



IMPLEMENTASI *HYBRID GENETIC ALGORITHM-SIMULATED ANNEALING* (GA-SA) DAN *VIRUS EVOLUTIONARY GENETIC ALGORITHM* (VEGA) PADA PERMASALAHAN DISTRIBUSI BARANG DUA TAHAP

SKRIPSI

Oleh:

**Dewinta Bunga Pasucindrawati
NIM 131810101011**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017**



IMPLEMENTASI *HYBRID GENETIC ALGORITHM-SIMULATED ANNEALING (GA-SA)* DAN *VIRUS EVOLUTIONARY GENETIC ALGORITHM (VEGA)* PADA PERMASALAHAN DISTRIBUSI BARANG DUA TAHAP

SKRIPSI

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

Oleh:

**Dewinta Bunga Pasucindrawati
NIM 13181010111**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2017**

PERSEMBAHAN

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan karunia-Nya kepada saya sehingga dapat menyelesaikan skripsi ini dengan lancar serta sholawat salam kepada Nabi besar Muhammad S.A.W. Skripsi ini saya persembahkan kepada:

1. Ayah Prayugo Sutomo, Umi Siti Aminah, dan Abah Harun serta keluarga besar terimakasih banyak atas kasih sayang, dorongan moral dan material , serta doa tulus ikhlas yang telah diberikan kepada saya.
2. Adik saya Annisa Zujjina Razifa yang telah memberikan motivasi kepada saya.
3. Guru-guru SD, SMP, SMA dan dosen-dosen Universitas Jember yang telah membimbing dan memberikan ilmu dengan penuh kesabaran.
4. Sahabat dan teman-teman Jurusan Matematika FMIPA Universitas Jember angkatan 2013 yang telah memotivasi dalam menyelesaikan skripsi ini.
5. Almamater Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

MOTTO

Apabila Anda berbuat kebaikan kepada orang lain, maka Anda telah berbuat baik terhadap diri sendiri.*



* Benyamen Franklin

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Dewinta Bunga Pasucindrawati

NIM : 131810101011

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul “Implementasi *Hybrid Genetic Algorithm-Simulated Annealing* dan *Virus Evolutonary Genetic Algorithm* pada Permasalahan Distribusi Barang Dua Tahap” 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, Juni 2017

Yang menyatakan,

Dewinta Bunga Pasucindrawati

NIM 131810101011

SKRIPSI

IMPLEMENTASI *HYBRID GENETIC ALGORITHM-SIMULATED ANNEALING* DAN *VIRUS EVOLUTIONARY GENETIC ALGORITHM* PADA PERMASALAHAN DISTRIBUSI BARANG DUA TAHAP

Oleh

Dewinta Bunga Pasucindrawati
NIM 1318101011

Pembimbing:

Dosen Pembimbing Utama : Kusbudiono, S.Si., M.Si.

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

PENGESAHAN

Skripsi berjudul “Implementasi *Hybrid Genetic Algorithm-Simulated Annealing* dan *Virus Evolutonary Genetic Algorithm* pada Permasalahan Distribusi Barang Dua Tahap” telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Tim Penguji:

Ketua,

Anggota I,

Kusbudiono, S.Si, M.Si.

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

NIP 197704302005011001

NIP 197211291998021001

Anggota II,

Anggota III,

Prof. Drs. I Made Tirta, M.Sc., Ph.D.

Drs. Rusli Hidayat, M.Sc.

NIP 195912201985031002

NIP 196610121993031001

Mengesahkan

Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Jember

Drs. Sujito, Ph.D.

NIP. 196102041987111001

RINGKASAN

Implementasi *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) pada Permasalahan Distribusi Barang Dua Tahap ; Dewinta Bunga Pasucindrawati, 131810101011; 2017; 68 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Distribusi merupakan proses pemindahan barang dari pabrik ke konsumen. Permasalahan distribusi yang utama adalah pada saat perusahaan memiliki wilayah pemasaran yang luas sehingga jalur pendistribusian harus diperhatikan. Tujuan dilakukan distribusi barang dua tahap adalah untuk menentukan jalur pendistribusian barang sehingga dapat meminimumkan biaya total distribusi.

Penelitian ini menggunakan data simulasi berupa kapasitas produsen, kapasitas distributor, kebutuhan agen, jarak produsen ke distributor, jarak distributor ke agen serta ditentukan biaya bahan bakar dan biaya pengangkutan per barang. Penyelesaian distribusi barang dua tahap pada penelitian ini menggunakan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA). Selain untuk mengetahui konsep dan hasil dari kedua algoritma tersebut, juga bertujuan untuk mengetahui hasil perbandingan kedua algoritma dalam menyelesaikan permasalahan distribusi barang dua tahap berdasarkan biaya total distribusi dan *running time*.

Berdasarkan hasil penelitian yang telah dilakukan, *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) didapatkan hasil total biaya distribusi paling minimal Rp 2.490.500,00 dan *Virus Evolutionary Genetic Algorithm* (VEGA) sebesar Rp 2.474.500,00. Ditinjau dari hasil tersebut dapat disimpulkan *Virus Evolutionary Genetic Algorithm* (VEGA) lebih baik daripada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA). Jika dilihat dari segi *running time* yang dibutuhkan oleh kedua algoritma, *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) memiliki *running time* lebih cepat daripada *Virus Evolutionary Genetic Algorithm* (VEGA).

PRAKATA

Puji syukur kehadiran Allah SWT atas segala kuasa dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi *Hybrid Genetic Algorithm-Simulated Annealing* dan *Virus Evolutonary Genetic Algorithm* pada Permasalahan Distribusi Barang Dua Tahap”. Penulisan skripsi ini dilakukan guna memenuhi salah satu syarat untuk menyelesaikan Program Studi Matematika (S1) dan mencapai gelar Sarjana Sains pada Jurusan Matematika FMIPA Universitas Jember.

Pada kesempatan ini penulis mengucapkan terima kasih atas bantuan dan bimbingan dalam penyusunan skripsi ini, kepada:

1. Bapak Kusbudiono, S.Si, M.Si. selaku dosen pembimbing utama dan Bapak Ahmad Kamsyakawuni, S.Si, M.Kom. selaku dosen pembimbing anggota.
2. Bapak Prof. Drs. I Made Tirta, M.Sc, Ph.D. selaku dosen penguji I dan Bapak Drs. Rusli Hidayat, M.Sc. selaku dosen penguji II.
3. Seluruh dosen dan karyawan Jurusan Matematika FMIPA Universitas Jember.
4. Keluarga besar Bapak Prayugo Sutomo yang telah memberikan semangat dan doa tulus ikhlas serta kasih sayangnya kepada saya.
5. Sahabat-sahabat “Maman” (Riska, Iris, dan Yulia) yang telah memberikan bentuk persahabatan yang indah.
6. Teman-teman ATLAS semuanya yang memberikan semangat dan motivasi dalam menyelesaikan skripsi ini
7. semua pihak yang telah membantu terselesaikannya skripsi ini.

Penulis menyadari bahwa penulisan ini masih terdapat kekurangan. Oleh karena itu, penulis mengharap kritik dan saran demi kesempurnaan penelitian selanjutnya. Semoga tugas akhir ini dapat bermanfaat bagi kita semua.

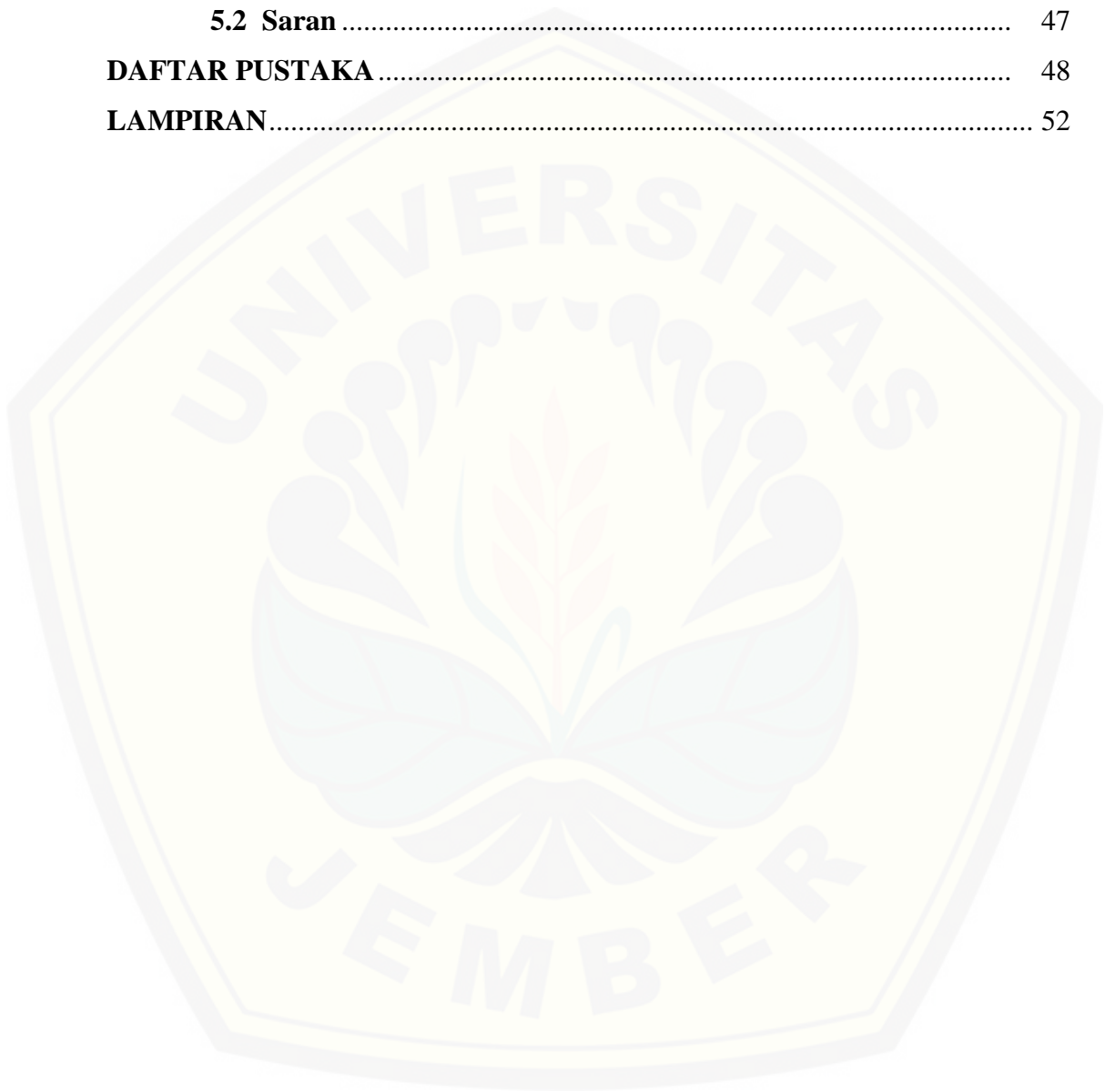
Jember, Juni 2017

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
DAFTAR ISI	vi
DAFTAR GAMBAR	x
DAFTAR TABEL	x
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
BAB 2. TINJAUAN PUSTAKA	4
2.1 Distribusi Barang Dua Tahap	4
2.2 Algoritma Genetika	6
2.2.1 Pengertian Individu.....	7
2.2.2 Komponen-komponen Utama Algoritma Genetika.....	8
2.3 <i>Simulated Annealing</i>	10
2.4 <i>Hybrid Genetic Algorithm-Simulated Annealing</i>	12
2.5 <i>Virus Evolutionary Genetic Algorithm</i>	15
BAB 3. METODE PENELITIAN	17
3.1 Data Penelitian	17
3.2 Langkah-langkah Penelitian	19
BAB 4. HASIL DAN PEMBAHASAN	17
4.1 Penyelesaian Distribusi Barang Dua Tahap Secara Manual.. 17	
4.1.1 Penyelesaian Menggunakan <i>Hybrid Genetic Algorithm-Simulated Annealing</i>	22
4.1.2 Penyelesaian Menggunakan <i>Virus Evolutionary Genetic Algorithm</i>	31
4.1.3 Hasil Program	33

