

**IMPLEMENTASI ALGORITMA GENETIKA DALAM PENENTUAN
RUTE TERBAIK PENDISTRIBUSIAN BBM PADA PT. PERTAMINA
BERBASIS ANDROID**

SKRIPSI



UIN SUNAN AMPEL
S U R A B A Y A

Disusun Oleh
GANESHAR BALENOREZKY DHIO PRASANDA
H02216006

PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL
SURABAYA

2019

PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini,

Nama : GANESHAR BALENOREZKY DHIO PRASANDA

NIM : H02216006

Program Studi : Matematika

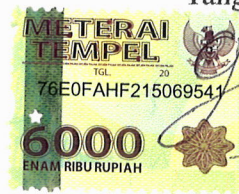
Angkatan : 2016

Menyatakan bahwa saya tidak melakukan plagiat dalam penulisan skripsi saya yang berjudul " IMPLEMENTASI ALGORITMA GENETIKA DALAM PENENTUAN RUTE TERBAIK PENDISTRIBUSIAN BBM PADA PT. PERTAMINA BERBASIS ANDROID ". Apabila suatu saat nanti terbukti saya melakukan tindakan plagiat, maka saya bersedia menerima sanksi yang telah ditetapkan.

Demikian pernyataan keaslian ini saya buat dengan sebenar-benarnya.

Surabaya, 20 Desember 2019

Yang menyatakan,



GANESHAR BALENOREZKY DHIO PRASANDA

NIM. H02216006

LEMBAR PERSETUJUAN PEMBIMBING

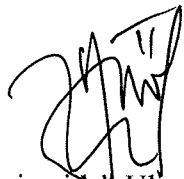
Skripsi oleh

Nama : GANESHAR BALENOREZKY DHIO PRASANDA
NIM : H02216006
Judul Skripsi : IMPLEMENTASI ALGORITMA GENETIKA DALAM
PENENTUAN RUTE TERBAIK PENDISTRIBUSIAN
BBM PADA PT. PERTAMINA BERBASIS ANDROID

telah diperiksa dan disetujui untuk diujikan.

Surabaya, 20 Desember 2019

Pembimbing



Nurissaidah Ulinnuha, M.Kom
NIP. 199011022014032204

PENGESAHAN TIM PENGUJI SKRIPSI

Skripsi oleh

Nama : GANESHAR BALENOREZKY DHIO PRASANDA
NIM : H02216006
Judul Skripsi : IMPLEMENTASI ALGORITMA GENETIKA DALAM
PENENTUAN RUTE TERBAIK PENDISTRIBUSIAN
BBM PADA PT. PERTAMINA BERBASIS ANDROID

Telah dipertahankan di depan Tim Penguji
pada tanggal Desember 2019

Mengesahkan,
Tim Penguji

Penguji I


Nurissaidah Dlinnuha, M.Kom
NIP. 199011022014032204

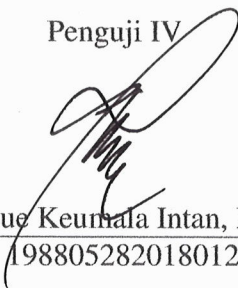
Penguji II


Dian C Rini Novitasari, M.Kom
NIP. 198511242014032001

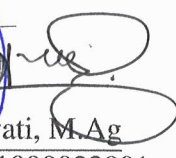
Penguji III


Aris Fanani, M.Kom
NIP. 198701272014031002

Penguji IV


Putroun Keurnala Intan, M.Si
NIP. 198805282018012001

Mengetahui,
Dekan Fakultas Sains dan Teknologi
UIN Sunan Ampel Surabaya



Dr. Eni Purwati, M.Ag
NIP. 196512211990022001



KEMENTERIAN AGAMA
UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA
PERPUSTAKAAN

Jl. Jend. A. Yani 117 Surabaya 60237 Telp. 031-8431972 Fax. 031-8413300
E-Mail: perpus@uinsby.ac.id

LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika UIN Sunan Ampel Surabaya, yang bertanda tangan di bawah ini, saya:

Nama : GANESHAR BALENOPEZKY DHIO PRASANDA .
NIM : H02216006 .
Fakultas/Jurusan : SAINTEK / MATEMATIKA .
E-mail address : ganesharprasanda@gmail.com .

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Perpustakaan UIN Sunan Ampel Surabaya, Hak Bebas Royalti Non-Eksklusif atas karya ilmiah :

☒ Sekripsi ☐ Tesis ☐ Desertasi ☐ Lain-lain (.....)

yang berjudul :

IMPLEMENTASI ALGORITMA GENETIKA DALAM PENENTUAN RUTE
TERBAIK PENDISTRIBUSIAN BBM PADA PT. PERTAMINA
BERBASIS ANDROID

beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini Perpustakaan UIN Sunan Ampel Surabaya berhak menyimpan, mengalih-media/format-kan, mengelolanya dalam bentuk pangkalan data (database), mendistribusikannya, dan menampilkan/mempublikasikannya di Internet atau media lain secara **fulltext** untuk kepentingan akademis tanpa perlu meminta ijin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan atau penerbit yang bersangkutan.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Perpustakaan UIN Sunan Ampel Surabaya, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah saya ini.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Surabaya, 30 Desember 2019 .

