



**PENERAPAN *FIREFLY ALGORITHM*
DAN *ANT COLONY OPTIMIZATION*
PADA *TRAVELLING SALESMAN PROBLEM***

SKRIPSI

Oleh
Rafika Ratnasari
NIM 121810101070

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016**



**PENERAPAN *FIREFLY ALGORITHM*
DAN *ANT COLONY OPTIMIZATION*
PADA *TRAVELLING SALESMAN PROBLEM***

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
Untuk menyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh
Rafika Ratnasari
NIM 121810101070

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016**

PERSEMBAHAN

Puji syukur kehadirat Allah SWT, skripsi ini saya persembahkan untuk:

1. kedua orang tua saya tercinta Ayahanda Suparji dan Ibunda Dwi Endah Wahyuningtyas serta saudara saya tercinta Rosyida Amini dan Indra Teguh Santosa , yang telah mendoakan dan memberi kasih sayang serta semangat;
2. seluruh guru dan dosen sejak taman kanak-kanak sampai perguruan tinggi, yang telah memberikan ilmu dan membimbing dengan penuh kesabaran;
3. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMA Negeri 2 Jember, SMP Negeri 6 Jember, SD Al-Baitul Amien (Full Day School) Jember, dan TK Sinar Nyata III.
4. teman-teman Bathics'12 yang selalu membantu dan memberi dukungan dan semangat.

MOTO

Gagal itu urusan nanti, yang penting kita berani mencoba dan terus mencoba¹

Hidup itu sederhana, jika kamu senang tersenyumlah dan jika kamu sedih tertawalah¹



¹ Pathdaily

PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Rafika Ratnasari

NIM : 121810101070

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Penerapan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*” 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, 29 Desember 2016

Yang menyatakan,

Rafika Ratnasari

NIM 121810101070

SKRIPSI

**PENERAPAN *FIREFLY ALGORITHM*
DAN *ANT COLONY OPTIMIZATION*
PADA *TRAVELLING SALESMAN PROBLEM***

Oleh

Rafika Ratnasari
NIM. 121810101070

Pembimbing

Dosen Pembimbing Utama : Ahmad Kamsyakawuni, S.Si., M.Kom.

Dosen Pembimbing Anggota : M. Ziaul Arif, S.Si., M.Sc.

PENGESAHAN

Skripsi berjudul "Penerapan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*" telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas MIPA Universitas Jember.

Tim Penguji :

Ketua,

Anggota I,

Ahmad Kamsyakawuni, S.Si., M.Kom.
NIP 197211291998021001

M. Ziaul Arif, S.Si., M.Sc.
NIP 198501112008121002

Anggota II,

Anggota III,

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

Drs. Rusli Hidayat, M.Sc.
NIP 196610121993031001

Mengesahkan

Dekan,

Drs. Sujito, Ph.D.
NIP 196102041987111001

RINGKASAN

Penerapan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*; Rafika Ratnasari, 121810101070; 2016; halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Travelling Salesman Problem merupakan persoalan optimasi untuk mencari nilai optimum dengan meminimumkan jarak total rute yang dikunjungi dengan mengatur urutan tempat dalam melakukan pendistribusian barang. Persoalan seperti ini terkadang muncul pada perusahaan yang mengelola dibidang pendistribusian. Seorang sales harus menentukan rute terpendek agar pendistribusian barang dapat menjangkau semua daerah yang akan dituju dan dapat mengoptimalkan jarak, waktu dan biaya selama melakukan perjalanan.

Data yang digunakan dalam penelitian ini adalah data sekunder yang diambil dari Christofides dan Eilon (1969). Dalam data ini diasumsikan bahwa kota sebagai titik (*vertex*) pada graf dan jarak antar kota diasumsikan sebagai sisi (*edge*) pada graf. Jumlah kota yang akan diteliti sejumlah 101 titik. Data akan diselesaikan menggunakan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*. Penelitian ini bertujuan untuk mengetahui konsep penerapan dua algoritma kedalam *Travelling Salesman Problem* tersebut dan membandingkan hasil yang sudah didapat berupa total jarak tempuh terpendek pada setiap perjalanan *salesman*.

Berdasarkan hasil dan pembahasan yang telah dilakukan dapat disimpulkan bahwa *Firefly Algorithm* dan *Ant Colony Optimization* dapat menemukan rute terpendek dengan jarak 477,1729 km pada data yang digunakan.

PRAKATA

Puji syukur kehadiran Allah SWT, yang telah melimpahkan rahmat, dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Penerapan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*”. Skripsi ini disusun untuk 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 berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Ahmad Kamsyakawuni, S.Si., M.Kom selaku Dosen Pembimbing Utama dan M. Ziaul Arif, S.Si., M.Sc. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Kosala Dwidja Purnomo, S.Si., M.Si. dan Drs. Rusli Hidayat, M.Sc selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun demi kesempurnaan skripsi ini;
3. Ika Hesti Agustin, S.Si., M.Si. selaku Dosen Pembimbing Akademik yang selalu memberikan motivasi dan pengarahan selama penulis menjadi mahasiswa;
4. seluruh Dosen dan Karyawan Jurusan Matematika Fakultas MIPA Universitas Jember;
5. semua pihak yang tidak dapat disebutkan satu per satu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Desember 2016

Penulis

DAFTAR ISI

	Halaman
HALAMAN SAMPUL	i
PERSEMBAHAN	ii
MOTO	iii
PERNYATAAN	iv
HALAMAN PEMBIMBING	v
PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xvi
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan	4
1.5 Manfaat	4
BAB 2. TINJAUAN PUSTAKA	5
2.1 Teori Dasar Graf	5
2.1.1 Graf Berarah (<i>Directed Graph</i>).....	5
2.1.2 Graf Tak Berarah (<i>Undirected Graph</i>).....	5
2.2 Lintasan Hamilton	6
2.3 Travelling Salesman Problem (TSP)	6
2.4 Metaheuristik	8
2.5 Firefly Algorithm	8
2.6 Ant Colony Optimization	13

BAB 3. METODE PENELITIAN	17
3.1 Data Penelitian	17
3.2 Langkah-langkah Penelitian	18
BAB 4. HASIL DAN PEMBAHASAN	21
4.1 Hasil	21
4.1.1 Penyelesaian <i>Travelling Salesman Problem</i> dengan <i>Firefly</i> <i>Algorithm</i>	21
4.1.2 Penyelesaian <i>Travelling Salesman Problem</i> dengan <i>Ant Colony</i> <i>Optimization</i>	29
4.1.3 Penyelesaian <i>Travelling Salesman Problem</i> dengan Program 37	
4.2 Pembahasan	41
BAB 5. KESIMPULAN	77
5.1 Kesimpulan	77
5.2 Saran	77
DAFTAR PUSTAKA	78
LAMPIRAN	80

DAFTAR TABEL

	Halaman
3.1 Data Penelitian	17
4.1 Jarak antar kota	21
4.2 Populasi Awal	22
4.3 Pengkodean Populasi Awal.....	22
4.4 Hasil Perhitungan Fungsi Tujuan.....	23
4.5 Hasil Perhitungan Intensitas Cahaya.....	23
4.6 Hasil Perhitungan Populasi Baru	28
4.7 Pengkodean Populasi Baru.....	28
4.8 Nilai Fungsi Tujuan yang Baru	29
4.9 Nilai Visibilitas (η_i, j)	30
4.10 Jumlah Feromon pada Semut 1	30
4.11 Penentuan Rute pada Semut 1.....	33
4.12 Penentuan Rute pada Semut 2.....	33
4.13 Penentuan Rute pada Semut 3.....	34
4.14 Penentuan Rute pada Semut 4.....	34
4.15 Penentuan Rute pada Semut 5.....	35
4.16 Total Jarak Tempuh pada Semut.....	35
4.17 Feromon baru	36
4.18 Hasil <i>Running</i> pada Percobaan Pertama	42
4.19 Hasil <i>Running</i> pada Percobaan Kedua	44
4.20 Hasil <i>Running</i> pada Percobaan Ketiga.....	45
4.21 Hasil <i>Running</i> pada Percobaan Keempat	46
4.22 Hasil <i>Running</i> pada Percobaan Kelima.....	48
4.23 Hasil <i>Running</i> pada Percobaan Keenam	48
4.24 Hasil <i>Running</i> pada Percobaan Ketujuh.....	48
4.25 Hasil <i>Running</i> pada Percobaan Kedelapan	48
4.26 Hasil <i>Running</i> pada Percobaan Kesembilan	48

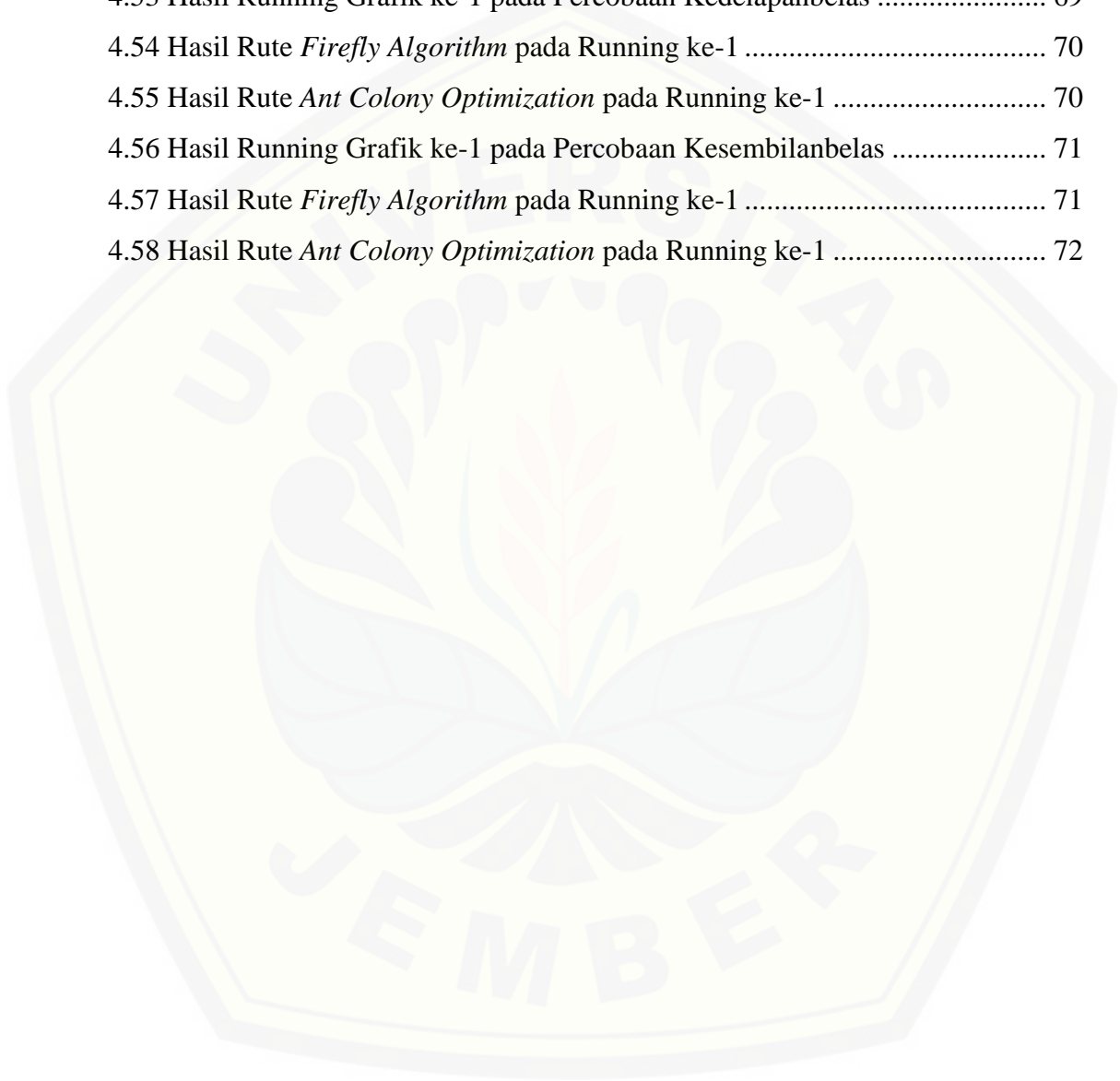
4.27 Hasil <i>Running</i> pada Percobaan Kesepuluh	48
4.28 Hasil <i>Running</i> pada Percobaan Kesebelas	48
4.29 Hasil <i>Running</i> pada Percobaan Keduabelas	48
4.30 Hasil <i>Running</i> pada Percobaan Ketigabelas.....	48
4.31 Hasil <i>Running</i> pada Percobaan Keempatbelas	48
4.32 Hasil <i>Running</i> pada Percobaan Kelimabelas.....	48
4.33 Hasil <i>Running</i> pada Percobaan Keenambelas	48
4.34 Hasil <i>Running</i> pada Percobaan Ketujuhbelas	48
4.35 Hasil <i>Running</i> pada Percobaan Kedelapanbelas	48
4.36 Hasil <i>Running</i> pada Percobaan Kesembilanbelas	48
4.37 Rangkuman Sembilanbelas Kali Percobaan pada <i>Firefly Algorithm</i>	73
4.38 Rangkuman Sembilanbelas Kali Percobaan pada <i>Ant Colony Optimization</i>	74

