

## BAB III

### METODOLOGI PENELITIAN

Pada bab ini dibahas mengenai metodologi dan perancangan untuk implementasi Algoritma Genetika dengan variasi mutasi dalam *Capacitated Vehicle Routing Problem with Time Windows (CVRPTW)* multi objektif pada distribusi beras bersubsidi Perum Bulog yang meliputi deskripsi masalah, tahapan penelitian, model optimasi dan teknik penyelesaian.

#### 3.1 Deskripsi Masalah

Penelitian ini membahas masalah penentuan rute distribusi beras bersubsidi, yaitu bagaimana beras dapat didistribusikan ke sejumlah pelanggan dengan tambahan kendala berupa adanya *capacitated* dan *time windows* pada masing-masing tujuan. Dalam hal ini, *capacitated* merujuk pada batasan jumlah barang atau kapasitas angkut maksimum yang dapat diangkut oleh setiap kendaraan selama satu perjalanan. *Time windows* merujuk pada batasan waktu ketersediaan pelanggan di mana pelanggan hanya dapat dilayani selama rentang waktu tersebut. Selain itu, *time windows* juga merujuk pada waktu yang diperlukan untuk melayani pelanggan. Suatu kendaraan harus mengunjungi pelanggan dalam rentang waktu tertentu, apabila melebihi rentang waktu yang ditentukan maka akan dikenakan penalti. Setiap kendaraan berangkat dari tempat awal (depot), kemudian mengunjungi sejumlah pelanggan dengan batas waktu pelayanan yang dimiliki setiap pelanggan, dan kembali lagi ke depot. Masalah penentuan rute pendistribusian beras bersubsidi diselesaikan dengan tujuan untuk mengoptimalkan rute pendistribusian setiap kendaraan dari depot ke pelanggan.

Penelitian ini meneliti masalah penentuan rute pendistribusian untuk memenuhi permintaan pelanggan dengan mempertimbangkan kendala kapasitas dan waktu pelayanan sehingga diperoleh total waktu, jumlah pelanggan terlewat, dan total biaya bahan bakar yang minimum. Penentuan rute pendistribusian pada masalah CVRPTW multi objektif akan diselesaikan menggunakan Algoritma Genetika dengan pemberian bobot pada setiap fungsi tujuan.

### 3.2 Tahapan Penelitian

Tahapan penelitian yang dilakukan adalah sebagai berikut:

#### 1. Studi Pustaka

Dalam penelitian ini, dilakukan studi pustaka dengan mempelajari teori yang berkaitan dengan Algoritma Genetika, CVRPTW, WSM, dan analisis sensitivitas dari berbagai sumber seperti buku dan jurnal untuk meningkatkan pengetahuan dan pemahaman dalam sebuah proses penelitian.

#### 2. Pengumpulan Data

Data yang digunakan pada penelitian ini adalah data yang berhubungan dengan pendistribusian beras bersubsidi oleh Perum Bulog.

#### 3. Penyelesaian Masalah

Pada tahapan ini akan dibangun model masalah CVRPTW multi objektif pada pendistribusian beras bersubsidi yang akan diselesaikan menggunakan Algoritma Genetika.

#### 4. Validasi

Sebelum dilakukan implementasi pada masalah CVRPTW multi objektif pada suatu perusahaan, perlu dilakukan validasi terlebih dahulu. Validasi merupakan proses yang dilakukan untuk menguji apakah model optimasi dan Algoritma Genetika yang digunakan sudah benar atau tidak. Validasi dilakukan dengan cara melihat solusi yang diperoleh dari metode penyelesaian dengan menggunakan Python sudah sesuai dengan tujuan penelitian atau belum. Jika belum sesuai, maka akan ditindaklanjuti lagi.

#### 5. Implementasi

Model optimasi CVRPTW multi objektif tersebut selanjutnya akan diimplementasikan untuk menyelesaikan masalah penentuan rute optimal pada pendistribusian beras bersubsidi.

#### 6. Penarikan Kesimpulan

Pada penarikan kesimpulan, dirangkum kembali hasil penelitian yang sudah dilakukan. Hasil penelitian tersebut meliputi kesesuaian hasil penelitian dengan rumusan masalah serta hasil implementasi berupa penentuan rute optimal yang dapat dibentuk dari data-data penelitian.

### 3.3 Data Penelitian

Data yang digunakan untuk penelitian ini merupakan data sekunder yang diambil dari penelitian Putri dkk. (2014) berupa data distribusi beras bersubsidi Perum Bulog yang meliputi data kendaraan berupa jumlah & kapasitas kendaraan, serta data pelanggan berupa data jumlah pelanggan, jumlah permintaan, *time windows*, dan data jarak pelanggan.

### 3.4 Model Optimisasi CVRPTW

Dalam masalah CVRPTW, terdapat  $K$  buah kendaraan dengan kapasitas berbeda yang akan mendistribusikan barang kepada  $N$  buah pelanggan. Setiap pelanggan  $i$  memiliki permintaan sebanyak  $q_i$ , waktu pelayanan  $s_i$ , dan pelayanan *time windows*  $z_i = [e_i, l_i]$  dengan  $e_i$  adalah waktu paling awal untuk melakukan pelayanan (*lower bound*) dan  $l_i$  adalah waktu paling lambat untuk melakukan pelayanan (*upper bound*). Permintaan  $q_i$  dari pelanggan harus dipenuhi dengan sekali pelayanan saja dalam batas *time windows* ( $z_i$ ).