Penulis

(GANESHAR B.D.P .)
nama terang dan tanda tangan

IMPLEMENTASI ALGORITMA GENETIKA DALAM PENENTUAN RUTE TERBAIK PENDISTRIBUSIAN BBM PADA PT. PERTAMINA BERBASIS ANDROID

Kata kunci: Graf, *Travelling Salesman Problem*, Algoritma Genetika, Asimetris

2.6.3.	Fungsi Evaluasi	16
2.6.4.	Seleksi	16
2.6.5.	Operator Genetika	17
2.6.6.	Penyilangan banyak titik (<i>Multi-point crossover</i>	18
2.6.7.	Penentuan Parameter	20
2.7.	Penerapan Algoritma Genetika Terhadap Kasus TSP	21
2.7.1.	Menentukan Populasi Awal	23
2.7.2.	Evaluasi <i>Fitness</i>	24
2.7.3.	Seleksi	26
2.7.4.	<i>Crossover</i>	28
2.7.5.	Mutasi	31
2.7.6.	<i>Elitism</i>	33
III	METODE PENELITIAN	35
3.1.	Metode Pengolahan	35
3.2.	<i>Flowchart System</i>	36
3.3.	<i>Flowchart</i> Algoritma Genetika	37
3.4.	Perancangan Antarmuka (<i>Interface</i>)	37
IV	HASIL DAN PEMBAHASAN	43
4.1.	Pencarian Rute Optimal Menggunakan Algoritma Genetika	43
4.1.1.	Menentukan Populasi Awal	45
4.1.2.	Evaluasi <i>Fitness</i>	47
4.1.3.	Seleksi	48
4.1.4.	<i>Crossover</i>	51
4.1.5.	Mutasi	55
4.1.6.	<i>Elitism</i>	60
4.1.7.	Hasil Algoritma Genetika	62
4.2.	Implementasi Algoritma Genetika Pada Sistem Aplikasi	63
4.2.1.	Halaman Menu Utama	63
4.2.2.	Halaman Bantuan	63
4.2.3.	Halaman Tentang	64

Pencarian jalan/rute terpendek adalah suatu contoh dari optimasi untuk menemukan suatu jalur antara dua simpul sedemikian hingga jumlah bobot dari busur penyusunnya dapat seminimal mungkin (Sukatoni, 2011). Pada umumnya jalur terpendek dapat diperoleh dengan cara menghitung jarak dari kota asal ke kota tujuan ataupun menghitung waktu yang telah ditempuh. Pencarian jalur terpendek akan semakin rumit jika alternatif jalur ke kota tujuan semakin banyak jumlahnya. (Saptono & Hidayat, 2007). Pencarian rute terpendek (rute optimum) sangatlah dibutuhkan oleh orang-orang yang masih belum mengetahui jalur yang akan dilewati untuk menuju tempat tujuannya, seperti kurir dari perusahaan jasa pengantaran barang ataupun dari perusahaan untuk pendistribusian suatu produk, yang biasa disebut dengan *Travelling Salesman Problem* (Baharudin, 2018).

Ketersediaan Bahan Bakar Minyak (BBM) di Indonesia sudah menjadi suatu kebutuhan penting bagi masyarakat, tidak hanya digunakan untuk bahan bakar kendaraan bermotor, akan tetapi BBM juga dibutuhkan dalam industri dan

Pegoptimalan rute terpendek dalam kasus pendistribusian BBM ke sejumlah SPBU di kota Surabaya adalah contoh dari permasalahan TSP Asimetris di dalam graf berbobot. TSP Asimetris merupakan salah satu jenis dari TSP dimana seorang *salesman* hanya mengunjungi setiap tujuan tepat satu kali kecuali titik awal yang menjadi titik akhir dari perjalanan *salesman* dan jalur antara satu titik dengan titik yang lain memiliki perbedaan jarak (bobot) yang berbeda. Banyak sekali jenis-jenis algoritma yang dapat digunakan untuk menyelesaikan permasalahan TSP Asimetris, seperti algoritma-algoritma yang terdapat pada Metode Deterministik dan Metode Heuristik. Akan tetapi dalam skripsi ini akan dipergunakan metode heuristik yang merupakan salah satu metode untuk memecahkan permasalahan optimasi (Napitipulu, 2018).

Genetic Algorithm (Algoritma Genetika) termasuk di dalam salah satu jenis metode heuristik yang merupakan cabang dari Algoritma Evolusi (*Evolutionary Algorithm*) (Napitipulu , 2018). Algoritma Genetika adalah metode optimasi