DAFTAR GAMBAR

	Halaman
2.1 Graf Berarah.....	5
2.2 Graf Tak Berarah.....	6
2.3 (a) Graf yang memiliki lintasan Hamilton: 4, 2, 1, 3 (b) Graf yang memiliki sirkuit Hamilton: 1, 2, 4, 3, 1 (c) Graf yang tidak memiliki lintasan dan sirkuit Hamilton.....	6
2.4 Ilustrasi Perilaku Kunang-kunang.....	9
2.5 <i>Flowchart Firefly Algorithm</i>	12
2.6 Ilustrasi Perilaku Semut	13
2.7 <i>Flowchart Ant Colony Optimazation</i>	16
3.1 <i>Flowchart</i> Langkah-langkah Penelitian	20
4.1 Tampilan Program.....	37
4.2 Hasil Running Grafik ke-1 pada Percobaan Pertama.....	41
4.3 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	41
4.4 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	42
4.5 Hasil Running Grafik ke-1 pada Percobaan Kedua	43
4.6 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	43
4.7 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	43
4.8 Hasil Running Grafik ke-1 pada Percobaan Ketiga	44
4.9 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	45
4.10 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	45
4.11 Hasil Running Grafik ke-1 pada Percobaan Keempat	46
4.12 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	46
4.13 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	47
4.14 Hasil Running Grafik ke-1 pada Percobaan Kelima	48
4.15 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	48
4.16 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	48
4.17 Hasil Running Grafik ke-1 pada Percobaan Keenam	49

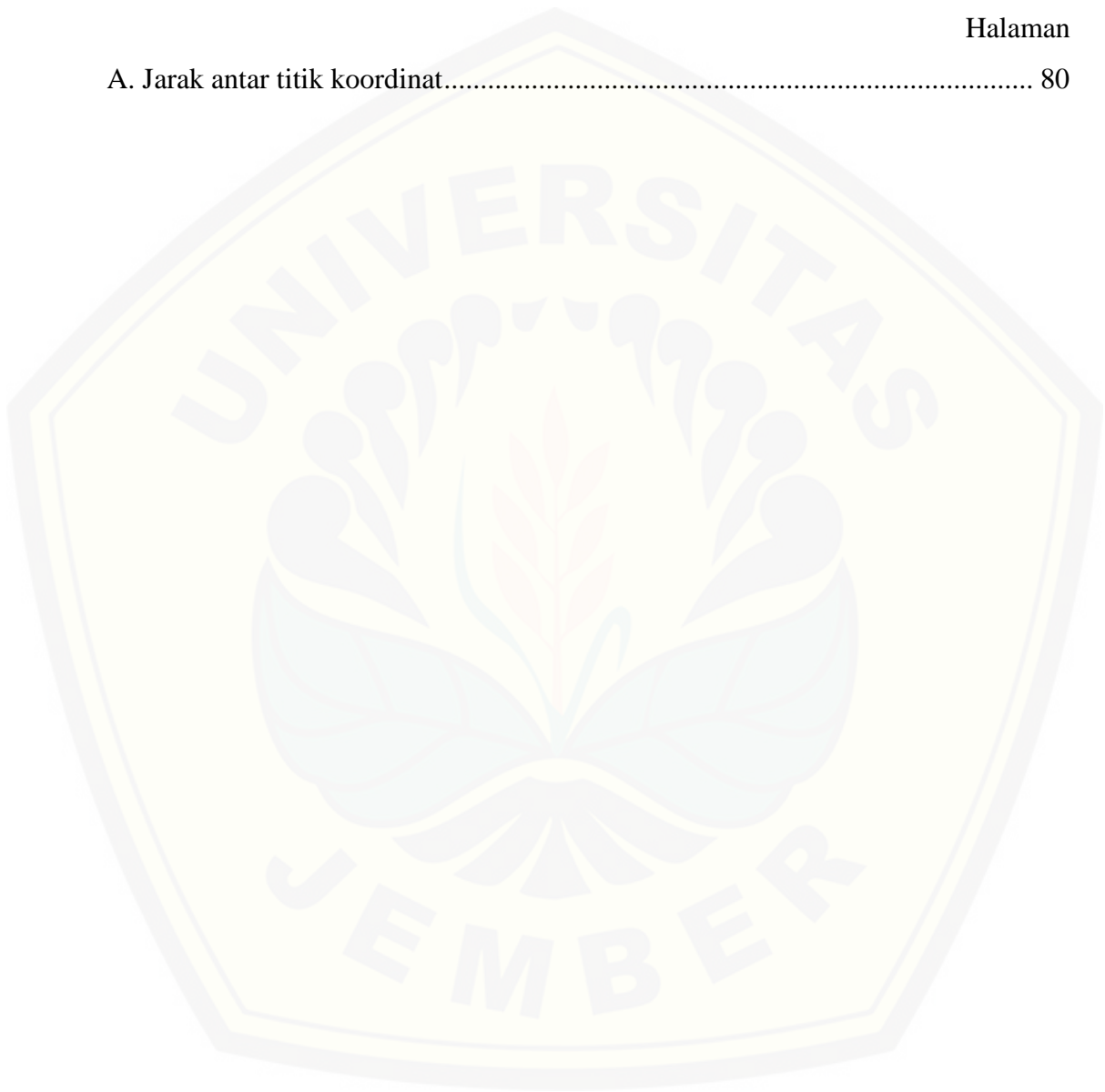
4.18 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	50
4.19 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	50
4.20 Hasil Running Grafik ke-1 pada Percobaan Ketujuh.....	51
4.21 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	51
4.22 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	52
4.23 Hasil Running Grafik ke-1 pada Percobaan Kedelapan.....	53
4.24 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	53
4.25 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	53
4.26 Hasil Running Grafik ke-1 pada Percobaan Kesembilan.....	54
4.27 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	55
4.28 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	55
4.29 Hasil Running Grafik ke-1 pada Percobaan Kesepuluh.....	56
4.30 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	56
4.31 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	57
4.32 Hasil Running Grafik ke-1 pada Percobaan Kesebelas.....	58
4.33 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	58
4.34 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	58
4.35 Hasil Running Grafik ke-1 pada Percobaan Keduabelas	59
4.36 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	60
4.37 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	60
4.38 Hasil Running Grafik ke-1 pada Percobaan Ketigabelas.....	61
4.39 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	61
4.40 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	62
4.41 Hasil Running Grafik ke-1 pada Percobaan pertama.....	63
4.42 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	63
4.43 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	63
4.44 Hasil Running Grafik ke-1 pada Percobaan Kelimabelas.....	64
4.45 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	65
4.46 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	65
4.47 Hasil Running Grafik ke-1 pada Percobaan Keenambelas	66
4.48 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	66

4.49 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	67
4.50 Hasil Running Grafik ke-1 pada Percobaan Ketujuhbelas.....	68
4.51 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	68
4.52 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	68
4.53 Hasil Running Grafik ke-1 pada Percobaan Kedelapanbelas	69
4.54 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	70
4.55 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	70
4.56 Hasil Running Grafik ke-1 pada Percobaan Kesembilanbelas	71
4.57 Hasil Rute <i>Firefly Algorithm</i> pada Running ke-1	71
4.58 Hasil Rute <i>Ant Colony Optimization</i> pada Running ke-1	72



DAFTAR LAMPIRAN

	Halaman
A. Jarak antar titik koordinat.....	80



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Distribusi barang adalah beberapa rangkaian kegiatan pengiriman hasil produksi kepada konsumen untuk dipasarkan dengan tujuan mempermudah pemasaran produk. Ketika sistem distribusi tidak mempunyai kontrol terhadap pendistribusian barang maka akan menyebabkan kerugian bagi perusahaan. Dalam pendistribusian barang, kondisi rute dan pemilihan rute terpendek sangat diperlukan. Jika seorang sales salah memilih rute maka akan menghambat proses pengiriman dan mengakibatkan biaya penggunaan bahan bakar menjadi lebih besar serta waktu yang dibutuhkan untuk pengiriman atau pendistribusian barang menjadi lama.

Travelling Salesman Problem merupakan persoalan optimasi untuk mencari nilai optimum dengan meminimumkan jarak total rute yang dikunjungi dengan mengatur urutan tempat dalam melakukan pendistribusian barang. Persoalan seperti ini terkadang muncul pada perusahaan yang mengelola dibidang pendistribusian. Seorang sales harus menentukan rute terpendek agar pendistribusian barang dapat menjangkau semua daerah yang akan dituju dan dapat mengoptimalkan jarak, waktu dan biaya selama melakukan perjalanan. Dalam pengiriman barang seorang salesman harus mengunjungi beberapa kota tanpa harus mendatangi kota yang sama lebih dari satu kali dan kembali pada kota asal pemberangkatan.

Menurut Dorigo dan Stutzle (2004) untuk menentukan rute terpendek dapat dilakukan dengan dua metode, yaitu metode konvensional dan metode metaheuristik. Metode konvensional adalah metode yang menggunakan perhitungan matematika secara manual dan membutuhkan waktu yang lama dalam melakukan perhitungan. Metode ini hanya bisa melakukan pencarian dengan lima sampai sepuluh rute saja. Sedangkan metode metaheuristik sebuah kerangka algoritma umum yang juga melakukan perubahan dalam pengadaptasian pada

sebuah masalah khusus. Penggunaan metaheuristik meningkatkan kemampuan pencarian solusi dengan kualitas tinggi yang berhubungan dengan masalah optimasi kombinasi.

Banyak peneliti telah menyelesaikan permasalahan optimasi berkaitan dengan *Travelling Salesman Problem* (TSP) yang diterapkan dengan beberapa algoritma misalnya *Firefly Algorithm* (FA), *A*star Algorithm*, *Artificial Bee Colony*, *Ant Colony Optimization*, dan *Bat Algorithm*. *Ant Colony Optimization* merupakan metode metaheuristik yang pertama kali diterapkan oleh Dorigo dan Gambardella (1996) untuk menyelesaikan persoalan *Travelling Salesman Problem*. Dalam penerapan *Ant Colony Optimization*, diketahui bahwa setiap semut yang berada dalam kawanan akan meninggalkan feromon pada setiap rute yang dilaluinya. Feromon ini berfungsi sebagai sinyal bagi sesama semut. Rute yang terpendek akan menyisakan feromon yang lebih kuat, sehingga semut berikutnya dapat memutuskan rute mana yang harus dipilih. Biasanya semut akan memilih rute yang memiliki feromon paling kuat, karena akan ditemui rute yang terpendek karena banyak semut yang melalui rute tersebut. Semakin kuat feromon yang ada maka semakin banyak semut yang melalui jalur tersebut. Banyak peneliti yang telah membahas tentang *Ant Colony Optimization* seperti menurut Apriliani (2011) yang membahas tentang penyelesaian *Travelling Salesman Problem* (TSP) menggunakan *Ant Colony Optimization* dan Algoritma *Cheapest Insertion Heuristic* (CIH), didapatkan bahwa Algoritma *Cheapest Insertion Heuristic* (CIH) menghasilkan jalur yang lebih optimal dibandingkan dengan *Ant Colony Optimization* untuk jumlah titik kurang dari 7. Sedangkan untuk titik lebih dari 7, algoritma semut lebih optimal dibandingkan Algoritma *Cheapest Insertion Heuristic* (CIH).

Firefly Algorithm merupakan salah satu metode metaheuristik yang membahas tentang pemecahan persoalan optimasi. Menurut Yang X.S (2010) algoritma ini terinspirasi oleh pergerakan kunang-kunang terutama pada cahaya yang ditimbulkan oleh kunang-kunang. Formulasi umum dari algoritma ini disajikan bersama-sama dengan pemodelan matematika analisis untuk memecahkan masalah optimasi. Ada beberapa peneliti yang sudah menerapkan

Firefly Algorithm ke dalam beberapa permasalahan seperti menurut Sharad dan Gopal (2013) pada penelitiannya tentang pemecahan *Travelling Salesman Problem* menggunakan *Firefly Algorithm*. Pada penelitian ini didapatkan bahwa *Firefly Algorithm* memberikan hasil jarak yang lebih baik dari pada *Genetic Algorithm* (GA) dan *Simulated Annealing* (SA) dalam proses penentuan rute transportasi. Sedangkan pada penelitian yang dilakukan oleh Annisa (2016) pada penelitiannya tentang penerapan algoritma *Firefly* pada permasalahan *Knapsack 0-1*. Hasil dari penelitian ini menghasilkan dari segi *running time* lebih cepat mencapai optimal meskipun untuk mencapai konvergen lebih lama, hal ini dikarenakan bilangan random sangat berpengaruh dalam mencapai solusi optimal. Berdasarkan dari beberapa permasalahan yang telah diselesaikan diatas, penulis tertarik untuk memecahkan persoalan optimasi dengan judul “Penerapan *Firefly Algorithm* dan *Ant Colony Optimization* pada *Travelling Salesman Problem*”.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas dapat ditarik beberapa rumusan masalah, yaitu:

- a. Bagaimana menyelesaikan masalah *Travelling Salesman Problem* menggunakan *Firefly Algorithm* dan *Ant Colony Optimization*?
- b. Bagaimana hasil perbandingan dari penerapan *Firefly Algorithm* dan *Ant Colony Optimization* berdasarkan total jarak minimum rute?

