terdapat algoritma lain yang dapat digunakan untuk menyelesaikan masalah ini yaitu *Improved Discrete Bat Algorithm* (IDBA). Algoritma ini merupakan algoritma optimasi yang terinspirasi dari perilaku kelelawar. Osaba, *et al* (2016) telah mengembangkan BA menjadi IDBA.

Dari uraian di atas, IDBA memiliki potensi untuk diaplikasikan pada OVRPTW karena algoritma ini hanya memilih solusi paling baik dari perhitungannya dengan analogi kelelawar yaitu ketika memangsa menentukan lokasi bertengger serta menghindari rintangan saat terbang dengan menggunakan gelombang suara yang dikeluarkannya yang disebut ekolokasi. Penulis berharap penerapan algoritma ini ke dalam OVRPTW dapat menghasilkan solusi yang optimal untuk mencari rute minimum kendaraan pada data real suatu perusahaan yang menggunakan kendaraan sewa.

1.2 Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini adalah sebagai berikut:

- a. Bagaimana cara menerapkan IDBA dalam menyelesaikan permasalahan OVRPTW?
- b. Bagaimana hasil pada permasalahan OVRPTW menggunakan IDBA?

1.4 Tujuan

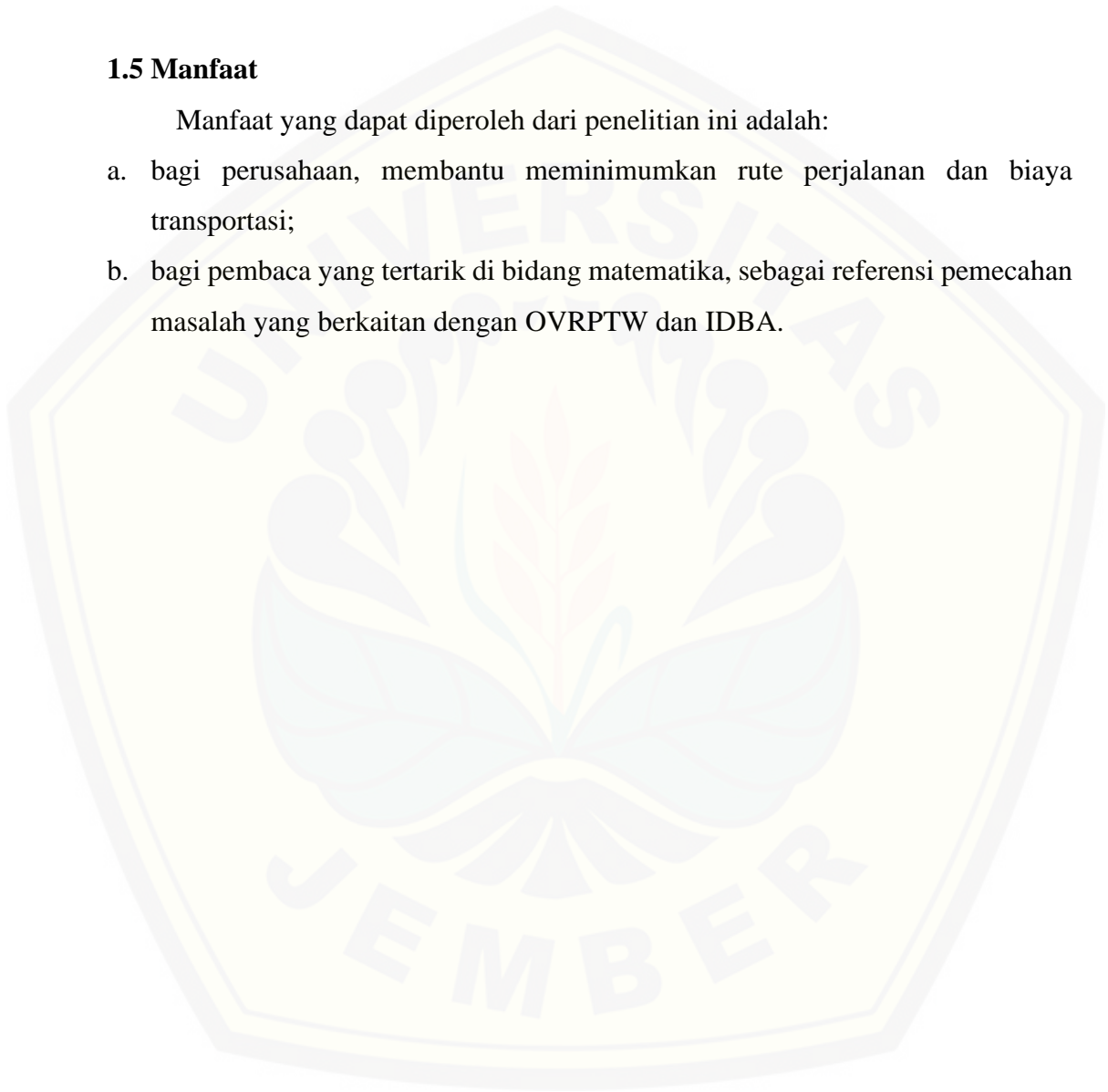
Tujuan dari penelitian ini adalah sebagai berikut:

- a. Untuk mengetahui cara penerapan IDBA pada permasalahan OVRPTW.
- b. Untuk mengetahui hasil pada permasalahan OVRPTW menggunakan IDBA.

1.5 Manfaat

Manfaat yang dapat diperoleh dari penelitian ini adalah:

- a. bagi perusahaan, membantu meminimumkan rute perjalanan dan biaya transportasi;
- b. bagi pembaca yang tertarik di bidang matematika, sebagai referensi pemecahan masalah yang berkaitan dengan OVRPTW dan IDBA.



BAB 2. TINJAUAN PUSTAKA

2.1 *Vehicle Routing Problem (VRP)*

Vehicle Routing Problem (VRP) pertama kali diperkenalkan oleh Dantzig dan Ramser, pada tahun 1959. VRP adalah sebuah permasalahan yang terdapat pada kendaraan untuk mencari rute-rute yang menghubungkan depot ke beberapa pelanggan. Rute tersebut merupakan rute yang memenuhi setiap kendala yang berkaitan dengan masalah kapasitas kendaraan, permintaan pelanggan, waktu, dan sebagainya. Setiap pelanggan dalam permasalahan VRP dikunjungi oleh tepat satu kendaraan (Moolman, *et al*, 2010).

Berikut ini merupakan komponen-komponen yang terdapat pada *vehicle routing problem (VRP)*:

a. Jaringan Kerja (*link*)

Jaringan kerja (*link*) merupakan setiap jalan yang tersedia dalam transportasi pada suatu rute. Setiap *link* berkaitan dengan, jarak tempuh, jenis kendaraan dan waktu perjalanan yang dilakukan. *Link* mempengaruhi biaya transportasi yang digunakan..

b. *Costumers*

Karakteristik khusus dari *costumers* adalah sebagai berikut:

- 1) Jumlah permintaan (*demand*) dari *costumers* berbeda-beda, yaitu jumlah permintaan yang diketahui secara pasti (kasus deterministik) dan juga jumlah permintaan yang diketahui tidak pasti (kasus stokastik).
- 2) Ada *costumers* yang mempunyai *time windows* yaitu periode waktu yang menunjukkan jangka waktu *costumers* dapat dilayani.

c. Depot

Depot merupakan titik awal dari suatu rute yang akan dilewati oleh kendaraan dalam melakukan pengiriman barang ke *costumers*. Setiap depot memiliki banyak kendaraan serta jumlah barang produksi yang tersedia.

d. Kendaraan (*vehicle*)

Karakteristik khusus dari kendaraan (*vehicle*) adalah sebagai berikut:

- 1) Mempunyai kapasitas kendaraan maksimum (berat dan volume maksimum) dalam mengangkut barang.
 - 2) Mempunyai total waktu kerja dari awal keberangkatan dari depot sampai kedatangan pelanggan terakhir, sesuai peraturan yang diberlakukan oleh perusahaan sesuai jam operasionalnya.
 - 3) Memerlukan biaya untuk melakukan pengiriman, biaya penggunaan kendaraan dihitung berdasarkan per unit jarak, per unit waktu, dan per rute.
- e. Pengemudi (*driver*)

Pengemudi yang mengoperasikan kendaraan harus memenuhi semua kendala yang ditetapkan dalam kontrak kerja dan aturan dari perusahaan. (Maria, 2000).

Berikut ini merupakan jenis-jenis cabang *vehicle routing problem* (VRP):

a. *Capacitated VRP* (CVRP)

CVRP atau *Capacitated Vehicle Routing Problem* adalah sebuah VRP yang diberikan sejumlah kendaraan dengan kapasitas tersendiri dan harus melayani sejumlah permintaan pelanggan yang telah diketahui dengan biaya transit minimum. Tujuan dari CVRP adalah meminimalisir jumlah kendaraan dan total waktu perjalanan, dan total permintaan barang untuk tiap rute tidak boleh melebihi kapasitas kendaraan yang melewati rute tersebut.

b. *VRP With Time Windows* (VRPTW)

VRPTW atau *Vehicle Routing Problem with Time Window* memiliki batas tambahan yaitu sebuah jangka waktu pelanggan harus disuplai, dan interval waktu di depot tersebut sebagai batas penjadwalan. Tujuannya adalah meminimalisir jumlah kendaraan dan total waktu perjalanan serta waktu menunggu yang dibutuhkan untuk menyuplai semua pelanggan pada jam-jam tertentu. Rute harus mulai dan berhenti dalam jangka waktu yang berkaitan dengan depot. OVRPTW adalah model VRPTW dengan rute terbuka atau kendaraan tidak kembali ke depot. OVRPTW akan dibahas lebih detail pada subbab 2.2.

c. *Multiple Depot VRP* (MDVRP)

MDVRP membutuhkan pengaturan para pelanggan ke depot-depot yang ada. Tiap kendaraan pergi dari satu depot melayani pelanggan-pelanggan yang sudah

ditentukan akan dilayani oleh depot tersebut, dan kembali lagi ke depot tersebut. Tujuan *MDVRP* adalah meminimalisasir jumlah kendaraan dan total waktu perjalanan dan total permintaan barang yang harus dilakukan dari beberapa depot.

d. *VRP With Pick-Up and Delivering (VRPPD)*

Vehicle Routing Problem with Pick-Up and Delivering atau *VRPPD* adalah sebuah *VRP* yang memiliki peluang kejadian pelanggan mengembalikan barang yang sudah diantarkan. Tujuannya adalah meminimalisasir jumlah kendaraan dan total waktu perjalanan dengan batasan bahwa kendaraan yang digunakan harus punya kapasitas yang cukup untuk mengantarkan barang ke pelanggan dan pengembalian barang ke depot.

e. *Split Delivery VRP (SDVRP)*

Split Delivery Vehicle Routing Problem atau *SDVRP* adalah perluasan *VRP* jika tiap pelanggan dapat dilayani dengan kendaraan yang berbeda andaikan biayanya dapat berkurang. Perluasan ini dilakukan jika jumlah permintaan pelanggan sama besar dengan kapasitas dari kendaraan. Tujuannya untuk meminimalisasir jumlah kendaraan dan total waktu perjalanan untuk pelayanan.

f. *Stochastic VRP (SVRP)*

Stochastic Vehicle Routing Problem atau *SPRV* adalah variasi *VRP* yang terjadi jika faktor sampingnya yang muncul bersifat acak. Dalam *SPRV* untuk mendapatkan solusi, masalah harus dibagi menjadi dua tahap. Solusi pertama ditentukan sebelum variabel random diketahui. Pada tahap kedua, pengoreksian dilakukan jika nilai dari variabel random sudah diketahui. Tujuannya adalah meminimalisasir jumlah kendaraan dan total waktu perjalanan untuk melayani pelanggan dengan nilai random untuk tiap pengantaran (pelanggan, permintaan, waktu).

g. *Periodic VRP*

Periodic Vehicle Routing Problem atau *PVRP*, *VRP* digeneralisasi dengan memperluas rentang perencanaan pengiriman menjadi M hari, dari semula hanya dalam rentang sehari. Tujuannya adalah meminimalisasir jumlah kendaraan dan total waktu perjalanan untuk melayani tiap pelanggan. Ditambah

dengan keadaan bahwa sebuah kendaraan tidak boleh kembali ke depot pada satu hari yang sama (Prana, 2007).