Model optimisasi dibangun berdasarkan referensi model CVRPTW yang telah ada dengan memperhatikan beberapa asumsi, yakni:

1. Hanya terdapat satu depot sebagai titik awal sekaligus titik akhir pendistribusian
2. Setiap titik distribusi yang terhubung memiliki jarak yang simetris, artinya jarak dari titik  $i$  ke titik  $j$  sama dengan jarak dari titik  $j$  ke titik  $i$ .
3. Kendaraan yang digunakan untuk pendistribusian mempunyai tipe yang sama, artinya kapasitas kendaraan yang digunakan sama dengan kecepatan konstan rata-rata 30 km/jam.
4. Biaya bahan bakar Rp1.000,00/km
5. Setiap kendaraan berangkat pada waktu yang sama yaitu 07.00.
6. Tidak terjadi kemacetan, kondisi jalan tidak rusak serta kendaraan dalam kondisi bagus.
7. Kendaraan akan menunggu jika waktu sampai tidak sesuai dengan waktu buka pelanggan dan akan memperoleh penalti keterlambatan jika kendaraan sampai melebihi waktu tutup pelanggan.
8. Setiap pelanggan dikunjungi tepat satu kali oleh satu kendaraan.

Tahapan pertama dalam pemodelan adalah mendefinisikan himpunan, parameter, dan variabel keputusan yang akan digunakan pada model optimisasi ini.

Berikut adalah himpunan dan parameter yang digunakan:

- $N$  : Himpunan pelanggan  
 $D$  : Himpunan depot  
 $K$  : Himpunan kendaraan  
 $C_k$  : Kapasitas kendaraan ke- $k$   
 $q_i$  : Permintaan pelanggan ke- $i$   
 $z_i$  : *Time windows* pelanggan ke- $i$   
 $s_i$  : Waktu pelayanan pelanggan ke- $i$   
 $t_{ij}$  : Waktu tempuh dari pelanggan ke- $i$  ke pelanggan ke- $j$   
 $c_{ij}$  : Biaya distribusi dari pelanggan ke- $i$  ke pelanggan ke- $j$

dengan variabel keputusan sebagai berikut:

1. Variabel  $X_{ijk}$ ,  $\forall i, j \in N, \forall k \in K, i \neq j$ .

Variabel  $x_{ijk}$  didefinisikan untuk mengetahui ada atau tidaknya perjalanan dari pelanggan ke- $i$  ke pelanggan ke- $j$  oleh kendaraan ke- $k$ . Dengan demikian, variabel keputusan ini didefinisikan sebagai berikut.

$$X_{ijk} = \begin{cases} 1, & \text{jika terdapat perjalanan dari pelanggan } i \text{ ke } j \text{ oleh kendaraan } k \\ 0, & \text{yang lainnya.} \end{cases}$$

2. Variabel  $T_{ik}$ ,  $\forall i \in N, \forall k \in K$ .

Variabel  $T_{ik}$  menyatakan waktu dimulainya pelayanan pada konsumen ke- $i$  oleh kendaraan ke- $k$ .

3. Variabel  $Y_{ik}$ ,  $\forall i, j \in N, \forall k \in K$ .

Variabel  $Y_{ik}$  menyatakan kapasitas total kendaraan ke- $k$  setelah melayani konsumen ke- $i$ .

4. Variabel  $M_i$ ,  $\forall i \in N$

Variabel  $M_i$  menyatakan apakah pelanggan ke- $i$  terlewat. Dengan demikian, variabel keputusan ini didefinisikan sebagai berikut.

$$M_i = \begin{cases} 1, & \text{jika pelanggan } i \text{ terlewat} \\ 0, & \text{yang lainnya.} \end{cases}$$

Masalah CVRPTW dapat diformulasikan ke dalam bentuk model matematika dengan tujuan meminimumkan total waktu distribusi, meminimumkan jumlah pelanggan yang terlewat, dan meminimumkan total biaya dalam melayani semua pelanggan. Jika  $z$  adalah fungsi tujuan untuk masalah CVRPTW, maka:

$$\min Z_1 = \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^K t_{ij} X_{ijk}$$

$$\min Z_2 = \sum_{i=1}^N M_i$$

$$\min Z_3 = \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^K c_{ij} X_{ijk}$$

Fungsi tujuan multi objektif dikombinasikan menjadi satu fungsi tujuan dengan metode pembobotan (*Weighted Sum Model*) sebagai berikut:

$$\min Z = w_1 \cdot Z_1 + w_2 \cdot Z_2 + w_3 \cdot Z_3$$

dengan  $w_1$ ,  $w_2$ , dan  $w_3$  adalah bobot masing-masing tujuan .