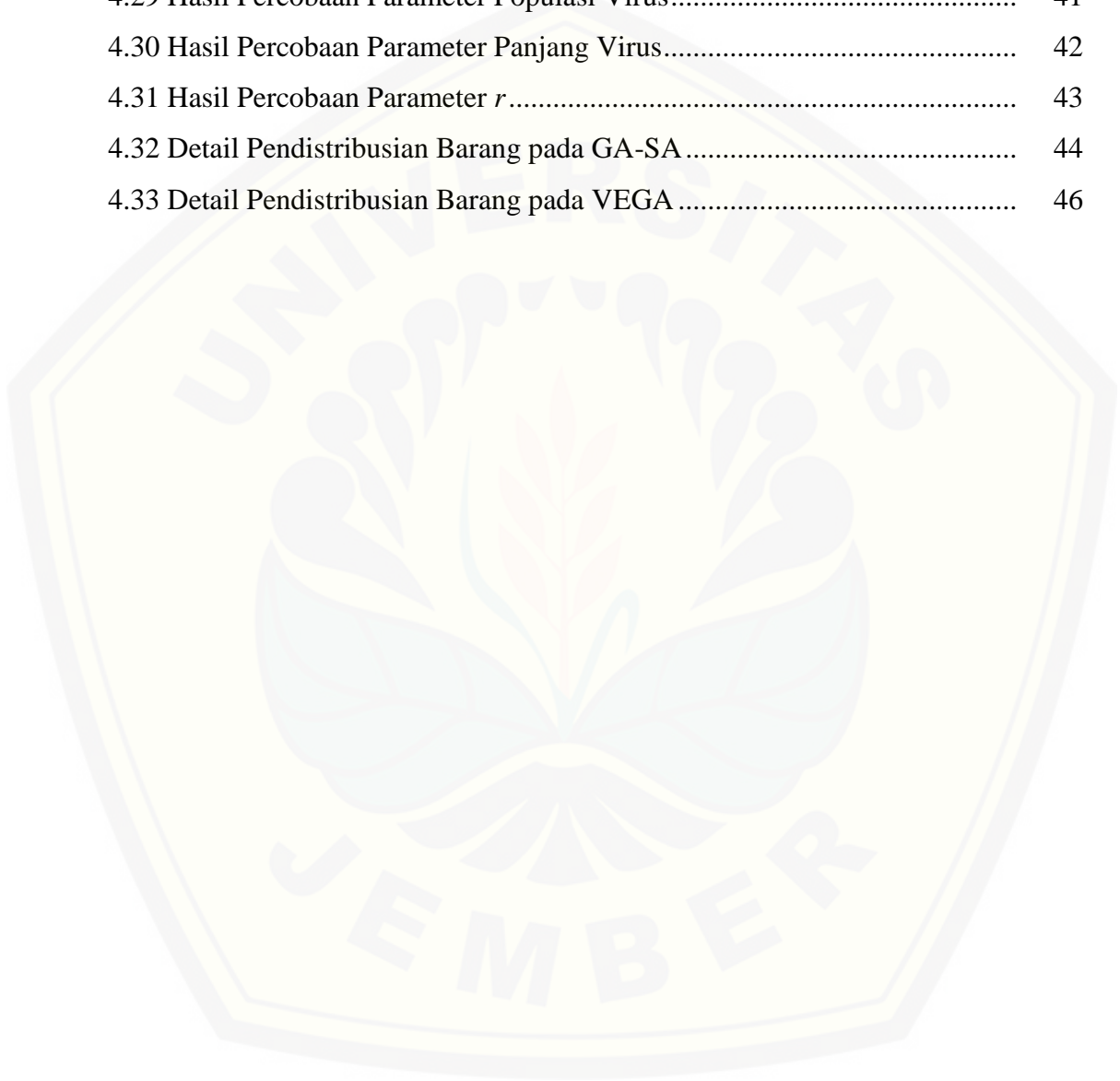
4.1.4 Hasil Percobaan	38
4.2 Pembahasan.....	43
BAB 5. KESIMPULAN DAN SARAN	17
5.1 Kesimpulan.....	47
5.2 Saran	47
DAFTAR PUSTAKA	48
LAMPIRAN.....	52



DAFTAR TABEL

	Halaman
3.1 Tabel Kapasitas Produsen dan Kebutuhan Distributor dan Agen	18
3.2 Tabel Jarak Produsen ke Distributor	18
3.3 Tabel Jarak Jarak Distributor ke Agen	18
4.1 Tabel Kapasitas Produsen dan Kebutuhan Distributor dan Agen (Perhitungan Manual)	21
4.2 Tabel Jarak Produsen ke Distributor (Perhitungan Manual)	21
4.3 Tabel Jarak Jarak Distributor ke Agen (Perhitungan Manual)	21
4.4 Posisi Gen pada Kromosom Segmen 1	22
4.5 Posisi Gen pada Kromosom Segmen 2	22
4.6 Bentuk Kromosom	23
4.7 Biaya Distribusi Tahap 2 Kromosom x_1	23
4.8 Biaya Distribusi Tahap 1 Kromosom x_1	24
4.9 Kromosom Baru	25
4.10 Proses <i>Crossover</i>	25
4.11 Hasil <i>Crossover</i>	26
4.12 Biaya Distribusi Tahap 2 Kromosom Anak 1	26
4.13 Biaya Distribusi Tahap 1 Kromosom Anak 1	26
4.14 Proses dan Hasil Mutasi	27
4.15 Biaya Distribusi Tahap 2 Kromosom HM_1	28
4.16 Biaya Distribusi Tahap 1 Kromosom HM_1	28
4.17 Kromosom Setelah Evaluasi	29
4.18 Hasil Operator SA	30
4.19 Populasi <i>Host</i>	31
4.20 Populasi Virus	31
4.21 Biaya Distribusi Tahap 2 Kromosom x_3'	32
4.22 Biaya Distribusi Tahap 1 Kromosom x_3'	32
4.23 Hasil Percobaan Parameter Populasi	38
4.24 Hasil Percobaan Parameter P_c	39

4.25 Hasil Percobaan Parameter Pm	39
4.26 Hasil Percobaan Parameter Iterasi	39
4.27 Hasil Percobaan Parameter Suhu Awal.....	40
4.28 Hasil Percobaan Parameter a	41
4.29 Hasil Percobaan Parameter Populasi Virus.....	41
4.30 Hasil Percobaan Parameter Panjang Virus.....	42
4.31 Hasil Percobaan Parameter r	43
4.32 Detail Pendistribusian Barang pada GA-SA.....	44
4.33 Detail Pendistribusian Barang pada VEGA.....	46



DAFTAR GAMBAR

	Halaman
2.1 Model Distribusi Barang Dua Tahap	4
2.2 Ilustrasi Representasi Penyelesaian Permasalahan dalam Algoritma Genetika	8
2.3 Pengkodean Permutasi	9
2.4 Ilustrasi <i>Crossover Two Cut Point</i>	9
2.5 Ilustrasi Proses <i>Exchange Mutation</i>	10
2.6 <i>Flowchart GA-SA</i>	14
2.7 <i>Flowchart VEGA</i>	16
3.1 Skema Langkah-langkah Penelitian	19
4.1 Tampilan Awal Program	34
4.2 Tampilan Hasil Program	36
4.3 Tampilan <i>Input Data</i>	37
4.4 Tampilan <i>Input Nilai Parameter</i>	37
4.5 Grafik Nilai <i>Fitness GA-SA</i>	44
4.6 Grafik Nilai <i>Fitness VEGA</i>	45

DAFTAR LAMPIRAN

	Halaman
A. Hasil Percobaan Parameter Populasi.....	52
B. Hasil Percobaan Parameter <i>Crossover</i>	54
C. Hasil Percobaan Parameter Mutasi.....	55
D. Hasil Percobaan Parameter Iterasi	57
E. Hasil Percobaan Parameter Suhu Awal	59
F. Hasil Percobaan Parameter <i>a</i>	61
G. Hasil Percobaan Parameter Populasi Virus.....	19
H. Hasil Percobaan Parameter Panjang Virus.....	34
I. Hasil Percobaan Parameter <i>r</i>	36

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Distribusi adalah proses pemindahan barang dari sumber (pabrik) ke konsumen. Manajemen distribusi merupakan hal penting yang harus diperhatikan oleh setiap perusahaan. Manajemen distribusi dilakukan agar pendistribusian barang berjalan dengan baik.

Permasalahan distribusi yang utama yaitu pada saat perusahaan memiliki wilayah pemasaran yang luas sehingga penentuan jalur pendistribusian barang harus diperhatikan. Karena hal ini akan berdampak pada biaya transportasi, semakin jauh jarak dari tempat produsen menuju tempat tujuan maka biaya distribusi juga akan semakin meningkat. Untuk itu, dilakukan distribusi barang dua tahap. Tujuan dilakukan distribusi barang dua tahap adalah untuk menentukan rute pengiriman barang secara tepat sehingga dapat meminimumkan biaya total distribusi.

Permasalahan distribusi barang dua tahap dapat diselesaikan dengan menggunakan beberapa algoritma, contohnya dengan menggunakan Algoritma Genetika oleh Sulistyorini & Mahmudy (2015) dan Algoritma *Evolution Strategies* oleh Vista & Mahmudy (2015). Pada penelitian yang dilakukan oleh Sulistyorini & Mahmudy (2015) dapat disimpulkan bahwa Algoritma Genetika mampu menyelesaikan permasalahan optimasi distribusi barang dua tahap. Dan pada penelitian Vista & Mahmudy (2015) dapat disimpulkan bahwa Algoritma *Evolution Strategies* mampu meminimalkan total biaya distribusi barang dua tahap.

