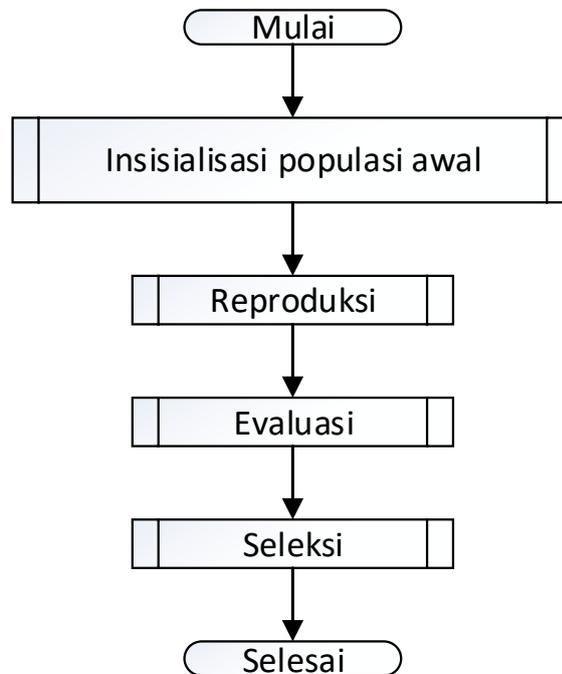


BAB 4 PERANCANGAN

Pada bab ini membahas mengenai perancangan aplikasi sistem pengiriman barang pada perusahaan J&T EXPRESS menggunakan Algoritme Genetika. Perancangan ini dilakukan meliputi tiga tahap, yaitu proses perancangan sistem, proses perhitungan manual, perancangan UI (*User Interface*) serta perancangan pengujian.

4.1 Siklus Algoritme Genetika

Proses untuk menemukan sebuah rute pengiriman barang yang optimum dapat dilakukan dengan mengolah data yang menggunakan metode algoritme genetika. Algoritme genetika dapat menjadi solusi alternatif untuk perhitungan matematika yang kompleks. Algoritme Genetika memiliki alur proses utama yaitu Inisialisasi Kromosom, Reproduksi, Evaluasi dan seleksi. Proses tersebut akan ditampilkan pada Gambar 4.1



Gambar 4.1 Flowchart Algoritme Genetika

Pada Gambar 4.1 dapat dilihat bahwa dalam penerapan Algoritme Genetika untuk pengiriman barang, perlu melalui beberapa langkah, yaitu:

1. Inisialisasi Parameter

Parameter yang digunakan pada proses algoritme genetika sendiri ada 2 jenis, yaitu yang pertama adalah parameter yang digunakan untuk menyelesaikan permasalahan dari pengiriman barang perusahaan. Kemudian yang ke-dua adalah

parameter yang digunakan atau dibutuhkan oleh proses Algoritme genetika itu sendiri.

Untuk parameter pertama kita membutuhkan 3 data, yaitu:

- a. Data alamat dari perusahaan dan alamat tujuan pengiriman yang nantinya digunakan untuk mencari waktu tempuh serta tingkat kemacetan yang ada.
- b. Waktu tempuh antar alamat perusahaan dengan alamat tujuan pengiriman serta waktu tempuh antar alamat tujuan pengiriman yang ada.
- c. Tingkat kemacetan antar alamat perusahaan dengan alamat tujuan pengiriman serta tingkat kemacetan antar alamat tujuan pengiriman yang ada.

Sedangkan untuk parameter ke-dua, yaitu parameter untuk menyelesaikan proses Algoritme genetika, sebagai berikut:

- a. Populasi (*PopSize*) merupakan sebuah penampungan atau himpunan dari solusi baru yang terdiri dari sejumlah individu atau kromosom (Mahmudy, 2015). Populasi dibutuhkan pada saat proses reproduksi dan seleksi.
- b. *Crossover Rate* (*cr*) merupakan sebuah bilangan acak yang digunakan untuk menentukan jumlah *offspring* pada proses reproduksi *crossover*. Semakin tinggi nilai *cr* yang digunakan semakin banyak proses *crossover* yang dilakukan.
- c. *Mutation Rate* (*mr*) merupakan sebuah bilangan acak yang digunakan untuk menentukan jumlah *offspring* pada proses reproduksi *crossover*. Semakin tinggi nilai *cr* yang digunakan semakin banyak proses *crossover* yang dilakukan.
- d. Generasi merupakan banyaknya iterasi atau generasi yang terjadi pada proses perhitungan algoritme genetika, biasanya digunakan untuk mengetahui perubahan hasil yang diperoleh.