Kendala-kendala dari model matematika pada permasalahan CVRPTW multi objektif adalah sebagai berikut:

1. Setiap pelanggan hanya dikunjungi tepat satu kali oleh kendaraan yang sama.

$$\sum_{k=1}^K \sum_{j=1}^N X_{i,j}^k = 1$$

2. Total jumlah permintaan pelanggan dalam satu rute tidak melebihi kapasitas kendaraan yang melayani rute tersebut. Misalkan terdapat lintasan dari  $i$  ke  $j$  dengan kendaraan  $k$ , maka:

$$Y_{ik} + q_j = Y_{jk}, \forall i, j \in N, \forall k \in K.$$

$$Y_{jk} \leq C_k, \forall j \in N, \forall k \in K.$$

3. Waktu memulai pelayanan di pelanggan ke- $j$  lebih dari atau sama dengan waktu kendaraan ke- $k$  memulai pelayanan di pelanggan ke- $i$  ditambah waktu tempuh perjalanan dari pelanggan ke- $i$  ke pelanggan ke- $j$ .

$$T_{ik} + s_{ik} + t_{ij} \leq T_{jk}, \forall i, j \in N, \forall k \in K.$$

4. Waktu kendaraan untuk memulai pelayanan di pelanggan ke- $i$  harus berada pada selang waktu  $[e_i, l_i]$ .

$$e_i \leq T_{ik} \leq l_i, \forall i \in N, \forall k \in K.$$

5. Kekontinuan rute, artinya kendaraan yang mengunjungi setiap pelanggan akan meninggalkan pelanggan tersebut setelah selesai waktu pelayanan.

$$\sum_{i=1}^N x_{ijk} - \sum_{i=1}^N x_{ijk} = 0, \forall i, j \in N, \forall k \in K.$$

6. Variabel keputusan  $X_{ijk}$  merupakan integer biner

$$X_{ijk} \in \{0,1\}, \forall i, j \in N, \forall k \in K.$$

Selengkapnya model CVRPTW dituliskan sebagai model optimasi berikut:

Minimumkan

$$Z = w_1 \cdot Z_1 + w_2 \cdot Z_2 + w_3 \cdot Z_3 \quad (3.1)$$

terhadap :

$$\sum_{k=1}^K \sum_{j=1}^N X_{i,j}^k = 1 \quad (3.2)$$

$$Y_{ik} + q_j = Y_{jk}, \forall i, j \in N, \forall k \in K \quad (3.3)$$

$$Y_{jk} \leq C_k, \forall j \in N, \forall k \in K$$

$$T_{ik} + S_{ik} + t_{ij} \leq T_{jk}, \forall i, j \in N, \forall k \in K \quad (3.4)$$

$$e_i \leq T_{ik} \leq l_i, \forall i \in N, \forall k \in K \quad (3.5)$$

$$\sum_{i=1}^N X_{ijk} - \sum_{i=1}^N X_{ijk} = 0, \forall i, j \in N, \forall k \in K \quad (3.6)$$

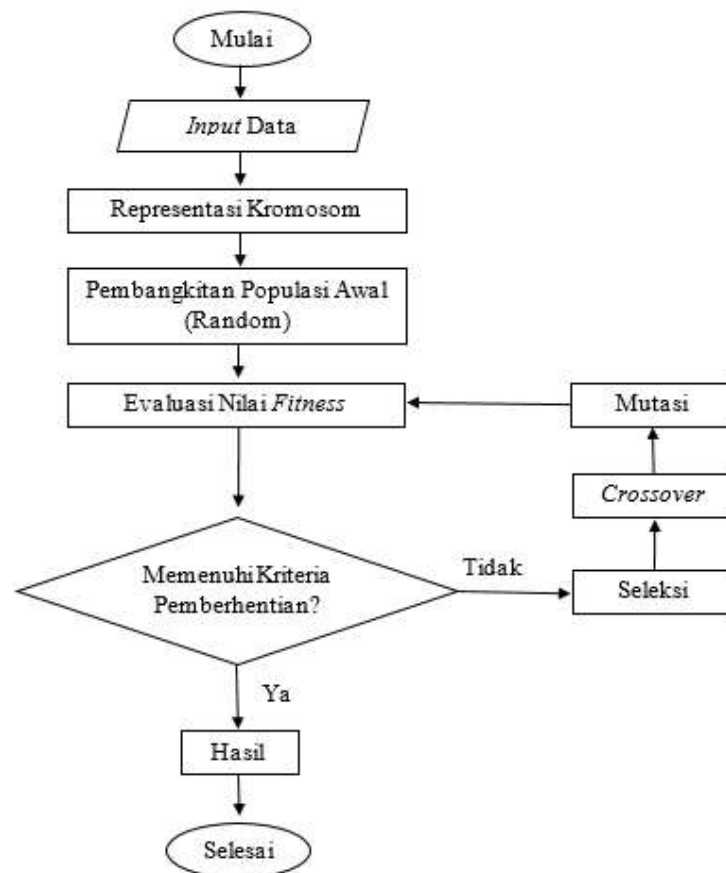
$$X_{ijk} \in \{0,1\}, \forall i, j \in N, \forall k \in K \quad (3.7)$$

Dengan demikian, model ini mengoptimalkan tiga tujuan utama dengan menggunakan metode WSM untuk menggabungkannya menjadi satu fungsi tujuan yang mempertimbangkan total waktu, jumlah pelanggan yang terlewat, dan total biaya bahan bakar pada pendistribusian.