Selain kedua algoritma diatas, terdapat algoritma optimasi lain, salah satunya *Genetic Algorithm-Simulated Annealing (GA-SA)*. Algoritma GA-SA merupakan penggabungan dari Algoritma Genetika dan *Simulated Annealing*. Algoritma GA-SA telah diterapkan dalam permasalahan Penjadwalan *Multipurpose Batch Chemical Plant* oleh Sofianti (2004). Pada penelitiannya

dapat disimpulkan bahwa solusi yang dihasilkan Algoritma GA-SA lebih baik dibandingkan solusi yang dihasilkan menggunakan Algoritma Genetika atau *Simulated Annealing* saja. Algoritma Genetika memiliki kelemahan yaitu sifat konvergennya yang prematur, suatu kondisi pada saat populasi pada Algoritma Genetika mencapai suatu keadaan dimana sebagian besar operator-operator genetik tidak lagi menghasilkan *offspring* yang lebih baik dari *parent*-nya dan adanya kemungkinan terjebak dalam solusi lokal optimum. Tujuan dari penggabungan Algoritma Genetika dan *Simulated Annealing* adalah untuk menutupi kelemahan Algoritma Genetika tersebut. *Simulated Annealing* merupakan algoritma yang memiliki kelebihan mampu bertahan menghadapi lokal optimum (Sofianti, 2004).

Selain algoritma GA-SA, terdapat algoritma optimasi lainnya yaitu *Virus Evolutionary Genetic Algorithm* (VEGA). VEGA merupakan algoritma gabungan dari Algoritma Genetika dan infeksi virus (Fukuda, 1999). Penggabungan VEGA dan metode *zero crossing* telah diterapkan pada permasalahan Persamaan Non-Linier oleh Anwar (2016). VEGA juga telah diaplikasikan pada tiga jenis permasalahan optimasi yaitu *Traveling Salesman Problem* (TSP), *Knapsack*, dan optimasi fungsi oleh Kubota et al. (1996) yang memberikan hasil VEGA lebih baik daripada *Steady State Genetic Algorithm* (SSGA). Selain itu, VEGA juga telah diaplikasikan pada optimasi proses mesin pemahat oleh Fountas (2013) yang menyatakan bahwa VEGA mampu memberikan hasil yang global optimum, sedangkan algoritma genetika terjebak pada pencarian lokal optimum.

Berdasarkan uraian diatas penulis akan menerapkan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) pada permasalahan distribusi barang dua tahap kemudian membandingkan hasil penerapan kedua algoritma tersebut berdasarkan biaya total distribusi. Selain itu, kedua algoritma tersebut juga dibandingkan berdasarkan *running time*.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dirumuskan permasalahan sebagai berikut:

- a. Bagaimana menerapkan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) pada permasalahan distribusi barang dua tahap?
- b. Bagaimana perbandingan hasil penerapan GA-SA dan VEGA pada permasalahan distribusi barang dua tahap berdasarkan biaya total distribusi dan *running time*?

1.3 Batasan Masalah

Batasan masalah digunakan agar penelitian ini dapat terfokus, diantaranya adalah:

- a. Produk yang didistribusikan hanya satu macam.
- b. Jumlah kendaraan dan kapasitas kendaraan tidak terbatas.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

- a. Untuk menerapkan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) dalam menyelesaikan masalah distribusi barang dua tahap.
- b. Untuk mengetahui perbandingan terhadap solusi yang dihasilkan menggunakan GA-SA dan VEGA berdasarkan biaya total distribusi dan *running time*.

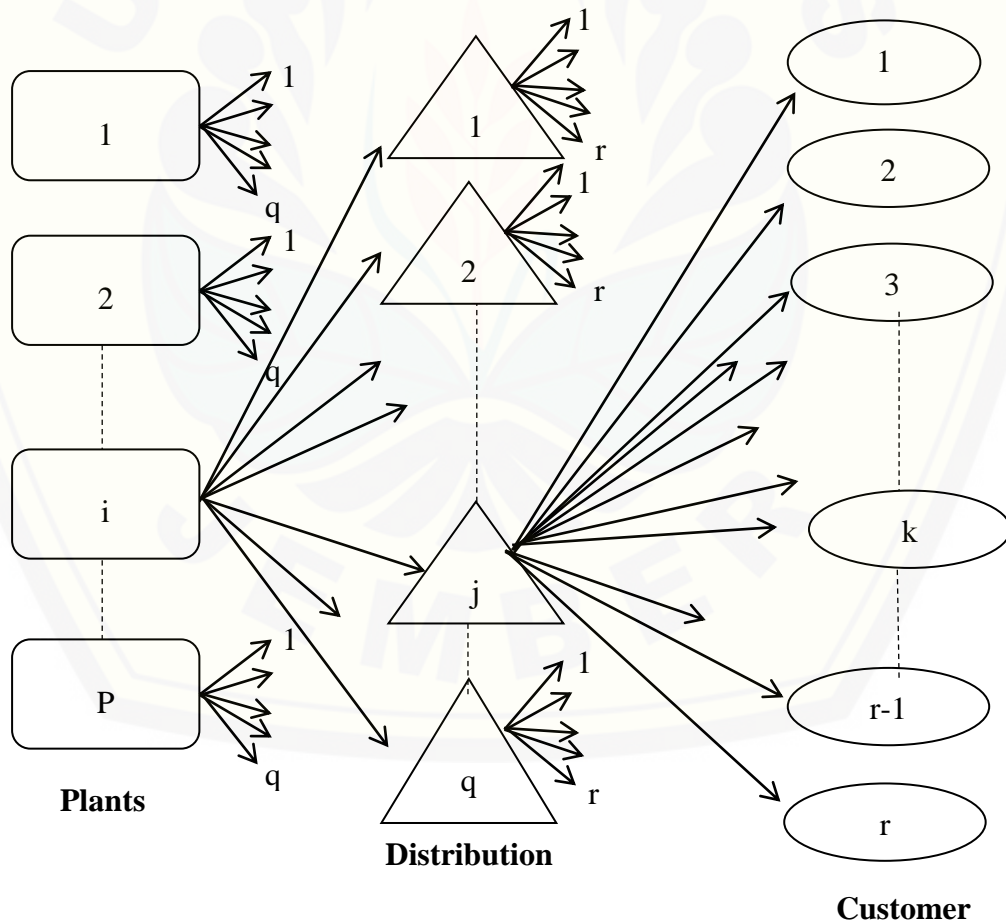
1.5 Manfaat Penelitian

Penelitian ini menerapkan dua algoritma yaitu *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) yang dapat membantu perusahaan untuk menentukan jalur optimal distribusi barang dua tahap.

BAB 2. TINJAUAN PUSTAKA

2.1 Distribusi Barang Dua Tahap

Model distribusi barang dua tahap yaitu adanya sumber dan beberapa tujuan, jumlah produk yang diangkut dari sumber harus sesuai dengan permintaan tujuan dan jumlah barang yang diminta harus sesuai dengan jumlah yang diproduksi oleh sumber (Sulistiyorini & Mahmudy, 2015). Tujuan dari distribusi barang dua tahap yaitu mengoptimalkan biaya transportasi dalam proses distribusi (Vista & Mahmudy, 2015). Untuk pemodelan distribusi barang dua tahap dapat dilihat pada Gambar 2.1.



Gambar 2.1: Model Distribusi Barang Dua Tahap

(Sumber: Jawahar dan Balaji, 2009)

Permasalahan transportasi pada distribusi barang dua tahap dengan pencarian biaya optimum dapat dinyatakan secara sistematis.

Fungsi pencarian biaya optimal dalam permasalahan distribusi barang dua tahap dapat dinyatakan pada Persamaan 2.1

$$Z = \sum_{i=1}^I \sum_{j=1}^J t_{ij} x_{ij} + \sum_{j=1}^J \sum_{k=1}^K c_{jk} y_{jk} \quad (2.1)$$

Distribusi barang tidak boleh melebihi maksimal kapasitas produksi yang dinyatakan dalam Persamaan 2.2 dan 2.3

$$\sum_{j=1}^J x_{ij} \leq a_i, \quad \forall i \quad (2.2)$$

$$\sum_{k=1}^K y_{jk} \leq b_j z_j, \quad \forall j \quad (2.3)$$

Jumlah distributor yang dimiliki perusahaan tidak melebihi jumlah maksimum distributor yang ditentukan, dinyatakan dalam Persamaan 2.4

$$\sum_{j=1}^J z_j \leq w \quad (2.4)$$

Semua permintaan konsumen harus dapat dipenuhi yang dapat dinyatakan dalam Persamaan 2.5

$$\sum_{j=1}^J y_{jk} \leq d_k, \quad \forall k \quad (2.5)$$

Penawaran total diasumsikan memenuhi kondisi seimbang yaitu penawaran total sama dengan permintaan total yang dapat dinyatakan dalam Persamaan 2.6 dan 2.7.

$$\sum_{i=1}^I \sum_{j=1}^J x_{ij} = \sum_{j=1}^J \sum_{k=1}^K y_{jk} \quad (2.6)$$

$$x_{ij}, y_{jk} \geq 0, \quad \forall i, j, k \quad (2.7)$$

Untuk penyelesaiannya variabel-variabel yang digunakan yaitu (Gen *et al.*, 2006):

- i : Jumlah sumber ($i = 1, 2, \dots, I$).
- j : Jumlah distributor ($j = 1, 2, \dots, J$).
- k : Jumlah konsumen ($k = 1, 2, \dots, K$).
- w : Jumlah maksimal distributor yang ditentukan.

- e. t_{ij} : Biaya perjalanan dari sumber menuju distributor.
- f. x_{ij} : Jumlah barang yang dikirimkan dari sumber menuju distributor.
- g. c_{jk} : Biaya perjalanan dari distributor menuju konsumen.
- h. y_{jk} : Jumlah barang yang dikirimkan dari distributor menuju konsumen.
- i. d_k : Jumlah permintaan dari konsumen k .

2.2 Algoritma Genetika

Pencipta algoritma genetika adalah John Holland. Algoritma ini menggunakan analogi secara langsung dari kebiasaan yang alami yaitu seleksi alam (Nurjanah, 2015).

Algoritma genetika merupakan salah satu tipe dari algoritma evolusi yang paling dikenal dan banyak digunakan untuk menyelesaikan permasalahan kompleks. Pada saat ini representasi dalam algoritma genetika sudah mengalami perkembangan yang awalnya hanya dapat representasi string biner, sekarang dapat menggunakan vektor bilangan integer dan pecahan (Mahmudy, 2013).