2. Inisialisasi Populasi Awal

Inisialisasi populasi awal adalah sebuah proses yang dilakukan untuk menentukan kumpulan individu individu yang dibentuk dalam sebuah kromosom. Kromosom sendiri dibentuk dengan memanfaatkan fungsi *random* yang ada pada *java*, sehingga individu yang sudah terpilih tidak terpilih lagi.

3. Reproduksi

Proses reproduksi sendiri dibagi menjadi dua yaitu *Crossover* dan *Mutation*.

- a. *Crossover* merupakan sebuah proses untuk menemukan kromosom baru, dengan menukarkan gen pada beberapa kromosom sehingga menghasilkan kromosom baru. Banyaknya proses ini sendiri dilakukan berdasarkan nilai *Crossover Rate*.
- b. *Mutation* merupakan proses yang dilakukan untuk menghasilkan kromosom baru hanya dengan memilih salah satu dari kromosom awal. Proses perubahan gen pada kromosom dilakukan dengan menukarkan posisi dari beberapa gen atau melakukan penggeseran untuk beberapa gen.

4. Evaluasi

Proses ini dilakukan dengan menggabungkan kromosom awal dengan kromosom baru hasil dari reproduksi. Setelah itu menghitung nilai *fitness*, nilai *fitness* pada sistem ini menggunakan MTSPTW untuk memisiah gen pada kromosom dengan sejumlah *sales* yang ada.

5. Seleksi

Pada proses ini dilakukan dengan mengurutkan hasil dari perhitungan nilai *fitness* pada proses evaluasi. Sehingga kromosom dengan nilai *fitness* yang lebih baik akan diproses untuk generasi selanjutnya, dan untuk kromosom dengan nilai *fitness* yang kurang baik akan dihapus. Bisa dikatakan proses ini adalah proses pembentukan populasi baru, dengan kromosom yang memiliki nilai *fitness* terbaik.