### 3.5 Teknik Penyelesaian

Permasalahan CVRPTW termasuk dalam kategori *NP-hard problem*, yang artinya solusi *feasible* yang dihasilkan akan semakin banyak seiring dengan bertambahnya variabel, sehingga akan membutuhkan waktu yang lama jika menggunakan solusi eksak. Oleh karena itu, diperlukan metode pendekatan untuk menyelesaikan masalah CVRPTW secara efisien. Dalam penelitian ini, metode pendekatan yang digunakan adalah Algoritma Genetika dengan WSM.

Algoritma Genetika digunakan untuk mengoptimisasi rute pendistribusian beras bersubsidi sehingga diperoleh rute distribusi yang optimal. Pada Algoritma Genetika, solusi dari permasalahan yang dihadapi dikodekan sebagai kromosom-kromosom. Tahap awal yang dilakukan adalah dengan membangkitkan secara acak kromosom-kromosom tersebut dalam sebuah populasi. Individu-individu dalam suatu populasi dianalogikan sebagai himpunan-himpunan solusi yang mungkin dalam suatu permasalahan optimisasi. Kemudian, himpunan-himpunan solusi dievaluasi tingkat penyesuaian diri terhadap lingkungannya dengan nilai *fitness*. Nilai *fitness* ini berhubungan erat dengan fungsi tujuan dari permasalahan yang akan diselesaikan. Pada masalah minimasi, semakin besar nilai *fitness* suatu individu maka akan semakin besar kesempatannya untuk bertahan dalam suatu populasi. Setelah mengevaluasi nilai *fitness* dilakukan proses seleksi menggunakan metode *roulette wheel selection*. Pada proses ini, individu dengan nilai *fitness* terbesar memiliki peluang yang lebih tinggi untuk dipilih sebagai induk pada proses *crossover* dan dipilih untuk proses mutasi. Pada penelitian ini digunakan PMX *crossover* dan tiga jenis mutasi yang berbeda yaitu *swapping mutatio*, *inversion mutation*, dan *insert mutation* untuk melihat mutasi mana yang menghasilkan rute paling optimal. Individu baru hasil *crossover* dan mutasi akan digabungkan dengan beberapa individu terbaik yang dipertahankan dari populasi saat ini kemudian dievaluasi nilai *fitness*-nya. Setelah itu, dipilih individu dengan nilai *fitness* terbaik sebanyak ukuran populasi sebelumnya, dan kembali lagi diseleksi pada generasi selanjutnya. Proses tersebut akan terus berjalan sampai syarat berhenti dipenuhi. Tahapan dalam Algoritma Genetika dapat dilihat pada Gambar 3.1 berikut.



Gambar 3. 1 Tahapan Algoritma Genetika

Dengan demikian, urutan langkah-langkah penyelesaian masalah CVRPTW dengan menggunakan Algoritma Genetika adalah sebagai berikut :

### 1. Representasi Kromosom

Representasi kromosom adalah proses pengkodean dari penyelesaian masalah. Pada masalah CVRPTW, satu kromosom merepresentasikan satu solusi. Dalam hal ini, satu kromosom merupakan suatu rute pendistribusian. Satu kromosom memuat sejumlah gen di mana satu gen mewakili satu pelanggan, sehingga panjang kromosom sama dengan banyaknya pelanggan. Pada penelitian ini, kromosom direpresentasikan menggunakan *permutation encoding*, di mana representasi kromosom dari teknik ini adalah kumpulan dari nilai integer yang mewakili suatu posisi dalam sebuah urutan.



Representasi kromosom menggunakan *permutation encoding* dapat dilihat pada Gambar 3.2 berikut ini:

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Gambar 3. 2 Representasi Kromosom

Pada Gambar 3.2 pelanggan direpresentasikan sebagai bilangan integer dari 1 – 10, yang berarti rute pendistribusian dari depot akan dimulai dengan urutan 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8 – 9 – 10.

## 2. Inisialisasi Parameter

Algoritma Genetika bekerja bergantung pada beberapa nilai parameter yang harus ditetapkan diawal proses. Parameter-parameter tersebut adalah sebagai berikut:

### a. Banyaknya generasi (iterasi)

Banyaknya jumlah generasi pada satu siklus reproduksi Algoritma Genetika berupa seleksi, *crossover*, dan mutasi yang setiap generasinya menghasilkan solusi permasalahan.

### b. Ukuran populasi awal

Banyaknya jumlah individu solusi yang akan diproses bersama dalam satu satuan siklus evolusi.