Variabel pada algoritma genetika adalah (Indrianingsih, 2010) :

- a. *Fitness* pada masing-masing individu digunakan untuk menentukan tingkat kesesuaian individu tersebut dengan kriteria yang akan dicapai.
- b. Populasi jumlah individu.
- c. Probabilitas *crossover* yang akan terjadi pada setiap generasi.
- d. Probabilitas mutasi yang akan terjadi pada setiap individu.
- e. Jumlah generasi yang dibentuk menentukan lamanya proses algoritma genetika berlangsung.

Siklus dari algoritma genetika adalah (Mahmudy, 2013) :

- a. Inisialisasi yaitu proses membangkitkan individu secara acak yang memiliki *string chromosome* dan ditempatkan pada sebuah populasi. Pada saat tahap ini harus ditentukan ukuran dari populasi (*popSize*). Berdasarkan presisi variabel yang dicari maka bisa menghitung panjang setiap *string chromosome*.
- b. Reproduksi dilakukan untuk menghasilkan keturunan dari semua individu dalam populasi dan ditempatkan pada *offspring*. Pada tahap ini harus

ditentukan nilai tingkat *crossover* (*crossover rate/pc*) yang menyatakan rasio *offspring* yang dihasilkan dari proses *crossover* terhadap ukuran populasi. Selain itu, nilai tingkat mutasi juga harus ditentukan yang menyatakan rasio *offspring* yang dihasilkan dari proses mutasi.

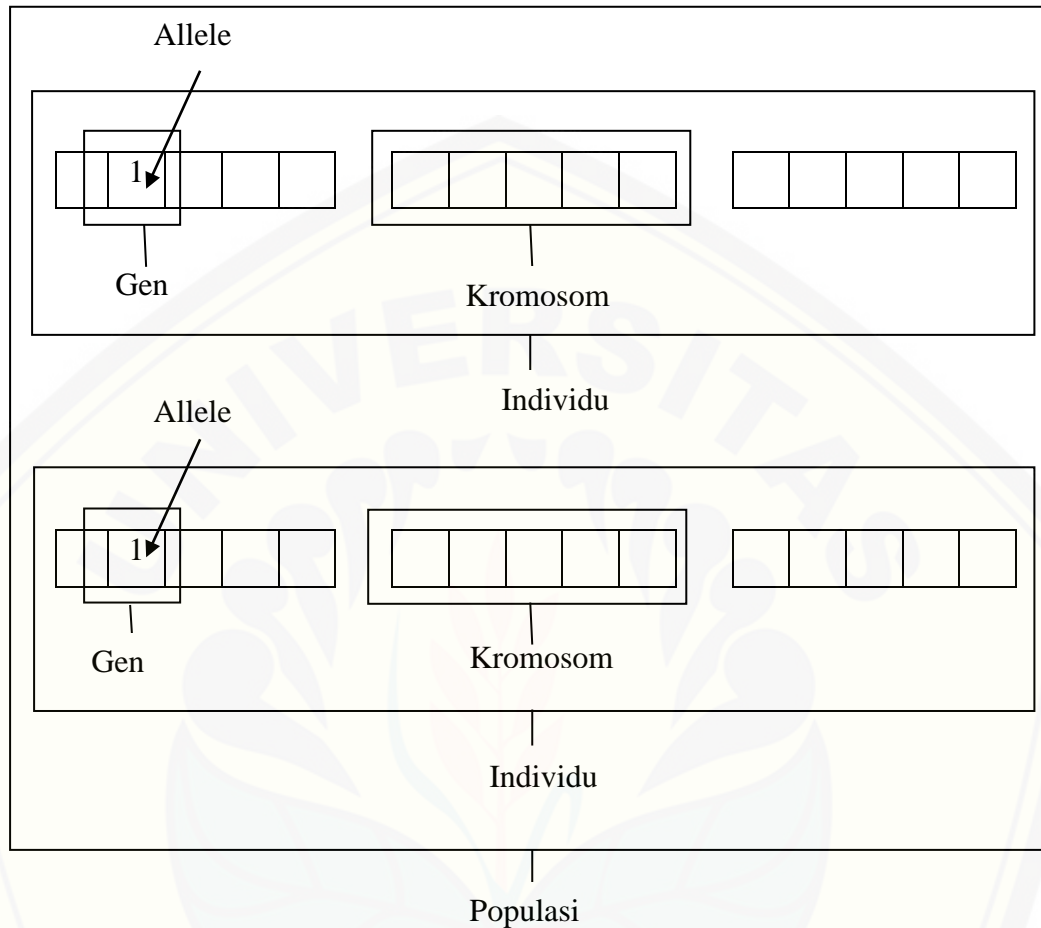
- c. Evaluasi dilakukan untuk menghitung *fitness* dari setiap kromosom. Semakin besar nilai *fitness* maka semakin baik kromosom tersebut dijadikan calon solusi.
- d. Seleksi dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan untuk melakukan proses genetik pada generasi berikutnya. Semakin besar nilai *fitness* dari suatu kromosom maka peluang untuk dipertahankan juga semakin besar.

2.2.1 Pengertian Individu

Individu bisa dikatakan sebagai kromosom yang merupakan kumpulan dari gen. Beberapa definisi penting dalam menyelesaikan Algoritma Genetika adalah sebagai berikut (Nurjanah, 2015):

- a. *Genotype* (gen), merupakan nilai yang menyatakan satuan dasar yang membentuk arti tertentu dalam satu kesatuan gen yang dinamakan kromosom.
- b. *Allele*, merupakan nilai dari gen.
- c. Kromosom, merupakan gabungan dari gen-gen yang membentuk suatu arti tertentu.
- d. Individu, merupakan suatu nilai sebagai salah satu solusi yang mungkin.
- e. Populasi, merupakan sekumpulan individu yang akan diproses dalam satu siklus proses evolusi.
- f. Generasi, merupakan satu siklus proses evolusi atau satu iterasi dalam Algoritma Genetika.

Istilah-istilah di atas akan diilustrasikan pada Gambar 2.2.



Gambar 2.2 Ilustrasi Representasi Penyelesaian Permasalahan dalam Algoritma Genetika

(Sumber: Nurjanah, 2015)

2.2.2 Komponen-komponen Utama Algoritma Genetika

- Pengkodean merupakan teknik untuk merepresentasikan populasi awal sebagai calon solusi pada suatu permasalahan dalam bentuk kromosom (Indrianingsih, 2010). Pada permasalahan distribusi barang dua tahap menggunakan pengkodean permutasi. Pengkodean permutasi akan diilustrasikan pada gambar 2.3.

Node	1	2	3	4
Kromosom	2	3	1	4

Gambar 2.3 Pengkodean Permutasi

(Sumber: Vista & Mahmudy, 2015)

- b. Membangkitkan populasi awal adalah membangkitkan sejumlah individu secara acak atau dengan prosedur tertentu (Nurjanah, 2015).
- c. Menghitung nilai *fitness* digunakan untuk mengukur kebaikan solusi dalam individu (Vista & Mahmudy, 2015). Fungsi *fitness* untuk minimasi dapat dilakukan menggunakan Persamaan 2.8.

$$fitness = \frac{c}{f(x)} \quad (2.8)$$

- d. Pindah silang (*crossover*) adalah memilih dua buah kromosom untuk dijadikan *parent* yang dipilih secara acak. Setelah itu dilakukan penukaran segmen kromosom induk untuk menghasilkan *offspring* (Sulistiyorini & Mahmudy, 2015). Probabilitas *crossover* memiliki nilai $0 \leq Pc \leq 1$. Nilai probabilitas yang baik berkisar antara 0,6 sampai 1 (Gen & Cheng, 1997). Pada permasalahan ini, dilakukan metode *crossover two cut point*. Proses *crossover* dengan metode *crossover two cut point* diilustrasikan pada Gambar 2.4.

1.

J2	J4	J3	J1	J5
----	----	----	----	----

 Induk 1

J3	J1	J5	J4	J2
----	----	----	----	----

 Induk 2

2. *mapping*

J3→J5

J1→J4

3.

J2	J1	J5	J4	J3
----	----	----	----	----

 Anak 1

J5	J4	J3	J1	J2
----	----	----	----	----

 Anak 2

Gambar 2.4 Ilustrasi *Crossover Two Cut Point*

(Sumber: Nurjanah, 2015)

- e. Mutasi digunakan untuk menghasilkan perubahan acak pada suatu kromosom. Probabilitas mutasi memiliki nilai $0 \leq P_m \leq 1$. Nilai probabilitas yang baik berkisar antara 0,001 sampai 0,2 (Gen & Cheng, 1997). Pada permasalahan distribusi barang dua tahap digunakan metode *exchange mutation*. Metode *exchange mutation* diilustrasikan pada Gambar 2.5.

<i>Parent</i>	2	3	1	4
<i>Offspring</i>	1	3	2	4

Gambar 2.5 Ilustrasi Proses *Exchange Mutation*

(Sumber: Vista & Mahmudy, 2015)

2.3 *Simulated Annealing*

Simulated Annealing adalah teknik pencarian secara acak yang menggunakan analogi bagaimana pendinginan besi dan membekukannya dalam bentuk kristal dan mencari nilai minimum global serta membentuk basis untuk teknik optimasi untuk permasalahan kombinatorial dan lainnya. *Simulated annealing* digunakan untuk mengatasi masalah optimum lokal agar mendapatkan solusi global optimum (Juniarto *et al.*, 2013).

Pada Algoritma *Simulated Annealing* (SA) akan dilakukan pemanasan untuk pembuatan kristal suatu material. Pemanasan tersebut dilakukan sampai mencapai titik tertentu. Saat material tersebut berada pada keadaan yang panas, maka atom-atom akan bergerak bebas dengan energi yang tinggi. setelah material mencapai titik tertentu, maka dilakukan penurunan suhu secara perlahan dengan harapan atom-atom tersebut berada pada posisi optimum sehingga menghasilkan energi minimum (Bertsimas & Tsitsiklis, 1993).

SA terdiri dari beberapa suhu dimana setiap suhu memiliki beberapa iterasi. Pertama, suhu awal dipilih dari solusi awal secara acak. Nilai dari fungsi biaya

berdasarkan solusi awal akan dihitung untuk meminimalkan fungsi biaya. Setelah itu akan didapatkan solusi baru. Nilai fungsi biaya berdasarkan solusi baru kemudian akan dihitung dan dibandingkan dengan nilai fungsi biaya berdasarkan solusi awal. Jika nilai fungsi biaya baru kurang dari nilai fungsi biaya saat ini, maka akan diterima. Sedangkan apabila jika perbedaan antara nilai fungsi biaya saat ini dan nilai fungsi biaya pada solusi baru (ΔE) sama dengan atau lebih besar nol bilangan acak δ berada pada selang $[0,1]$ dihasilkan dari distribusi seragam. Jika

$$\exp\left(-\left(\frac{\Delta E}{T}\right)\right) \geq \delta \quad (2.9)$$

terpenuhi, maka solusi baru yang dihasilkan dapat diterima (Orkcu, 2013)..