1.3 Batasan Masalah

Batasan masalah dalam penerapan *Firefly Algorithm* dan *Ant Colony Optimization* adalah:

- a. Titik keberangkatan *sales* berawal dan berakhir di titik yang sama.
- b. Jarak antar kota dianggap simetris, dimana jarak dari kota 1 ke kota 2 sama dengan jarak dari kota 2 ke kota 1.
- c. Setiap titik terhubung ke titik lain.

1.4 Tujuan

Tujuan dari penulisan skripsi ini adalah:

- a. Menyelesaikan masalah *Travelling Salesman Problem* menggunakan *Firefly Algorithm* dan *Ant Colony Optimization*.
- b. Mengetahui hasil perbandingan dari penerapan *Firefly Algorithm* dan *Ant Colony Optimization* berdasarkan total jarak minimum rute.

1.5 Manfaat

Manfaat dari penulisan skripsi ini adalah:

- a. Mendapatkan total jarak minimum rute pada permasalahan *Travelling Salesman Problem* dengan menerapkan *Firefly Algorithm* dan *Ant Colony Optimization*.
- b. Memberikan wawasan lebih dalam bagi pembaca tentang permasalahan *Travelling Salesman Problem* dengan menerapkan *Firefly Algorithm* dan *Ant Colony Optimization*.

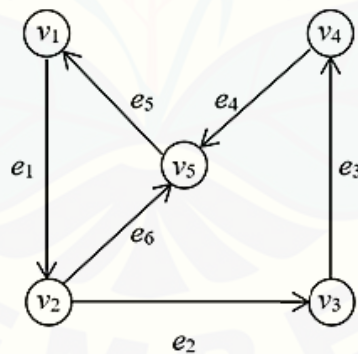
BAB 2. TINJAUAN PUSTAKA

2.1 Teori Dasar Graf

Sebuah graf G didefinisikan sebagai pasangan himpunan $(V(G), E(G))$ dengan $V = \{v_1, v_2, \dots, v_n\}$ adalah himpunan tak kosong dari elemen-elemen yang disebut titik (*vertex*) dan $E = \{e_1, e_2, \dots, e_m\}$ adalah himpunan boleh kosong dari elemen-elemen yang disebut sisi (*edge*) dimana sisi adalah pasangan tak terurut dari dua titik di $G(v_i, v_j)$ atau dapat ditulis $e = v_i, v_j$. Salah satu contoh graf G dapat dilihat pada Gambar 2.1 berikut ini (Chartrand dan Oelermann, 1993).

2.1.1 Graf Berarah (*Directed Graph*)

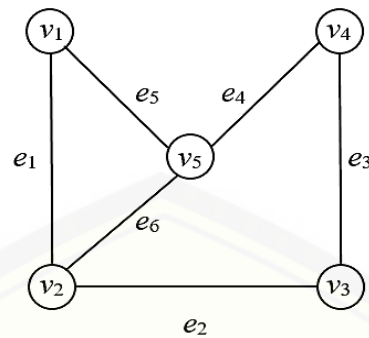
Graf berarah G terdiri dari suatu himpunan V dari *vertex-vertex* dan suatu himpunan $E(A)$ dari *arc* sedemikian rupa sehingga setiap *arc* $a \in A$ menghubungkan pasangan *vertex* terurut. Contoh graf berarah dapat dilihat pada Gambar 2.1 dibawah ini



Gambar 2.1 Graf Berarah

2.1.2 Graf Tak Berarah (*Undirected Graph*)

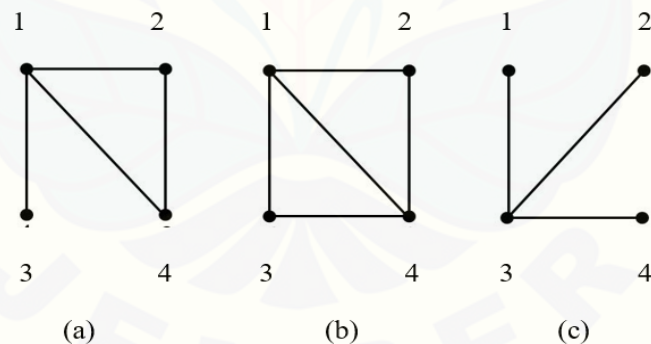
Graf tak berarah G terdiri dari suatu himpunan V dari *vertex-vertex* dan suatu himpunan E dari *edge-edge* sedemikian rupa sehingga setiap sisi $e \in E$ dikaitkan pasangan *vertex* tak terurut. Contoh graf tak berarah dapat dilihat pada Gambar 2.3 dibawah ini,



Gambar 2.2 Graf Tak Berarah

2.2 Lintasan Hamilton

Lintasan Hamilton adalah lintasan yang melewati tiap simpulnya tepat satu kali. Sedangkan sirkuit Hamilton sirkuit (lintasan tertutup) yang melewati tiap simpul didalam graf tepat satu kali, kecuali simpul asal yang sekaligus merupakan simpul akhir yang dilalui dua kali. Graf yang memiliki sirkuit Hamilton dinamakan graf Hamilton, sedangkan graf yang hanya memiliki lintasan Hamilton disebut graf semi-Hamilton (Fonda, 2011).



Gambar 2.3 (a) Graf yang memiliki lintasan Hamilton: 4, 2, 1, 3 (b) Graf yang memiliki sirkuit Hamilton: 1, 2, 4, 3, 1 (c) Graf yang tidak memiliki lintasan dan sirkuit Hamilton

2.3 Travelling Salesman Problem (TSP)

Travelling Salesman Problem (TSP) adalah salah satu permasalahan kombinatorial. Permasalahan *Travelling Salesman Problem* (TSP)-nya adalah menemukan siklus Hamilton dengan bobot minimum, yang secara sederhana disebut siklus Hamilton minimum dari graf. Pada dasarnya persoalan *Travelling*

Salesman Problem (TSP) dapat dinyatakan sebagai perjalanan seorang *sales* yang berangkat dari kota asal mengunjungi tiap-tiap kota tepat satu kali dan kembali lagi ke kota asalnya. Jika diketahui jarak antar kota, maka fungsi objektif *Travelling Salesman Problem* (TSP) adalah untuk menemukan rute perjalanan seorang sales dengan total jarak yang minimum (Chartrand dan Oelermann, 1993). Secara matematis *Travelling Salesman Problem* (TSP) dapat diformulasikan seperti persamaan (2.1),

$$\text{Min } z(x) = \sum_{i=1}^n \sum_{j=1}^n c_{i,j} x_{i,j} \quad (2.1)$$

dengan kendala,

$$\sum_{i=1}^n x_{i,j} = 1, \quad j = 2, \dots, n \quad (2.2)$$

$$\sum_{j=1}^n x_{i,j} = 1, \quad i = 2, \dots, n \quad (2.3)$$

keterangan:

$x_{i,j} = 1$ apabila ada perjalanan *sales* dari simpul i menuju simpul j dan 0 jika tidak ada perjalanan

$c_{i,j}$ = jarak dari simpul i menuju simpul j

Terdapat dua jenis *Travelling Salesman Problem* (TSP), yaitu simetris dan asimetris. Pada *Travelling Salesman Problem* (TSP) simetris, biaya dari kota 1 ke kota 2 sama dengan biaya dari kota 2 ke kota 1 dan membentuk graf tak berarah. Sedangkan *Travelling Salesman Problem* (TSP) asimetris, biaya dari kota 1 ke kota 2 tidak sama dengan biaya dari kota 2 ke kota 1 dan membentuk graf berarah. Untuk *Travelling Salesman Problem* (TSP) simetris jumlah jalur yang mungkin dapat diperoleh dengan menggunakan rumus permutasi seperti pada persamaan (2.4),

$$\frac{n!}{2n} = \frac{(n-1)!}{2} \quad (2.4)$$

Sedangkan untuk *Travelling Salesman Problem* (TSP) asimetris jumlah jalur yang mungkin dapat diperoleh dengan menggunakan rumus permutasi seperti pada persamaan (2.5),

$$\frac{n!}{n} = (n-1)! \quad (2.5)$$

keterangan:

n = jumlah seluruh kota

(Wiyanti, 2013).

2.4 Metaheuristik

Metaheuristik adalah sebuah kerangka algoritma umum yang juga melakukan perubahan dalam pengadaptasian pada sebuah masalah khusus. Sebuah kerangka dapat diartikan sebagai langkah-langkah penyelesaian masalah secara sistematis. Penggunaan metaheuristik meningkatkan kemampuan pencarian solusi dengan kualitas tinggi yang berhubungan dengan masalah optimasi kombinasi (Dorigo dan Stutzle, 2004).

Metaheuristik bersifat *problem independent*, artinya metaheuristik tidak bergantung pada jenis permasalahannya, sehingga dapat diaplikasikan untuk berbagai jenis permasalahan. Contoh dari metode metaheuristik adalah *Genetic Algorithm* (GA), *Particle Swam Optimization* (PSO), *Ant Colony Optimization* (ACO), *Soccer Games Optimization* (SGO) dan *Firefly Algorithm* (FA). Meskipun metaheuristik dapat digunakan untuk berbagai jenis permasalahan, tetapi kemampuan mengadopsi untuk jenis masalah tertentu berpengaruh besar terhadap kualitas penyelesaian yang dihasilkan.

2.5 Firefly Algorithm

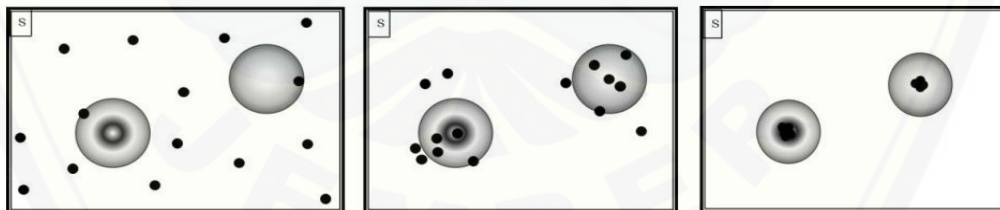
Firefly Algorithm atau Algoritma Kunang-Kunang termasuk salah satu algoritma pada bidang *Artificial Intelligence* atau kecerdasan buatan. Pada bidang *Artificial Intelligence* atau kecerdasan buatan adalah istilah *Swarm Intelligence* yang diartikan sebagai desain algoritma atau alat *problem solving* terdistribusi yang terinspirasi oleh perilaku sosial kolektif koloni serangga dan koloni binatang. *Firefly Algorithm* merupakan salah satu dari *Swarm Intelligence* tersebut. *Firefly Algorithm* adalah sebuah algoritma metaheuristik yang terinspirasi dari perilaku berkedip kunang-kunang.

Formulasi umum dari *Firefly Algorithm* disajikan bersama-sama dengan pemodelan matematika analisis untuk memecahkan masalah optimasi. Secara

khusus, *Firefly Algorithm* memiliki banyak kemiripan dengan algoritma lain yang didasarkan pada kecerdasan kawanan, seperti *Particle Swarm Optimization* (PSO), *Artificial Bee Colony optimization* (ABC), dan *Bacterial Foraging Algorithm* (BFA), memang jauh lebih sederhana baik dalam konsep dan implementasi. Algoritma ini sangat efisien dan dapat mengungguli algoritma konvensional lainnya, seperti algoritma genetika, untuk memecahkan banyak masalah optimasi (Yang, 2010).

Yang (2010) merumuskan *Firefly Algorithm* sebagai berikut:

- Semua kunang-kunang itu unisex, jadi suatu kunang-kunang akan tertarik pada kunang-kunang yang lain.
- Daya tarik sebanding dengan tingkat kecerahan kunang-kunang, kunang-kunang dengan tingkat kecerahan lebih rendah akan tertarik dan bergerak ke kunang-kunang dengan tingkat kecerahan lebih tinggi, kecerahan dapat berkurang seiring dengan bertambahnya jarak dan adanya penyerapan cahaya akibat faktor udara.
- Kecerahan atau intensitas cahaya kunang-kunang ditentukan oleh nilai fungsi tujuan dari masalah yang diberikan. Untuk masalah maksimisasi, intensitas cahaya sebanding dengan nilai fungsi tujuan.