Sejumlah penelitian mengatakan bahwa algoritma genetika merupakan salah satu metode yang sering digunakan untuk menyelesaikan model TSP dikarenakan algoritma genetika memiliki operator-operator evolusi yang sangat efektif pada pencarian global dan algoritma genetika juga bekerja dengan populasi titik, bukan 1 titik. Sebagai contoh judul dari penelitian sebelumnya yang diteliti oleh Lusina yaitu *“Pencarian Rute Terbaik Pada Travelling Salesman Problem (TSP) Menggunakan Algoritma Genetika pada Dinas Kebersihan dan Pertamanan Kota Pekanbaru”* yang menghasilkan solusi untuk Dinas Kebersihan dan Pertamanan Kota Pekanbaru dalam mencari rute optimal pada kegiatan pengangkutan sampah yang ingin dikelola (Lusiana & Hasibuan, 2015). Penelitian mengenai TSP yang menggunakan metode Algoritma Genetika juga dilakukan oleh Reza Kurniawan, yang berjudul *“Penerapan Algoritma Genetika Untuk Pencarian Lokasi ATM Terdekat Berbasis Android”* yang dapat memberikan informasi berupa jalur terpendek untuk lokasi ATM terdekat pada perangkat *mobile phone* dengan baik. Dengan adanya hasil dari penelitian sebelumnya maka dapat diketahui bahwa salah satu algoritma yang sangat cocok untuk menyelesaikan model TSP adalah algoritma genetika (Kurniawan & Rathomi, 2018).

SP kedalam bentuk aplikasi *Smartphone* Android

Masalah

hasil pencarian rute optimal pendistribusian BBM ke s
aya dengan menggunakan algoritma genetika?

hasil implementasi algoritma genetika ke dalam
droid untuk mencari rute optimal menuju ke sejumlah
jungi di kota Surabaya?

Analisis

hasil pencarian rute optimal menggunakan algoritma

BBM di wilayah SPBU di kota Surabaya

SP kedalam bentuk aplikasi *Smartphone* Android

Masalah

hasil pencarian rute optimal pendistribusian BBM ke s
aya dengan menggunakan algoritma genetika?

hasil implementasi algoritma genetika ke dalam
droid untuk mencari rute optimal menuju ke sejumlah
jungi di kota Surabaya?

Analisis

hasil pencarian rute optimal menggunakan algoritma

BBM di wilayah SPBU di kota Surabaya

- SP kedalam bentuk aplikasi *Smartphone* Android
- Masalah**
- hasil pencarian rute optimal pendistribusian BBM ke s
aya dengan menggunakan algoritma genetika?
- hasil implementasi algoritma genetika ke dalam
droid untuk mencari rute optimal menuju ke sejumlah
jungi di kota Surabaya?
- Penelitian**
- hasil pencarian rute optimal menggunakan algoritma
- BBM di wilayah SPBU di kota Surabaya

SP kedalam bentuk aplikasi *Smartphone* Android

Masalah

hasil pencarian rute optimal pendistribusian BBM ke s
aya dengan menggunakan algoritma genetika?

hasil implementasi algoritma genetika ke dalam
droid untuk mencari rute optimal menuju ke sejumlah
jungi di kota Surabaya?

Penelitian

hasil pencarian rute optimal menggunakan algoritma

BBM di wilayah SPBU di kota Surabaya

- SP kedalam bentuk aplikasi *Smartphone* Android
- Masalah**
- hasil pencarian rute optimal pendistribusian BBM ke s
aya dengan menggunakan algoritma genetika?
- hasil implementasi algoritma genetika ke dalam
droid untuk mencari rute optimal menuju ke sejumlah
jungi di kota Surabaya?
- Penelitian**
- hasil pencarian rute optimal menggunakan algoritma
- BBM di wilayah SPBU di kota Surabaya

1. Studi Kasus dari penelitian ini adalah pendistribusian BBM Pertamina ke setiap SPBU di Kota Surabaya.
2. Data lokasi SPBU hanya mencakup wilayah Kota Surabaya yang disediakan PT. Pertamina (Persero) dan jumlah yang digunakan adalah 96 SPBU.
3. Bobot yang digunakan adalah jarak.
4. Parameter yang digunakan adalah ukuran populasi (PopSize), Probabilitas *crossover* (Pc), Probabilitas mutasi (Pm), dan Maksimum Generasi (MaxGen).
5. Dalam penelitian ini digunakan PopSize sebesar 20, Pc sebesar 0,5, Pm sebesar 0,01 dan MaxGen sebesar 1000.

- latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

latar belakang penulisan masalah, manfaat penelitian

TINJAUAN PUSTAKA

Graf yang dipermasalahkan bukanlah grafik, melainkan representasi objek diskrit dan hubungannya dengan mereka. Secara visual, representasi graf adalah untuk mendeklarasikan objek sebagai titik atau lingkaran, dan hubungan antara objek tersebut dinyatakan oleh garis. Titik atau bulatan pada graf disebut juga dengan simpul, noktah, node atau vertex dan untuk garis disebut juga dengan sisi, *arc* atau *edge* (Napitipulu, 2018).

Graf dapat dilambangkan $G = (V, E)$, dimana V adalah seperangkat poin(himpunan titik), simpul noktah, vertex atau nodes dari G yaitu $V = (v_1, v_2, v_3, \dots, v_n)$. Sedangkan E adalah himpunan rusuk *edges*, *arc* atau sisi dari G , yaitu $E = (e_1, e_2, e_3, \dots, e_n)$ (Lestari , 2016).

$V(G)$ dan $E(G)$ dalam graf jika dipasangkan akan menjadi graf G dengan $V(G)$ yang merupakan objek-objek yang disebut titik (*node*) dari himpunan tak kosong dan berhingga, sedangkan $E(G)$ adalah titik-titik yang berbeda di $V(G)$ yang disebut sisi dari himpunan (mungkin kosong) pasangan tak berurutan (Napitipulu, 2018).