Jumlah solusi baru yang dihasilkan sama dengan jumlah iterasi pada suhu yang dibatasi oleh kondisi terminasi. Setelah semua iterasi pada suhu lengkap, suhu akan diturunkan secara perlahan. Pada saat penurunan suhu, semua iterasi yang diperlukan harus diselesaikan terlebih dahulu sebelum pindah ke suhu berikutnya. Proses ini akan terus diulangi sampai kriteria penghentian terpenuhi. Hasil simulasi SA berkaitan dengan jumlah iterasi pada setiap suhu dan kecepatan mengurangi suhu. Aturan penurunan suhu pada Persamaan 2.10 :

$$T_{baru} = \alpha \times T_{awal} \quad (2.10)$$

dimana T_{awal} = temperatur awal

T_{baru} = temperatur baru

α = faktor reduksi suhu ($\alpha < 1$) (Emanuel & Aritonang, 2008).

Ada tiga komponen dalam algoritma SA (Tospornsampan *et al.*, 2007) :

a. Proses *Annealing*

Proses *annealing* dilakukan agar tidak terjebak dalam lokal minimum.

Parameter pada proses annealing adalah:

1) Suhu awal

Suhu awal ditentukan setinggi mungkin untuk memperluas penerimaan solusi baru.

2) Jumlah iterasi pada setiap suhu

Ditentukan terlebih dahulu jumlah iterasi pada setiap suhu. Jika sudah mencapai iterasi yang ditentukan, maka suhu akan diturunkan. Jumlah iterasi dapat konstan atau berubah-ubah pada setiap suhunya.

3) Pemilihan parameter untuk menurunkan suhu

Setelah iterasi yang ditentukan tercapai, maka akan dilakukan penurunan suhu dengan parameter α yang disebut *cooling rate* yang berkisar antara 0 dan 1. Nilai α yang disarankan oleh Kirkpatrick *et al* (1983) adalah antara 0,8 sampai 0,99.

b. Penyusunan Ulang

Penyusunan ulang dilakukan secara acak untuk mengubah solusi saat ini dengan solusi yang baru.

c. Penghentian Algoritma

Kriteria penghentian harus ditentukan sejak awal. Kriteria dapat berupa banyaknya iterasi dimana proses akan berhenti jika tidak didapatkan solusi baru yang dapat diterima. Selain itu kriteria juga dapat berupa suhu minimum dimana proses akan berhenti setelah mencapai suhu minimum tersebut.

2.4 Hybrid Genetic Algorithm-Simulated Annealing

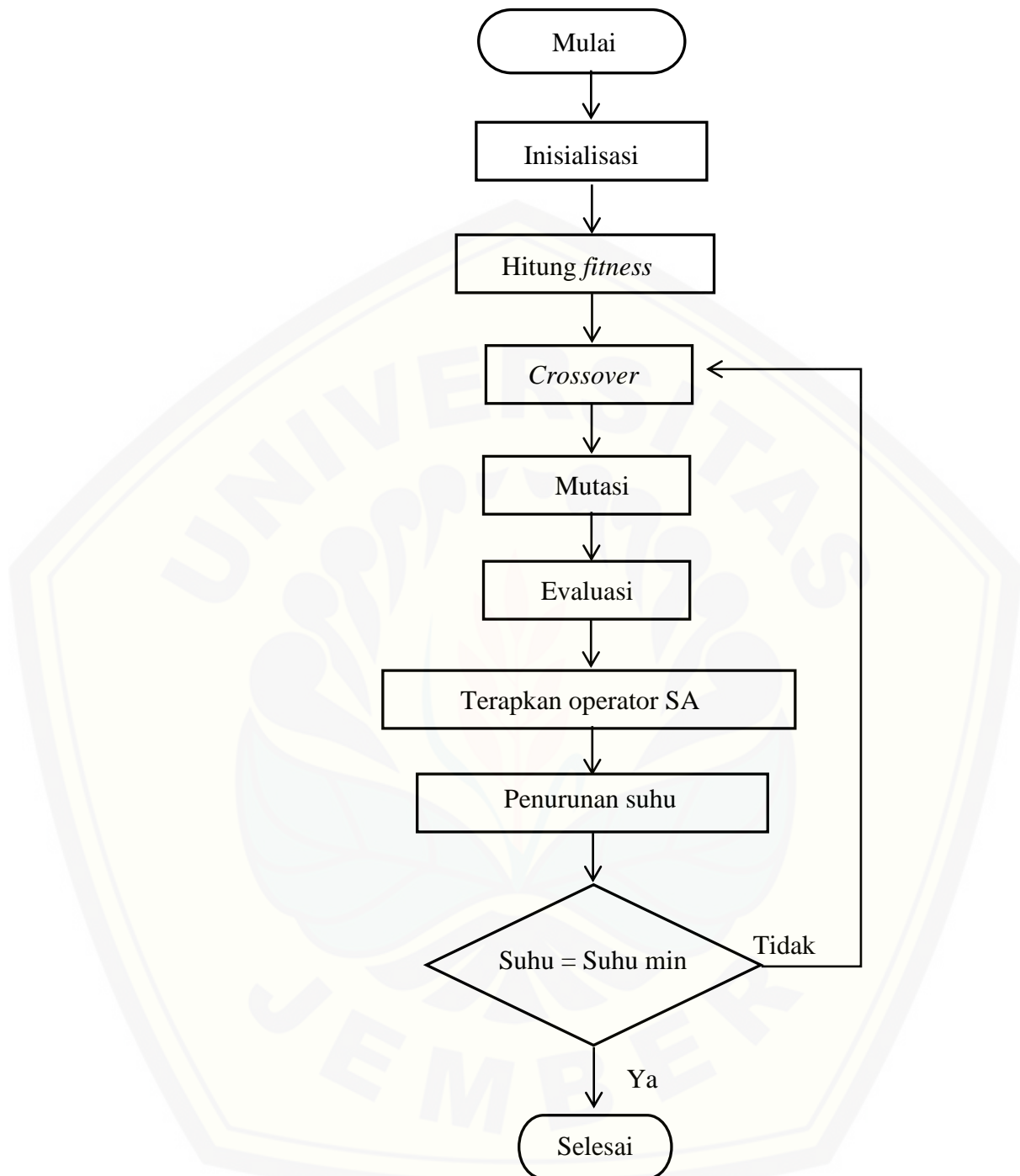
Hybrid Genetic Algorithm-Simulated Annealing merupakan algoritma gabungan dari Algoritma Genetika dan *Simulated Annealing*. Algoritma Genetika memiliki kelemahan yaitu sifat konvergennya yang prematur, suatu kondisi pada saat populasi pada Algoritma Genetika mencapai suatu keadaan dimana sebagian besar operator-operator genetik tidak lagi menghasilkan *offspring* yang lebih baik dari *parent*-nya dan adanya kemungkinan terjebak dalam solusi lokal optimum. Tujuan dari penggabungan Algoritma Genetika dan *Simulated Annealing* adalah untuk menutupi kelemahan Algoritma Genetika tersebut. *Simulated Annealing* merupakan algoritma yang memiliki kelebihan mampu bertahan menghadapi lokal optimum (Sofianti, 2004).

Siklus *Hybrid Genetic Algorithm-Simulated Annealing* adalah :

- a. Representasi kromosom secara acak dalam sebuah populasi.
- b. *Fitness* setiap kromosom dievaluasi

- c. Kromosom direproduksi memiliki nilai *fitness* yang proporsional
- d. *Crossover* dilakukan pada kromosom yang sudah dipilih untuk menghasilkan kromosom baru.
- e. Mutasi dilakukan pada kromosom baru.
- f. Kromosom baru dievaluasi.
- g. Operator SA digunakan untuk memilih dari *parent* dan anak yang akan tetap diterapkan.
- h. Suhu akan menurun seperti pada Persamaan 2.10.
- i. Jika kriteria yang sudah ditentukan tercapai, akan diberikan solusi terbaik. Jika tidak, lanjutkan ke langkah c (Orkcu, 2013).

Operator SA diterapkan sebagai berikut: Pertama, induk 1 dan anak 1 dibandingkan. Apabila anak 1 lebih baik, maka induk 1 diganti dengan anak 1. Jika *parent* 1 lebih baik, maka induk 1 oleh anak 1 juga akan diganti dengan probabilitas tertentu. Prosedur yang sama akan diterapkan pada induk 2 dan anak 2. Misalkan perbedaan antara nilai-nilai *fitness* yang sesuai dengan anak dan induk (ΔE) adalah sama dengan atau lebih besar nol (induk lebih baik) dan suhu saat ini adalah T . Bilangan acak δ berada pada selang $[0,1]$ dihasilkan dari distribusi seragam dan jika (2.9) terpenuhi maka anak tersebut akan diterima. Dengan demikian, anak dapat dipilih dengan beberapa probabilitas bahkan jika itu lebih rendah dari induk. Awalnya saat suhu tinggi, ada kemungkinan lebih tinggi menerima anak yang memiliki nilai *fitness* lebih rendah dari induk. Jika anak yang dihasilkan lebih baik, maka itu diterima (Orkcu, 2013). Langkah-langkah pada GA-SA dapat diilustrasikan pada *flowchart* di bawah ini:



Gambar 2.6 Flowchart GA-SA

2.5 Virus Evolutionary Genetic Algorithm (VEGA)

Virus Evolutionary Genetic Algorithm (VEGA) merupakan penggabungan algoritma genetika dan infeksi virus (Fukuda, 1999). Algoritma ini diajukan oleh Kubota *et al.* (1996) yang merupakan pengembangan dari teori evolusi Darwin yang menyatakan seleksi alam. VEGA terdiri dari dua populasi yaitu populasi *host* dan populasi virus. Populasi *host* pada algoritma VEGA identik dengan populasi pada algoritma genetika. Populasi virus adalah *substring* dari populasi *host* yang digunakan untuk menginfeksi populasi *host*. Populasi virus memiliki dua operator infeksi virus yaitu operator *reverse transcription* dan operator *transduction*.

Reverse transcription dalam biologi merupakan proses dari virus yang menggunakan enzim untuk mengganti RNA mereka menjadi DNA. Operator *reverse transcription* pada VEGA juga akan melakukan hal yang sama. Setiap generasi, beberapa virus memiliki kesempatan bahwa *substring*-nya akan menjadi kromosom populasi *host*. Operator *transduction* adalah proses pengambilan DNA dari individu *host* yang digunakan untuk membuat virus baru (Ling, 2006).