2.2 Open Vehicle Routing Problem Time Windows (OVRPTW)

Open Vehicle Problem With Time Windows (OVRPTW) adalah salah satu variasi dari *Vehicle Routing Problem* (VRP) dengan mempertimbangkan batasan waktu dan kendala kendaraan tidak kembali ke depot. Rute yang terbentuk pada OVRPTW berawal dari depot dan berakhir pada pelanggan terakhir. Sedangkan batasan waktu atau *time windows* pada OVRPTW yang digunakan pada tiap pelanggan adalah *range* waktu dimana pelanggan dapat menerima barang kiriman. *Time windows* terdiri dari waktu buka dan waktu tutup pelanggan dan depot. Pada varian ini juga terdapat waktu pelayanan, yang mempresentasikan waktu yang dibutuhkan untuk melayani pelanggan, dapat berupa waktu untuk menurunkan barang dari setiap kendaraan.

Variabel keputusan untuk meminimumkan total jarak tempuh kendaraan adalah

$$x_{ij}^k = \begin{cases} 1 & , \text{jika kendaraan } k \text{ dari pelanggan } i \text{ ke pelanggan } j \\ 0 & , \text{jika lainnya} \end{cases}$$

Dengan fungsi tujuan:

$$\text{Meminimumkan } f(x) = K*Op + BBM* \sum_{k=1}^K \sum_{i=0}^N \sum_{j=0}^N c_{ij} x_{ij}^k \quad (2.1)$$

Dan fungsi kendala:

$$\sum_{k=1}^K \sum_{i=0}^N x_{ij}^k = 1; j = 1, 2, \dots, N \quad (2.2)$$

$$\sum_{i=0}^N \sum_{j=1}^N x_{ij}^k q_j \leq Q; k = 1, 2, \dots, K \quad (2.3)$$

$$\sum_{j=0}^N x_{oj}^k = 1; k = 1, 2, \dots, K \quad (2.4)$$

$$\sum_{i=0}^N x_{ih}^k - \sum_{j=0}^N x_{hj}^k = 0; h = 1, 2, \dots, N; k = 1, 2, \dots, K \quad (2.5)$$

$$\sum_{i=1}^n t_i + \sum_{i=0}^n \sum_{j=0}^n t_{ij} \leq T_w \quad (2.6)$$

$$b_i \leq t_i \leq e_i \quad (2.7)$$

$$x_{ij}^k \in \{0,1\}; i, j = 0, 1, \dots, N; k = 1, 2, \dots, K \quad (2.8)$$

Persamaan 2.1 menjelaskan fungsi tujuan perjalanan dari depot kemudian mengunjungi semua pelanggan. Kendala (2.2) menjelaskan setiap pelanggan hanya

dapat dilayani tepat satu kali. Kendala (2.3) menjelaskan total permintaan tidak boleh melebihi kapasitas kendaraan. Kendala (2.4) – (2.5) menjelaskan setiap kendaraan berangkat dari depot dan setelah melayani pelanggan maka kendaraan akan meninggalkan tempat. Kendala (2.6) – (2.7) menjelaskan pengiriman barang dilakukan selama jam operasional depot. Kendala (2.8) merupakan variabel keputusan.

(Kallehauge, *et al*, 2001).

2.3 Bat Algorithm (BA)

Bat Algorithm (BA) merupakan algoritma metaheuristik yang terinspirasi dari perilaku kelelawar. BA diperkenalkan oleh Xin She Yang pada tahun 2010. Kelelawar ketika memangsa, menentukan lokasi bertengger serta menghindari rintangan saat terbang dengan menggunakan gelombang suara yang dikeluarkannya yang disebut ekolokasi.

Beberapa asumsi yang harus diterapkan dalam mengimplementasikan algoritma kelelawar yaitu:

- a. Semua kelelawar menggunakan ekolokasi untuk memperkirakan jarak dan dapat membedakan antara mangsa dan rintangan
- b. Setiap kelelawar terbang dengan kecepatan yang tidak tetap (*random*) pada posisi dan frekuensi tertentu
- c. Tingkat kekerasan dan *pulse rate* gelombang suara tiap kelelawar berubah-ubah bergantung pada kedekatan jarak dengan target (Yang, 2010).

Beberapa tahapan dalam algoritma kelelawar yaitu:

- a. Ekolokasi

Ekolokasi atau disebut juga biosonar adalah sonar biologi yang digunakan oleh kelelawar. Binatang yang memiliki kemampuan ekolokasi mengeluarkan bunyi dan mendengarkan pantulan bunyi tersebut yang dipantulkan oleh objek – objek di sekitarnya. Dengan menggunakan bunyi pantulan tersebut, binatang itu bisa mengidentifikasi keberadaan objek. Ekolokasi digunakan binatang sebagai alat navigasi untuk berburu. Gelombang suara yang dipancarkannya sangat beragam bergantung pada

target dan jenis spesiesnya serta kelelawar cenderung menggunakan frekuensi yang konstan. Beberapa jenis kelelawar memiliki penglihatan yang baik serta indera penciuman yang tajam, tetapi mereka tetap menggunakan ekolokasi pada saat terbang atau memadukan antara indera penglihatan dan penciuman.

b. Pergerakan kelelawar (*Movement of Bats*)

Setiap kelelawar bergantung pada kecepatan v_i^t dan lokasi x_i^t dengan menyesuaikan frekuensi Q_i pada iterasi t dalam dimensi d pada solusi. Diantara seluruh kelelawar terdapat satu solusi (posisi) global terbaik sementara x_g .

c. Pencarian solusi local (*local search*)

Pada proses pencarian solusi local, dimulai dengan penentuan solusi terbaik dari setiap kelelawar yang bersesuaian atau sering disebut sebagai *personal best bat* x_p .

d. Perubahan kebisingan (*loudness*) dan *pulse rate*

Setiap kelelawar memiliki kebisingan (A_i) dan *pulse rate* r_i yang berbeda-beda dan akan diperbaharui sesuai dengan proses iterasi berlangsung. Ketika *pulse rate* meningkat dan kebisingan gelombang suara yang dikeluarkan menurun, saat itulah kelelawar semakin dekat dengan target (Yang, 2010).

2.4 Improved Discrete Bat Algorithm (IDBA)

Improved Discrete Bat Algorithm merupakan pengembangan dari Algoritma *BA (Bath Algorithm)*. IDBA diperkenalkan oleh Eneko, *et al* pada tahun 2015. Algoritma ini merupakan hasil perbaikan yang lebih efektif dari algoritma kelelawar sebelumnya. Menurut filosofi dasar BA, semua kelelawar melakukan gerakan dengan cara yang sama (Suharto, *et al*, 2015). Strategi ini tidak digunakan dalam IDBA, namun kelelawar dianugerahi "kecerdasan" tertentu. Dengan cara ini, kelelawar menggunakan skema pergerakan yang berbeda tergantung pada titik ruang solusi di mana mereka berada. Semua parameter yang terkait dengan masing-masing kelelawar diinisialisasi dan ditentukan. Parameter tersebut adalah

kecepatan v_i , *pulse rate* r_i dan *loudness* A_i . *Loudness* A_i biasanya menurun saat kelelawar menemukan mangsa, sedangkan tingkat denyut emisi r_i meningkat.

Ketika $A_0 = 1 \rightarrow$ *loudness* awal

$A_{min} = 0 \rightarrow$ *loudness* akhir \rightarrow kelelawar menemukan mangsa (solusi optimal)

Persamaan yang digunakan adalah sebagai berikut:

$$A_i^{t+1} = \alpha A_i^t \quad (2.9)$$

$A_i \rightarrow 0$ (mendekati 0)

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (2.10)$$

Dimana $t \rightarrow \infty, 0 < \alpha < 1, \gamma > 0$

A_i dan r_i digunakan untuk meindahakan algoritma dari *exploration* (pencarian global) ke *exploitation* (pencarian lokal).

A_i, r_i, α , dan γ sangat berpengaruh terhadap proses perpindahan, yaitu :

1. Jika perpindahan dari *exploration* ke *exploitation* lambat, maka algoritma semakin lama untuk konvergen, karena solusi terus teracak.
2. jika perpindahan tersebut terlalu cepat, maka algoritma akan terjebak pada *local optimum*, karena algoritma lebih banyak melakukan pencarian lokal dibanding global. Tidak mencakup keseluruhan kemungkinan solusi.

$A_i = [0,7, 1]$; $r_i = [0, 0,4]$; $\alpha = \gamma = 0,98$ adalah nilai terbaiknya (Yang, 2010).

Perhitungan banyak solusi dilakukan sebanyak v_i . Perolehan nilai v_i setiap kelelawar dilakukan secara *random* dengan batas $[a, b]$, dalam pembahasan ini batas bilangan asli yang digunakan adalah $a = 1, b = HammingDistance$. Nilai *HammingDistance* adalah jumlah urutan nilai yang berbeda antara X_i dengan X^* .

Apabila nilai *random* v_i kurang dari $\frac{n}{2}$ maka langkah selanjutnya menggunakan 2 *opt*. Apabila nilai *random* v_i lebih dari $\frac{n}{2}$ maka langkah selanjutnya menggunakan 3 *opt*.

Berikut ini adalah Pseudocode *Improved Discrete Bat Algorithm*:

1. Definisikan fungsi tujuan $f(x)$
2. Inisialisasi parameter
3. Inisialisasi populasi bat $X = X_1, X_2, X_3, \dots, X_n$
4. repeat
5. for setiap bat X_i
6. Bangkitkan solusi baru (v_i, x_i)
7. If $rand > r_i$
8. Pilih satu dari beberapa solusi terbaik
9. Bangkitkan local solusi
10. if $rand < A_i$ dan $f(x_i) < f(x^*)$
11. Terima solusi baru
12. Tingkatkan r_i dan kurangi A_i
13. until kriteria pemberhentian dicapai ($t = Tmax$)
14. Solusi akhir = solusi terbaik

(Osaba, *et al*, 2015).

BAB 3. METODE PENELITIAN

3.1 Data Penelitian

Data yang diambil di penelitian ini adalah data dari CV Medi Pety Nusantara Jember yang memproduksi obat-obatan untuk hewan peliharaan. Barang produksi tersebut dikirimkan ke pelanggan yang ada di beberapa kota di Jawa Timur. Data yang diperlukan yaitu sebagai berikut:

- a. jumlah pelanggan
- b. jarak antara depot dengan pelanggan dan jarak antara pelanggan satu dengan pelanggan lainnya
- c. jumlah permintaan barang setiap toko
- d. waktu buka dan tutup depot dan toko

3.2 Langkah-langkah Penelitian

Langkah-langkah yang dilakukan dalam penelitian ini untuk menyelesaikan masalah *Open Vehicle Routing Problem Time Windows* menggunakan *Improved Discrete Bat Algorithm* adalah:

- a. Studi Literatur

Langkah awal untuk melakukan penelitian ini yaitu melakukan studi literatur dari beberapa jurnal dan skripsi tentang *Open Vehicle Routing Problem Time Windows* dan *Improved Discrete Bat Algorithm*

- b. Pengambilan Data

Pengambilan data dan pengumpulan data diambil dari data primer CV Medipety Nusantara yang bertempat di Kecamatan Tanggul, Kabupaten Jember.

- c. Penginputan Data

Selanjutnya dilakukan penginputan data yang telah diperoleh, data yang digunakan yaitu berupa jarak dari depot ke setiap pelanggan maupun antar pelanggan, waktu tempuh t_i , jam operasional toko (*time windows*), jumlah permintaan setiap pelanggan (q), biaya BBM, dan biaya kendaraan.

d. Perancangan sistem

Perancangan sistem dilakukan dengan cara menghitung secara manual data yang telah diidentifikasi menggunakan *Improved Discrete Bat Algorithm* dengan langkah–langkah sebagai berikut:

- 1) Mendefinisikan fungsi tujuan $f(x)$
- 2) Menginisialisasi parameter
- 3) Menginisialisasi kandidat rute $X = x_1, x_2, \dots, x_n$;
- 4) Membangkitkan solusi baru
- 5) Memilih satu solusi terbaik
- 6) Membangkitkan lokal solusi

e. Implementasi Sistem.