Gambar 2.4 Ilustrasi Perilaku Kunang-kunang

Ada dua hal yang berkaitan dan sangat penting dalam *Firefly Algorithm* yaitu intensitas cahaya dan fungsi keatraktifan. Dalam hal ini banyak dari kita berasumsi bahwa keatraktifan dipengaruhi oleh tingkat intensitas cahaya. Untuk kasus yang paling sederhana contohnya masalah optimasi minimum, tingkat intensitas cahaya pada sebuah kunang-kunang x dapat dilihat seperti pada persamaan (2.6),

$$I(x) = \frac{1}{f(x)} \quad (2.6)$$

keterangan:

$I(x)$ = Intesitas cahaya

$f(x)$ = Fungsi tujuan yang akan dicari

Dengan nilai I merupakan tingkat itensitas cahaya pada x kunang-kunang yang sebanding terhadap solusi fungsi tujuan permasalahan yang akan dicari $f(x)$. Keatraktifan β yang bernilai relatif, karena itensitas cahaya yang harus dilihat dan dinilai oleh kunang-kunang lain. Dengan demikian, hasil penilaian akan berbeda tergantung dari jarak antara kunang-kunang yang satu dengan yang lainnya r_{ij} . Selain itu, itensitas cahaya akan menurun dilihat dari sumbernya dikarenakan terserap oleh media contohnya udara γ . Fungsi keatraktifan dapat menggunakan persamaan (2.8),

$$\beta(r) = \beta_0 e^{(-\gamma r^2)} \quad (2.7)$$

keterangan:

$\beta(r)$ = Fungsi daya tarik

β_0 = Koefisien ketertarikan pada posisi awal

γ = Koefisien penyerapan cahaya

r = Jarak antar kunang-kunang

Jarak antara kunang-kunang i dan j pada lokasi x , x_i dan x_j dapat ditentukan ketika dilakukanya peletakan titik dimana *firefly* tersebut disebar secara acak dalam diagram kartesius yang dapat di rumuskan seperti persamaan (2.9),

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2} \quad (2.8)$$

Dimana $x_{i,k}$ adalah komponen ke- k dari lokasi koordinat x_i terhadap kunang-kunang ke- i . Pada bidang 2D dapat dirumuskan seperti persamaan (2.9),

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2.9)$$

Pergerakan kunang-kunang i yang bergerak menuju tingkat intensitas cahaya yang terbaik dapat dilihat dari persamaan (2.10),

$$x_i' = x_i + \beta_0 e^{(-\gamma r_{ij}^2)} (x_j - x_i) + \alpha \left(rand - \frac{1}{2} \right) \quad (2.10)$$

keterangan:

x_i' = Posisi kunang-kunang i yang baru

x_i = Posisi kunang-kunang i yang sekarang

β_0 = Koefisien ketertarikan pada posisi awal

γ = Koefisien penyerapan cahaya

r_{ij} = Jarak antara kunang-kunang i dan j

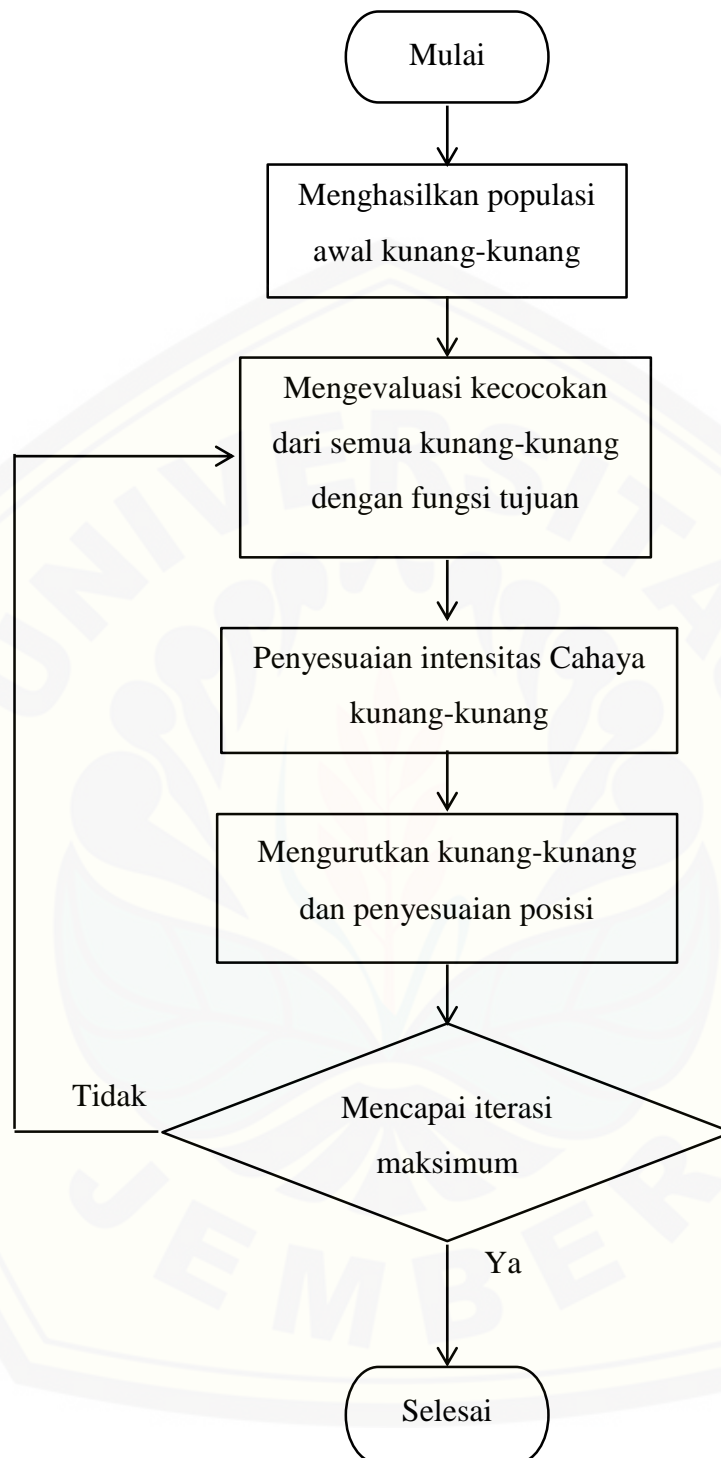
x_j = Kunang-kunang j

x_i = Kunang-kunang i

α = [0,1].

rand = bilangan acak pada selang [0,1]

Dimana variabel awal x_i menunjukkan posisi awal kunang-kunang yang berada pada lokasi x , kemudian persamaan (2.8) yang terdiri dari variabel $\beta_0=1$ variabel ini merupakan nilai keaktratifan awal pada *firefly*, variabel (*exp*) bilangan eksponensial, variabel $\gamma=0,01$ merupakan nilai untuk tingkat penyerapan pada lingkungan sekitar *firefly* yaitu udara dan terakhir r_{ij} merupakan variabel selisih jarak awal antara *firefly* i dan j . Semua variabel pada persamaan (2.8) tersebut diberikan dari fungsi keatraktifan *firefly* yang mana menentukan tingkat kecerahan. Selanjutnya persamaan (2.9) terdiri dari selisih nilai solusi pada *firefly* i terhadap *firefly* j . Kemudian fungsi persamaan pergerakan *firefly* secara *random* (*rand*) yang menunjukkan adanya bilangan random yang kisarannya antara [0,1]. Variabel α yang memiliki kisaran antara [0,1]. Semua variabel yang terbentuk pada persamaan pergerakan *firefly* menjamin cara kerja algoritma cepat menuju solusi yang optimal.

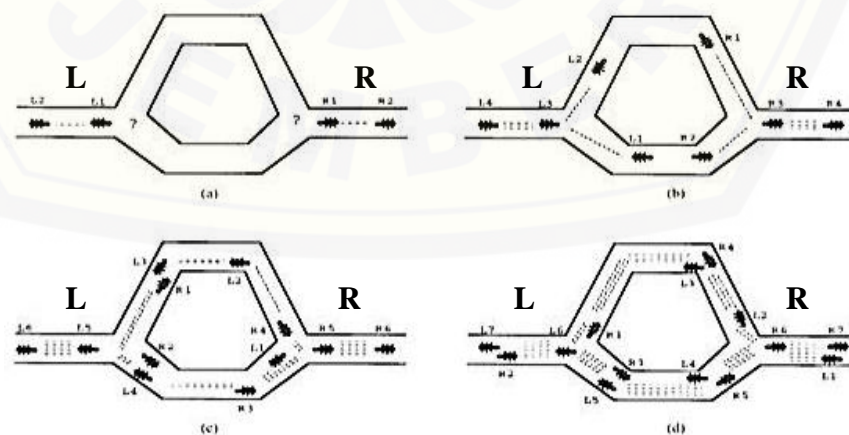


Gambar 2.5 Flowchart Firefly Algorithm

2.6 Ant Colony Optimization

Ant Colony Optimization (ACO) diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut. Semut mampu mengindera lingkungannya yang kompleks untuk mencari makanan dan kemudian kembali ke sarangnya dengan meninggalkan zat feromon pada rute-rute yang mereka lalui.

Feromon merupakan suatu senyawa kimia yang berasal dari kelenjar endokrin dan digunakan oleh makhluk hidup untuk mengenali sesama jenis, individu lain, kelompok, dan untuk membantu proses reproduksi. Berbeda dengan hormone, feromon menyebar ke luar tubuh dan hanya dapat mempengaruhi dan dikenali oleh individu lain yang sejenis. Senyawa ini mempunyai sifat yang mudah menguap. Koloni semut dapat menemukan rute optimal antara sarang dan sumber makanan berdasarkan feromon yang ditinggalkan pada lintasan yang telah dilalui. Semakin banyak semut yang melalui suatu lintasan, maka semakin banyak feromon yang ditinggalkan. Hal ini menyebabkan lintasan yang dilalui semut dalam jumlah sedikit, semakin lama semakin berkurang kepadatan semut yang melewatinya, atau bahkan tidak dilewati sama sekali. Sebaliknya, lintasan yang dilalui semut dalam jumlah banyak, semakin lama akan semakin bertambah kepadatan semut yang melewatinya, atau bahkan semua semut melalui lintasan tersebut. Pada Gambar 2.7 memperlihatkan bagaimana semut dapat menemukan rute optimal tersebut.



Gambar 2.6 Ilustrasi Perilaku Semut

Pada Gambar 2.7 (a) terdapat dua kelompok yang melakukan perjalanan. Kelompok semut L berangkat dari arah kiri ke kanan dan kelompok semut R berangkat dari kanan ke kiri.

Gambar 2.7 (b) kedua kelompok berangkat dari titik yang sama dan dalam posisi pengambilan keputusan jalan sebelah mana yang akan diambil. Kelompok L membagi dua kelompok lagi. Sebagian melalui jalan atas dan sebagian lagi melalui jalan bawah. Hal ini juga berlaku pada kelompok R.

Gambar 2.7 (c) menunjukkan bahwa kelompok semut berjalan pada kecepatan yang sama dengan meninggalkan feromon di jalan yang telah dilalui. Feromon yang ditinggalkan oleh kumpulan semut yang melalui jalan atas telah mengalami banyak penguapan karena jarak yang ditempuh lebih panjang daripada jalan bawah. Sedangkan feromon yang berada di jalan bawah penguapannya cenderung lebih lama. Karena jarak yang ditempuh lebih pendek sehingga feromon masih tersimpan dan dapat diikuti oleh semut-semut lainnya.

Gambar 2.7 (d) menunjukkan bahwa semut-semut yang lain pada akhirnya memutuskan untuk melewati jalan bawah karena feromon yang ditinggalkan masih banyak. Sedangkan feromon pada jalan atas sudah banyak menguap sehingga semut-semut tidak memilih jalan atas tersebut. Semakin banyak semut yang melalui jalan maka semakin banyak semut yang mengikutinya, semakin sedikit semut yang melalui jalan, maka feromon yang ditinggalkan semakin berkurang bahkan hilang. Dari sinilah kemudian terpilih jalur terpendek antara sarang dan sumber makanan (Leksono, 2009).