Pada Gambar 2.1 dapat diketahui bahwa himpunan $V(G)$ adalah (v_1, v_2, v_3, v_4) dan $E(G)$ adalah $(e_1, e_2, e_3, e_4, e_5, e_6)$

[illegible]

2.4. Algoritma

Algoritma merupakan suatu prosedur langkah-demi-langkah atau urutan langkah yang terbatas dari awal hingga akhir yang disusun secara logis dan sistematis untuk mendapatkan solusi dari suatu masalah, baik memecahkannya benar atau salah. (Untoro , 2010).

1. Algoritma memiliki langkah yang terbatas, yang berarti suatu algoritma harus berhenti setelah mengerjakan serangkaian tugas.
2. Setiap langkah tidak memiliki makna ganda atau tidak dapat diperbaiki (*not ambiguous*) dengan kata lain setiap langkah harus didefinisikan secara tepat.
3. Memiliki kondisi awal atau masukan (*input*).
4. Memiliki kondisi akhir atau keluaran (*output*).
5. Algoritma yang diikuti dengan benar pasti akan menyelesaikan masalah (efektif).

[illegible]

Adanya teori evolusi *Darwin*, yang menyatakan bahwa kelangsungan hidup suatu makhluk dipengaruhi oleh aturan telah menjadi inspirasi dari algoritma ini. “yang kuatlah yang bertahan”. Hidup dapat dipertahankan melalui proses produksi, kawin silang (*crossover*), dan mutasi itu adalah pernyataan yang dinyatakan oleh *Darwin*. Hal ini lah yang mendasari terbentuknya algoritma genetika (Napitipulu , 2018).

Algoritma Genetika bekerja dengan menggunakan aturan probabilistik, yang berarti bahwa algoritma genetika menciptakan banyak solusi dengan secara acak. Algoritma Genetika akan mendapatkan populasi yang lebih baik atau populasi yang mewakili solusi dari permasalahan dengan harapan semakin dekat kepada solusi masalah yang ada dengan cara menciptakan populasi baru melalui iterasi secara terus-menerus terhadap populasi awal (Napitipulu , 2018).

Terdapat beberapa hal yang harus dilakukan terlebih dahulu dalam penggunaan algoritma genetika, (Napitipulu , 2018), yaitu:

1. Menentukan kromosom, representasi solusi untuk masalah yang harus dipecahkan.
2. Mencari nilai *fitness*, nilai ini akan menjadi pandangan apakah solusi yang didapat baik atau tidak.
3. Pembentukan populasi awal. proses ini biasanya dilakukan dengan mudah bilangan acak seperti *random-walk*.
4. Menentukan seleksi yang digunakan.
5. Menentukan proses perkawinan silang (*crossover*).
6. Menentukan mutasi gen yang digunakan.

Terdapat beberapa komponen yang merupakan proses utama algoritma genetika, seperti *encoding*, prosedur inisialisasi, fungsi evaluasi, seleksi, operator genetika, dan penentuan parameter (Napitipulu, 2018).

Teknik yang digunakan untuk menyandikan suatu gen dari kromosom adalah teknik pengkodean, dimana gen merupakan bagian dari kromosom dan akan mewakili variabel. Solusi alternatif harus dikodekan ke dalam kromosom terlebih dahulu agar dapat diproses melalui algoritma genetika (Hardi , 2014). Teknik ini meliputi penyandian gen dan kromosom. Bentuk *string bit*, pohon, *array* bilangan real, daftar aturan, elemen permutasi, elemen program atau representasi lain biasa digunakan untuk menyajikan suatu gen, yang kemudian dapat diimplementasikan untuk operator genetika. Demikian juga, kromosom dapat disajikan dalam bentuk string bit, bilangan real, elemen permutasi, daftar aturan, elemen program, dan struktur lainnya (Napitipulu , 2018). Ada pun contoh dari representasi pembentukan kromosom adalah sebagai berikut:

- String bit : 10011, 01101, 11101, dst
- Bilangan real : 65.66, -67.98, 562.88, dst
- Elemen permutasi : E1, E2, E10, dst
- Daftar aturan : pemrograman genetika
- Struktur lainnya.

2.6.2. Prosedur Inisialisasi

Prosedur inisialisasi adalah suatu proses untuk menghasilkan populasi awal. Inisialisasi kromosom atau menghasilkan populasi awal dapat dilakukan secara acak, tetapi harus memperhatikan domain solusi dan kendala masalah yang ada (Napitipulu , 2018).

2.6.3. Fungsi Evaluasi

Evaluasi fungsi objektif (fungsi tujuan) dan konversi fungsi objektif ke dalam fungsi *fitness* merupakan dua hal yang harus dilakukan dalam algoritma genetika. Nilai yang dimiliki oleh setiap individu yang digunakan untuk menentukan tingkat kesesuaian seorang individu dengan kriteria atau tujuan (objektif) dari masalah yang ingin dicapai disebut juga dengan *Fitness* (Mahmudy, Marian & Luong , 2014)

2.6.4. Seleksi

Proses menentukan individu mana yang akan dipilih untuk dilakukan kawin silang (*crossover*) dan bagaimana keturunan terbentuk dari individu-individu yang dipilih tersebut merupakan proses dari seleksi (Napitipulu , 2018). Tujuan seleksi adalah untuk memberikan peluang reproduksi yang lebih besar untuk anggota populasi terbaik (Napitipulu , 2018). Berikut adalah metode seleksi yang dapat digunakan:

a. *Rank Based Fitness*

Dalam metode ini, nilai objektif akan dijadikan sudut pandang untuk mengurutkan populasi. Posisi dari masing-masing individu akan diurutkan berdasarkan nilai *fitness* dari setiap individu (Napitipulu , 2018).