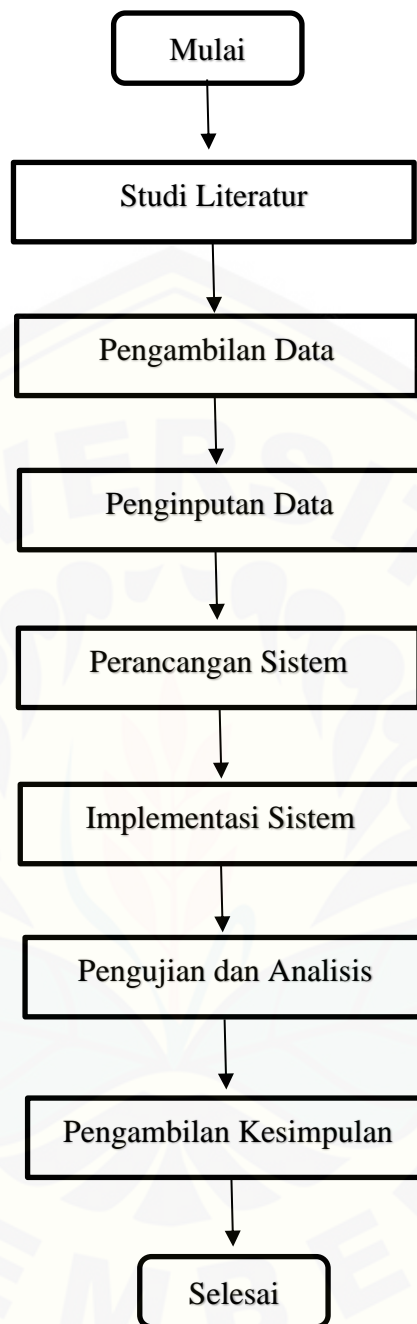
Proses implementasi dilakukan dengan membuat program sesuai dengan perancangan sistem yang telah dilakukan. Program yang akan digunakan adalah MATLAB pada menu *guide*.

f. Pengujian dan Analisis

Pengujian dilakukan dengan menggunakan parameter yaitu kapasitas kendaraan = 240; waktu servis = (0 300), $\alpha = 0,98$; $\gamma = 0,98$; serta NPop dan Iterasi yang bervariasi. Analisis dilakukan untuk mencari hasil yang optimal dan pengaruh parameter pada waktu komputasi program.

g. Pengambilan Kesimpulan

Membuat kesimpulan berdasarkan hasil implementasi program yang diperoleh.



Gambar 3.1 Skema metode penelitian

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan yang telah dilakukan maka dapat diperoleh kesimpulan sebagai berikut:

- a. *Improved Discrete Bat Algorithm* (IDBA) dapat diterapkan pada permasalahan *Open Vehicle Routing With Time Windows* (OVRPTW) dengan langkah-langkah yaitu mendefinisikan fungsi tujuan, inisialisasi parameter, membangkitkan solusi baru, mengevaluasi fungsi tujuan, menentukan solusi terbaik (X^*) sementara, mencari *hamming distance*, membangkitkan nilai *rand* untuk setiap *velocity* (v_i), *loudness* (A_i), dan *pulse rate* r_i , membangkitkan lokal solusi, melakukan update solusi, mengulangi proses dari membandingkan nilai *rand* sampai dengan menentukan solusi terbaik hingga iterasi maksimum terpenuhi.
- b. Hasil implementasi program pada data pelanggan Jember diperoleh solusi terbaik dengan total jarak tempuh 42,9 km dan total biaya Rp286.465, pada data pelanggan Jember dan Banyuwangi diperoleh solusi terbaik dengan total jarak tempuh 192,9 km dan total biaya Rp663.965, pada data semua pelanggan diperoleh solusi terbaik dengan total jarak tempuh 621,3 km dan total biaya Rp1.778.105. Penyelesaian permasalahan OVRPTW menggunakan algoritma IDBA dapat membantu perusahaan untuk menghemat biaya transportasi sebesar Rp277.105 atau sekitar 13,48% pada data semua pelanggan.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, diharapkan penelitian selanjutnya mampu menerapkan *Improved Discrete Bat Algorithm* pada kasus *Vehicle Routing Problem* yang lain seperti *Vehicle Routing With Time Windows* atau *Biobjective Vehicle Routing Problem*, serta diharapkan juga melakukan *hybrid Improved Discrete Bat Algorithm* dengan algoritma lainnya yang memungkinkan mendapatkan hasil yang lebih baik.

DAFTAR PUSTAKA

- Kallehauge, B., J. Larsen dan O. B. G. Madsen. 2001. Lagrangean Duality Applied on Vehicle Routing Problem With Time Window. *Informatic and Mathematical Modelling (IMM)*.
- Maria, L. 2000. *Ant Colony Optimization for vehicle routing in advanced logistics systems*. IDSIA Galleria: Switzerland.
- Moolman, A. J., K. Koen, dan J. V. Wethuizen. 2010. Activity Based Costing For Vehicle Routing Problems. *South African Journal of Industrial Engineering*. 21(2): 161-171.
- Osaba, E., Xin-She, Y., Fernando, D., Pedro, L.G., dan R. Carballedo. 2015. *An Improved Discrete Bat Algorithm for Symetric and Asymmetric Traveling Salesman Problems*. London:University of Deusto.
- Prana, R. 2007. *Aplikasi Kombinatorial pada Vehicle Routing Problem*. Bandung: Jurusan Teknik Informatika, ITB Bandung.
- Suharto., Robandi, I., dan A, Priyadi. 2015. Penalaan *Power System Stabilizer (PSS)* Untuk Perbaikan Stabilitas Dinamik pada Sistem Tenaga Listrik Menggunakan *Bat Algorithm (BA)*. *Jurnal Teknik ITS*. 4(1):337-3539.
- Suyono, L.A. 2018. Penyelesaian Biobjective Vehicle Routing Problem With Time Windows dengan Algoritma Genetika dan Algoritma Cat Swarm Optimization. *Skripsi*. Jember:Universitas Jember.
- Taqwiya, U.A.C. 2017. Aplikasi Algoritma Genetika pada Kasus Vehicle Routing Problem with Time Windows. *Skripsi*. Jember: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.
- Yang, X.S. 2010. *A new metaheuristic bat-inspired algorithm. Nature Inspired Cooperative Strategies for Optimization*. Springer. 284:65-74.
- Yansyah, D.M.2016.Aplikasi Algoritma Cheapest Insertion Heuristic dalam Kasus Vehicle Routing Problem With Time Windows pada Pengangkutan Sampah. *Skripsi*. Jember: Universitas Jember.

LAMPIRAN

A. Data depot dan pelanggan

Node	Nama Toko	Alamat	Jam operasional	Latitude	Longitude	Permintaan
0	CV Medipety Nusantara / Depot	Dusun Krajan. Desa Patemon. Kec Tanggul. Jember	08.00 – 22.00	-8.155036	113.451609	0
1	Blessing petshop	Jl. Pajajaran Blk. Gg. No.12. Lingkungan Sumberdand. Kebonsari. Kec. Sumpersari. Kabupaten Jember	06.00 – 20.00	-8.181901	113.708129	10
2	Hani petshop	Jl. gatot subroto . gang iv. Tembaan. Kapatihan. Kec. Kaliwates. Kabupaten Jember	07.00 – 21.00	-8.173177	113.699185	15
3	Satwa sehat petshop	Jl. Karimata. Gumuk Kerang. Sumpersari. Kec. Sumpersari. Kabupaten Jember	07.00 – 21.00	-8.177745	113.716452	12
4	Satwa sehat 2 petshop	Jl. PB Sudirman No.57. Pagah. Jemberlor. Kec. Patrang. Kabupaten Jember	07.00 – 21.00	-8.162283	113.707162	12
5	Kapron petshop	Jl. Matrip No.52. Krajan Timur. Sumpersari. Kec. Sumpersari. Kabupaten Jember	07.00 – 21.30	-8.159022	113.717844	50
6	Hokky petshop	Jl. Letjen S.Parman No.40. Lingkungan Sadengan. Kebonsari. Kec. Sumpersari. Kabupaten Jember	07.00 – 22.00	-8.182767	113.718195	20
7	Zahra petshop	Jl. Teuku Umar No.114. Tegal Besar Kulon. Tegal Besar. Kec. Kaliwates. Kabupaten Jember	07.00 – 21.00	-8.186718	113.689007	15
8	Pratama petshop	Lingkungan Sumberdand. Kebonsari. Kec. Kaliwates. Kabupaten Jember	09.00 – 21.30	-8.183997	113.703146	25
9	Vendy petshop	Jl. Gajah Mada. Kidul Ps.. Rambipuji. Kec. Rambipuji. Kabupaten Jember	07.00 – 21.00	-8.206218	113.606381	20
10	Phoenix petshop	Jl. Jend. Gatot Subroto No.60. Tembaan. Kapatihan. Kec. Kaliwates. Kabupaten Jember	07.00 – 21.00	-8.171528	113.699344	35

Node	Nama Toko	Alamat	Jam operasional	Latitude	Longitude	Permintaan
11	Sahabat satwa petshop	Jl. KH Imam Bahri No.65. Dusun Kopen. Genteng Kulon. Kec. Genteng. Kabupaten Banyuwangi Dusun Krajan.	10.00 – 21.00	-8.346996	114.151758	20
12	Kanaya petshop	Genteng Wetan. Kec. Genteng. Kabupaten Banyuwangi	08.00 – 21.00	-8.356806	114.160345	40
13	Garfield petshop	Jl. Lugonto. RT. 01 / RW. 01. Rogojampi. Maras. Rogojampi. Kabupaten Banyuwangi	08.00 – 20.00	-8.304013	114.288855	12
14	Anugrah petshop	Dusun Yosowinangun. Jajag. Kec. Gambiran. Kabupaten Banyuwangi	06.00 – 21.00	-8.1243881	114.192603	10
15	Alam jaya petshop	Jl. Moh. Husni Thamrin No.4. Singotrunan. Giri. Kec. Banyuwangi. Kabupaten Banyuwangi	07.30 – 20.00	-8.204373	114.373989	54
16	Twins petshop	Jl. Letjen S Parman No.89. Pakis. Kec. Banyuwangi. Kabupaten Banyuwangi	08.00 – 20.00	-8.240903	114.354139	20
17	Banyuwangi petshop	Kebalenan. Kec. Banyuwangi. Kabupaten Banyuwangi	08.00 – 21.00	-8.233018	114.355556	10
18	Namira petshop	Jl. Nasional III No.18. Dusun Krajan. Kalirejo. Kec. Kabat. Kabupaten Banyuwangi	07.00 – 17.00	-8.245808	114.349138	20
19	Petshop trisula	Jl. Brigpol Sudarlan. Baralian. Nangkaan. Curah Dami. Kabupaten Bondowoso	07.00 -20.00	-7.925789	113.786753	25
20	Cenderawasih petshop	Area Sawah/Kebun. Nangkaan. Kec. Bondowoso. Kabupaten Bondowoso	08.00 – 21.00	-7.924991	113.809173	68
21	Cenderawasih petshop 2	Jl. Ahmad Yani. Kauman. Kotakulon. Kec. Bondowoso. Kabupaten Bondowoso	07.00 – 21.00	-7.913297	113.820152	34
22	Gemini poultry shop	Kademangan Kulon. Kademangan. Kec. Bondowoso. Kabupaten Bondowoso	09.00 – 18.00	-7.915525	113.830223	16
23	Dianufa	Jl. Basuki Rahmat No.134. Mimbaan Tengah. Mimbaan.	09.00 – 16.30	-7.699502	114.020145	102