Ant Colony Optimization terdiri dari 2 (dua) aspek yaitu:

a. Aturan Keadaan Transisi

Aturan keadaan transisi merupakan suatu aturan yang digunakan untuk memilih titik berikutnya yang akan dilalui. Aturan keadaan transisi yang digunakan oleh *Ant Colony Optimization* disebut *random proportional rule* yang ditunjukkan oleh persamaan berikut. $P_k(i, j)$ merupakan probabilitas dari semut k pada titik i yang memilih untuk menuju ke titik j dengan persamaan (2.11),

$$P_k(i, j) = \begin{cases} \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{j \in M_k} [\tau(i, j)]^\alpha [\eta(i, j)]^\beta} & \text{jika } j \in M_k \\ 0 & \text{untuk lainnya} \end{cases} \quad (2.11)$$

keterangan:

$\tau(i, j)$ = Jumlah feromon pada sisi dari simpul i ke simpul j

$\eta(i, j) = \frac{1}{c_{ij}}$ merupakan *visibility* (dimana c_{ij} merupakan jarak dari simpul i menuju simpul j)

$M_k(i)$ = Himpunan yang berisi simpul-simpul yang telah dikunjungi oleh semut k yang sedang berada pada simpul i

β = Parameter perbandingan jumlah feromon relatif terhadap jarak ($\beta > 0$).

α = Parameter pengontrol pengaruh $\tau(i, j)$

b. Aturan Pembaharuan Feromon

Jumlah feromon pada sisi yang dilalui oleh semut akan diperbarui jumlahnya, yakni dikurangi atau ditambahi. Aturan pembaharuan feromon ini digunakan untuk mengurangi dan menambahkan feromon yang ditinggalkan oleh semut. Jumlah feromon $\tau'(i, j)$ akan berubah sesuai dengan menggunakan persamaan (2.12),

$$\tau'(i, j) = (1 - \rho)\tau(i, j) + \rho \cdot \Delta\tau_{i,j}^k \quad (2.12)$$

keterangan:

$\tau'(i, j)$ = Jumlah feromon yang baru

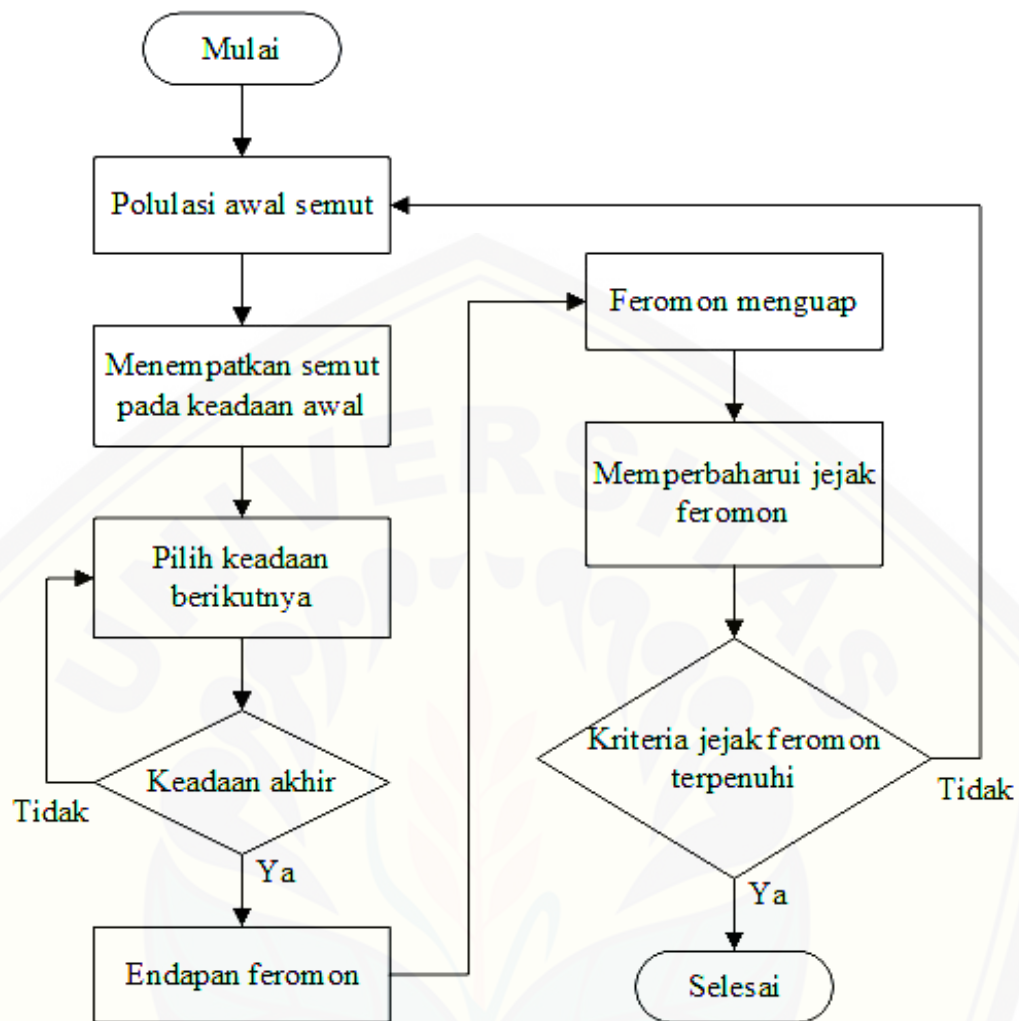
$\tau(i, j)$ = Jumlah feromon pada sisi dari simpul i ke simpul j

ρ = Tingkat penguapan feromon ($0 < \rho < 1$)

$\Delta\tau_{i,j}^k = \begin{cases} \frac{1}{L_k} & \text{jika } (i, j) \in \text{jarak tempuh terpendek yang dilakukan semut } k \\ 0 & \text{lainnya} \end{cases}$

L_k = panjang jarak tempuh terpendek yang dilakukan semut k

(Waliprana, 2009)



Gambar 2.7 Flowchart Ant Colony Optimazation

BAB 3. METODE PENELITIAN

3.1 Data Penelitian

Data yang digunakan dalam penelitian ini adalah data sekunder yang diambil dari Christofides dan Eilon (1969). Dalam data ini diasumsikan bahwa kota sebagai titik (*vertex*) pada graf dan jarak antar kota diasumsikan sebagai sisi (*edge*) pada graf. Jumlah kota yang akan diteliti sejumlah 50 titik yaitu

Tabel 3.1 Data Penelitian

Kota	x	y	Kota	x	y
1	41	49	35	63	65
2	35	17	36	2	60
3	55	45	37	20	20
4	55	20	38	5	5
5	15	30	39	60	12
6	25	30	40	40	25
7	20	50	41	42	7
8	10	43	42	24	12
9	55	60	43	23	3
10	30	60	44	11	14
11	20	65	45	6	38
12	50	35	46	2	48
13	30	25	47	8	56
14	15	10	48	13	52
15	30	5	49	6	68
16	10	20	50	47	47
17	5	30			
18	20	40			

3.2 Langkah-langkah Penelitian

Langkah-langkah yang dilakukan dalam penelitian ini untuk menyelesaikan *Travelling Salesman Problem* dengan menerapkan *Firefly Algorithm* dan *Ant Colony Optimization* adalah:

a. Studi Literatur

Langkah awal untuk melakukan penelitian ini yaitu melakukan studi literatur dari beberapa jurnal dan skripsi tentang *Firefly Algorithm* dan *Ant Colony Optimization*.

b. Pengambilan Data

Pengambilan data diambil dari Christofides dan Eilon (1969) yang menggunakan 50 titik kota.

c. Pengidentifikasian Data

Pada langkah ketiga ini dilakukan pengidentifikasian data berupa jarak antar kota.

d. Penerapan kedalam Graf.

Pengaplikasian data yang diasumsikan bahwa kota sebagai titik pada graf dan jarak antar kota diasumsikan sebagai sisi pada graf.

e. Penerapan *Firefly Algorithm* dan *Ant Colony Optimization*.

Langkah-langkah penerapan *Firefly Algorithm*:

- 1) Menginisialisasi parameter dan membentuk populasi awal *firefly*.
- 2) Menghitung fungsi tujuan sebagai penentuan tingkat intensitas cahaya awal yang diasumsikan untuk mencari rute awal.
- 3) Membandingkan tiap *firefly* dari nilai fungsi tujuan untuk mengetahui intensitas cahaya yang paling besar.
- 4) Memperbarui pergerakan tiap *firefly* dan menentukan jarak antar *firefly*.
- 5) Lakukan langkah (4) sampai mendapatkan batas iterasi untuk mendapatkan rute terpendek.

Langkah-langkah penerapan *Ant Colony Optimization*:

- 1) Menentukan parameter.
- 2) Membangun rute

Setiap semut dari masing-masing titik membangun rute, dengan menerapkan aturan keadaan transisi yaitu aturan kemungkinan solusi dari semut k pada titik i yang memilih untuk menuju titik j . Selanjutnya memperbaharui jumlah feromon pada sisi yang telah dilalui oleh semut k .

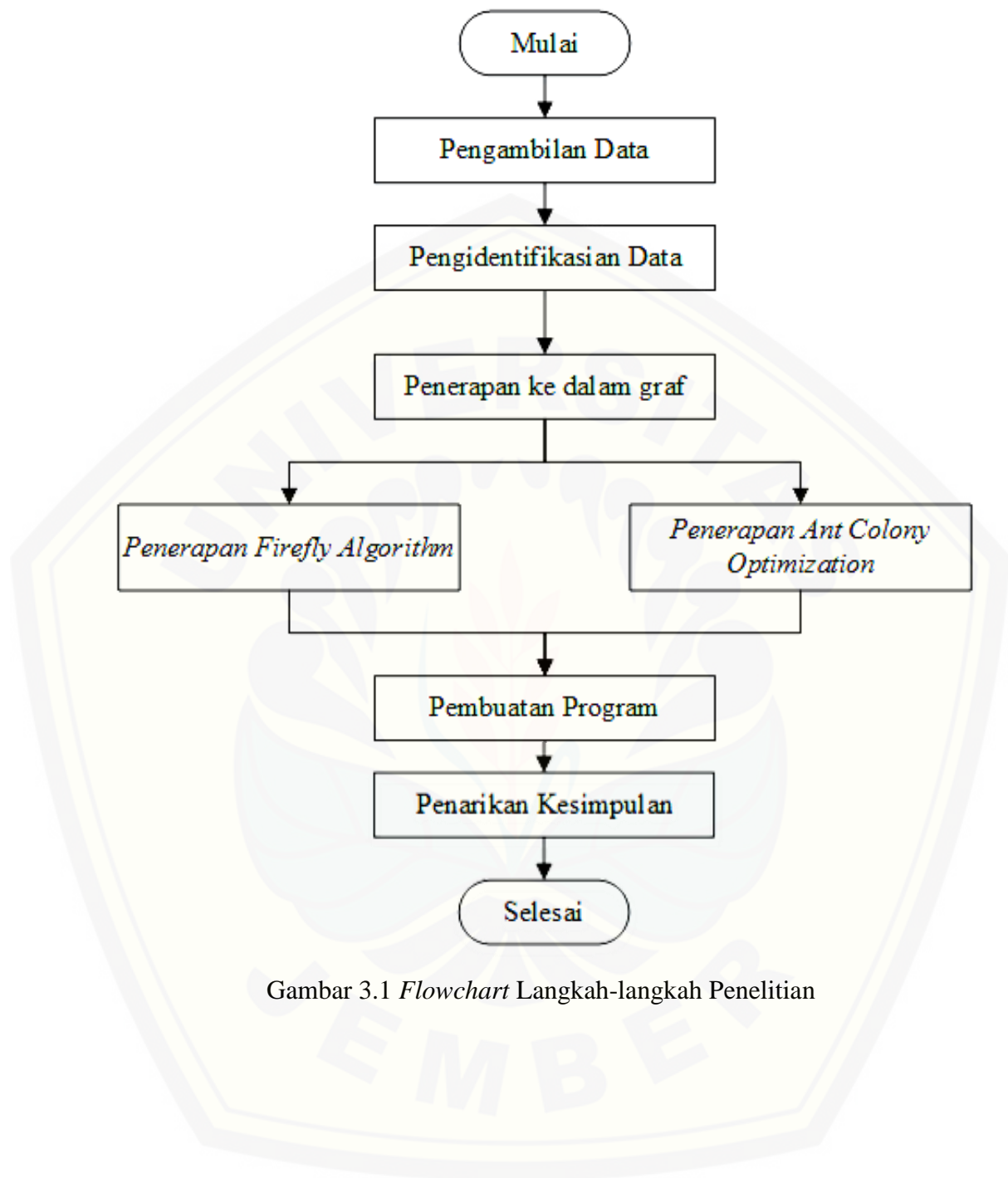
- 3) Ulangi langkah (2) sampai iterasi yang telah ditentukan.
- 4) Jika semua semut telah menyelesaikan rute masing-masing, selanjutnya hitung nilai total jarak tempuh pada semut.
- 5) Setelah menghitung nilai total jarak tempuh pada semut, pilih nilai total jarak tempuh minimum lalu perbarui jumlah feromon pada sisi-sisi yang dilewati semut terbaik.

f. Pembuatan Program

Pembuatan program *Firefly Algorithm* dan *Ant Colony Optimization* dengan menggunakan *software* MATLAB. Pada langkah ini penulis membuat skrip program dan desain tampilan sesuai algoritma yang digunakan.

g. Penarikan Kesimpulan

Menyimpulkan hasil dari penelitian yang telah dilakukan.