Misalkan virus ditunjukkan oleh i . Diasumsikan f_{ihost_j} dan $f'_{ihost'_j}$ masing-masing merupakan nilai *fitness* dari individu *host* sebelum dan sesudah diinfeksi oleh virus i . Selisih antara f_{ihost_j} dan $f'_{ihost'_j}$ dapat ditunjukkan sebagai $fitvirus_{i,j}$. Dan $fitvirus_i$ dapat ditunjukkan sebagai jumlah dari nilai *fitness* semua *host* yang terinfeksi oleh virus i (Ling, 2006). Uraian di atas dapat dituliskan pada Persamaan 2.12 dan Persamaan 2.13.

$$fitvirus_{i,j} = f'_{ihost'_j} - f_{ihost_j} \quad (2.11)$$

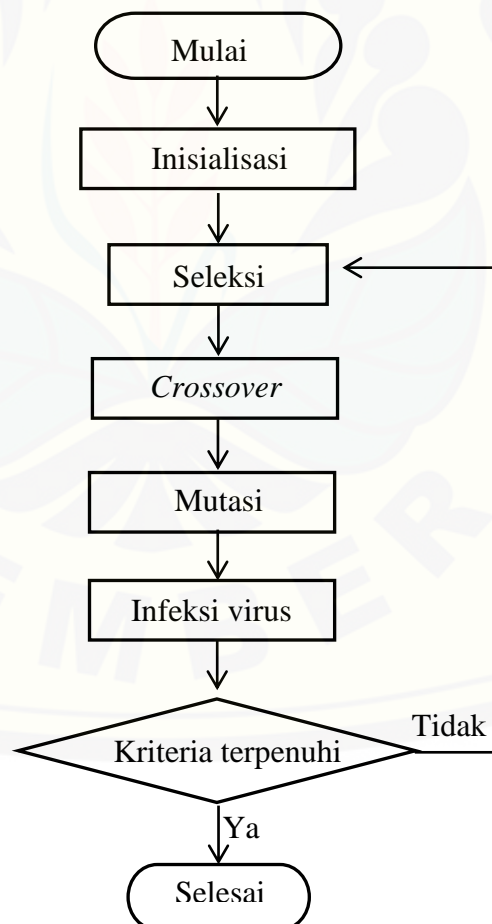
$$fitvirus_i = \sum_{j \in S} fitvirus_{i,j} \quad (2.12)$$

dengan i adalah virus dan S adalah himpunan individu *host* yang terinfeksi oleh virus i . Nilai *fitness* virus menunjukkan pengaruh keseluruhan dari virus terhadap setiap *host* yang diinfeksi. Jika bernilai positif, berarti virus tersebut memiliki efek positif pada individu *host*. Sementara itu, jika bernilai negatif, maka virus tidak dapat memperbaiki individu *host*. Nilai *fitness* virus juga mengontrol seberapa

mampu virus akan bertahan yang dapat disebut sebagai kekuatan hidup virus (Ling, 2006). Kekuatan hidup virus dapat dihitung dengan Persamaan 2.14.

$$life_i = r \times life_{i,t-1} + fitvirus_i \quad (2.13)$$

dengan t dan r masing-masing merupakan generasi virus dan tingkat penurunan daya hidup virus. Jika $life_i$ bernilai negatif, berarti virus tidak memberikan pengaruh positif terhadap individu *host*. Maka dari itu, operator *transduction* akan memperbarui kromosom virus dengan mengambil *substring* individu *host* secara acak. Jika bernilai $life_i$ positif, individu virus memperoleh sebagian *substring* dari individu *host* yang diinfeksi (Anwar, 2016). Langkah-langkah pada *Virus Evolutionary Genetic Algorithm* (VEGA) dapat diilustrasikan pada *flowchart* di bawah ini:



Gambar 2.8 Flowchart Virus Evolutionary Genetic Algorithm (VEGA)

BAB 3. METODE PENELITIAN

3.1 Data Penelitian

Pada penelitian ini, data yang akan digunakan adalah data simulasi berupa data acak yang dibuat oleh peneliti sendiri.

Data simulasi distribusi barang dua tahap:

Tabel 3.1 Tabel Kapasitas Produsen dan Kebutuhan Distributor dan Agen

Node	Produsen	Distributor	Agen
1	750	200	60
2	750	120	78
3		150	60
4		150	58
5		100	42
6		165	84
7		200	66
8		100	80
9		165	54
10		150	48
11			54
12			50
13			46
14			52
15			54
16			78
17			60
18			60
19			66
20			50
Total	1500	1500	1300

Tabel 3.2 Tabel Jarak Produsen ke Distributor

Produsen	Distributor									
	1	2	3	4	5	6	7	8	9	10
1	13	20	7	16	4	8	11	23	2	19
2	8	7	12	16	4	19	21	11	15	9

Tabel 3.3 Tabel Jarak Jarak Distributor ke Agen

Distributor	Agen									
	1	2	3	4	5	6	7	8	9	10
1	13	6	17	10	16	21	7	20	9	23
2	6	22	2	7	20	15	13	18	6	19
3	24	17	14	7	24	15	6	22	13	8
4	13	17	18	16	23	23	2	20	24	20
5	14	25	13	14	10	15	5	6	10	4
6	6	2	16	11	5	1	7	25	15	13
7	3	11	6	6	20	20	6	3	16	16
8	2	13	16	22	4	16	16	17	18	18
9	21	15	10	22	5	13	21	6	11	10
10	20	3	3	8	10	7	20	7	22	18

Distributor	Agen									
	11	12	13	14	15	16	17	18	19	20
1	22	3	24	8	9	10	22	17	22	2
2	2	10	17	3	24	7	22	14	12	2
3	4	13	8	15	24	2	6	7	5	25
4	22	15	14	5	6	12	11	2	20	17
5	11	24	1	3	19	16	22	7	12	10
6	11	13	22	15	16	8	7	10	4	13
7	24	11	11	8	23	13	24	16	22	19
8	19	25	8	10	15	7	16	25	16	2
9	25	7	24	18	18	8	5	6	3	18
10	7	17	12	22	19	17	21	19	23	18

Diasumsikan biaya bahan bakar per kilometer sebesar Rp 500,-

Diasumsikan biaya pengangkutan per barang sebesar Rp 1000,-

3.2 Langkah-langkah Penelitian

Adapun langkah-langkah yang akan dilakukan pada penelitian ini untuk menyelesaikan distribusi dua tahap sebagai berikut:

a. Studi Literatur

Langkah pertama dalam penelitian ini adalah pengumpulan literatur tentang permasalahan distribusi barang dua tahap, algoritma genetika, *simulated annealing*, *hybrid genetic algorithm-simulated annealing* (GA-SA) dan *virus evolutionary genetic algorithm* (VEGA). Kemudian dilakukan studi terhadap literatur tersebut.

b. Pengambilan Data

Pada tahap ini, data yang diambil berupa data simulasi yang dibuat oleh peneliti sendiri.

c. Penerapan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm*

Data yang telah dibangkitkan pada langkah b, kemudian diolah dan diselesaikan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* dengan langkah-langkah sesuai pada subbab 2.4 dan 2.5.

d. Pembuatan Program

Pada tahap ini, peneliti akan penerapan *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* untuk menyelesaikan permasalahan distribusi barang dua tahap menggunakan *software* MATLAB. Program dibuat dalam bentuk *Graphical User Interface* (GUI).

e. Kesimpulan

Hasil dari simulasi program akan ditarik kesimpulan untuk menjawab rumusan masalah.

BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan dapat disimpulkan bahwa *Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) dapat diterapkan pada permasalahan distribusi barang dua tahap. Pada data yang digunakan, total biaya distribusi yang dihasilkan oleh VEGA yaitu sebesar Rp 2.474.500,00, sedangkan total biaya distribusi yang dihasilkan oleh GA-SA yaitu sebesar Rp 2.490.500,00, sehingga dapat disimpulkan bahwa pada data tersebut, VEGA lebih baik daripada GA-SA dalam penyelesaian distribusi barang dua tahap. Dari grafik kekonvergenannya, grafik yang dihasilkan oleh GA-SA naik turun artinya GA-SA lebih sulit menemukan solusi yang optimal dibandingkan dengan grafik yang dihasilkan oleh VEGA yang hanya naik dan lurus. Berdasarkan *running time*, VEGA membutuhkan *running time* yang lebih lama dibandingkan GA-SA dalam satuan detik.

5.2 Saran

Biaya distribusi yang digunakan pada penelitian ini hanya memperhatikan faktor beban gaji supir dan standar bahan bakar. Untuk penelitian selanjutnya dapat memperhatikan faktor lainnya seperti jumlah kendaraan yang dipakai dan jadwal pengiriman barang.

DAFTAR PUSTAKA

- Anwar, Z. Penerapan Penggabungan Metode Zero Crossing dan Virus Evolutionary Genetic Algorithm (VEGA) pada Penyelesaian Persamaan Non-Linier. <http://repository.unej.ac.id/bitstream/handle/123456789/76386/Zainul%20Anwar%20-%20121810101025%20-1.pdf?sequence=1>. [Diakses pada 28 Maret 2017].
- Bertsimas, D. & J. Tsitsiklis. 1993. Simulted Annealing. *Statistical Science*. 8(1): 10-15.
- Chun, F. & Yi, Z. 2009. A Genetic Algorithm of Two-stage Supply Chain Distribution Problem Associated with Fixed Charge and Multiple Transportation Modes. *Fifth International Conference on Natural Computation*. 76-80.
- Emanuel, Am W, dan Aritonang, A.F. 2008. Aplikasi Desktop Pencarian Rute Jalan dengan Algoritma Simulated Annealing. *Jurnal Informatika*. 4(2):93-103.
- Gen, M., F. Altiparmak, & L. Lin. 2006. A Genetic Algorithm For Two-Stage Transportastion Problem Using Priority-Based Encoding. *OR Spektrum*. 28: 337-354.
- Gen, M. & R. Cheng. 1997. *Genetic Algorithm and Engineering Design*. USA: John Wile & Sons, Inc.
- Fukuda, T., K. Shimojima, & N. Kubota. 1999. *Virus Evolutionary Genetic Algorithm and Its Applications To Travelling Salesman Problem*. Editor : Xin Yao. Singapura : World Scientific.
- Indrianingsih, Y. 2010. Algoritma Genetik Untuk Menyelesaikan Masalah Optimasi Fungsi Berkendala Dengan Pengkodoan Bilangan Bulat. *Jurnal Angkasa Sekolah Tinggi Teknologi Adisutjipto (STTA)*. 2(1): 67-76.
- Jawahar, N. & A.N. Balaji. 2009. A Genetic Algorithm For The Two-Stage Supply Chain Distribution Problem Associated With A Fixed Charge. *European Journal of Operational Research*. 194:496-537.