Node	Nama Toko	Alamat	Jam operasional	Latitude	Longitude	Permintaan
24	Panther poultry petshop	Kec. Panji. Kabupaten Situbondo Jl. Jawa. Krajan Mimbaan. Mimbaan. Kec. Panji. Kabupaten Situbondo	07.00 – 21.00	-7.704641	114.013063	30
25	Sherena aquatic petshop	Jl. WR. Supratman No.69. Mulyautama. Patokan. Kec. Situbondo. Kabupaten Situbondo	10.00 – 21.00	-7.701346	113.998367	26
26	Petshop lumajang	Jl. Raya Pasirian No.218. Joho. Pasirian. Kabupaten Lumajang	07.00 – 21.00	-8.208511	113.117049	15
27	Bintang petshop	Jl. Mayor Jendral Sukartiko No.204. Jogotrunan. Kec. Lumajang. Kabupaten Lumajang	08.00 – 20.00	-8.135127	113.233398	10
28	Nina petshop	Pasar Senggol No. 24B. Jalan Iskandar Muda. Tompokersan. Kecamatan Lumajang. Tompokersan. Kec. Lumajang. Kabupaten Lumajang	06.00 – 22.00	-8.124732	113.224758	15
29	Rumah kucing petshop	Jl. Brigjen Katamso No.24. Mangunharjo. Kec. Mayangan. Kota Probolinggo	08.30 – 20.00	-7.746727	113.220147	40
30	Petshop moly	Jl. RA Kartini No.45. Bangilan. Panggungrejo. Kota Pasuruan	09.00 – 20.00	-7.638868	112.911438	35
31	Belang tiga petshop	Jl veteran kios no 9. Pekuncen. Panggungrejo. Pekuncen. Panggungrejo. Kota Pasuruan.	08.30 – 21.00	-7.642238	112.914075	50
32	Paris petshop	Ruko 2. No. 2B. Jl. Bendungan Sutami. Sumbersari. Kec. Lowokwaru. Kota Malang	08.00 – 21.00	-7.956272	112.613193	40
33	Momo petshop	Jl. Candi Panggung No.32. Mojolangu. Kec. Lowokwaru. Kota Malang	08.00 – 21.00	-7.933795	112.621199	68
34	Kiddie petshop	Jl. Laksda Adi Sucipto No.60. Blimbing. Kec. Blimbing. Kota Malang	08.00 – 21.00	-7.937868	112.646099	50
35	Goldy petshop & pet care grooming	Jl. Bondowoso No.28c. Gading Kasri. Kec. Klojen. Kota Malang	09.00 – 20.00	-7.964591	112.614962	50

B. Data jarak antar pelanggan (km)

D	0	1	2	3	4	5	6	7	8	9	10	11
0	0	33.1	33	33.9	31.8	33.4	33.8	30.3	32.6	19.4	30.8	102
1	33.1	0	2.9	1.5	4	4.5	1.5	3.1	0.8	13.6	3.2	67.6
2	33	2.9	0	3.2	2	3.6	3.2	4	2.3	13	0.5	69.3
3	33.9	1.5	3.2	0	3.6	3	0.9	4.7	2.4	15.6	4.7	67
4	31.8	4	2	3.6	0	1.6	3.5	4.4	2.7	13.3	2.9	69.7
5	33.4	4.5	3.6	3	1.6	0	3.8	5.9	3.6	14.9	4.5	68.7
6	33.8	1.5	3.2	0.9	3.5	3.8	0	4.3	2.4	15.1	4.6	66.1
7	30.3	3.1	4	4.7	4.4	5.9	4.3	0	2.3	10.9	3.4	70.4
8	32.6	0.8	2.3	2.4	2.7	3.6	2.4	2.3	0	12.8	2.4	68.5
9	19.4	13.6	13	15.6	13.3	14.9	15.1	10.9	12.8	0	11.4	82.4
10	30.8	3.2	0.5	4.7	2.9	4.5	4.6	3.4	2.4	11.4	0	68.9
11	102	67.6	69.3	67	69.7	68.7	66.1	70.4	68.5	82.4	68.9	0
12	103	69.2	70.9	68.6	71.3	70.3	67.8	72	70.1	84	70.5	1.7
13	124	90.2	91.9	89.6	92.3	91.3	88.7	93	91.1	105	91.5	22.3
14	113	78.7	80.4	78.1	80.8	79.8	77.2	81.5	79.6	93.5	80	12.8
15	141	107	109	106	109	108	105	110	108	122	108	38.9
16	137	103	105	102	105	104	101	106	104	118	104	34.9
17	137	102	104	102	104	103	101	105	103	117	104	34.5
18	132	98.2	99.9	99.8	102	101	98.9	103	101	115	102	32.4
19	65.2	36.8	35.2	36.1	33.2	38	36.9	37.5	36	45.5	34.8	89
20	62.8	34.5	32.9	33.8	30.9	30.8	34.6	35.2	33.7	43.2	32.5	86.7

D	0	1	2	3	4	5	6	7	8	9	10	11
21	63.8	35.4	33.8	34.7	31.8	31.7	35.6	36.2	34.6	44.2	33.4	87.6
22	66.1	37.8	36.2	37.1	34.1	34.1	37.9	38.5	37	46.5	35.8	90
23	99.6	71.2	69.6	70.5	67.6	67.6	71.4	72	70.5	80	69.2	131
24	99.3	70.9	69.3	70.2	67.3	67.2	71.1	71.7	70.1	79.7	68.9	132
25	99.7	71.4	69.8	70.7	67.7	67.7	71.5	72.1	70.6	80.1	69.3	120
26	55.8	81.1	84	86.6	84.4	82.4	82.6	78.3	83.9	71.1	83.6	154
27	35.2	66.5	65.9	68.5	66.2	67.8	68.4	63.8	65.7	52.9	65.4	126
28	34.8	66.1	65.4	68	65.8	67.4	67.6	63.4	65.2	52.3	64.9	135
29	66.9	98.2	97.5	100	97.9	99.5	100	95.5	97.4	84.6	97.1	181
30	109	140	139	142	140	141	142	137	139	126	139	209
31	110	141	140	143	141	142	143	138	140	127	140	210
32	178	209	208	211	209	210	211	206	208	195	208	278
33	174	205	204	207	205	206	207	202	204	191	204	274
34	171	202	202	204	202	203	204	199	201	189	201	271
35	177	208	208	210	208	210	210	206	208	195	207	277

D	12	13	14	15	16	17	18	19	20	21	22	23
0	103	124	113	141	137	137	132	65,2	62,8	63,8	66,1	99,6
1	69,2	90,2	78,7	107	103	102	98,2	36,8	34,5	35,4	37,8	71,2
2	70,9	91,9	80,4	109	105	104	99,9	35,2	32,9	33,8	36,2	69,6
3	68,6	89,6	78,1	106	102	102	99,8	36,1	33,8	34,7	37,1	70,5
4	71,3	92,3	80,8	109	105	104	102	33,2	30,9	31,8	34,1	67,6
5	70,3	91,3	79,8	108	104	103	101	38	30,8	31,7	34,1	67,6

D	12	13	14	15	16	17	18	19	20	21	22	23
6	67,8	88,7	77,2	105	101	101	98,9	36,9	34,6	35,6	37,9	71,4
7	72	93	81,5	110	106	105	103	37,5	35,2	36,2	38,5	72
8	70,1	91,1	79,6	108	104	103	101	36	33,7	34,6	37	70,5
9	84	105	93,5	122	118	117	115	45,5	43,2	44,2	46,5	80
10	70,5	91,5	80	108	104	104	102	34,8	32,5	33,4	35,8	69,2
11	1,7	22,3	12,8	38,9	34,9	34,5	32,4	89	86,7	87,6	90	131
12	0	21,2	11,1	37,8	32,8	33,4	31,3	90,6	88,3	89,2	91,6	130
13	21,2	0	21,7	16,6	12,6	12,2	10,2	112	109	110	113	109
14	11,1	21,7	0	38,3	34,3	33,9	31,9	100	97,8	98,7	101	130
15	37,8	16,6	38,3	0	4,8	4,4	5,5	104	102	99,5	98	92,1
16	32,8	12,6	34,3	4,8	0	3,2	0,8	102	100	97,9	96,4	99,6
17	33,4	12,2	33,9	4,4	3,2	0	2,6	103	100	98,2	96,8	98,6
18	31,3	10,2	31,9	5,5	0,8	2,6	0	104	99,3	97,1	95,6	98,5
19	90,6	112	100	104	102	103	104	0	2,5	4,8	7,2	40,6
20	88,3	109	97,8	102	100	100	99,3	2,5	0	2,5	4,9	38,3
21	89,2	110	98,7	99,5	97,9	98,2	97,1	4,8	2,5	0	2,6	35,9
22	91,6	113	101	98	96,4	96,8	95,6	7,2	4,9	2,6	0	35,7
23	130	109	130	92,1	99,6	98,6	98,5	40,6	38,3	35,9	35,7	0
24	131	110	132	93,3	101	99,8	99,7	40,3	38	35,5	35,4	1,2
25	122	113	134	96	104	103	102	40,7	38,4	36	35,8	4
26	155	176	165	192	187	189	186	118	115	116	117	152
27	137	158	147	174	169	171	168	99,4	97,1	98,1	99,1	133
28	137	158	146	173	169	171	168	99	96,5	97,5	98,7	133
29	182	208	192	191	199	198	197	93,5	95,9	93,1	94,1	99

D	12	13	14	15	16	17	18	19	20	21	22	23
30	211	231	220	229	236	235	235	131	134	131	151	137
31	205	226	215	228	236	234	234	130	133	130	131	136
32	279	300	289	317	324	323	323	209	222	219	220	225
33	276	297	285	313	321	320	320	216	218	215	216	221
34	273	294	282	310	318	317	317	213	215	212	213	218
35	279	300	289	316	324	323	323	219	221	218	219	224

D	24	25	26	27	28	29	30	31	32	33	34	35
0	99,3	99,7	55,8	35,2	34,8	66,9	109	110	178	174	171	177
1	70,9	71,4	81,1	66,5	66,1	98,2	140	141	209	205	202	208
2	69,3	69,8	84	65,9	65,4	97,5	139	140	208	204	202	208
3	70,2	70,7	86,6	68,5	68	100	142	143	211	207	204	210
4	67,3	67,7	84,4	66,2	65,8	97,9	140	141	209	205	202	208
5	67,2	67,7	82,4	67,8	67,4	99,5	141	142	210	206	203	210
6	71,1	71,5	82,6	68,4	67,6	100	142	143	211	207	204	210
7	71,7	72,1	78,3	63,8	63,4	95,5	137	138	206	202	199	206
8	70,1	70,6	83,9	65,7	65,2	97,4	139	140	208	204	201	208
9	79,7	80,1	71,1	52,9	52,3	84,6	126	127	195	191	189	195
10	68,9	69,3	83,6	65,4	64,9	97,1	139	140	208	204	201	207
11	132	120	154	126	135	181	209	210	278	274	271	277
12	131	122	155	137	137	182	211	205	279	276	273	279
13	110	113	176	158	158	208	231	226	300	297	294	300
14	132	134	165	147	146	192	220	215	289	285	282	289

D	24	25	26	27	28	29	30	31	32	33	34	35
15	93,3	96	192	174	173	191	229	228	317	313	310	316
16	101	104	187	169	169	199	236	236	324	321	318	324
17	99,8	103	189	171	171	198	235	234	323	320	317	323
18	99,7	102	186	168	168	197	235	234	323	320	317	323
19	40,3	40,7	118	99,4	99	93,5	131	130	209	216	213	219
20	38	38,4	115	97,1	96,5	95,9	134	133	222	218	215	221
21	35,5	36	116	98,1	97,5	93,1	131	130	219	215	212	218
22	35,4	35,8	117	99,1	98,7	94,1	151	131	220	216	213	219
23	1,2	4	152	133	133	99	137	136	225	221	218	224
24	0	3,8	152	134	133	98,9	137	136	225	221	218	224
25	3,8	0	147	134	128	95,9	134	133	222	218	215	221
26	152	147	0	20,4	19,7	66,4	108	103	102	173	170	102
27	134	134	20,4	0	2	47,9	89,5	84,3	158	155	152	158
28	133	128	19,7	2	0	46,7	88,4	83,2	157	154	151	157
29	98,9	95,9	66,4	47,9	46,7	0	39,2	38,3	120	117	114	120
30	137	134	108	89,5	88,4	39,2	0	0,9	80,6	76,9	74	80,1
31	136	133	103	84,3	83,2	38,3	0,9	0	80,5	76,8	73,9	80
32	225	222	102	158	157	120	80,6	80,5	0	5,2	7,3	1
33	221	218	173	155	154	117	76,9	76,8	5,2	0	3,3	6,4
34	218	215	170	152	151	114	74	73,9	7,3	3,3	0	7,8
35	224	221	102	158	157	120	80,1	80	1	6,4	7,8	0