Gambar 3.1 *Flowchart* Langkah-langkah Penelitian

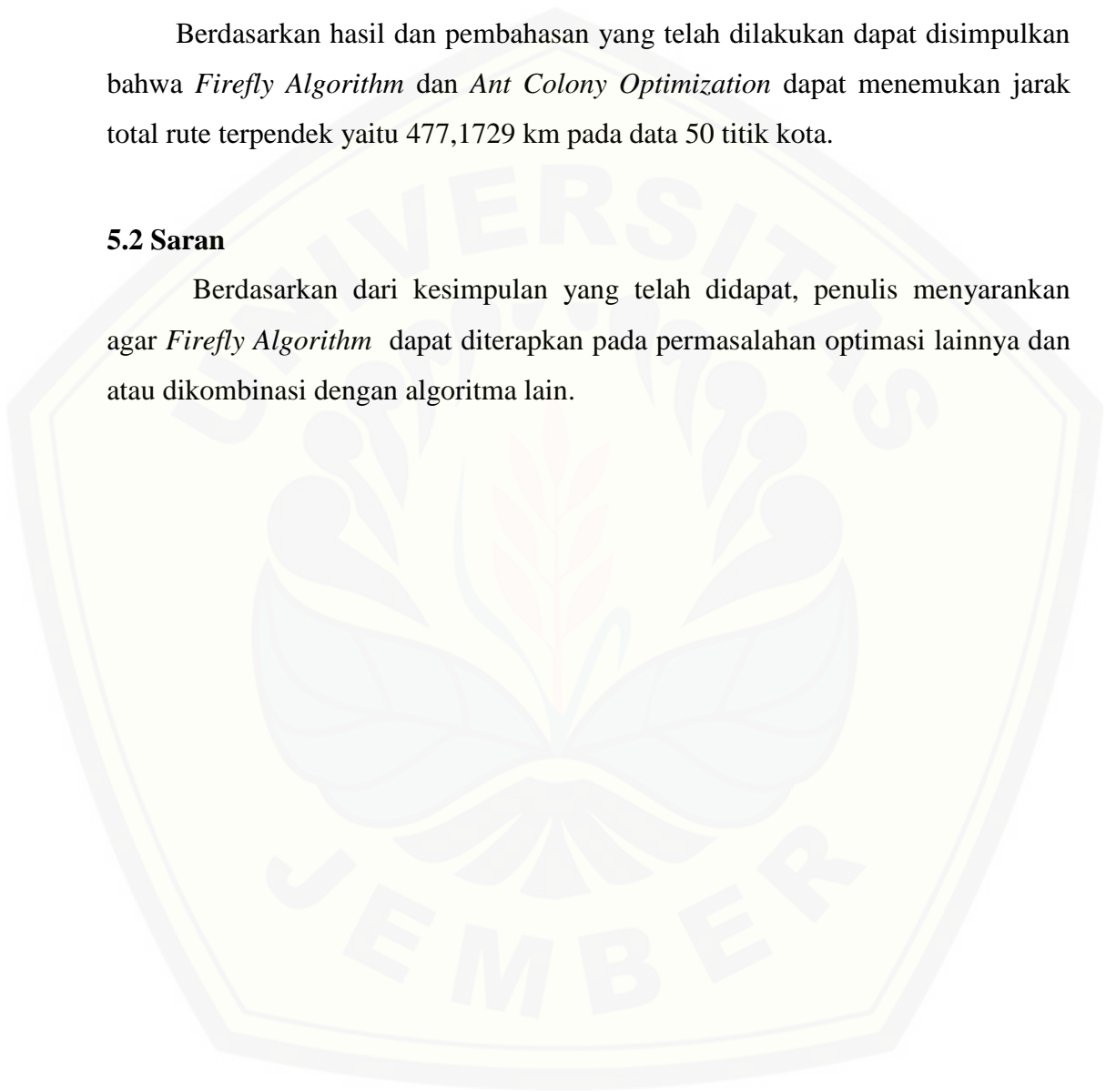
BAB 5. KESIMPULAN

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan yang telah dilakukan dapat disimpulkan bahwa *Firefly Algorithm* dan *Ant Colony Optimization* dapat menemukan jarak total rute terpendek yaitu 477,1729 km pada data 50 titik kota.

5.2 Saran

Berdasarkan dari kesimpulan yang telah didapat, penulis menyarankan agar *Firefly Algorithm* dapat diterapkan pada permasalahan optimasi lainnya dan atau dikombinasi dengan algoritma lain.



DAFTAR PUSTAKA

- Annisa, A. 2016. *Penerapan Algoritma Firefly pada Permasalahan Knapsack 0-1*. Tidak Dipublikasikan. Skripsi. Jember: Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember.
- Apriliani, I. 2011. *Penyelesaian Travelling Salesman Problem (TSP) Menggunakan Algoritma Semut Dan Algoritma Cheapest Insertion Heuristic (CIH)*. Tidak Dipublikasikan. Skripsi. Jember: Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember.
- Chartrand, G dan O.R Oellermann. 1993. *Applied and Algorithmic Graph Theory*. New York: McGraw Hill, Inc.
- Christofides, N dan Eilon, S. 1969. Expected Distances in Distribution Problems. *Operational Research Quarterly*. Vol. 20, Hal:437-443.
- Dorigo, M. dan Gambardella, L.M. 1996. *Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem*. Universite libre de Bruxelles.
- Dorigo, M. dan Gambardella, L.M. 1997. *Ant Colony for the Travelling Salesman Problem System*. TR/IRIDIA/1996-3. Universite libre de Bruxelles. Belgium.
- Dorigo, M. dan Stutzle, T. 2004. *Ant Colony Optimization A Bradford Book*. The MIT Press Cambridge. Massachusetts London. England.
- Fonda, W. 2011. Penerapan Sirkuit Hamilton dalam Perencanaan Lintasan Trem di ITB. Program Studi Teknik Informatika, ITB, Bandung.
- Ilyas. M. 2007. *Perbandingan Penggunaan Algoritma Greedy Dan Algoritma Genetika Dalam Penyelesaian Masalah Perjalanan Salesman*. Tidak Dipublikasikan. Skripsi. Jember: Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Jember.
- Leksono, A. 2009. *Algoritma Ant Colony Optimization (ACO) untuk menyelesaikan Travelling Salesman Problem*. Tidak Dipublikasikan. Skripsi. Semarang: Program Studi Jurusan Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Diponegoro.

- Maria, A., Sinaga, E. Y., dan Iwo, M. H. 2005. Penyelesaian Masalah Travelling Salesman Problem Menggunakan Ant Colony Optimization (ACO). Disertasi. Laboratorium Ilmu dan Rekayasa Komputasi Departemen Teknik Informatika ITB. Bandung.
- Munir, Rinaldi. 2003. *Matematika Diskrit*. Edisi Kedua. Bandung: Informatika.
- Sharad dan Gopal. 2013. Solving Travelling Salesman Problem using Firefly Algorithm. *International Journal for Research in Science & Advance Technologies* 2(2): 2319-2690.
- Waliprana, W. E. 2003. Ant Colony Optimization. *Jurnal Ilmu Komputer dan Teknologi Informasi*, Vol. III, No. 2.
- Wiyanti, D. T. 2013. Algoritma Optimasi Untuk Penyelesaian Travelling Salesman Problem. *Jurnal Transformatika*, Vol. 11, No. 1, hal: 1-6.
- Wulandari, D. 2015. *Sistem Optimasi Rute Terpendek Pengangkutan Sampah di Surabaya Menggunakan Ant Colony Optimization (ACO)*. Tidak Dipublikasikan. Skripsi. Jember: Program Studi Sistem Informasi Universitas Jember.
- Yang, X. S. 2010. *Naturazed-Inspired Metaheuristic Algorithm*. Second Edition. United Kingdom: Luniver Press.

LAMPIRAN

Lampiran A. Jarak antar titik koordinat

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
V1	0	32,5576	14,5602	32,2025	32,2025	24,8395	21,0238	31,5753	17,8045	15,5563	23,7065	16,6433	26,4008	46,8722	45,3542
V2	32,5576	0	34,4093	20,2237	23,8537	16,4012	36,2491	36,0694	47,4236	43,2897	45,5412	23,4307	9,43398	21,1896	13
V3	14,5602	34,4093	0	25	42,72	33,541	35,3553	45,0444	15	29,1548	38,0789	11,1803	32,0156	53,1507	47,1699
V4	32,2025	20,2237	25	0	41,2311	31,6228	46,0977	50,5371	40	47,1699	53,1507	15,8114	25,4951	41,2311	29,1548
V5	32,2025	23,8537	42,72	41,2311	0	10	20,6155	13,9284	50	33,541	30,4138	35,3553	15,8114	20	29,1548
V6	24,8395	16,4012	33,541	31,6228	10	0	20,6155	19,8494	42,4264	30,4138	30,4138	25,4951	7,07107	22,3607	25,4951
V7	21,0238	36,2491	35,3553	46,0977	20,6155	20,6155	0	12,2066	36,4005	14,1421	10	33,541	26,9258	40,3113	46,0977
V8	31,5753	36,0694	45,0444	50,5371	13,9284	19,8494	12,2066	0	48,1041	26,2488	19,7231	40,7922	26,9072	33,3766	42,9418
V9	17,8045	47,4236	15	40	50	42,4264	36,4005	48,1041	0	25	35	25,4951	43,0116	64,0312	60,4152
V10	15,5563	43,2897	29,1548	47,1699	33,541	30,4138	14,1421	26,2488	25	0	10	32,0156	35	52,2015	55
V11	23,7065	45,5412	38,0789	53,1507	30,4138	30,4138	10	19,7231	35	10	0	39,0512	36,4005	50,2494	55,9017
V12	16,6433	23,4307	11,1803	15,8114	35,3553	25,4951	33,541	40,7922	25,4951	32,0156	39,0512	0	22,3607	43,0116	36,0555
V13	26,4008	9,43398	32,0156	25,4951	15,8114	7,07107	26,9258	26,9072	43,0116	35	36,4005	22,3607	0	21,2132	20
V14	46,8722	21,1896	53,1507	41,2311	20	22,3607	40,3113	33,3766	64,0312	52,2015	50,2494	43,0116	21,2132	0	15,8114
V15	45,3542	13	47,1699	29,1548	29,1548	25,4951	46,0977	42,9418	60,4152	55	55,9017	36,0555	20	15,8114	0
V16	42,45	25,1794	51,4782	45	11,1803	18,0278	31,6228	23	60,208	44,7214	41,2311	42,72	20,6155	11,1803	25
V17	40,7063	32,6956	52,2015	50,9902	10	20	25	13,9284	58,3095	39,0512	33,541	45,2769	25,4951	22,3607	35,3553
V18	22,8473	27,4591	35,3553	40,3113	11,1803	11,1803	10	10,4403	40,3113	22,3607	20	30,4138	18,0278	30,4138	36,4005
V19	28,2312	47,4236	42,72	56,5685	30	31,6228	11,1803	17,72	40	15	5	43,0116	38,0789	50	57,0088
V20	16,4924	49,0306	22,3607	46,0977	46,0977	40,3113	29,1548	41,3401	11,1803	15,8114	25,4951	30,4138	42,72	62,6498	61,8466