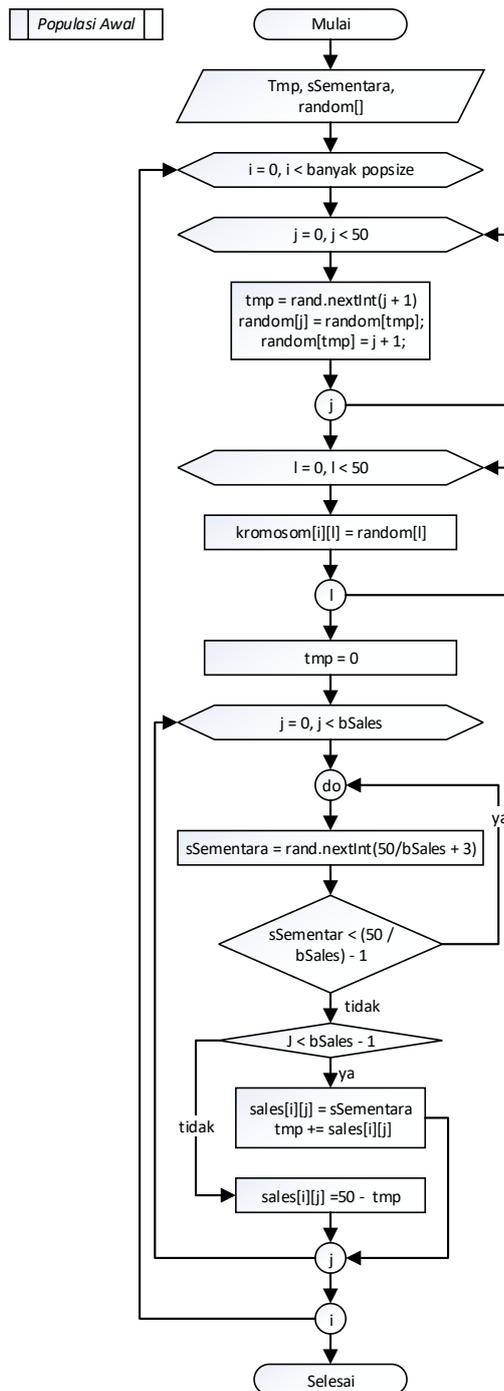
4.2 Proses Perhitungan Manual

Pada penelitian yang akan diselesaikan yaitu pada permasalahan pengiriman barang yang dilakukan pada perusahaan ekspedisi J&T *Express* di kota Surabaya dengan menggunakan algoritme genetika. Tujuan dari penelitian ini adalah untuk mendapatkan hasil rute paling minimal dengan mempertimbangkan waktu perjalanan pada proses pengiriman barang. Proses pengiriman barang yang dilakukan oleh perusahaan ekspedisi J&T *Express* menggunakan lebih dari satu *sales* karena banyaknya pelanggan yang harus dikunjungi dan terdapat pertimbangan waktu, sehingga masalah yang akan diselesaikan merupakan masalah *Multiple Travelling Salesman Problem (M-TSP)*. Namun pada penelitian ini terdapat pertimbangan waktu perjalanan sehingga adanya penambahan variabel *Time Window*.

Penelitian ini menggunakan metode Algoritme genetika dikarenakan metode ini mempunyai kemampuan dalam memecahkan suatu masalah dengan formulasi matematika yang kompleks untuk memberikan solusi yang mendekati optimum. Dalam perhitungan manual yang telah dilakukan, dapat membuktikan bahwa Algoritme genetika yang digabungkan dengan MTSPTW bisa digunakan untuk menentukan rute dari pengiriman barang, berdasarkan jarak atau waktu tempuh dan tingkat kemacetan yang ada di kota Surabaya.

4.2.1 Inisialisasi Populasi Awal

Inisialisasi populasi awal merupakan salah satu proses dalam algoritme genetika yang terjadi pembentukan himpunan individu atau kromosom baru pada sebuah populasi yang terbentuk sebanyak *popSize* yang telah ditentukan. Pada inisialisasi populasi, individu pada populasi merupakan alternatif solusi dari sebuah permasalahan.



Gambar 4.2 Diagram Alir Inisialisasi Populasi Awal

Setiap individu atau kromosom pada penelitian ini menggunakan representasi permutasi yaitu representasi berupa angka-angka *integer* untuk menyatakan nomor pada setiap simpul yang merepresentasikan nomor tiap wilayah yang dituju. Individu terbentuk dari beberapa gen yang merupakan urutan dari rute-rute yang akan dikunjungi untuk proses pengiriman barang. Terdapat dua segmen pada setiap individu atau kromosom, yaitu segmen pertama urutan

rute pelanggan yang selalu dimulai dari kantor pusat dan yang kedua jumlah yang harus dikunjungi setiap *sales*.

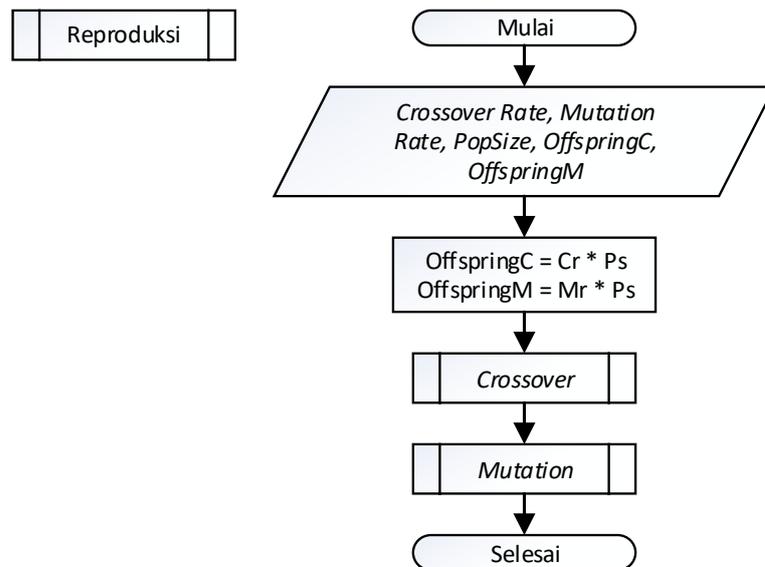
Individu dalam populasi merupakan pembentukan secara *random* sehingga hasil yang diperoleh selalu berbeda-beda, hal ini terjadi karena algoritme genetika memiliki sifat *stochastic* (Mahmudy, 2015). Hasil atau solusi yang didapatkan dari perhitungan algoritme genetika memang selalu berubah-ubah, tetapi hasil akhir tersebut adalah alternatif solusi dari setiap generasi yang merupakan individu yang paling optimum. Pada penelitian ini menggunakan *popSize* sebanyak 5 sehingga pada inisialisasi populasi yang terbentuk adalah 5 individu dan diuraikan pada Tabel 4.1.