Populasi awal dapat dibentuk dengan membangkitkan populasi acak. Untuk membangkitkan populasi digunakan rumus pembangkitan bilangan random.

$$Pop = random(nk, N) \quad (2.1)$$

Pop = populasi

nk = banyaknya gen pada 1 kromosom

N = banyaknya kromosom dalam 1 populasi

Dari rumus didapatkan arti, untuk membentuk 1 populasi maka akan dibangkitkan sejumlah N kromosom dalam 1 populasi, dimana 1 kromosom terdapat nk gen.

Dengan ketentuan parameter yang ada, maka akan terdapat 4 kromosom di dalam 1 populasi dan di dalam 1 kromosom terdapat 7 gen. Pada contoh kasus ini kromosom dipresentasikan dengan menggunakan elemen permutasi, yang diwakili dengan titik A, B, C, D, E dan F.

Kemudian hitung jarak total setiap kromosom berdasarkan jarak pada Tabel

2.1, seperti yang ditunjukkan di bawah ini:

$$K_1 = AB + BD + DC + CE + EF + FA$$

$$= 4 + 1 + 2 + 5 + 3 + 6$$

$$= 21$$

$$F_1 = 1/21$$

$$= 0.048$$

Lakukan langkah perhitungan seperti diatas untuk mendapatkan nilai *fitness* kromosom lain. Sehingga akan didapatkan nilai *fitness* untuk semua kromosom seperti yang ditunjukkan pada Tabel 2.3.

Tabel 2.3 Nilai *Fitness*

Kromosom(K)	Total Jarak	<i>Fitness</i>
1	21	0.048
2	20	0.050
3	24	0.042
4	20	0.050
Total <i>Fitness</i>		0.19

Sebagaimana yang terlihat pada Tabel 2.3, kromosom 2 dan kromosom 4 mempunyai nilai *fitness* tertinggi pada generasi saat ini yaitu 0.050 yang masing-masing susunan gennya adalah [A B C F E D A] dan [A D B C E F A], serta jarak yang didapatkan oleh kedua kromosom ini adalah 20. Untuk itu, kromosom 2 dan 4 akan disimpan sebagai kromosom elit yang kemudian akan digunakan di populasi selanjutnya pada proses *elitism*.

Total *fitness* dari semua kromosom telah diketahui pada proses sebelumnya, yaitu sebesar 0.19. Kemudian akan dicari *fitness* relatif pada setiap kromosom. Metode seleksi yang digunakan pada skripsi ini adalah *Roulette Wheel* dengan rumus untuk menentukan *fitness* relatifnya adalah sebagai berikut:

Dimana:

F = nilai *fitness* dari masing-masing kromosom

Maka didapatkan sebagai berikut:

$$P_4 = \frac{0.05}{0.19} = 0.2631 \approx 0.263$$

Kemudian hitung nilai *fitness* kumulatif dengan rumus ($C_k = P_{k-1} + P_k$) dengan C_k adalah *fitness* kumulatifnya, sehingga didapat.

2.7.4. Crossover

Proses *crossover* ini tidak akan dialami oleh seluruh kromosom induk, tetapi hanya sebagian saja. Ukuran dari probabilitas *crossover* (P_c) yang akan menentukan berapa banyak kromosom yang akan dilakukan *crossover*. P_c yang digunakan telah ditentukan sebelumnya, yaitu 0.5. Maka kromosom yang akan mengalami *crossover* adalah $0.5 \times 4 = 2$ kromosom. Metode *crossover* yang digunakan pada skripsi ini adalah *Partially Mapped Crossover* (PMX), yang mana dalam proses ini langkah pertama yang diambil adalah mengurutkan kromosom sesuai dengan nilai *fitness* terbaik. Karena sebelumnya telah diketahui bahwa hanya 2 kromosom yang akan mengalami proses *crossover* dan jumlah kromosom dalam populasi adalah 4 kromosom maka pilih 2 kromosom dengan nilai *fitness* terbaik, agar menyisakan 2 kromosom *fitness* terendah.

Kromosom Baru	Kromosom Terpilih	Susunan Gen
1	Kromosom 4	A D B C E F A
2	Kromosom 3	A B D E C F A
3	Kromosom 2	A B C F E D A
4	Kromosom 4	A D B C E F A

Tabel 2.5 Urutan Kromosom Berdasarkan Nilai *Fitness*

Kromosom	Susunan Gen	<i>Fitness</i>
1	A D B C E F A	0.050
3	A B C F E D A	0.050
4	A D B C E F A	0.050
2	A B D E C F A	0.042

Berdasarkan Tabel 2.5 dapat diketahui bahwa kromosom yang memiliki nilai *fitness* terbaik adalah kromosom 1 dan kromosom 3. Kemudian kedua kromosom tersebut akan disimpan untuk digabungkan dengan kromosom *offspring*. Lalu untuk 2 kromosom yang memiliki nilai *fitness* terendah akan di acak posisinya dan akan dilakukan proses *crossover*.