C. Data waktu perjalanan (jam)

D	0	1	2	3	4	5	6	7	8	9	10	11
0	0	0,9	0,89	0,9	0,85	0,9	0,9	0,78	0,86	0,45	0,82	2,62
1	0,9	0	0,12	0,06	0,16	0,17	0,06	0,11	0,03	0,42	0,12	1,72
2	0,89	0,12	0	0,08	0,09	0,15	0,12	0,16	0,09	0,4	0,03	1,79
3	0,9	0,06	0,08	0	0,15	0,12	0,03	0,17	0,09	0,48	0,2	1,7
4	0,85	0,16	0,09	0,15	0	0,06	0,12	0,16	0,11	0,42	0,11	1,81
5	0,9	0,17	0,15	0,12	0,06	0	0,15	0,23	0,14	0,52	0,2	1,81
6	0,9	0,06	0,12	0,03	0,12	0,15	0	0,16	0,09	0,49	0,21	1,71
7	0,78	0,11	0,16	0,17	0,16	0,23	0,16	0	0,07	0,33	0,12	1,82
8	0,86	0,03	0,09	0,09	0,11	0,14	0,09	0,07	0	0,42	0,09	1,79
9	0,45	0,42	0,4	0,48	0,42	0,52	0,49	0,33	0,42	0	0,35	2,18
10	0,82	0,12	0,03	0,2	0,11	0,2	0,21	0,12	0,09	0,35	0	1,81
11	2,62	1,72	1,79	1,7	1,81	1,81	1,71	1,82	1,79	2,18	1,81	0
12	2,68	1,77	1,86	11,55	1,86	1,88	1,77	1,86	1,85	2,24	1,85	0,05
13	3,14	2,25	2,33	2,22	2,33	2,37	2,26	2,37	2,34	2,73	2,35	0,55
14	2,9	2	2,07	1,96	2,09	2,1	2	2,1	2,09	2,47	2,1	0,39
15	3,6	2,71	2,8	2,67	2,8	2,84	2,73	2,84	2,82	3,22	2,84	0,97
16	3,47	2,57	2,64	2,53	2,64	2,68	2,57	2,69	2,67	3,06	2,7	0,83
17	3,46	2,55	2,62	2,52	2,62	2,68	2,57	2,68	2,66	3,04	2,68	0,83
18	3,35	2,5	2,56	2,2	2,3	2,26	2,18	2,28	2,55	2,88	2,55	0,76
19	1,63	0,9	0,85	0,73	0,62	0,53	0,72	0,72	0,85	1,11	0,82	2,13
20	1,57	0,83	0,76	0,67	0,57	0,57	0,65	0,65	0,8	1,06	0,78	2,06

D	0	1	2	3	4	5	6	7	8	9	10	11
21	1,6	0,87	0,79	0,69	0,58	0,58	0,69	0,67	0,83	1,09	0,8	2,1
22	1,68	0,94	0,9	0,74	0,65	0,64	0,75	0,75	0,92	1,16	0,87	2,17
23	2,48	1,75	1,7	1,4	1,31	1,31	1,46	1,48	1,68	1,96	1,65	2,9
24	2,48	1,76	1,7	1,38	1,31	1,29	1,45	1,46	1,68	1,95	1,65	2,94
25	2,5	1,76	1,7	1,4	1,31	1,32	1,48	1,48	1,68	1,96	1,67	2,93
26	1,37	2,18	2,17	1,72	1,67	1,69	1,73	1,6	2,03	1,61	2,03	3,79
27	0,79	1,65	1,65	1,34	1,31	1,31	1,32	1,22	1,52	1,1	1,5	3,27
28	0,83	1,7	1,68	1,38	1,32	1,36	1,38	1,26	1,54	1,12	1,54	3,32
29	1,99	2,82	2,8	2,29	2,21	2,23	2,25	2,14	2,57	2,14	2,61	4,35
30	2,42	3,27	3,24	2,8	2,71	2,75	2,77	2,66	2,98	2,53	3,01	4,81
31	2,45	3,3	3,28	2,82	2,74	2,79	2,8	2,7	3,02	2,58	3,07	4,83
32	3,38	4,22	4,18	3,72	3,63	3,7	3,72	3,46	4,02	3,55	4,04	5,81
33	3,24	4,09	4,08	3,63	3,54	3,6	3,62	3,52	3,8	3,34	3,86	5,64
34	3,13	3,98	3,96	3,54	3,46	3,51	3,54	3,44	3,69	3,22	3,75	5,51
35	3,33	4,18	4,17	3,68	3,62	3,68	3,71	3,58	3,97	3,52	4,01	5,76

D	12	13	14	15	16	17	18	19	20	21	22	23
0	2,68	3,14	2,9	3,6	3,47	3,46	3,35	1,63	1,57	1,6	1,68	2,48
1	1,77	2,25	2	2,71	2,57	2,55	2,5	0,9	0,83	0,87	0,94	1,75
2	1,86	2,33	2,07	2,8	2,64	2,62	2,56	0,85	0,76	0,79	0,9	1,7
3	11,55	2,22	1,96	2,67	2,53	2,52	2,2	0,73	0,67	0,69	0,74	1,4
4	1,86	2,33	2,09	2,8	2,64	2,62	2,3	0,62	0,57	0,58	0,65	1,31
5	1,88	2,37	2,1	2,84	2,68	2,68	2,26	0,53	0,57	0,58	0,64	1,31

D	12	13	14	15	16	17	18	19	20	21	22	23
6	1,77	2,26	2	2,73	2,57	2,57	2,25	0,74	0,67	0,71	0,77	1,5
7	1,86	2,37	2,1	2,84	2,69	2,68	2,28	0,72	0,65	0,67	0,75	1,48
8	1,85	2,34	2,09	2,82	2,67	2,66	2,55	0,85	0,8	0,83	0,92	1,68
9	2,24	2,73	2,47	3,22	3,06	3,04	2,88	1,11	1,06	1,09	1,16	1,96
10	1,85	2,35	2,1	2,84	2,7	2,68	2,55	0,82	0,78	0,8	0,87	1,65
11	0,05	0,55	0,39	0,97	0,83	0,83	0,76	2,13	2,06	2,1	2,17	2,9
12	0	0,52	0,34	0,96	0,82	0,82	0,74	2,17	2,11	2,15	2,37	3,1
13	0,52	0	0,56	0,44	0,3	0,3	0,24	2,83	2,76	2,8	2,87	2,53
14	0,34	0,56	0	1,03	0,88	0,88	0,8	2,56	2,5	2,52	2,61	3,19
15	0,96	0,44	1,03	0	0,17	0,16	0,18	2,82	2,74	2,68	2,61	2,07
16	0,82	0,3	0,88	0,17	0	0,11	0,11	2,83	2,76	2,7	2,64	2,35
17	0,82	0,3	0,88	0,16	0,11	0	0,07	2,77	2,7	2,65	2,58	2,3
18	0,74	0,24	0,8	0,18	0,11	0,07	0	2,93	2,85	2,8	2,71	2,48
19	2,17	2,83	2,56	2,82	2,83	2,77	2,93	0	0,08	0,15	0,22	1,06
20	2,11	2,76	2,5	2,74	2,76	2,7	2,85	0,08	0	0,08	0,17	1,01
21	2,15	2,8	2,52	2,68	2,7	2,65	2,8	0,15	0,08	0	0,1	0,88
22	2,37	2,87	2,61	2,61	2,64	2,58	2,71	0,22	0,17	0,1	0	0,88
23	3,1	2,53	3,19	2,07	2,35	2,3	2,48	1,06	1,01	0,88	0,88	0
24	3,15	2,58	3,24	2,12	2,41	2,35	2,52	1,06	0,96	0,88	0,88	0,05
25	3,18	2,67	3,35	2,21	2,5	2,44	2,64	1,07	0,98	0,9	0,9	0,13
26	4,16	4,63	4,39	5	4,83	4,91	4,93	3,11	2,91	2,95	2,98	3,88
27	3,59	4,12	3,86	4,52	4,35	4,42	4,47	2,54	2,35	2,42	2,45	3,33
28	3,65	4,16	3,9	4,54	4,39	4,47	4,52	2,59	2,4	2,45	2,49	3,37
29	4,68	5,12	4,9	4,67	4,94	4,89	5,1	2,42	3,97	2,32	2,37	2,4

D	12	13	14	15	16	17	18	19	20	21	22	23
30	5,25	5,69	5,45	5,53	5,79	5,73	5,99	3,35	3,26	3,18	3,17	3,26
31	5,21	5,64	5,4	5,48	5,76	5,7	5,94	3,32	3,21	3,13	3,2	3,21
32	6,19	6,63	6,39	6,46	6,74	6,67	6,87	3,93	4,25	4,16	4,2	4,23
33	6,08	6,51	6,27	6,34	5,83	6,56	6,77	4,11	4,08	4	4,05	4,06
34	5,97	6,42	6,19	6,23	6,51	6,46	6,68	4	3,97	3,87	3,92	3,94
35	6,17	6,6	6,38	6,42	6,7	6,64	6,85	4,23	4,23	4,13	4,18	4,2

D	24	25	26	27	28	29	30	31	32	33	34	35
0	2,48	2,5	1,37	0,79	0,83	1,99	2,42	2,45	3,38	3,24	3,13	3,33
1	1,76	1,76	2,18	1,65	1,7	2,82	3,27	3,3	4,22	4,09	3,98	4,18
2	1,7	1,7	2,17	1,65	1,68	2,8	3,24	3,28	4,18	4,08	3,96	4,17
3	1,38	1,4	1,72	1,34	1,38	2,29	2,8	2,82	3,72	3,63	3,54	3,68
4	1,31	1,31	1,67	1,31	1,32	2,21	2,71	2,74	3,63	3,54	3,46	3,62
5	1,29	1,32	1,69	1,31	1,36	2,23	2,75	2,79	3,7	3,6	3,51	3,68
6	1,49	1,52	1,79	1,36	1,42	2,32	2,85	2,88	3,81	3,71	3,62	3,81
7	1,46	1,48	1,6	1,22	1,26	2,14	2,66	2,7	3,46	3,52	3,44	3,58
8	1,68	1,68	2,03	1,52	1,54	2,57	2,98	3,02	4,02	3,8	3,69	3,97
9	1,95	1,96	1,61	1,1	1,12	2,14	2,53	2,58	3,55	3,34	3,22	3,52
10	1,65	1,67	2,03	1,5	1,54	2,61	3,01	3,07	4,04	3,86	3,75	4,01
11	2,94	2,93	3,79	3,27	3,32	4,35	4,81	4,83	5,81	5,64	5,51	5,76
12	3,15	3,18	4,16	3,59	3,65	4,68	5,25	5,21	6,19	6,08	5,97	6,17
13	2,58	2,67	4,63	4,12	4,16	5,12	5,69	5,64	6,63	6,51	6,42	6,6
14	3,24	3,35	4,39	3,86	3,9	4,9	5,45	5,4	6,39	6,27	6,19	6,38

D	24	25	26	27	28	29	30	31	32	33	34	35
15	2,12	2,21	5	4,52	4,54	4,67	5,53	5,48	6,46	6,34	6,23	6,42
16	2,41	2,5	4,83	4,35	4,39	4,94	5,79	5,76	6,74	5,83	6,51	6,7
17	2,35	2,44	4,91	4,42	4,47	4,89	5,73	5,7	6,67	6,56	6,46	6,64
18	2,52	2,64	4,93	4,47	4,52	5,1	5,99	5,94	6,87	6,77	6,68	6,85
19	1,06	1,07	3,11	2,54	2,59	2,42	3,35	3,32	3,93	4,11	4	4,23
20	0,96	0,98	2,91	2,35	2,4	3,97	3,26	3,21	4,25	4,08	3,97	4,23
21	0,88	0,9	2,95	2,42	2,45	2,32	3,18	3,13	4,16	4	3,87	4,13
22	0,88	0,9	2,98	2,45	2,49	2,37	3,17	3,2	4,2	4,05	3,92	4,18
23	0,05	0,13	3,88	3,33	3,37	2,4	3,26	3,21	4,23	4,06	3,94	4,2
24	0	0,13	3,91	3,35	3,39	2,38	3,25	3,2	4,21	4,05	3,92	4,18
25	0,13	0	3,93	2,97	3,37	2,28	3,16	3,12	4,11	3,97	3,84	4,08
26	3,91	3,93	0	0,65	0,63	2,16	2,55	2,52	3,56	3,41	3,29	3,49
27	3,35	2,97	0,65	0	0,1	1,51	1,91	1,89	2,95	2,76	2,64	3,56
28	3,39	3,37	0,63	0,1	0	1,51	1,91	1,89	2,93	2,76	2,63	2,9
29	2,38	2,28	2,16	1,51	1,51	0	0,88	0,86	1,94	1,73	1,6	1,9
30	3,25	3,16	2,55	1,91	1,91	0,88	0	0,15	1,41	1,21	1,11	1,4
31	3,2	3,12	2,52	1,89	1,89	0,86	0,15	0	1,43	1,21	1,11	1,41
32	4,21	4,11	3,56	2,95	2,93	1,94	1,41	1,43	0	0,08	0,31	0,05
33	4,05	3,97	3,41	2,76	2,76	1,73	1,21	1,21	0,08	0	0,14	0,29
34	3,92	3,84	3,29	2,64	2,63	1,6	1,11	1,11	0,31	0,14	0	0,36
35	4,18	4,08	3,49	3,56	2,9	1,9	1,4	1,41	0,05	0,29	0,36	0