### c. *Crossover rate*

Parameter penentu jumlah individu dalam populasi yang akan mengalami *crossover*. Parameter yang disarankan berada pada *range* 0,6 – 0,7. Pemilihan *crossover rate* yang terlalu besar menyebabkan eksplorasi yang berlebihan karena algoritma terlalu fokus pada eksplorasi ruang solusi dengan menggabungkan banyak materi genetik dari induk yang berbeda dan akan memperlambat konvergensi sehingga membutuhkan waktu yang lebih lama untuk mencapai solusi yang optimal. Apabila *crossover rate* yang dipilih terlalu kecil akan menyebabkan stagnasi atau populasi terlalu homogen dan tidak mengalami banyak perubahan dari generasi ke generasi.

### d. *Mutation rate*

Parameter penentu jumlah individu yang akan mengalami mutasi. Parameter yang disarankan berada pada *range* 0,01 – 0,3. Pemilihan *mutation rate* yang terlalu besar menyebabkan perubahan yang drastis pada individu dan pemilihan

*mutation rate* yang terlalu kecil menyebabkan peluang individu yg dimutasi sangat kecil sehingga individu akan cenderung mirip dengan induknya.

### 3. Pembangkitan Populasi Awal (*Spanning*)

Pada tahap ini, populasi awal dibangkitkan secara acak untuk membentuk suatu kesatuan individu. Setiap individu yang dibangkitkan dibentuk dalam representasi permutasi dengan jumlah kromosom pada suatu individu diasumsikan sebanyak jumlah pelanggan yang akan dikunjungi. Pembangkitan individu disertai dengan pembangkitan kapasitas individu. Pembangkitan kapasitas individu memiliki tujuan untuk pembagian rute berdasarkan kapasitas kendaraan pengangkut yang digunakan sehingga memenuhi kendala yang diberikan. Setiap kendaraan  $k$  akan mendistribusikan beras bersubsidi untuk memenuhi permintaan pelanggan. Satu kromosom akan menghasilkan rute bagi setiap kendaraan. Untuk mendapatkan rute yang dilalui oleh setiap kendaraan pada satu kromosom dilakukan langkah-langkah sebagai berikut. Misalkan elemen ke- $i$  dari suatu kromosom dengan panjang  $n$  adalah  $g_i$ .

1. Inisialisasi rute pertama  $k_1$  dengan memasukan gen urutan pertama, menjadi  $k_1 = \{g_1\}$ .
2. Hitung total permintaan gen di  $k_1$ .
3. Jika total permintaan gen kurang dari atau sama dengan kapasitas  $k_i, i = 1,2,3,4,5$  maka tambahkan gen urutan selanjutnya  $g_{i+1}$ , jika tidak maka kurangi gen urutan terakhir yang dimasukan dari  $k_i$  dan masukan gen tersebut ke dalam rute baru  $k_{i+1}$ .

lakukan langkah ke tiga sampai semua gen dalam satu kromosom sudah terpilih semua. Setelah semua individu dalam populasi awal memenuhi kendala yang diberikan, maka proses Algoritma Genetika akan dilanjutkan ke proses evaluasi nilai *fitness*.

### 4. Evaluasi Nilai *Fitness*

Setelah dilakukan pembangkitan populasi awal, langkah selanjutnya yaitu menentukan nilai *fitness* dari setiap individu. Nilai *fitness* digunakan untuk mengukur kelayakan sebuah individu untuk dipertahankan atau tidak. Penggunaan nilai *fitness* bertujuan untuk menentukan rute pendistribusian dengan meminimumkan total waktu, jumlah pelanggan yang terlewat, dan total biaya bahan

bakar. Dalam perhitungan nilai *fitness* terdapat total jarak (km) yang diakumulasikan dalam satuan waktu (menit) dengan persamaan berikut:

$$\text{waktu tempuh} = \frac{\text{jarak (km)}}{30} \times 60 \quad (3.8)$$

Pada penelitian ini, akan dihitung nilai *fitness* tiap kromosom sesuai dengan kendaraan yang digunakan. Nilai *fitness* suatu individu merupakan hasil penjumlahan dari semua nilai *fitness* kromosom setiap kendaraan dalam individu tersebut, dan nilai *fitness* diperoleh dengan rumus berikut.

$$\text{fitness} = \frac{1}{f(x) + c} \quad (3.9)$$

dengan nilai  $f(x)$  merupakan nilai dari fungsi tujuan, yaitu total waktu distribusi yang dijumlahkan dengan waktu penalti dan waktu tunggu, total pelanggan yang terlewat, serta total biaya bahan bakar yang dikalikan dengan bobot dari masing-masing fungsi tujuan. Waktu penalti yaitu waktu pelayanan yang melebihi jangka waktu yang tersedia, dan  $c$  adalah bilangan yang sangat kecil yang sudah ditentukan sebelumnya untuk menghindari masalah ketika  $f(x)$  bernilai 0.

Nilai dari total waktu distribusi, jumlah pelanggan yang terlewat, dan total biaya bahan bakar berada pada rentang nilai yang berbeda sehingga dilakukan normalisasi data terlebih dahulu untuk menghindari dominasi dari salah satu tujuan. Normalisasi dilakukan dengan membagi nilai yang diperoleh dari masing-masing tujuan oleh nilai maksimum dari masing-masing tujuan tersebut. Misalkan nilai maksimum untuk total waktu distribusi adalah 2000 menit, nilai maksimum untuk jumlah pelanggan yang terlewat adalah 10, dan nilai maksimum untuk total biaya adalah Rp200.000,00.