Tabel 2.6 Hasil Acak Kromosom

Kromosom	Susunan Gen
Kromosom 4	A D B C E F A
Kromosom 2	A B C F E D A

Kromosom yang telah di acak pada Tabel 2.6 akan menjadi *parent* yang kemudian akan dilakukan *crossover* dan menghasilkan *offspring*. Cara pemilihan *parent* dapat dilakukan dengan memasangkan kromosom pada urutan pertama dengan kromosom terakhir, kemudian memasangkan kromosom kedua pertama dengan kromosom kedua terakhir dan akan berjalan terus sampai memasangkan kromosom terakhir dengan kromosom pertama kembali. Lalu untuk menentukan batas gen yang akan dilakukan *crossover* dapat diperoleh dengan membangkitkan bilangan acak sebanyak dua kali dimulai dari 1 sampai panjang kromosom.

Bilangan acak terkecil akan menjadi batas awal gen *crossover* dan bilangan acak terbesar akan menjadi batas akhir gen *crossover*. Setelah diketahui batas awal dan batas akhir gen, batas tersebut akan digunakan untuk mengambil sebagian gen pada *parent* 1 dimulai dari batas awal sampai batas akhir. Gen yang diambil tersebut akan menjadi bagian dari *offspring* yang letak gennya sama seperti *parent* 1. Kemudian akan digunakan gen pada *parent* 2 untuk mengisi secara berurutan gen *offspring* yang belum terisi, sehingga gen pada *offspring* dapat terlengkapi.

Tabel 2.7 Proses *Crossover*

<i>Parent</i>	Susunan Gen	Bilangan Acak	<i>Offspring</i>
1	A D B C E F A	2 dan 4	A D B C F E A
2	A B C F E D A		
1	A B C F E D A	3 dan 5	A D C F E B A
2	A D B E C F A		

Setelah didapatkan kromosom *offspring* maka populasi baru dapat dibentuk dengan menggabungkan kromosom yang memiliki nilai *fitness* terbaik dengan *offspring* seperti yang terdapat pada Tabel 2.8.

Tabel 2.8 Populasi Kromosom Baru Setelah *Crossover*

Kromosom Baru	Kromosom Terpilih	Susunan Gen
1	Kromosom 1	A D B C E F A
2	Kromosom 3	A B C F E D A
3	<i>Offspring 1</i>	A D B C F E A
4	<i>Offspring 2</i>	A D C F E B A

Populasi baru yang diperoleh dari hasil *crossover* akan dilanjutkan proses mutasi dengan nilai P_m yang telah ditentukan sebelumnya, yaitu 0.1. Metode mutasi yang digunakan adalah *Swapping Mutation*, dengan langkah awal menghitung total gen pada satu populasi dengan rumus:

Dimana:

g = jumlah gen dalam kromosom

 k = jumlah kromosom
$$P = 7 \times 4$$

$$P = 28$$

Untuk menentukan banyaknya gen yang akan termutasi digunakan ukuran Pm yang telah ditentukan sebelumnya dengan panjang total gen yang telah didapat. Maka banyak gen yang termutasi adalah $28 \times 0.1 = 2.8$. Telah didapatkan hasil 2.8 maka dilakukanlah pembulatan menjadi 3. Sehingga ada 3 gen yang mengalami mutasi. Kemudian untuk menentukan posisi gen yang akan mengalami mutasi diperoleh dengan membangkitkan bilangan acak dimulai dari 1 hingga total panjang gen, yaitu 1-28. Setelah dilakukan pengacakan, dimisalkan posisi gen 11, 17 dan 25 yang akan mengalami mutasi. Maka, didapatlah populasi kromosom

- $$AD + DC + CE + EF + FB + BA$$

$$= 3 + 2 + 5 + 3 + 5 + 4 = 22$$

$$Fitness \text{ kromosom } 4 = \frac{1}{22} = 0.045$$

2.7.6. *Elitism*

Terdapat suatu kemungkinan bahwa kromosom terbaik yang ada pada generasi sebelumnya tidak termasuk dalam populasi baru atau kromosom dengan nilai *fitness* terbaik telah menurun karena proses *crossover* dan proses mutasi oleh karena itu proses *elitism* ini dibutuhkan.