D. Script Program Algoritma Improved Discrete Bat

```

clc;
cla(handles.axes1, 'reset');
set(handles.axes1, 'XLim', [0 1000], 'YLim', [0
500], 'FontSize', 8, 'Fontweight', 'bold');
axes(handles.axes1);
xlabel('Iterasi (t)'); ylabel('Total Biaya');
set(handles.listbox1, 'string', '    ', 'value', 1, 'UserData', []);
set(handles.text10, 'string', '0 s');
pause(0.001);

tic;
% - Data -
Data_Cust = get(handles.uitable1, 'UserData');
if ~isempty(Data_Cust)
    Data_Coor = Data_Cust(:, 1:2);
    Data_Demand = Data_Cust(:, 3);
    Data_OC = jam2waktu(Data_Cust(:, 4:5))';
    BukaDepot = Data_OC(1, 1);
    Data_OC = Data_OC - BukaDepot;
end
Data = get(handles.uitable2, 'UserData');
if ~isempty(Data)
    Data_Dist = Data{1};
    Data_Time = Data{2};
end
Capacity = str2num(get(handles.edit1, 'string'));
TService = str2num(get(handles.edit2, 'string'));
BBM = str2num(get(handles.edit7, 'string'));
BKend = str2num(get(handles.edit8, 'string'));

% - Parameter -
Npop = str2num(get(handles.edit3, 'string'));
Tmax = str2num(get(handles.edit4, 'string'));
alpha = str2num(get(handles.edit5, 'string'));
gamma = str2num(get(handles.edit6, 'string'));

if ~isempty(Data_Cust) && ~isempty(Data) && ~isempty(Npop) &&
~isempty(Tmax) ...
    && ~isempty(alpha) && ~isempty(gamma)

    % - node -
    node = size(Data_Dist, 1) - 1;

    % - Inisialisasi -
    for i = 1 : Npop
        X(i, :) = randperm(node);
        [Rute{i}, TotJar{i}, TotBar{i}, Wber{i}, Wsel{i}] =
TSP2OVRPTW(X(i, :), Data_Dist, Data_Time, Data_OC, Data_Demand, Capacity,
TService);
        Nfx(i) = BBM*sum(TotJar{i}) + BKend*size(Rute{i}, 2);
    end
    ri0 = ones(1, Npop)*(rand*0.4);
    ri = ri0;
    Ai = ones(1, Npop)*(rand*3+0.7);
    vi = ones(1, Npop);

    % - Bat terbaik (X*) -

```

```

best = find(Nfx == min(Nfx));
Xstar = X(best(1),:);
Nfstar = Nfx(best(1));

% - non-improvement iteration & best-so-far -
bsf(1) = Nfstar;
nonim = 0;

% - Iterasi -
t = 0;
while t < Tmax
    for i = 1 : Npop
        % - Bangkitkan solusi baru -
        Hdist = sum(X(i,:) ~= Xstar); % Hamming Distance
        vi(i) = ceil(rand*(Hdist-1)+1);
        if vi(i) == 0
            vi(i) = 1;
        end
        Xnew = []; NfxN = [];
        for j = 1 : vi(i)
            if vi(i) > node/2
                Xnew(j,:) = Opt3(X(i,:));
            else
                Xnew(j,:) = Opt2(X(i,:));
            end
            [RuteN, TotJarN, TotBarN, WberN, WselN] =
TSP2OVRPTW(Xnew(j,:), Data_Dist, Data_Time, Data_OC, Data_Demand, Capacity,
TService);
            NfxN(j) = BBM*sum(TotJarN) + BKend*size(RuteN,2);
        end
        best = find(NfxN == min(NfxN));
        New = Xnew(best(1),:);
        Newfx = NfxN(best(1));

        rand1 = rand;
        % - Local Search
        if rand1 > ri(i)
            [bsfx, idx] = sort(Nfx);
            bs = ceil(rand*Npop/2);
            k = idx(bs); % one solution among the best (half of
population)
            LS = Opt2(X(k,:));
            [RuteLS, TotJarLS, TotBarLS, WberLS, WselLS] =
TSP2OVRPTW(LS, Data_Dist, Data_Time, Data_OC, Data_Demand, Capacity,
TService);
            NfxLS = BBM*sum(TotJarLS) + BKend*size(RuteLS,2);
            if NfxLS < Newfx
                New = LS;
                Newfx = NfxLS;
            end
        end
    end

    % - Acceptance & Update ri, Ai
    if rand1 < Ai(i) && Newfx < Nfx(i)
        X(i,:) = New;
        Nfx(i) = Newfx;
        ri(i) = ri0(i) * (1 - exp(-gamma * t));
        Ai(i) = alpha * Ai(i);
    end
end
end

```

```

% - Bat terbaik (X*) -
best = find(Nfx == min(Nfx));
Xstar = X(best(1),:);
Nfstar = Nfx(best(1));

% - non-improvement iteration & best-so-far -
bsf(t+2) = Nfstar;
if bsf(t+2) ~= bsf(t+1)
    nonim = t+1;
end

% - Plot -
plot(0:t+1,bsf,'r','LineWidth',2);

line(nonim,bsf(nonim+1),'Marker','s','MarkerEdgeColor','k','MarkerFaceColor',
'y','MarkerSize',5);
    set(handles.axes1,'YLim',[min(bsf)-0.1*max(bsf)
1.1*max(bsf)],'FontSize',8,'FontWeight','bold');
xlabel('Iterasi (t)'); ylabel('Total Biaya');
pause(0.0001);

% - Iterasi selanjutnya
t = t + 1;
end

% - Hasil -
Final = Xstar;
[RuteF, TotJarF, TotBarF, WberF, WselF] = TSP2OVRPTW(Final, Data_Dist,
Data_Time, Data_OC, Data_Demand, Capacity, TService);
Hasil{1} = ['Total Biaya = Rp ' num2str(BBM*sum(TotJarF) +
BKend*size(RuteF,2))];
Hasil{2} = ['Banyak Kendaraan = ' num2str(size(RuteF,2))];
Hasil{3} = ['Total Jarak Tempuh = ' num2str(sum(TotJarF)) ' km'];
for k = 1:size(RuteF,2)
    RouteLine = ['Rute ' num2str(k) ' : 0-'];
    for i = 1:length(RuteF{k})-1
        RouteLine = [RouteLine num2str(RuteF{k}(i)) '-'];
    end
    RouteLine = [RouteLine num2str(RuteF{k}(end))];
    TberF = WberF + BukaDepot;
    TselF = WselF + BukaDepot;
    Hasil{(6*k-5)+3} = ' ';
    Hasil{(6*k-4)+3} = RouteLine;
    Hasil{(6*k-3)+3} = ['Jarak tempuh = ' num2str(TotJarF(k)) ' km'];
    Hasil{(6*k-2)+3} = ['Total permintaan = ' num2str(TotBarF(k))];
    Hasil{(6*k-1)+3} = ['Waktu Berangkat = ' waktu2jam(TberF(k))];
    Hasil{(6*k)+3} = ['Waktu Selesai = ' waktu2jam(TselF(k))];
end
Hasil{6*size(RuteF,2)+4} = ' ';
Hasil{6*size(RuteF,2)+5} = ['Iterasi konvergen = ' num2str(nonim)];
set(handles.listbox1,'string',char(Hasil),'UserData',{RuteF,Hasil});
set(handles.text10,'string',[num2str(toc) ' s']);
end

```

E. Script Program Open Vehicle Routing With Time Windows

```

function [Rute, TotJar, TotBar, Wber, Wsel] = TSP2OVRPTW(TSP, Data_Dist,
Data_Time, Data_OC, Data_Demand, Capacity, TService)
if length(TService) == 1
    TService = [TService 0];
end
N = length(TSP); a = 1; uk = 1;
for i = 1 : N
    temp = [0 TSP(a:i)];
    %Perhitungan Waktu
    tw = 1;
    timeA = Data_OC(1,1);
    for j = 1 : length(temp)-1
        timeA = max(timeA + Data_Time(temp(j)+1,temp(j+1)+1),
Data_OC(1,temp(j+1)+1));
        timeA = timeA +
(Data_Demand(temp(j+1)+1)*TService(1)+TService(2))/3600;
    end
    if timeA > Data_OC(2,temp(j+1)+1) %melebihi waktu tutup toko
        tw = 0;
    else
        timeB = timeA;
        for j = length(temp) : -1 : 2
            timeB = timeB -
(Data_Demand(temp(j)+1)*TService(1)+TService(2))/3600;
            timeB = timeB - Data_Time(temp(j-1)+1,temp(j)+1);
            if timeB > Data_OC(2,temp(j-1)+1) %tidak memenuhi time windows
                tw = 0;
                break;
            end
        end
    end
    timeP = timeA;
    if timeP > Data_OC(2,1) %selesai melebihi jam kerja
        tw = 0;
    end
    %Perhitungan Jumlah Barang
    cc = 1;
    Jbar = sum(Data_Demand(temp+1));
    if Jbar > Capacity
        cc = 0;
    end
    %Penyimpanan solusi
    if tw == 0 || cc == 0
        Rute{uk} = temp(2:end-1);
        tjar = Data_Dist(1,Rute{uk}(1)+1);
        for j = 1 : length(Rute{uk})-1
            tjar = tjar + Data_Dist(Rute{uk}(j)+1,Rute{uk}(j+1)+1);
        end
        TotJar(uk) = tjar;
        TotBar(uk) = sum(Data_Demand(Rute{uk}+1));
        temp2 = [0 Rute{uk}];
        %Perhitungan Waktu
        timeA = Data_OC(1,1);
        for j = 1 : length(temp2)-1
            timeA = max(timeA + Data_Time(temp2(j)+1,temp2(j+1)+1),
Data_OC(1,temp2(j+1)+1));
            timeA = timeA +
(Data_Demand(temp2(j+1)+1)*TService(1)+TService(2))/3600;
        end
    end
end
end

```

```

timeB = timeA;
for j = length(temp2) : -1 : 2
    timeB = timeB -
(Data_Demand(temp2(j)+1)*TService(1)+TService(2))/3600;
    timeB = timeB - Data_Time(temp2(j-1)+1,temp2(j)+1);
end
Wber(uk) = timeB; %Waktu Berangkat
Wsel(uk) = timeA; %Waktu Selesai
uk = uk + 1;
a = i;
if i == N
    Rute{uk} = temp(end);
    tjar = Data_Dist(1,Rute{uk}(1)+1);
    TotJar(uk) = tjar;
    TotBar(uk) = sum(Data_Demand(Rute{uk}+1));
    temp2 = [0 Rute{uk}];
    %Perhitungan Waktu
    timeA = Data_OC(1,1);
    for j = 1 : length(temp2)-1
        timeA = max(timeA + Data_Time(temp2(j)+1,temp2(j+1)+1),
Data_OC(1,temp2(j+1)+1));
        timeA = timeA +
(Data_Demand(temp2(j+1)+1)*TService(1)+TService(2))/3600;
    end
    timeB = timeA;
    for j = length(temp2) : -1 : 2
        timeB = timeB -
(Data_Demand(temp2(j)+1)*TService(1)+TService(2))/3600;
        timeB = timeB - Data_Time(temp2(j-1)+1,temp2(j)+1);
    end
    Wber(uk) = timeB; %Waktu Berangkat
    Wsel(uk)=timeA; %Waktu Selesai
end
else
    if i == N
        Rute{uk} = temp(2:end);
        tjar = Data_Dist(1,Rute{uk}(1)+1);
        for j = 1 : length(Rute{uk})-1
            tjar = tjar + Data_Dist(Rute{uk}(j)+1,Rute{uk}(j+1)+1);
        end
        TotJar(uk) = tjar;
        TotBar(uk) = sum(Data_Demand(Rute{uk}+1));
        temp2 = [0 Rute{uk}];
        timeA = Data_OC(1,1);
        for j = 1 : length(temp2)-1
            timeA = max(timeA + Data_Time(temp2(j)+1,temp2(j+1)+1),
Data_OC(1,temp2(j+1)+1));
            timeA = timeA +
(Data_Demand(temp2(j+1)+1)*TService(1)+TService(2))/3600;
        end
        timeB = timeA;
        for j = length(temp2) : -1 : 2
            timeB = timeB -
(Data_Demand(temp2(j)+1)*TService(1)+TService(2))/3600;
            timeB = timeB - Data_Time(temp2(j-1)+1,temp2(j)+1);
        end
        Wber(uk) = timeB; %Waktu Berangkat
        Wsel(uk) = timeA; %Waktu Selesai
    end
end
end
end
end