Sehingga langkah pertama yang dilakukan untuk menghitung nilai *fitness* adalah membagi pelanggan ke dalam kendaraan sesuai dengan kapasitas pelanggan seperti pada Gambar 3.3 berikut.

P1	1	2	3	4	5	6	7	8	9	10
Pr	180	190	160	160	150	155	190	180	155	180
TP	840					860				
K	1					2				

Gambar 3. 3 Pembagian Rute Berdasarkan Kapasitas Kendaraan

Keterangan istilah yang digunakan pada Gambar 3.3 :

P1 : Kromosom yang digunakan sebagai contoh perhitungan

PP : Permintaan pelanggan

TP : Total permintaan yang disesuaikan dengan kendaraan

K : Kendaraan yang digunakan.

Tahapan selanjutnya untuk menghitung nilai *fitness* yaitu menghitung total waktu, pelanggan yang terlewat, dan biaya bahan bakar dari setiap kendaraan yang digunakan, seperti pada Tabel 3.1 – Tabel 3.2 berikut.

Tabel 3. 1 Tujuan Kendaraan 1

Kendaraan 1									
Pelanggan	Waktu (menit)	Sampai	Buka	Tunggu	Mulai	Layanan	Selesai	Tutup	Penalti
1	18	07.18	08.00	42	08.00	100	09.40	13.00	0
2	28	10.08	07.00	0	10.08	94	11.42	12.30	0
3	64	12.06	09.00	0	12.46	105	14.31	13.30	61
4	14	14.45	08.30	0	0	0	14.45	14.30	15
5	46	15.31	07.30	0	0	0	15.31	14.00	91
Depot	45	16.17	0	0	0	0	16.17	0	0
<b>Total</b>							<b>557</b>		<b>167</b>
<b>Total Waktu (Menit)</b>							<b>724</b>		
<b>Pelanggan Terlewat</b>							<b>2</b>		
<b>Total Jarak (km)</b>							<b>88</b>		
<b>Total Biaya Bahan bakar</b>							<b>88000</b>		

Tabel 3. 2 Tujuan Kendaraan 2

Kendaraan 2									
Pelanggan	Waktu (menit)	Sampai	Buka	Tunggu	Mulai	Layanan	Selesai	Tutup	Penalti
6	22	07.22	13.00	348	13.00	94	14.34	16.00	0
7	36	15.10	07.30	0	0	0	15.10	14.00	70
8	22	15.32	08.00	0	15.32	113	17.25	17.30	0
9	16	17.42	08.00	0	0	0	17.42	14.00	222
10	40	18.22	10.00	0	0	0	18.22	15.30	112
Depot	14	18.36	0	0	0	0	18.36	0	0
<b>Total</b>							<b>696</b>		<b>404</b>
<b>Total Waktu (Menit)</b>							<b>1100</b>		
<b>Pelanggan Terlewat</b>							<b>3</b>		
<b>Total Jarak (km)</b>							<b>75</b>		
<b>Total Biaya Bahan bakar</b>							<b>75000</b>		

Setelah total waktu, pelanggan terlewat dan biaya bahan bakar diketahui, selanjutnya dilakukan normalisasi pada nilai tersebut seperti pada tabel 3.3 berikut.

Tabel 3. 3 Normalisasi Nilai Masing-Masing Tujuan

Hasil	Nilai Masing-Masing Tujuan					
	Total Waktu Distribusi (menit)		Jumlah pelanggan Terlewat		Total Biaya Bahan Bakar (Rp)	
	Kendaraan 1	Kendaraan 2	Kendaraan 1	Kendaraan 2	Kendaraan 1	Kendaraan 2
	724	1100	2	3	88000	75000
Nilai Maksimum	2000		10		200000	
Nilai Hasil Normalisasi	0,362	0,55	0,2	0,3	0,44	0,375

Setelah normalisasi nilai dari masing-masing tujuan, selanjutnya adalah menghitung nilai *fitness* dengan menggunakan kombinasi bobot  $W_1 = 0,33$ ;  $W_2 = 0,33$ ;  $W_3 = 0,33$  pada setiap fungsi tujuan seperti pada Tabel 3.4 berikut.

Tabel 3. 4 Menghitung Nilai *Fitness*

Kendaraan	Nilai Tujuan yang Dinormalisasi				Nilai <i>Fitness</i>
	Bobot	Total Waktu (menit)	Pelanggan Terlewat	Total biaya	
1	0,33	0,362	0,2	0,44	0,11946
2	0,33	0,55	0,3	0,375	0,1815
<i>Fitness P1</i>					0,30096

Setelah menghitung nilai *fitness* dari masing-masing individu, maka diperoleh nilai *fitness* terbaik dari populasi awal. Individu dengan nilai *fitness* terbaik dari populasi generasi pertama akan dipertahankan dan dibawa ke generasi selanjutnya. Langkah selanjutnya yaitu melakukan seleksi untuk menentukan individu sebagai induk.