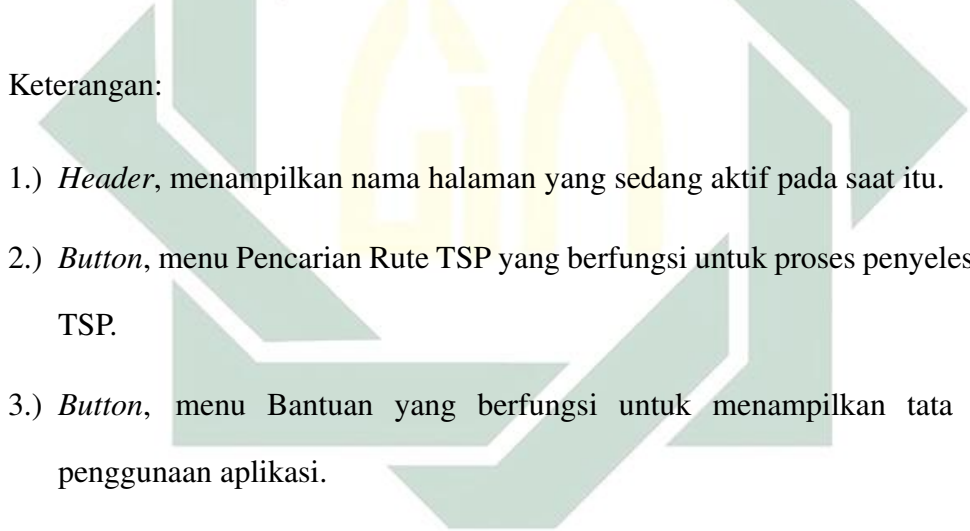
Pada proses evaluasi *fitness* di populasi awal sebelumnya, terdapat dua kromosom yang memiliki nilai *fitness* tertinggi. Namun, pada proses *elitism* ini hanya satu kromosom saja yang akan digunakan. Penulis memilih kromosom 2 pada proses evaluasi *fitness* sebagai kromosom pengganti untuk kromosom 4 yang telah mengalami proses *crossover* dan mutasi. Sehingga akan didapatkan populasi baru seperti pada Tabel 2.10.

Masukan Koordinat SPBU



Perancangan antarmuka adalah proses membuat perancangan dari halaman yang akan ditampilkan pada layar. Hal ini bertujuan untuk memudahkan pengguna untuk dapat melakukan interaksi dengan aplikasi (Napitipulu , 2018)

Pada Halaman ini, rancangan aplikasi akan berisi tiga menu utama, yaitu Pencarian Rute TSP, Bantuan, dan Tentang. Sebagaimana terlihat pada Gambar 3.4 yang merupakan rancangan halaman menu utama pada aplikasi.



Keterangan:

- 1.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
- 2.) *Button*, menu Pencarian Rute TSP yang berfungsi untuk proses penyelesaian TSP.
- 3.) *Button*, menu Bantuan yang berfungsi untuk menampilkan tata penggunaan aplikasi.

Keterangan:

- 1.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
- 2.) *Button*, menu Pencarian Rute TSP yang berfungsi untuk proses penyelesaian TSP.
- 3.) *Button*, menu Bantuan yang berfungsi untuk menampilkan tata penggunaan aplikasi.

- Keterangan:
- 1.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
 - 2.) *Button*, menu Pencarian Rute TSP yang berfungsi untuk proses penyelesaian TSP.
 - 3.) *Button*, menu Bantuan yang berfungsi untuk menampilkan tata penggunaan aplikasi.

Keterangan:

- 1.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
- 2.) *Button*, menu Pencarian Rute TSP yang berfungsi untuk proses penyelesaian TSP.
- 3.) *Button*, menu Bantuan yang berfungsi untuk menampilkan tata penggunaan aplikasi.

Keterangan:

- 1.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
- 2.) *Button*, menu Pencarian Rute TSP yang berfungsi untuk proses penyelesaian TSP.
- 3.) *Button*, menu Bantuan yang berfungsi untuk menampilkan tata penggunaan aplikasi.



- 1.) *Button*, untuk kembali ke halaman sebelumnya.
- 2.) *Header*, menampilkan nama halaman yang sedang aktif pada saat itu.
- 3.) *TextView*, menampilkan keterangan pengguna harus memilih SPBU yang akan dikunjungi.
- 4.) *ListView*, menampilkan alamat SPBU.
- 5.) *Button*, untuk memulai proses pencarian rute TSP yang ditampilkan pada halaman selanjutnya.

Halaman ini akan muncul ketika hasil proses dilakukan pada halaman sebelumnya (halaman Pilih SPBU), telah dilakukan. Halaman ini akan menampilkan hasil proses Algoritma Genetika, yang berisi jarak sebelum dan sesudah proses algoritma, dan solusi rute yang direkonsiliasi akan muncul. Desain halaman Hasil Proses TSP dapat dilihat pada Gambar 3.6.

Dari ketentuan parameter yang ditetapkan, maka akan terdapat 20 kromosom didalam satu populasi dan terdapat 7 gen didalam satu kromosom. Pada perhitungan ini, gen A akan digunakan sebagai titik awal dan akhir dari rute perjalanan. Berdasarkan contoh pembentukan kromosom diatas maka dimisalkan didapat populasi awal serta total jarak dari masing-masing kromosom sebagai berikut.

Kromosom 10 = [A-E-F-C-D-B-A] = 58.2

Crossover atau pindah silang dilakukan dengan menggunakan *Partilly Mapped Crossover* (PMX) dengan probabilitas *crossover* (P_c) sebesar 0.5, maka kromosom yang akan mengalami *crossover* hanya 10 kromosom saja. Untuk memulai proses *crossover*, terlebih dahulu urutkan kromosom berdasarkan nilai *fitness* dari yang terbaik, seperti pada tabel 4.4.