```


F. Hasil *running* data pelanggan Jember

Npop 25 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	32	38,1074
2	1	43,4	286890	35	24,0013
3	1	42,9	286465	75	20,7939
4	1	42,9	286465	20	23,4688
5	1	42,9	286465	15	20,9171
6	1	42,9	286465	40	63,3578
7	1	42,9	286465	15	20,4732
8	1	42,9	286465	27	19,7536
9	1	42,9	286465	61	19,9248
10	1	42,9	286465	24	19,9746
Rata – Rata		42,95	286507,5	34,4	27,07725

Npop 50 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	16	24,6969
2	1	42,9	286465	26	24,3738
3	1	42,9	286465	12	22,4669
4	1	42,9	286465	14	24,4951
5	1	42,9	286465	16	23,6574
6	1	42,9	286465	11	23,9346
7	1	42,9	286465	22	29,1342
8	1	42,9	286465	12	26,4824
9	1	42,9	286465	9	22,6295
10	1	42,9	286465	12	24,3141
Rata – Rata		42,9	286465	15	24,61849

Npop 100 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	6	32,9653
2	1	42,9	286465	19	32,4991
3	1	42,9	286465	16	32,303
4	1	42,9	286465	8	33,1296
5	1	42,9	286465	10	35,366
6	1	42,9	286465	13	33,0871
7	1	42,9	286465	7	35,8389
8	1	42,9	286465	13	34,7169
9	1	42,9	286465	6	33,2205
10	1	42,9	286465	14	38,4816
Rata – Rata		42,9	286465	11,2	34,1608

Npop 200 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	5	58,9246
2	1	42,9	286465	12	62,7174
3	1	42,9	286465	4	56,4983
4	1	42,9	286465	7	49,941
5	1	42,9	286465	8	62,7061
6	1	42,9	286465	8	61,2464
7	1	42,9	286465	9	61,2287
8	1	42,9	286465	7	53,9906
9	1	42,9	286465	9	64,9451
10	1	42,9	286465	11	53,9083
Rata – Rata		42,9	286465	8	58,61065

Npop 25 Tmax 1000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	18	42,8897
2	1	42,9	286465	17	49,2572
3	1	42,9	286465	30	45,9651
4	1	42,9	286465	9	42,5024
5	1	42,9	286465	6	44,7579
6	1	42,9	286465	13	40,871
7	1	42,9	286465	7	49,7272
8	1	42,9	286465	18	56,8287
9	1	44	287400	11	46,1782
10	1	42,9	286465	18	45,5177
Rata – Rata		43,01	286558,5	14,7	46,44951

Npop 25 Tmax 2000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	1	42,9	286465	50	81,2425
2	1	42,9	286465	13	88,8466
3	1	42,9	286465	47	74,2585
4	1	42,9	286465	24	79,0443
5	1	42,9	286465	11	71,6486
6	1	42,9	286465	9	72,605
7	1	42,9	286465	13	74,0781
8	1	42,9	286465	11	72,4957
9	1	42,9	286465	98	72,578
10	1	42,9	286465	8	73,7373
Rata – Rata		42,9	286465	28,4	76,05346

G. Hasil *running* data pelanggan Jember dan Banyuwangi

Npop 25 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	194,3	665155	137	31,3758
2	2	193,8	664730	68	28,0796
3	2	195,2	665920	55	29,5022
4	2	192,9	663965	128	31,6608
5	2	192,9	663965	207	32,882
6	2	192,9	663965	44	28,4735
7	2	202,1	671785	37	31,8645
8	2	202,3	671955	53	26,3469
9	2	192,9	663965	100	28,3938
10	2	193,8	664730	54	31,3818
Rata – Rata		195,31	666013,5	88,3	29,99609

Npop 50 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	71	51,098
2	2	195,2	665920	83	45,3685
3	2	192,9	663965	80	56,0744
4	2	195,2	665920	220	47,0758
5	2	200,7	670595	78	45,4573
6	2	193,8	664730	80	47,4986
7	2	192,9	663965	62	38,4049
8	2	193,8	664730	53	46,0366
9	2	194,1	664985	195	56,5106
10	2	192,9	663965	66	43,9222
Rata – Rata		194,44	665274	98,8	47,74469

Npop 100 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	73	81,8324
2	2	194	664900	51	87,771
3	2	193,8	664730	51	139,3067
4	2	194	664900	66	99,4025
5	2	193,8	664730	123	151,3718
6	2	194	664900	57	68,2354
7	2	192,9	663965	84	83,724
8	2	192,9	663965	78	73,5414
9	2	194	664900	41	57,3889
10	2	194	664900	73	66,8717
Rata – Rata		193,63	664585,5	69,7	90,94458

Npop 200 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	106	141,2115
2	2	192,9	663965	75	146,7139
3	2	192,9	663965	89	148,7732
4	2	192,9	663965	124	238,2529
5	2	194	664900	54	122,8324
6	2	192,9	663965	72	165,5198
7	2	192,9	663965	149	160,9821
8	2	194	664900	39	128,5917
9	2	192,9	663965	276	221,2017
10	2	192,9	663965	199	189,3042
Rata – Rata		193,12	664152	118,3	166,33834

Npop 500 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	51	640,017
2	2	192,9	663965	37	312,6878
3	2	192,9	663965	35	313,7516
4	2	192,9	663965	60	416,4589
5	2	192,9	663965	62	537,1147
6	2	192,9	663965	106	426,3335
7	2	192,9	663965	58	467,2714
8	2	192,9	663965	41	330,0926
9	2	192,9	663965	37	373,8915
10	2	192,9	663965	51	396,0184
Rata – Rata		192,9	663965	53,8	421,36374

Npop 25 Tmax 1000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	195,2	665920	103	57,2037
2	2	196,9	667365	173	62,8426
3	2	195,2	665920	163	50,7619
4	2	200,7	670595	79	49,0204
5	2	196,6	667110	60	46,9011
6	2	193,8	664730	106	47,2083
7	2	200,7	670595	151	56,7272
8	2	194,3	665155	366	63,8291
9	2	193,8	664730	357	55,4935
10	2	194,1	664985	123	53,2202
Rata – Rata		196,13	666710,5	168,1	54,3208

Npop 500 Tmax 1000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	43	610,8474
2	2	192,9	663965	50	756,7486
3	2	192,9	663965	74	851,3335
4	2	192,9	663965	43	464,9345
5	2	192,9	663965	50	483,7458
6	2	192,9	663965	74	719,9466
7	2	192,9	663965	48	431,3326
8	2	192,9	663965	66	564,1437
9	2	192,9	663965	73	705,195
10	2	192,9	663965	51	648,1048
Rata – Rata		192,9	663965	57,2	623,63325

Npop 25 Tmax 2000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	195,2	665920	64	103,9101
2	2	192,9	663965	91	91,5101
3	2	196,5	667025	187	86,3697
4	2	192,9	663965	90	88,8056
5	2	194	664900	103	94,0053
6	2	194,9	665665	83	100,3478
7	2	194,1	664985	67	101,0023
8	2	194,9	665665	179	102,2855
9	2	200,7	670595	93	116,6215
10	2	195,3	666005	34	94,5322
Rata – Rata		195,14	665869	99,1	97,93901

Npop 200 Tmax 2000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	2	192,9	663965	61	414,7315
2	2	192,9	663965	73	592,7579
3	2	192,9	663965	52	381,0322
4	2	194,1	664985	54	381,144
5	2	192,9	663965	37	290,9262
6	2	194	664900	84	709,7513
7	2	194	664900	49	396,9762
8	2	192,9	663965	110	338,1499
9	2	192,9	663965	47	304,0796
10	2	192,9	663965	82	411,8741
Rata – Rata		193,24	664254	64,9	422,14229

H. Hasil *running* data semua pelanggan

Npop 25 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	753,3	1890305	456	59,0027
2	5	743,7	1882145	316	61,7464
3	5	806,5	1935525	368	61,7168
4	5	800,5	1930425	379	62,8962
5	5	745,8	1883930	464	65,3818
6	5	928,5	2039225	485	86,8294
7	5	734,3	1874155	476	70,4544
8	5	778,3	1911555	285	56,9325
9	5	779,1	1912235	439	58,8994
10	5	743,7	1882145	316	61,7464
Rata – Rata		781,7	1914165	398,4	64,5606

Npop 50 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	717,1	1859535	373	120,8979
2	5	688,5	1835225	217	145,2225
3	5	694	1839900	232	99,4865
4	5	698,8	1843980	338	107,9873
5	5	737,5	1876875	360	117,2051
6	5	831,4	1956690	326	134,1276
7	5	737,9	1877215	314	108,6497
8	5	750	1887500	426	107,9593
9	5	729,8	1870330	497	133,1995
10	5	628,4	1784140	485	147,1117
Rata – Rata		721,34	1863139	356,8	122,18471

Npop 100 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	736,8	1876280	500	302,9299
2	5	621,5	1778275	439	245,9268
3	5	732,3	1872455	252	198,7283
4	5	750,2	1887670	464	251,4861
5	5	688,9	1835565	426	218,3952
6	5	871,7	1990945	276	186,3569
7	5	751	1888350	258	209,4947
8	5	688,7	1835395	447	233,4452
9	5	660,3	1811255	257	222,3245
10	5	690,2	1836670	267	202,8188
Rata – Rata		719,16	1861286	358,6	227,19064

Npop 200 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	702,6	1847210	488	1002,6304
2	5	690,7	1837095	242	412,2495
3	5	621,3	1778105	485	398,0024
4	5	714,9	1857665	467	510,9184
5	5	647,8	1800630	457	848,6563
6	5	676,7	1825195	492	825,7662
7	5	624,7	1780995	398	453,5699
8	5	623,7	1780145	352	473,4895
9	5	682,4	1830040	248	428,4475
10	5	688,8	1835480	380	522,716
Rata – Rata		667,36	1817256	400,9	587,64461

Npop 5000 Tmax 500

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	621,3	1778105	354	16826,3198
2	5	621,3	1778105	155	11810,5569
3	5	621,3	1778105	336	19220,8959
4	5	621,3	1778105	186	15020,7331
5	5	621,3	1778105	262	17700,3286
6	5	621,3	1778105	230	20130,8089
7	5	621,3	1778105	447	25969,5152
8	5	621,3	1778105	336	25408,7376
9	5	621,3	1778105	209	25437,8018
10	5	621,3	1778105	384	29836,0421
Rata – Rata		621,3	1778105	289,9	20736,17399

Npop 25 Tmax 1000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	945,3	2053505	374	93,6983
2	5	694,2	1840070	373	91,5101
3	5	667,4	1817290	311	99,0714
4	5	734	1873900	838	122,1976
5	5	802,9	1932465	721	101,1913
6	5	841,2	1965020	382	90,7242
7	5	737,7	1877045	287	80,2327
8	5	737,4	1876790	741	110,4484
9	5	808,6	1937310	684	122,9474
10	5	963,5	2068975	572	115,3108
Rata – Rata		793,22	1924237	528,3	102,73322