- Juniarto, S.D., E. Martiana., A. Fariza. & Ira P. 2013, Optimasi Distribusi Barang Berdasarkan Rute dan Daya Tampung Menggunakan Metode Simulated Annealing. http://repo.pens.ac.id/1133/1/Makalah_PA_7407040021.pdf. [Diakses pada 05 Desember 2016].
- Kirkpatrick, S., C.D. Gelatt, dan M.P. Vecchi. 1983. Optimization by Simulated Annealing. *SCIENCE*. 220(4598): 671-680.
- Kubota, N., K. Shimojima, & T. Fukuda. 1996. The Role of Virus Infection in Virus-Evolutionary Genetic Algorithm. *IEEE*. 182-187.
- Ling, W. X. 2006. *Virus Transmission Genetic Algorithm*. Thesis. Athens : The University of Georgia.
- Mahmudy, W.F. 2013. *Algoritma Evolusi*. Malang: Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
- Nurjanah, W. D. 2015. Penerapan Algoritma Genetika Dan Simulated Annealing Pada Penjadwalan Flowshop. <http://repository.unej.ac.id/bitstream/handle/123456789/72173/Weny%20Duwi%20Nurjanah%20%20091810101054.pdf?sequence=1>. [Diakses pada 07 Desember 2016].
- Orkcu, H. H. 2013. Subset Selection In Multiple Linear Regression Models: A Hybrid Of Genetic And Simulated Annealing Algorithms. *Applied Mathematics and Computation*. 219: 11.018-11.028.
- Sofianti, T., D. Penjadwalan Multipurpose Batch Chemical Plant dengan Metode Optimasi Gabungan : Algoritma Genetika-Simulated Annealing. *Proceedings, Komputer dan Sistem Intelijen (KOMMIT2004)*.297-308.
- Sulistiyorini, R. & W.F. Mahmudy. 2015. Penerapan Algoritma Genetika Untuk Permasalahan Optimasi Distribusi Barang Dua Tahap. <http://wayanfm.lecture.ub.ac.id/files/2015/05/JurnalSkripsi-2014-2015-013-Riska-Sulistiyorini.pdf>. [Diakses pada 23 November 2016].
- Tospornsampan, J., I. Kita, M. Ishii, & Y. Kitamura. 2007. Split-Pipe Design of Water Distribution Using Simulated Annealing. *International Journal of Environmental, Chemical, Ecological, Geological and Geophysical Engineering*. 1(4): 28-38.

Vista, C.B. & W.F. Mahmudy. 2015. Penerapan Algoritma Evolution Strategies Untuk Optimasi Distribusi Barang Dua Tahap. <http://wayanfm.lecture.ub.ac.id/files/2015/05/JurnalSkripsi-2014-2015-011-Candra-Bella-Vista.pdf>. [Diakses pada 23 November 2016].



LAMPIRAN

LAMPIRAN A. Hasil Percobaan Parameter Populasi

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: $P_c = 1$, $P_m = 0,1$, 10000 iterasi, $T_{awal} = 900$, $T_{akhir} = 0$, $\alpha = 0.8$, populasi virus = 10, panjang virus = 0,6, $r = 0,9$. 2.524.000,00

A.1 Percobaan parameter 80 populasi

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.541.000,00	290,943	2.474.500,00	575,4858
2.	2.529.500,00	290,7095	2.483.500,00	597,3023
3.	2.524.000,00	270,1305	2.483.000,00	543,4648
4.	2.503.500,00	279,9332	2.478.500,00	559,0524
5.	2.534.000,00	286,6904	2.500.500,00	551,6509
6.	2.512.000,00	272,7082	2.480.500,00	550,5273
7.	2.536.000,00	295,9338	2.474.500,00	582,7561
8.	2.510.000,00	281,6898	2.491.500,00	551,803
9.	2.519.500,00	285,8835	2.492.500,00	551,4405
10.	2.525.000,00	285,6122	2.496.500,00	573,5538

A.2 Percobaan parameter 90 populasi

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.516.500,00	297,8645	2.482.500,00	579,9302
2.	2.527.000,00	329,1043	2.475.500,00	602,2645
3.	2.534.000,00	336,5125	2.495.000,00	591,9047

<i>Running ke-</i>	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
4.	2.511.500,00	308,9954	2.500.500,00	592,6887
5.	2.519.500,00	340,8975	2.486.000,00	592,123
6.	2.520.500,00	321,1643	2.484.000,00	625,784
7.	2.533.500,00	319,8428	2.491.500,00	626,0765
8.	2.520.500,00	295,6089	2.479.500,00	569,7498
9.	2.527.500,00	299,9316	2.487.000,00	579,1993
10.	2.508.500,00	308,9792	2.484.500,00	578,8502

A.3 Percobaan parameter 100 populasi

<i>Running ke-</i>	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.525.000,00	346,7128	2.482.500,00	602,1382
2.	2.511.000,00	345,739	2.489.500,00	596,7082
3.	2.512.500,00	345,1927	2.484.000,00	600,7308
4.	2.511.000,00	315,849	2.489.000,00	580,8334
5.	2.530.000,00	343,1487	2.481.500,00	608,7433
6.	2.536.500,00	365,6362	2.480.500,00	631,4202
7.	2.523.000,00	354,7186	2.481.500,00	661,3612
8.	2.515.500,00	367,1321	2.489.500,00	670,6771
9.	2.516.500,00	404,7341	2.485.500,00	604,5054
10.	2.515.000,00	340,7201	2.478.000,00	614,4487

LAMPIRAN B. Hasil Percobaan Parameter *Crossover*

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA)

diantaranya: Populasi = 60, $P_m = 0,1$, 3000 iterasi, $Tawal = 500$, $Takhir = 0$, $\alpha = 0.8$, populasi virus = 10, panjang virus = 0,4, $r = 0,9$

B.1 Percobaan parameter $P_c = 1$

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.527.500,00	41,01	2.485.000,00	94,7507
2.	2.540.000,00	40,0647	2.499.500,00	95,5892
3.	2.524.000,00	36,3682	2.493.000,00	87,514
4.	2.538.500,00	39,2004	2.489.000,00	86,4546
5.	2.558.000,00	37,288	2.489.500,00	86,0744
6.	2.545.500,00	40,6974	2.486.500,00	91,4962
7.	2.549.000,00	37,5827	2.484.000,00	88,211
8.	2.522.000,00	37,5181	2.478.000,00	87,5351
9.	2.541.000,00	40,6283	2.493.000,00	89,4715
10.	2.564.500,00	36,9521	2.532.500,00	81,5327

B.2 Percobaan parameter $P_c = 0.95$

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.537.000,00	38,3709	2.498.000,00	86,8587
2.	2.607.500,00	32,7858	2.516.500,00	76,2622
3.	2.576.500,00	33,5789	2.493.500,00	77,4753
4.	2.587.500,00	34,0765	2.483.500,00	80,0994
5.	2.584.500,00	33,5189	2.498.500,00	79,8416
6.	2.540.000,00	32,2886	2.507.000,00	76,7492
7.	2.593.500,00	32,6119	2.500.500,00	78,3198
8.	2.561.500,00	32,9615	2.497.500,00	78,0576

<i>Running ke-</i>	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
9.	2.570.000,00	34,3908	2.481.500,00	81,0792
10.	2.577.000,00	41,1605	2.499.500,00	92,4975

B.3 Percobaan parameter $P_c = 0.89$

<i>Running ke-</i>	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.543.000,00	38,0301	2.492.500,00	90,9661
2.	2.578.500,00	38,1582	2.481.500,00	90,7179
3.	2.586.000,00	40,0744	2.485.500,00	87,3289
4.	2.609.000,00	37,5302	2.513.000,00	84,9854
5.	2.557.500,00	36,7299	2.488.000,00	85,7042
6.	2.548.500,00	38,447	2.503.500,00	89,9913
7.	2.559.000,00	39,7393	2.502.500,00	90,0801
8.	2.586.000,00	39,0707	2.508.000,00	87,9525
9.	2.557.500,00	39,0673	2.493.500,00	88,173
10.	2.611.500,00	38,9265	2.513.500,00	88,4785

LAMPIRAN C. Hasil Percobaan Parameter Mutasi

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: Populasi = 60, $P_c = 1$, 3000 iterasi, $Tawal = 500$, $Takhir = 0$, $\alpha = 0.8$, populasi virus = 10, panjang virus = 0,4, $r = 0,9$.

C.1 Percobaan parameter $Pm = 0,05$

<i>Running</i> ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.550.500,00	24,681	2.505.000,00	81,7656
2.	2.573.500,00	25,7531	2.500.000,00	77,5227
3.	2.548.000,00	25,1623	2.513.000,00	79,2212
4.	2.589.500,00	25,3669	2.496.000,00	79,251
5.	2.577.500,00	23,781	2.503.500,00	76,4604
6.	2.546.500,00	24,372	2.499.500,00	81,7627
7.	2.553.000,00	24,5441	2.502.500,00	79,5161
8.	2.552.500,00	27,0904	2.514.000,00	81,9297
9.	2.546.500,00	27,1248	2.493.500,00	82,2704
10.	2.571.000,00	25,5647	2.484.500,00	82,3284

C.2 Percobaan parameter $Pm = 0,1$

<i>Running</i> ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.546.000,00	41,0424	2.509.000,00	91,0078
2.	2.537.500,00	40,804	2.476.500,00	94,006
3.	2.551.500,00	36,7702	2.495.000,00	86,5116
4.	2.528.500,00	36,6031	2.501.500,00	87,5612
5.	2.537.500,00	38,2113	2.503.000,00	86,449
6.	2.527.000,00	37,2418	2.486.000,00	87,2504
7.	2.557.500,00	37,0693	2.496.000,00	85,8537
8.	2.542.000,00	36,8397	2.512.000,00	89,7369
9.	2.554.500,00	36,9015	2.508.000,00	85,5316
10.	2.549.000,00	36,4318	2.507.000,00	87,0629

C.3 Percobaan parameter $P_m = 0,2$

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.547.000,00	65,211	2.503.000,00	107,7761
2.	2.535.000,00	62,337	2.487.500,00	106,1246
3.	2.590.500,00	66,453	2.485.500,00	102,6423
4.	2.554.000,00	61,9218	2.497.000,00	102,9737
5.	2.522.500,00	62,573	2.505.500,00	102,7724
6.	2.538.500,00	70,5726	2.508.000,00	109,4066
7.	2.542.500,00	70,2202	2.479.000,00	112,2437
8.	2.532.500,00	73,8415	2.489.000,00	117,2238
9.	2.552.500,00	77,4879	2.487.500,00	118,1074
10.	2.539.000,00	72,2813	2.484.500,00	113,1645

LAMPIRAN D. Hasil Percobaan Parameter Iterasi

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) dan *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$, $Tawal = 500$, $Takhir = 0$, $\alpha = 0,8$, populasi virus = 10, panjang virus = 0,4, $r = 0,9$.