Kromosom	Susunan Gen	<i>Fitness</i>	Kromosom	Susunan Gen	<i>Fitness</i>
15	A E B D C F A	0.023	20	A B C D E F A	0.019
17	A B D E C F A	0.021	4	A C D B F E A	0.017
18	A B D E C F A	0.021	8	A E F D B C A	0.017
19	A B D E C F A	0.021	10	A E F D B C A	0.017
12	A B D C E F A	0.020	14	A C D B F E A	0.017
3	A D B C E F A	0.020	5	A D B C F E A	0.017
6	A D B C E F A	0.020	11	A E F C D B A	0.017
7	A D B C E F A	0.020	13	A F B D E C A	0.017
16	A D B C E F A	0.020	9	A B C D F E A	0.015
2	A B C D E F A	0.019	1	A B F C E D A	0.015

Tabel 4.6 Proses *Crossover*

<i>Parent</i>	Susunan Gen	Bilangan Acak	<i>Offspring</i>
1	A C D B F E A	3 dan 5	A E D B F C A
2	A E F D B C A		
1	A B C D E F A	2 dan 5	A B C D E F A
2	A E F D B C A		
1	A F B D E C A	3 dan 4	A F B D C E A
2	A B F C E D A		
1	A B C D F E A	3 dan 6	A B C D F E A
2	A E F C D B A		
1	A D B C F E A	4 dan 6	A D B C F E A
2	A C D B F E A		
1	A C D B F E A	2 dan 3	A C D B F E A
2	A D B C F E A		
1	A E F C D B A	2 dan 4	A E F C B D A
2	A B C D F E A		
1	A B F C E D A	3 dan 5	A B F C E D A
2	A F B D E C A		
1	A E F D B C A	4 dan 6	A E F D B C A
2	A B C D E F A		
1	A E F D B C A	2 dan 5	A E F D B C A
2	A C D B F E A		

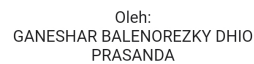
Tabel 4.8 Kromosom yang Mengalami Mutasi

Kromosom	Susunan Gen Lama	Susunan Gen Baru
1	A E B D C F A	A E B D C F A
2	A B D E C F A	A B D E C F A
3	A B D E C F A	A B D E C F A
4	A B D E C F A	A B D E C F A
5	A B D C E F A	A B D C E F A
6	A D B C E F A	A D B C E F A
7	A D B C E F A	A D B C E F A
8	A D B C E F A	A D B C E F A
9	A D B C E F A	A D B C E F A
10	A B C D E F A	A B C D E F A
11	A E D B F C A	A E D B F C A
12	A B C D E F A	A B C D E F A
13	A F B D C E A	A F B D C E A
14	A B C D F E A	A B F D E C A
15	A D B C F E A	A D B C F E A
16	A C D B F E A	A C D B F E A
17	A E F C B D A	A F E C B D A
18	A B F C E D A	A B F C E D A
19	A E F D B C A	A E F D B C A
20	A E F D B C A	A E F D B C A

Untuk melanjutkan ke proses selanjutnya perlu dilakukan kembali perhitungan nilai *fitness* sehingga dapat diketahui kromosom mana yang akan



Halaman ini berisi informasi tentang aplikasi, termasuk judul skripsi dan data singkat penulis. Halaman Tentang dapat dilihat pada Gambar 4.5.



Gambar 4.4 Halaman Tentang

Gambar 4.6 Halaman Hasil Proses TSP



Pada halaman ini akan ditampilkan jarak sebelum dan sesudah proses TSP serta menampilkan solusi rute yang diberikan. Seperti pada Gambar 4.7 dapat diketahui bahwa jarak sebelum proses TSP bernilai 45.80 Km dan jarak setelah proses TSP bernilai 42.20 Km dengan rute terbaik yang diperoleh berawal dari Kantor Pertamina (A) yang kemudian menuju SPBU 51.601.66 (C), kemudian menuju SPBU 51.601.77 (D), kemudian menuju SPBU 51.601.65 (B), kemudian menuju SPBU 54.601.03 (E), kemudian menuju SPBU 54.601.05 (F), yang kemudian kembali lagi ke Kantor Pertamina (A).

- Sukaton, R., 2011, *Penggunaan Algoritma Genetika Dalam Masalah Jalur Terpendek Pada Jaringan Data*.
- Supriana, I., 2017, *Implementasi Algoritma Genetika Dalam Penentuan Rute Terbaik Pendistribusian BBM Pada PT Burung Laut* Jurnal Teknologi Informasi dan Komputer, 285-294.
- Suyanto., 2010, *Algoritma Optimasi Deterministik atau Probabilistik* Yogyakarta: Graha ilmu.
- Untoro, F., 2010, *Algoritma dan Pemrograman dengan Bahasa java* Yogyakarta: Graha Imlu.
- Wahyu, P., 2013, *Optimasi Traveling Salesman Problem Dengan Algoritma Genetika Menggunakan Operator Partially Matched Crossover* Malang: Universitas Islam Negeri Maulana Malik Ibrahim Malang.
- Xu, J., Pei, L., & Zhu, R.-z., 2018, *Aplication Of Genetic Algorithm with Random Crossover and Dynamic Mutation on the Travelling Salesman Problem* Science Direct, 937-945.