Npop 500 Tmax 1000

No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	624,9	1781165	298	2255,4874
2	5	621,9	1778615	848	2578,1352
3	5	621,3	1778105	250	1497,6267
4	5	625,6	1781760	477	2284,3594
5	5	683,7	1831145	263	1407,9729
6	5	662,8	1813380	379	1740,7343
7	5	622,1	1778785	648	2399,2881
8	5	625,1	1781335	661	3368,8012
9	5	684,2	1831570	276	1387,9312
10	5	691,6	1837860	297	1458,6962
Rata – Rata		646,32	1799372	439,7	2037,90326

Npop 25 Tmax 2000

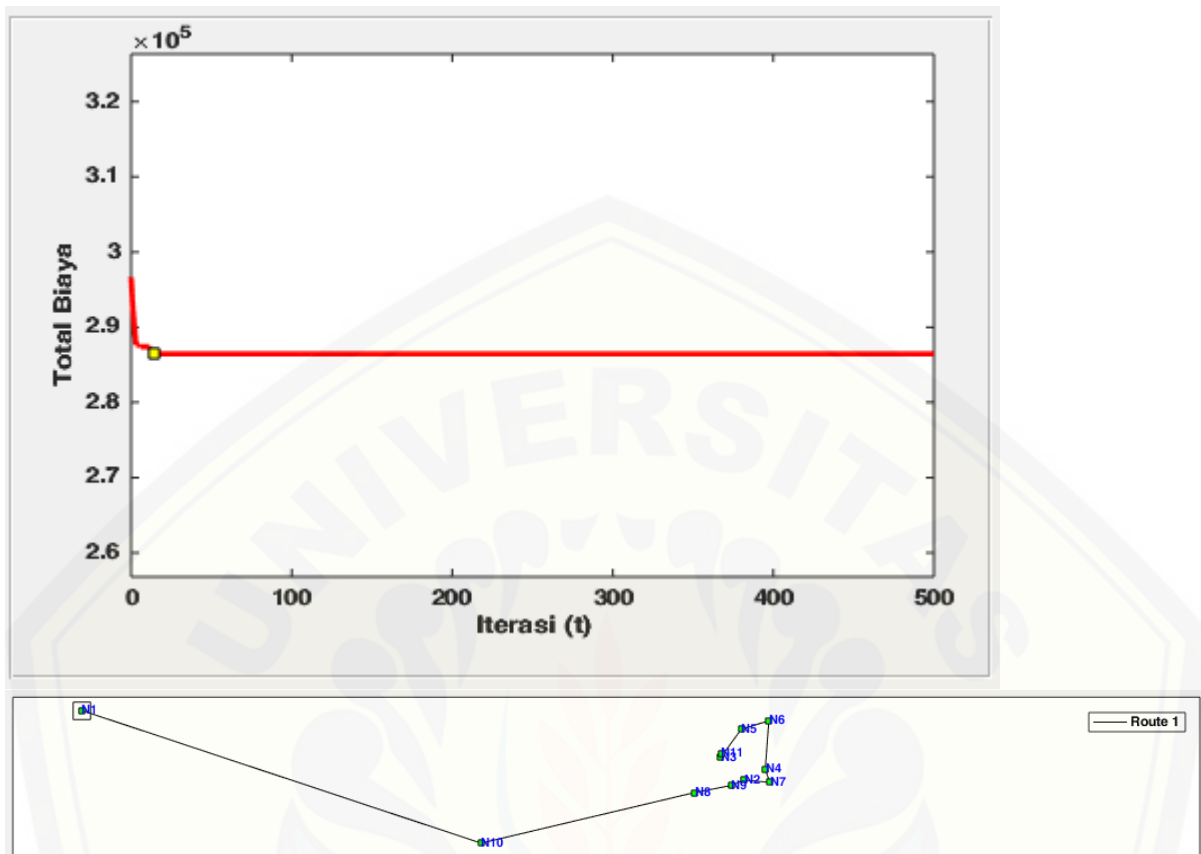
No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	861,4	1982190	923	148,3622
2	5	659,2	1810320	255	131,3324
3	5	743,2	1881720	1160	202,9985
4	5	691,5	1837775	1038	161,0908
5	5	824,9	1951165	291	140,2226
6	5	934,6	2044410	563	192,6708
7	5	695,4	1841090	806	164,4395
8	5	977,4	2080790	242	134,4444
9	5	768,1	1902885	515	159,9607
10	5	697,9	1843215	458	160,1347
Rata – Rata		785,36	1917556	625,1	159,56566

Npop 200 Tmax 2000

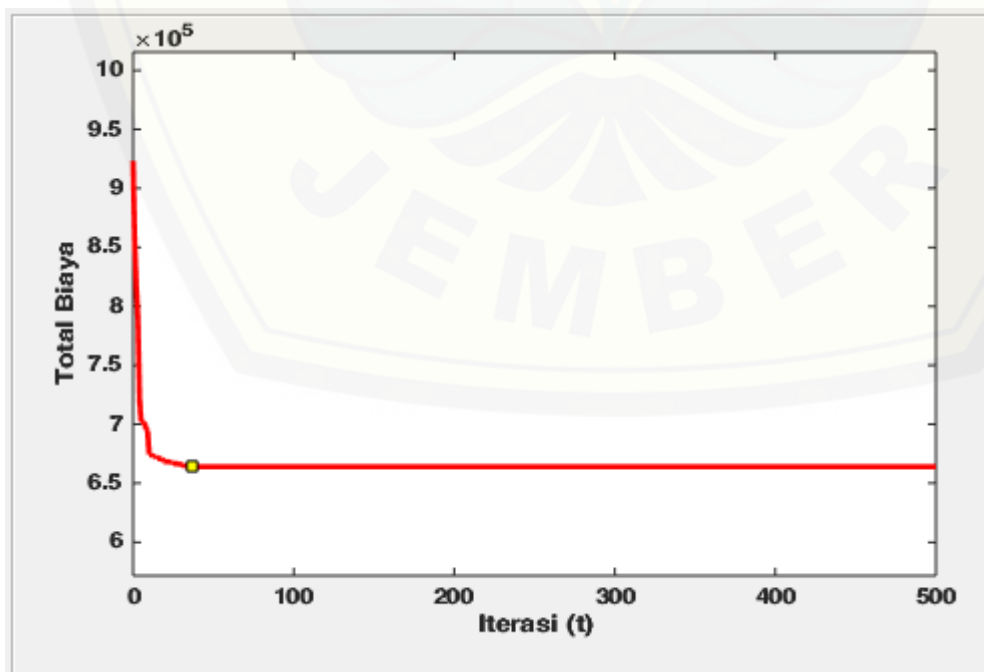
No	Banyak Kendaraan	Jarak Tempuh	Total Biaya	Iterasi Konvergen	Waktu Komputasi
1	5	622,6	1779210	413	981,8878
2	5	737,6	1876960	367	846,7022
3	5	725	1866250	255	717,124
4	5	727,4	1868290	268	784,8694
5	5	626,1	1782185	780	1487,3856
6	5	717,4	1859790	197	825,2095
7	5	768,3	1903055	150	1006,9263
8	5	675,2	1823920	196	747,456
9	5	630,7	1786095	1717	3540,9117
10	5	702,8	1847380	191	857,4062
Rata – Rata		693,31	1839313,5	453,4	1179,58787

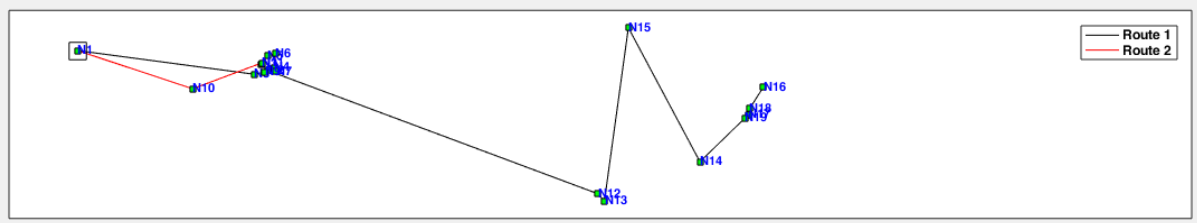
I. Gambar grafik kekonvergenan dan plot rute pada program

1. Pelanggan Jember

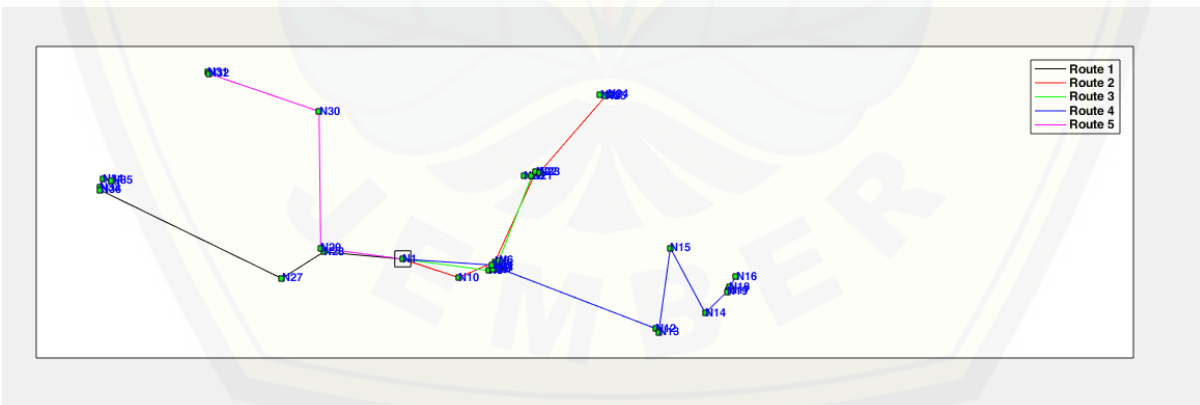
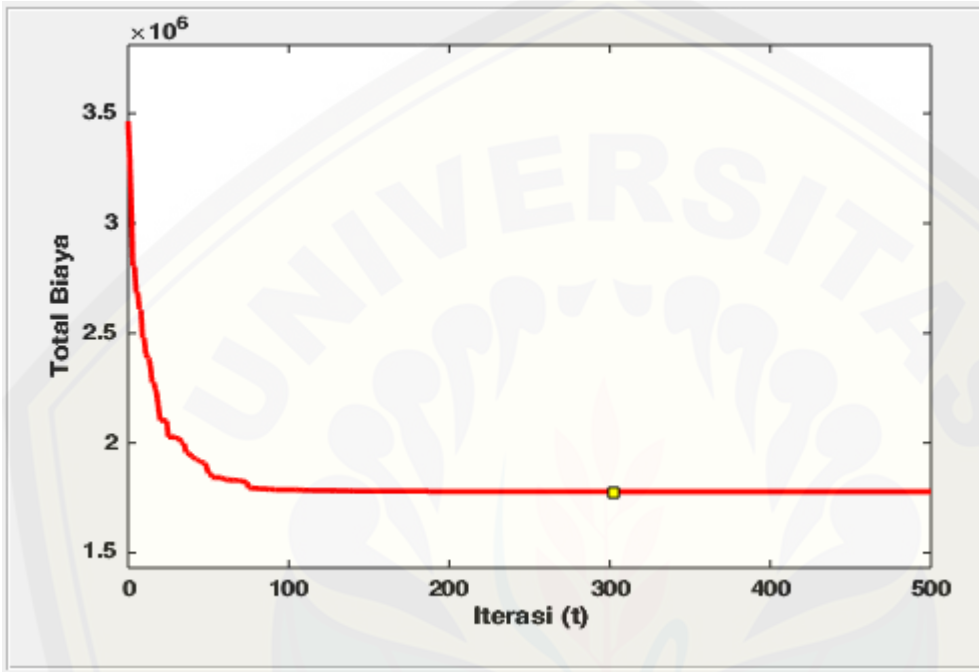


2. Data Pelanggan Jember dan Banyuwangi





3.Data semua pelanggan



J. Data rute awal perusahaan

X	Rute	Jarak Tempuh (km)	Total Permintaan (pack)	Fungsi Tujuan X_i
X_1	0-5-4-6-11-12-16-17-15	146,7	226	
X_2	0-9-7-8-24-23-25	107,9	218	
X_3	0-20-19-29-30-31-28	282,1	233	
X_4	0-27-26-35-32-33-34	167,1	233	
X_5	0-10-2-1-3-14-13-18-22-21	243,9	164	