D.1 Percobaan Iterasi = 10.000

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.523.500,00	114,3015	2.508.500,00	313,1748
2.	2.535.000,00	122,8937	2.491.500,00	300,1553
3.	2.519.500,00	147,0877	2.491.500,00	341,4781
4.	2.512.500,00	142,1558	2.483.000,00	330,208

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
5.	2.566.000,00	138,0305	2.474.500,00	337,4257
6.	2.515.000,00	153,2771	2.491.500,00	365,0209
7.	2.505.500,00	148,6553	2.493.000,00	363,2789
8.	2.534.500,00	113,7768	2.495.500,00	261,1638
9.	2.555.000,00	114,1893	2.481.500,00	261,7027
10.	2.536.500,00	110,3269	2.490.000,00	256,5973

D.2 Percobaan Iterasi = 20.000

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.521.000,00	216,1189	2.287.000,00	489,5532
2.	2.524.000,00	222,8192	2.483.000,00	513,6907
3.	2.524.500,00	221,3903	2.476.000,00	512,0488
4.	2.534.000,00	230,9102	2.500,500,00	559,9412
5.	2.523.000,00	264,7687	2.486.500,00	580,1531
6.	2.523.500,00	229,1907	2.489.000,00	520,1973
7.	2.521.000,00	224,8255	2.487.000,00	520,0575
8.	2.514.000,00	226,9583	2.483.000,00	526,148
9.	2.524.500,00	227,8533	2.476.000,00	523,0311
10.	2.534.000,00	253,211	2.500.500,00	536,734

D.3 Percobaan Iterasi = 50.000

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.532.500,00	627,2353	2.488.000,00	1579,482

Running ke-	Hasil GA-SA		Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
2.	2.517.500,00	634,2781	2.497.500,00	1468,2808
3.	2.517.000,00	633,9906	2.486.000,00	1920,1463
4.	2.513.500,00	683,7596	2.487.500,00	1500,3238
5.	2.502.000,00	601,0108	2.491.000,00	1142,0635
6.	2.490.500,00	605,9656	2.479.500,00	1379,659
7.	2.501.500,00	605,1402	2.484.500,00	1488,9239
8.	2.531.500,00	648,6723	2.497.000,00	1397,028
9.	2.501.500,00	608,5166	2.482.000,00	1408,7373
10.	2.533.000,00	572,319	2.486.000,00	1920,1463

LAMPIRAN E. Hasil Percobaan Parameter Suhu Awal (Tawal)

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$ 3000 iterasi, $Takhir = 0$, $\alpha = 0.8$.

E.1 Percobaan parameter Tawal = 500

Running ke-	Hasil GA-SA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.528.000,00	33,7199
2.	2.538.000,00	34,6303
3.	2.538.000,00	34,3105
4.	2.528.000,00	33,4569
5.	2.541.000,00	37,1189
6.	2.554.000,00	36,6012
7.	2.550.000,00	36,0671
8.	2.534.500,00	36,9864

Hasil GA-SA		
<i>Running ke-</i>	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
9.	2.584.000,00	35,9659
10.	2.546.500,00	37,9801

E.2 Percobaan parameter Tawal = 700

Hasil GA-SA		
<i>Running ke-</i>	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.553.000,00	35,3347
2.	2.535.500,00	36,06-8
3.	2.527.000,00	33,613
4.	2.532.000,00	32,8516
5.	2.553.000,00	32,5697
6.	2.530.500,00	33,0986
7.	2.547.000,00	37,4923
8.	2.576.000,00	33,3649
9.	2.536.000,00	33,292
10.	2.543.500,00	32,6472

E.2 Percobaan parameter Tawal = 900

Hasil GA-SA		
<i>Running ke-</i>	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.527.000,00	33,3304
2.	2.522.000,00	33,9011
3.	2.527.000,00	35,8085
4.	2.544.500,00	34,5862
5.	2.527.000,00	34,9277

Hasil GA-SA		
<i>Running ke-</i>	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
6.	2.522.000,00	36,6833
7.	2.536.500,00	33,3511
8.	2.544.500,00	36,4267
9.	2.522.000,00	35,3249
10.	2.525.000,00	53,5911

LAMPIRAN F. Hasil Percobaan Parameter α

Parameter-parameter yang digunakan pada *Hybrid Genetic Algorithm-Simulated Annealing* (GA-SA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$ 3000 iterasi, $T_{awal} = 900$, $T_{akhir} = 0$

F.1 Percobaan parameter $\alpha = 0.8$

Hasil GA-SA		
<i>Running ke-</i>	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.536.500,00	37,2484
2.	2.576.000,00	36,2725
3.	2.558.000,00	34,6652
4.	2.538.500,00	28,2611
5.	2.581.500,00	38,5333
6.	2.566.000,00	40,661
7.	2.533.500,00	40,1246
8.	2.539.000,00	38,149
9.	2.555.000,00	39,9224
10.	2.538.000,00	37,4928

F.2 Percobaan parameter $\alpha = 0.9$

<i>Running</i> ke-	Hasil GA-SA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.540.000,00	36,0817
2.	2.533.000,00	36,3223
3.	2.550.000,00	35,1145
4.	2.536.000,00	33,2418
5.	2.519.500,00	33,982
6.	2.547.500,00	43,1486
7.	2.543.000,00	35,3454
8.	2.556.000,00	33,5674
9.	2.536.500,00	34,2253
10.	2.535.500,00	35,8103

F.3 Percobaan parameter $\alpha = 0.98$

<i>Running</i> ke-	Hasil GA-SA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.551.500,00	34,2477
2.	2.537.500,00	34,5372
3.	2.559.500,00	34,4312
4.	2.530.500,00	34,1128
5.	2.547.500,00	35,5841
6.	2.527.000,00	36,0824
7.	2.527.000,00	36,0645
8.	2.545.000,00	36,2454
9.	2.547.500,00	37,615
10.	2.527.000,00	38,3192

LAMPIRAN G. Hasil Percobaan Parameter Populasi Virus

Parameter-parameter yang digunakan pada *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$ 3000 iterasi, panjang virus = 0,4, $r = 0,9$.

G.1 Percobaan parameter Populasi Virus = 10

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.490.500,00	73,9435
2.	2.500.500,00	75,3431
3.	2.508.000,00	74,1007
4.	2.484.500,00	76,7553
5.	2.494.000,00	81,5302
6.	2.493.500,00	84,2023
7.	2.489.000,00	89,4512
8.	2.494.000,00	92,2232
9.	2.504.500,00	87,9402
10.	2.508.000,00	88,7117

G.2 Percobaan parameter Populasi Virus = 25

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.487.000,00	174,9565
2.	2.490.000,00	171,8924
3.	2.503.000,00	158,7106
4.	2.506.000,00	160,4295
5.	2.495.500,00	160,3192

6.	2.510.000,00	161,3407
7.	2.496.500,00	168,7954
8.	2.490.000,00	160,7642
9.	2.488.000,00	161,7013
10.	2.516.000,00	161,4214

G.3 Percobaan parameter Populasi Virus = 35

Hasil VEGA		
Running ke-	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.497.500,00	204,7024
2.	2.484.500,00	217,6173
3.	2.497.500,00	229,2416
4.	2.495.000,00	222,1625
5.	2.497.500,00	216,0915
6.	2.484.500,00	221,9601
7.	2.502.000,00	221,2235
8.	2.495.000,00	311,0835
9.	2.484.500,00	232,7828
10.	2.525.500,00	312,8502

LAMPIRAN H. Hasil Percobaan Parameter Panjang Virus

Parameter-parameter yang digunakan *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$ 3000 iterasi, populasi virus = 10, $r = 0,9$.

H.1 Percobaan parameter Panjang Virus = 0.4

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.493.000,00	87,2326
2.	2.495.500,00	90,0745
3.	2.493.500,00	87,7473
4.	2.500.000,00	92,0701
5.	2.494.000,00	91,8375
6.	2.481.000,00	92,5018
7.	2.485.000,00	93,8516
8.	2.484.000,00	90,7646
9.	2.503.500,00	91,4735
10.	2.483.500,00	91,9841

H.2 Percobaan parameter Panjang Virus = 0.6

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.499.000,00	88,271
2.	2.527.000,00	86,252
3.	2.476.500,00	86,4274
4.	2.491.000,00	82,5389
5.	2.489.500,00	81,1834
6.	2.485.500,00	91,503
7.	2.493.500,00	83,5989
8.	2.500.500,00	77,7271
9.	2.494.500,00	82,3616
10.	2.502.500,00	82,9776

H.1 Percobaan parameter Panjang Virus = 0.8

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.496.000,00	79,0189
2.	2.480.000,00	94,5401
3.	2.484.500,00	79,0868
4.	2.503.000,00	78,6007
5.	2.503.500,00	75,7031
6.	2.491.500,00	81,4947
7.	2.500.500,00	84,3474
8.	2.507.000,00	84,1048
9.	2.494.000,00	81,7926
10.	2.491.500,00	90,1089

LAMPIRAN I. Hasil Percobaan Parameter r

Parameter-parameter yang digunakan pada *Virus Evolutionary Genetic Algorithm* (VEGA) diantaranya: Populasi = 60, $P_c = 1$, $P_m = 0,1$ 3000 iterasi populasi virus = 10, panjang virus = 0,6.

I.1 Percobaan parameter $r = 0,9$

<i>Running</i> ke-	Hasil VEGA	
	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.489.500,00	84,7752
2.	2.494.500,00	87,7451
3.	2.485.500,00	80,9544
4.	2.482.500,00	83,0922
5.	2.494.500,00	80,0799

Hasil VEGA		
Running ke-	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
6.	2.488.500,00	83,6314
7.	2.486.500,00	82,3475
8.	2.487.000,00	87,6366
9.	2.493.000,00	84,5162
10.	2.499.500,00	84,7221

I.2 Percobaan parameter $r = 0,7$

Hasil VEGA		
Running ke-	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.487.000,00	88,4923
2.	2.492.000,00	89,1604
3.	2.494.000,00	84,9902
4.	2.500.500,00	85,3546
5.	2.499.000,00	88,0867
6.	2.492.000,00	88,4604
7.	2.491.500,00	90,3352
8.	2.524.000,00	91,4654
9.	2.501.000,00	91,8004
10.	2.495.500,00	88,9244

I.3 Percobaan parameter $r = 0,5$

Hasil VEGA		
Running ke-	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
1.	2.484.000,00	88,7033

Hasil VEGA		
Running ke-	Biaya Total Distribusi (Rp)	Waktu Komputasi (detik)
2.	2.488.500,00	86,9019
3.	2.493.500,00	88,8504
4.	2.512.000,00	88,733
5.	2.483.500,00	90,0156
6.	2.492.000,00	89,5428
7.	2.506.500,00	101,5564
8.	2.502.000,00	87,6103
9.	2.490.500,00	89,8278
10.	2.487.000,00	89,382