## 5. Seleksi

Proses selanjutnya yaitu seleksi untuk memilih individu terbaik dari populasi untuk dimasukkan ke dalam generasi berikutnya. Individu yang terpilih pada proses seleksi akan menjadi induk pada proses selanjutnya. Dalam penelitian ini, metode seleksi yang digunakan adalah metode *roulette wheel selection*.

Metode *roulette wheel selection* dianalogikan seperti permainan roda putar. Pada permainan roda putar, lingkaran roda dibagi menjadi beberapa wilayah. Lebar suatu wilayah kromosom ditentukan menurut nilai *fitness*-nya. Semakin besar nilai *fitness* maka luas wilayahnya juga akan semakin besar dan peluang kromosom

untuk terpilih sebagai induk juga besar. Tahapan metode seleksi *roulette wheel* adalah sebagai berikut:

- Hitung nilai *fitness* masing-masing individu ( $f_i$ , dimana  $i$  adalah individu ke-1 sampai dengan ke- $n$ )
- Hitung total nilai *fitness* semua individu
- Hitung probabilitas masing-masing individu, dengan rumus probabilitas seleksi yaitu:

$$P_i = \frac{f_i}{\sum_{i=1}^n f_i} \times 100\%$$

Dari perhitungan, diperoleh nilai untuk setiap individu pada angka 1 sampai dengan 100.

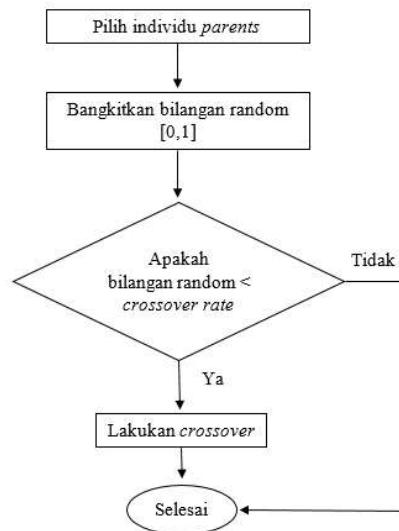
- Bangkitkan bilangan acak antara 1 sampai dengan 100.

Dari bilangan acak yang dihasilkan, dapat ditentukan individu mana yang terpilih dalam proses seleksi.

## 6. Kawin Silang (*Crossover*)

Setelah terpilih induk-induk dari proses seleksi, selanjutnya dilakukan proses *crossover* kepada induk-induk tersebut secara bergantian. *Crossover* pada dua induk akan menghasilkan individu baru yang disebut anak.

Tahapan dalam proses *crossover* dapat diilustrasikan seperti Gambar 3.4 berikut.



Gambar 3. 4 Proses *Crossover*

Pada penelitian ini menggunakan *Partially-Mapped Crossover* (PMX) yang dapat mencegah adanya keturunan mirip dengan munculnya gen ganda yang terdapat pada suatu individu. Langkah-langkah dalam metode PMX dapat dilihat pada Gambar 3.5 – 3.8 (Azmi dkk., 2012) :

- a. Tentukan dua titik secara random untuk membagi induk. Di antara dua titik tersebut akan terdapat area pemetaan yang disebut dengan *mapping*.

Induk 1	5	2	6	7	8	9	1	10	4	3
Induk 2	1	2	3	4	5	6	7	8	9	10

Gambar 3. 5 Area Pemetaan PMX

- b. Lakukan penukaran *substring* antara induk 1 dan induk 2, *substring* yang ditukar adalah *substring* yang berada dalam area *mapping*, maka akan terbentuk dua *ProtoChild*.

<i>ProtoChild 1</i>	5	2	6	4	5	6	7	10	4	3
<i>ProtoChild 2</i>	1	2	3	7	8	9	1	8	9	10

Gambar 3. 6 Hasil *Protochild* PMX

- c. Definisikan hubungan pemetaan yang berada dalam area *mapping*. Pemetaan dilakukan untuk mengganti nilai 4,5,6,7 menjadi 7,8,9,1.

4	5	6	7
↓	↓	↓	↓
7	8	9	1

Gambar 3. 7 Pemetaan PMX

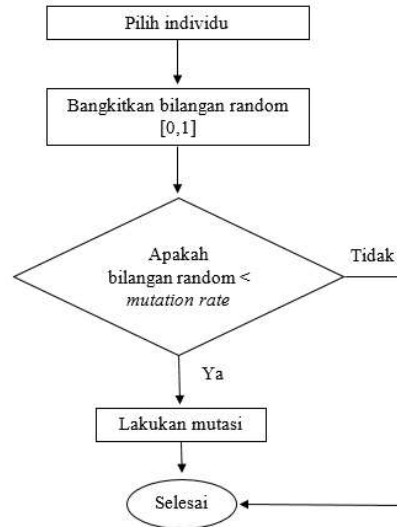
- d. Lakukan pemetaan kromosom dari awal sesuai *ProtoChild 1* dan *ProtoChild 2*, pertahankan *substring* yang berada dalam area pemetaan, jika terdapat angka yang sama pada area dalam dan luar pemetaan, maka angka yang harus diganti adalah angka yang berada diluar area pemetaan. Pergantian angka sesuai dengan hasil definisi hubungan pemetaan sebelumnya hingga menghasilkan keturunan dengan gen yang berbeda dengan induk.