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
V21	29,2746	10,4403	26,9258	10	31,6228	22,3607	39,0512	41,8808	41,2311	42,72	47,1699	15,8114	15,8114	31,6228	21,2132
V22	39,2046	12,2066	36,4005	14,1421	36,0555	28,2843	47,1699	48,1041	50,9902	52,2015	55,9017	25,4951	21,2132	30	15,8114
V23	44,1814	15,6205	41,2311	18,0278	39,0512	32,0156	51,4782	51,6624	55,9017	57,0088	60,4152	30,4138	25	30,4138	15
V24	27,7849	34,9857	14,1421	18,0278	50,2494	40,3113	47,4342	55,5788	26,9258	43,0116	51,4782	15	36,4005	55,9017	46,0977
V25	37,6431	30,1496	26,9258	10	50,9902	41,2311	54,0833	59,6154	41,2311	53,1507	60,208	21,2132	35,3553	50,9902	38,0789
V26	19,4165	16,4012	18,0278	14,1421	30	20	32,0156	37,3363	31,6228	33,541	39,0512	7,07107	15,8114	36,0555	29,1548
V27	10,8167	23	20,6155	28,2843	22,3607	14,1421	18,0278	25,1794	28,2843	20,6155	25	15,8114	15,8114	36,0555	35,3553
V28	12	20,8806	16,1245	22,0227	26,9258	17,4642	24,6982	31,5753	26,9258	25,4951	31,1448	9,21954	16,2788	37,4833	33,8378
V29	24,0416	38,2884	9,48683	23,7697	50,448	40,8044	44,7214	54,0093	20,1246	38,4708	47,5395	15,6525	38,0132	58,5235	50,2494
V30	11,0454	43,2897	21,2132	42,72	39,0512	33,541	22,3607	34,4819	15	10	20	26,9258	36,4005	55,9017	55,9017
V31	10,4403	35,2278	25	40	27,2029	22,8035	11,1803	22,8473	25,2982	8,06226	13,6015	25,4951	27,0185	44,9444	47,0106
V32	20,8806	52	31,241	52,9245	43,8292	40,2616	24,2074	36,0694	21,9317	10,2956	17,4929	37,1618	44,2832	62,2977	64,195
V33	12,3693	39,3573	7,28011	32,0624	43,909	35,609	33,0606	43,9318	8,24621	24,3516	33,9559	17,2627	35,4683	56,6392	52,3259
V34	24,7386	48,4149	14,1421	36,4005	55,9017	47,1699	45,2769	56,2939	11,1803	35,3553	45,2769	25	46,0977	67,2681	61,0328
V35	27,2029	55,5698	21,5407	45,7056	59,4054	51,6624	45,5412	57,3847	9,43398	33,3766	43,2897	32,6956	51,8556	73	68,4763
V36	40,5216	54,2033	55,0818	66,4003	32,6956	37,8021	20,5913	18,7883	53	28	18	54,1202	44,8219	51,6624	61,7171
V37	35,805	15,2971	43,0116	35	11,1803	11,1803	30	25,0799	53,1507	41,2311	40	33,541	11,1803	11,1803	18,0278
V38	56,8507	32,311	64,0312	52,2015	26,9258	32,0156	47,4342	38,3275	74,3303	60,4152	57,0088	54,0833	32,0156	11,1803	25
V39	41,5933	25,4951	33,3766	9,43398	48,4665	39,3573	55,1725	58,8303	48,2597	56,6039	62,482	25,0799	32,6956	45,0444	30,8058
V40	24,0208	9,43398	25	15,8114	25,4951	15,8114	32,0156	34,9857	38,0789	36,4005	40,3113	14,1421	10	29,1548	22,3607
V41	42,0119	12,2066	40,1622	18,3848	35,4683	28,6007	48,3011	48,1664	54,5711	54,3415	57,3847	29,1204	21,6333	27,1662	12,1655
V42	40,7185	12,083	45,2769	32,0156	20,1246	18,0278	38,2099	34,0147	57,1402	48,3735	48,1664	34,7131	14,3178	9,21954	9,21954
V43	49,3964	18,4391	52,8015	36,2353	28,1603	27,074	47,0956	42,0595	65,3682	57,4282	57,0789	41,8688	23,0868	10,6301	7,28011
V44	46,0977	24,1868	53,8238	44,4072	16,4924	21,2603	37,108	29,0172	63,6553	49,7695	46,8722	44,2945	21,9545	5,65685	21,0238
V45	36,6879	35,805	49,4975	52,2015	12,0416	20,6155	18,4391	6,40312	53,7122	32,5576	26,0768	44,1022	27,2947	29,4109	40,8044

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
V46	39,0128	45,2769	53,0848	59,9416	22,2036	29,2062	18,1108	9,43398	54,3415	30,4631	21,6333	49,7293	36,2353	40,1622	51,3128
V47	33,7343	47,4342	48,2701	59,203	26,9258	31,0644	13,4164	13,1529	47,1699	22,3607	12,6491	46,9574	38,0132	46,5296	55,5428
V48	28,1603	41,3401	42,5793	52,8015	22,0907	25,0599	7,28011	9,48683	42,7551	18,7883	10,6301	40,7185	31,9061	42,0476	49,98
V49	39,8246	58,6686	54,1295	68,593	39,0512	42,4853	22,8035	25,318	49,6488	25,2982	16,1245	55	49,2443	58,6941	67,4166
V50	6,32456	32,311	8,24621	28,1603	36,2353	27,8029	27,1662	37,2156	15,2643	21,4009	29,9666	12,3693	27,8029	48,9183	45,31

	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
V1	42,45	40,7063	22,8473	28,2312	16,4924	29,2746	39,2046	44,1814	27,7849	37,6431	19,4165	10,8167	12	24,0416	11,0454
V2	25,1794	32,6956	27,4591	47,4236	49,0306	10,4403	12,2066	15,6205	34,9857	30,1496	16,4012	23	20,8806	38,2884	43,2897
V3	51,4782	52,2015	35,3553	42,72	22,3607	26,9258	36,4005	41,2311	14,1421	26,9258	18,0278	20,6155	16,1245	9,48683	21,2132
V4	45	50,9902	40,3113	56,5685	46,0977	10	14,1421	18,0278	18,0278	10	14,1421	28,2843	22,0227	23,7697	42,72
V5	11,1803	10	11,1803	30	46,0977	31,6228	36,0555	39,0512	50,2494	50,9902	30	22,3607	26,9258	50,448	39,0512
V6	18,0278	20	11,1803	31,6228	40,3113	22,3607	28,2843	32,0156	40,3113	41,2311	20	14,1421	17,4642	40,8044	33,541
V7	31,6228	25	10	11,1803	29,1548	39,0512	47,1699	51,4782	47,4342	54,0833	32,0156	18,0278	24,6982	44,7214	22,3607
V8	23	13,9284	10,4403	17,72	41,3401	41,8808	48,1041	51,6624	55,5788	59,6154	37,3363	25,1794	31,5753	54,0093	34,4819
V9	60,208	58,3095	40,3113	40	11,1803	41,2311	50,9902	55,9017	26,9258	41,2311	31,6228	28,2843	26,9258	20,1246	15
V10	44,7214	39,0512	22,3607	15	15,8114	42,72	52,2015	57,0088	43,0116	53,1507	33,541	20,6155	25,4951	38,4708	10
V11	41,2311	33,541	20	5	25,4951	47,1699	55,9017	60,4152	51,4782	60,208	39,0512	25	31,1448	47,5395	20
V12	42,72	45,2769	30,4138	43,0116	30,4138	15,8114	25,4951	30,4138	15	21,2132	7,07107	15,8114	9,21954	15,6525	26,9258
V13	20,6155	25,4951	18,0278	38,0789	42,72	15,8114	21,2132	25	36,4005	35,3553	15,8114	15,8114	16,2788	38,0132	36,4005
V14	11,1803	22,3607	30,4138	50	62,6498	31,6228	30	30,4138	55,9017	50,9902	36,0555	36,0555	37,4833	58,5235	55,9017
V15	25	35,3553	36,4005	57,0088	61,8466	21,2132	15,8114	15	46,0977	38,0789	29,1548	35,3553	33,8378	50,2494	55,9017
V16	0	11,1803	22,3607	40,3113	57,0088	35	36,4005	38,0789	57,0088	55	36,4005	32,0156	35,3553	58,3095	50
V17	11,1803	0	18,0278	31,6228	53,1507	41,2311	44,7214	47,1699	60,208	60,8276	40	31,6228	36,6742	60,208	46,0977

	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
V18	22,3607	18,0278	0	20,6155	35,3553	32,0156	39,0512	43,0116	45,2769	49,2443	26,9258	15	21,2132	44,0454	28,2843
V19	40,3113	31,6228	20,6155	0	30,4138	50	58,3095	62,6498	55,9017	64,0312	42,4264	28,2843	34,7131	52,2015	25
V20	57,0088	53,1507	35,3553	30,4138	0	45	55	60	36,0555	49,2443	35	26,9258	28,2843	29,8329	7,07107
V21	35	41,2311	32,0156	50	45	0	10	15	25	20	10	22,3607	17,4642	29,0689	40,3113
V22	36,4005	44,7214	39,0512	58,3095	55	10	0	5	32,0156	22,3607	20	31,6228	27,2947	37,2156	50,2494
V23	38,0789	47,1699	43,0116	62,6498	60	15	5	0	36,0555	25	25	36,4005	32,249	41,5933	55,2268
V24	57,0088	60,208	45,2769	55,9017	36,0555	25	32,0156	36,0555	0	15	20,6155	30,4138	24,0832	7,07107	35,3553
V25	55	60,8276	49,2443	64,0312	49,2443	20	22,3607	25	15	0	22,3607	36,0555	29,4109	22,0227	47,1699
V26	36,4005	40	26,9258	42,4264	35	10	20	25	20,6155	22,3607	0	14,1421	8,06226	22,4722	30,4138
V27	32,0156	31,6228	15	28,2843	26,9258	22,3607	31,6228	36,4005	30,4138	36,0555	14,1421	0	6,7082	29,0689	20,6155
V28	35,3553	36,6742	21,2132	34,7131	28,2843	17,4642	27,2947	32,249	24,0832	29,4109	8,06226	6,7082	0	23,5372	23,0217
V29	58,3095	60,208	44,0454	52,2015	29,8329	29,0689	37,2156	41,5933	7,07107	22,0227	22,4722	29,0689	23,5372	0	30
V30	50	46,0977	28,2843	25	7,07107	40,3113	50,2494	55,2268	35,3553	47,1699	30,4138	20,6155	23,0217	30	0
V31	38,2753	34,0588	16,2788	17,8885	19,105	34,9285	44,2719	49,0408	38,0132	46,6905	26,0768	12,6491	18,0278	34,4819	12,0416
V32	55,0091	49,2037	32,6497	21,9317	10,7703	50,01	59,8415	64,7765	45,3431	57,4543	40,2616	29	32,5576	39,6232	10,2956
V33	53,6004	52,8015	35,1141	38,833	15,2643	32,9848	42,7551	47,676	20,8087	34,176	23,4094	21,6333	19,2094	14,8661	15,2643
V34	65,192	65	47,4342	50,2494	22,3607	40,3113	49,2443	53,8516	20	35	32,0156	33,541	30	13,0384	25,4951
V35	69,527	67,7422	49,7393	48,2597	18	48,4665	57,8705	62,6418	30,0666	45,0444	39,3573	37,5366	35,609	23,0217	23,5372
V36	40,7922	30,1496	26,9072	13	43,2897	58,7282	65,9469	69,814	67,7791	74,6257	52,4309	38,5876	45,2769	64,56	38
V37	10	18,0278	20	40,3113	51,4782	25	26,9258	29,1548	47,4342	45	26,9258	25	27,0185	49,1935	44,7214
V38	15,8114	25	38,0789	55,9017	72,111	42,72	40,3113	40	67,082	61,8466	47,1699	46,0977	48,1664	69,6419	65,192
V39	50,636	57,8705	48,8262	65,7951	55,0818	17	15,1327	16,5529	23,5372	9,43398	23,4307	37,5366	31,4006	30,2655	52
V40	30,4138	35,3553	25	43,0116	40,3113	7,07107	15,8114	20,6155	26,9258	25,4951	7,07107	15,8114	12,0416	29,4109	35
V41	34,5398	43,566	39,6611	59,4811	58,0775	13,3417	4,24264	3,60555	36,2353	26,4197	23,1948	33,7343	30,0167	41,3401	53,0377
V42	16,1245	26,1725	28,2843	48,8365	57,0088	22,4722	21,095	22,1359	47,0106	41,7732	27,6586	30,0832	30,2324	50	50,5964

	V16	V17	V18	V19	V20	V21	V22	V23	V24	V25	V26	V27	V28	V29	V30
V43	21,4009	32,45	37,1214	57,5587	65,7875	27,8029	23,0868	22,0907	52,8015	45,31	34,8281	38,8973	38,4708	56,5862	59,4811
V44	6,08276	17,088	27,5136	46,1736	61,2944	34,5254	34,2345	35,171	57,9396	54,3323	37,5766	35,3836	37,8021	59,9416	54,3783
V45	18,4391	8,06226	14,1421	23,7697	47,4342	42,9535	48,0104	51,0882	59,0762	61,6847	39,8121	29,0689	35,0143	58,1378	40,4969
V46	29,1204	18,2483	19,6977	17,6918	46,2385	51,3128	57,3847	60,8112	64,3273	68,942	46,6154	33,9559	40,5216	62,2896	39,8497
V47	36,0555	26,1725	20	8,06226	38,0789	51,6236	59,0339	63,0079	60,7454	67,4166	45,2217	31,3847	38,0789	57,7235	32,249
V48	32,1403	23,4094	13,8924	8,24621	34,5398	45,2548	52,8015	56,8595	54,7083	61,0574	38,833	25,0599	31,7648	51,9711	28,1603
V49	48,1664	38,0132	31,305	12,0416	39,1152	61,8466	69,8928	74,0945	67,6018	76,0592	54,4518	40,3113	46,7547	63,561	34,9285
V50	45,8039	45,31	27,8927	34,5398	18,1108	27,074	37,054	42,0476	21,6333	32,45	17,1172	13,8924	11,6619	17,72	14,7648