Tabel 4.1 Tabel Inisialisasi Populasi Awal

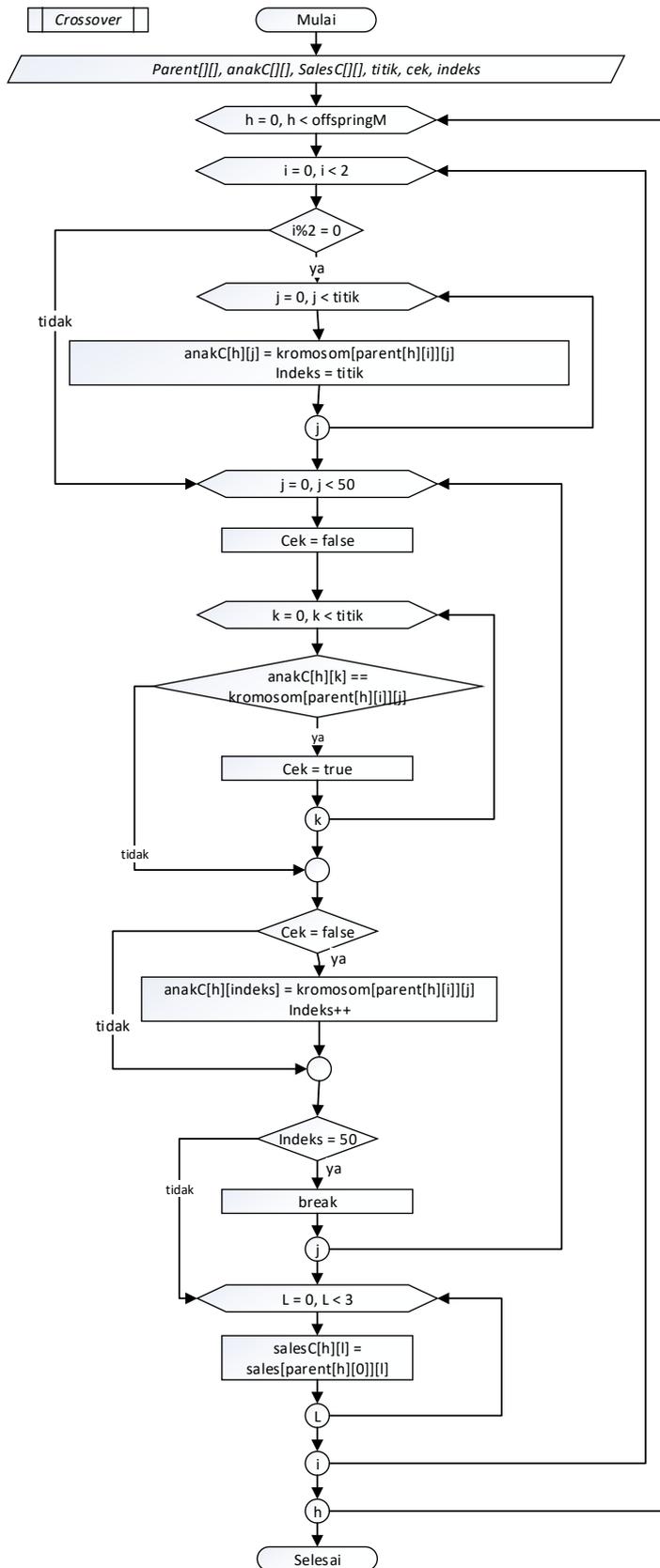
<i>P</i>	Kromosom													Sales		
<i>p1</i>	46	47	28	29	1	42	37	..	.	6	35	24	19	18	17	15
<i>p2</i>	10	49	33	15	44	24	6	..	.	20	22	48	38	19	17	14
<i>p3</i>	16	45	32	11	17	28	43	..	.	1	21	30	3	18	15	17
<i>p4</i>	9	41	2	19	16	42	48	..	.	33	39	1	27	17	16	17
<i>p5</i>	37	16	34	1	40	46	24	..	.	29	33	18	10	16	18	16