<i>Child 1</i>	8	2	9	4	5	6	7	10	1	3
<i>Child 2</i>	4	2	3	7	8	9	1	5	6	10

Gambar 3. 8 Hasil Keturunan PMX

## 7. Mutasi

Setelah proses *crossover* maka selanjutnya adalah mutasi gen. Proses mutasi dilakukan dengan tujuan untuk memperoleh individu baru sebagai kandidat solusi pada generasi selanjutnya dengan nilai *fitness* yang lebih baik dan lama-kelamaan akan menuju solusi optimal yang diinginkan. Tahapan dalam proses mutasi dapat diilustrasikan seperti Gambar 3.9 berikut.

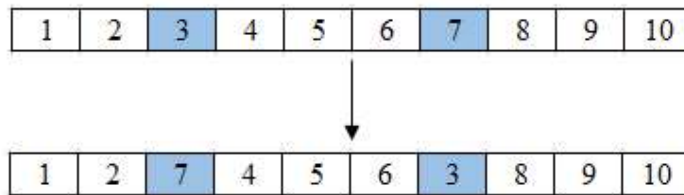


Gambar 3. 9 Proses Mutasi

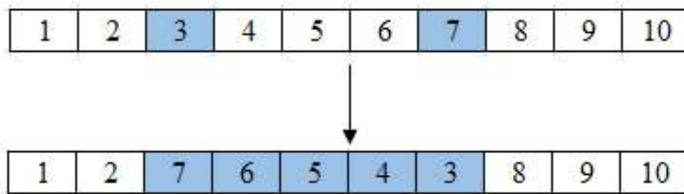
Dalam proses mutasi terdapat istilah *mutation rate*. Pada penelitian ini, nilai *mutation rate* akan menentukan banyaknya individu yang akan dimutasi. Jenis mutasi yang digunakan dalam penelitian ini yaitu *swapping mutation*, *inversion mutation*, dan *insert mutation*. Penggunaan ketiga variasi tersebut bertujuan untuk melihat mutasi mana yang memberikan hasil terbaik. Gen yang akan dimutasi ditentukan dengan membangkitkan dua buah nilai random  $[0, \text{pelanggan}]$ . Pada *swapping mutation*, dua buah nilai random yang dibangkitkan ditukar posisinya. Pada *inversion mutation*, pelanggan yang berada dalam rentang dua buah nilai random yang dibangkitkan dibalik posisinya. Sedangkan pada *insert mutation*, posisi pelanggan yg berada pada titik nilai random pertama akan mengisi posisi pelanggan yg berada pada titik nilai random kedua, pelanggan setelah pelanggan posisi nilai random pertama akan mengisi posisi nilai random pertama dan seterusnya sampai pada posisi nilai random kedua. Ilustrasi *swapping mutation*,



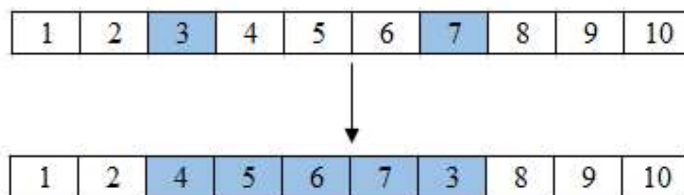
*inversion mutation*, dan *insert mutation* dapat dilihat pada Gambar 3.10 – 3.12 sebagai berikut.



Gambar 3. 10 *Swapping Mutation*



Gambar 3. 11 *Inversion Mutation*



Gambar 3. 12 *Insert Mutation*

Setelah menyelesaikan proses mutasi maka satu generasi atau iterasi telah terselesaikan sehingga diperoleh kromosom baru hasil seleksi, *crossover*, dan mutasi. Setiap kromosom akan dihitung nilai *fitness*-nya untuk dipilih sebagai populasi baru generasi selanjutnya. Untuk mendapatkan nilai *fitness* yang lebih baik maka tahapan generasi akan terus diulang sampai banyaknya generasi dan ukuran populasi yang ditentukan.

## 8. Pembentukan Populasi Baru

Tahap selanjutnya yaitu pembentukan populasi baru. Individu baru hasil *crossover* dan mutasi akan digabung dengan beberapa individu terbaik yang dipertahankan dari populasi saat ini kemudian dievaluasi nilai *fitness*-nya. Selanjutnya dipilih individu dengan nilai *fitness* terbaik sebanyak ukuran populasi untuk dipertahankan dan dibawa ke generasi selanjutnya.

## 9. Kriteria Pemberhentian

Algoritma Genetika akan terus berjalan dari generasi ke generasi sampai kriteria pemberhentian dipenuhi. Pada penelitian ini, kriteria pemberhentian yang akan digunakan adalah banyak generasi, di mana Algoritma Genetika akan berhenti setelah mencapai banyak generasi yang sudah ditentukan sebelumnya.