	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
V1	10,4403	20,8806	12,3693	24,7386	27,2029	40,5216	35,805	56,8507	41,5933	24,0208	42,0119	40,7185	49,3964	46,0977	36,6879
V2	35,2278	52	39,3573	48,4149	55,5698	54,2033	15,2971	32,311	25,4951	9,43398	12,2066	12,083	18,4391	24,1868	35,805
V3	25	31,241	7,28011	14,1421	21,5407	55,0818	43,0116	64,0312	33,3766	25	40,1622	45,2769	52,8015	53,8238	49,4975
V4	40	52,9245	32,0624	36,4005	45,7056	66,4003	35	52,2015	9,43398	15,8114	18,3848	32,0156	36,2353	44,4072	52,2015
V5	27,2029	43,8292	43,909	55,9017	59,4054	32,6956	11,1803	26,9258	48,4665	25,4951	35,4683	20,1246	28,1603	16,4924	12,0416
V6	22,8035	40,2616	35,609	47,1699	51,6624	37,8021	11,1803	32,0156	39,3573	15,8114	28,6007	18,0278	27,074	21,2603	20,6155
V7	11,1803	24,2074	33,0606	45,2769	45,5412	20,5913	30	47,4342	55,1725	32,0156	48,3011	38,2099	47,0956	37,108	18,4391
V8	22,8473	36,0694	43,9318	56,2939	57,3847	18,7883	25,0799	38,3275	58,8303	34,9857	48,1664	34,0147	42,0595	29,0172	6,40312
V9	25,2982	21,9317	8,24621	11,1803	9,43398	53	53,1507	74,3303	48,2597	38,0789	54,5711	57,1402	65,3682	63,6553	53,7122
V10	8,06226	10,2956	24,3516	35,3553	33,3766	28	41,2311	60,4152	56,6039	36,4005	54,3415	48,3735	57,4282	49,7695	32,5576
V11	13,6015	17,4929	33,9559	45,2769	43,2897	18	40	57,0088	62,482	40,3113	57,3847	48,1664	57,0789	46,8722	26,0768
V12	25,4951	37,1618	17,2627	25	32,6956	54,1202	33,541	54,0833	25,0799	14,1421	29,1204	34,7131	41,8688	44,2945	44,1022
V13	27,0185	44,2832	35,4683	46,0977	51,8556	44,8219	11,1803	32,0156	32,6956	10	21,6333	14,3178	23,0868	21,9545	27,2947
V14	44,9444	62,2977	56,6392	67,2681	73	51,6624	11,1803	11,1803	45,0444	29,1548	27,1662	9,21954	10,6301	5,65685	29,4109

	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
V15	47,0106	64,195	52,3259	61,0328	68,4763	61,7171	18,0278	25	30,8058	22,3607	12,1655	9,21954	7,28011	21,0238	40,8044
V16	38,2753	55,0091	53,6004	65,192	69,527	40,7922	10	15,8114	50,636	30,4138	34,5398	16,1245	21,4009	6,08276	18,4391
V17	34,0588	49,2037	52,8015	65	67,7422	30,1496	18,0278	25	57,8705	35,3553	43,566	26,1725	32,45	17,088	8,06226
V18	16,2788	32,6497	35,1141	47,4342	49,7393	26,9072	20	38,0789	48,8262	25	39,6611	28,2843	37,1214	27,5136	14,1421
V19	17,8885	21,9317	38,833	50,2494	48,2597	13	40,3113	55,9017	65,7951	43,0116	59,4811	48,8365	57,5587	46,1736	23,7697
V20	19,105	10,7703	15,2643	22,3607	18	43,2897	51,4782	72,111	55,0818	40,3113	58,0775	57,0088	65,7875	61,2944	47,4342
V21	34,9285	50,01	32,9848	40,3113	48,4665	58,7282	25	42,72	17	7,07107	13,3417	22,4722	27,8029	34,5254	42,9535
V22	44,2719	59,8415	42,7551	49,2443	57,8705	65,9469	26,9258	40,3113	15,1327	15,8114	4,24264	21,095	23,0868	34,2345	48,0104
V23	49,0408	64,7765	47,676	53,8516	62,6418	69,814	29,1548	40	16,5529	20,6155	3,60555	22,1359	22,0907	35,171	51,0882
V24	38,0132	45,3431	20,8087	20	30,0666	67,7791	47,4342	67,082	23,5372	26,9258	36,2353	47,0106	52,8015	57,9396	59,0762
V25	46,6905	57,4543	34,176	35	45,0444	74,6257	45	61,8466	9,43398	25,4951	26,4197	41,7732	45,31	54,3323	61,6847
V26	26,0768	40,2616	23,4094	32,0156	39,3573	52,4309	26,9258	47,1699	23,4307	7,07107	23,1948	27,6586	34,8281	37,5766	39,8121
V27	12,6491	29	21,6333	33,541	37,5366	38,5876	25	46,0977	37,5366	15,8114	33,7343	30,0832	38,8973	35,3836	29,0689
V28	18,0278	32,5576	19,2094	30	35,609	45,2769	27,0185	48,1664	31,4006	12,0416	30,0167	30,2324	38,4708	37,8021	35,0143
V29	34,4819	39,6232	14,8661	13,0384	23,0217	64,56	49,1935	69,6419	30,2655	29,4109	41,3401	50	56,5862	59,9416	58,1378
V30	12,0416	10,2956	15,2643	25,4951	23,5372	38	44,7214	65,192	52	35	53,0377	50,5964	59,4811	54,3783	40,4969
V31	0	17,4642	22	34,1321	34,5398	30,0832	33,8378	53,7122	49,4065	28,4605	46,3249	40,6079	49,6488	42,9418	28,6531
V32	17,4642	0	24,7588	33,1059	28,2843	34,2053	51,2445	70,6824	62,2415	44,2832	62,3939	58,0517	67,082	60,0083	42,45
V33	22	24,7588	0	12,3693	16,4012	51,6236	45,9674	67,1789	40,6079	29,9666	46,3249	49,4065	57,4543	56,6392	49,0408
V34	34,1321	33,1059	12,3693	0	10,198	63,1981	57,0088	78,1025	43,2897	39,0512	53,2259	59,4138	66,8431	67,8012	61,4003
V35	34,5398	28,2843	16,4012	10,198	0	61,2046	62,2415	83,4506	53,0848	46,1411	61,6847	65,8027	73,7835	72,8354	63,0714
V36	30,0832	34,2053	51,6236	63,1981	61,2046	0	43,8634	55,0818	75,2861	51,6624	66,4003	52,8015	60,7454	46,8722	22,3607
V37	33,8378	51,2445	45,9674	57,0088	62,2415	43,8634	0	21,2132	40,7922	20,6155	25,5539	8,94427	17,2627	10,8167	22,8035
V38	53,7122	70,6824	67,1789	78,1025	83,4506	55,0818	21,2132	0	55,4437	40,3113	37,054	20,2485	18,1108	10,8167	33,0151
V39	49,4065	62,2415	40,6079	43,2897	53,0848	75,2861	40,7922	55,4437	0	23,8537	18,6815	36	38,0789	49,0408	59,9333

	V31	V32	V33	V34	V35	V36	V37	V38	V39	V40	V41	V42	V43	V44	V45
V40	28,4605	44,2832	29,9666	39,0512	46,1411	51,6624	20,6155	40,3113	23,8537	0	18,1108	20,6155	27,8029	31,0161	36,4005
V41	46,3249	62,3939	46,3249	53,2259	61,6847	66,4003	25,5539	37,054	18,6815	18,1108	0	18,6815	19,4165	31,7805	47,5079
V42	40,6079	58,0517	49,4065	59,4138	65,8027	52,8015	8,94427	20,2485	36	20,6155	18,6815	0	9,05539	13,1529	31,6228
V43	49,6488	67,082	57,4543	66,8431	73,7835	60,7454	17,2627	18,1108	38,0789	27,8029	19,4165	9,05539	0	16,2788	38,9102
V44	42,9418	60,0083	56,6392	67,8012	72,8354	46,8722	10,8167	10,8167	49,0408	31,0161	31,7805	13,1529	16,2788	0	24,5153
V45	28,6531	42,45	49,0408	61,4003	63,0714	22,3607	22,8035	33,0151	59,9333	36,4005	47,5079	31,6228	38,9102	24,5153	0
V46	29,2746	39,1152	51,1566	63,3877	63,3246	12	33,2866	43,1045	68,2642	44,4185	57,28	42,19	49,6588	35,171	10,7703
V47	23,3452	29,9666	45,1774	57,0088	55,7315	7,2111	37,9473	51,0882	68,1175	44,5533	59,6406	46,8188	55,0818	42,107	18,1108
V48	18	27,8029	40	52,0865	51,6624	13,6015	32,7567	47,676	61,7171	38,1838	53,535	41,4849	50,01	38,0526	15,6525
V49	29,6816	29,0172	49,6488	60,4152	57,0789	8,94427	50	63,0079	77,7946	54,8179	70,8308	58,8218	67,1863	54,231	30
V50	16,7631	25,0599	7,81025	19,6977	24,0832	46,8402	38,1838	59,397	37,3363	23,0868	40,3113	41,8808	50,1199	48,8365	41,9762

	V46	V47	V48	V49	V50
V1	39,0128	33,7343	28,1603	39,8246	6,32456
V2	45,2769	47,4342	41,3401	58,6686	32,311
V3	53,0848	48,2701	42,5793	54,1295	8,24621
V4	59,9416	59,203	52,8015	68,593	28,1603
V5	22,2036	26,9258	22,0907	39,0512	36,2353
V6	29,2062	31,0644	25,0599	42,4853	27,8029
V7	18,1108	13,4164	7,28011	22,8035	27,1662
V8	9,43398	13,1529	9,48683	25,318	37,2156
V9	54,3415	47,1699	42,7551	49,6488	15,2643
V10	30,4631	22,3607	18,7883	25,2982	21,4009
V11	21,6333	12,6491	10,6301	16,1245	29,9666

	V46	V47	V48	V49	V50
V12	49,7293	46,9574	40,7185	55	12,3693
V13	36,2353	38,0132	31,9061	49,2443	27,8029
V14	40,1622	46,5296	42,0476	58,6941	48,9183
V15	51,3128	55,5428	49,98	67,4166	45,31
V16	29,1204	36,0555	32,1403	48,1664	45,8039
V17	18,2483	26,1725	23,4094	38,0132	45,31
V18	19,6977	20	13,8924	31,305	27,8927
V19	17,6918	8,06226	8,24621	12,0416	34,5398
V20	46,2385	38,0789	34,5398	39,1152	18,1108
V21	51,3128	51,6236	45,2548	61,8466	27,074
V22	57,3847	59,0339	52,8015	69,8928	37,054
V23	60,8112	63,0079	56,8595	74,0945	42,0476
V24	64,3273	60,7454	54,7083	67,6018	21,6333
V25	68,942	67,4166	61,0574	76,0592	32,45
V26	46,6154	45,2217	38,833	54,4518	17,1172
V27	33,9559	31,3847	25,0599	40,3113	13,8924
V28	40,5216	38,0789	31,7648	46,7547	11,6619
V29	62,2896	57,7235	51,9711	63,561	17,72
V30	39,8497	32,249	28,1603	34,9285	14,7648
V31	29,2746	23,3452	18	29,6816	16,7631
V32	39,1152	29,9666	27,8029	29,0172	25,0599
V33	51,1566	45,1774	40	49,6488	7,81025
V34	63,3877	57,0088	52,0865	60,4152	19,6977
V35	63,3246	55,7315	51,6624	57,0789	24,0832
V36	12	7,2111	13,6015	8,94427	46,8402

	V46	V47	V48	V49	V50
V37	33,2866	37,9473	32,7567	50	38,1838
V38	43,1045	51,0882	47,676	63,0079	59,397
V39	68,2642	68,1175	61,7171	77,7946	37,3363
V40	44,4185	44,5533	38,1838	54,8179	23,0868
V41	57,28	59,6406	53,535	70,8308	40,3113
V42	42,19	46,8188	41,4849	58,8218	41,8808
V43	49,6588	55,0818	50,01	67,1863	50,1199
V44	35,171	42,107	38,0526	54,231	48,8365
V45	10,7703	18,1108	15,6525	30	41,9762
V46	0	10	11,7047	20,3961	45,0111
V47	10	0	6,40312	12,1655	40,025
V48	11,7047	6,40312	0	17,4642	34,3657
V49	20,3961	12,1655	17,4642	0	46,0652
V50	45,0111	40,025	34,3657	46,0652	0