4.2.2 Reproduksi

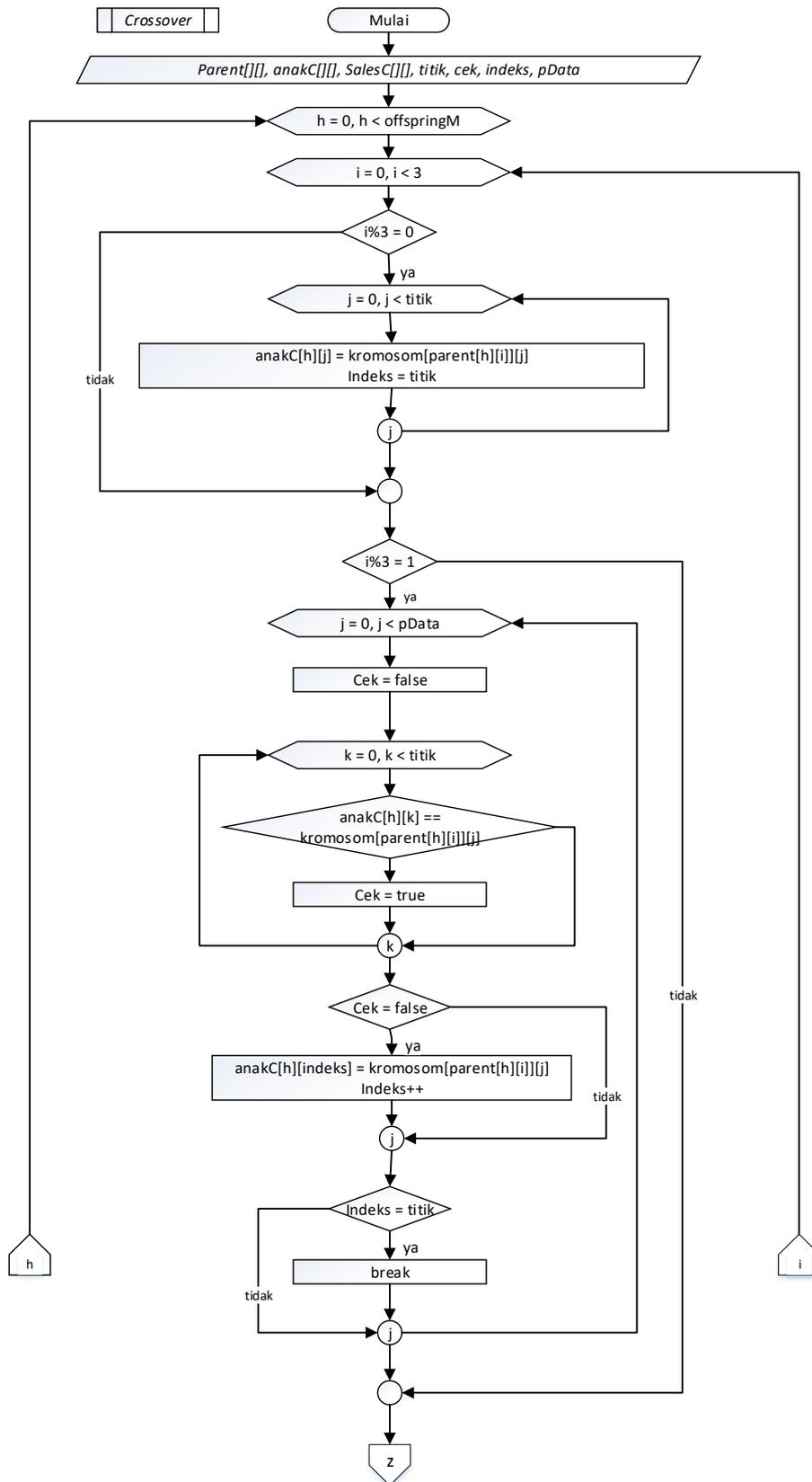
Proses reproduksi merupakan tahap selanjutnya dari algoritme genetika yaitu pembentukan individu-individu baru yang terbentuk dari individu awal (*parent*) yang sebelumnya telah diinisialisasi. Reproduksi bertujuan untuk penambahan variasi-variasi kromosom dari yang ada sebelumnya dan alternatif solusi menjadi lebih beragam, sehingga untuk lebih luas dalam mendapatkan solusi optimum. Proses reproduksi ada dua yaitu *crossover* dan *mutation*. Diagram alur reproduksi ditunjukkan pada Gambar 4.3.



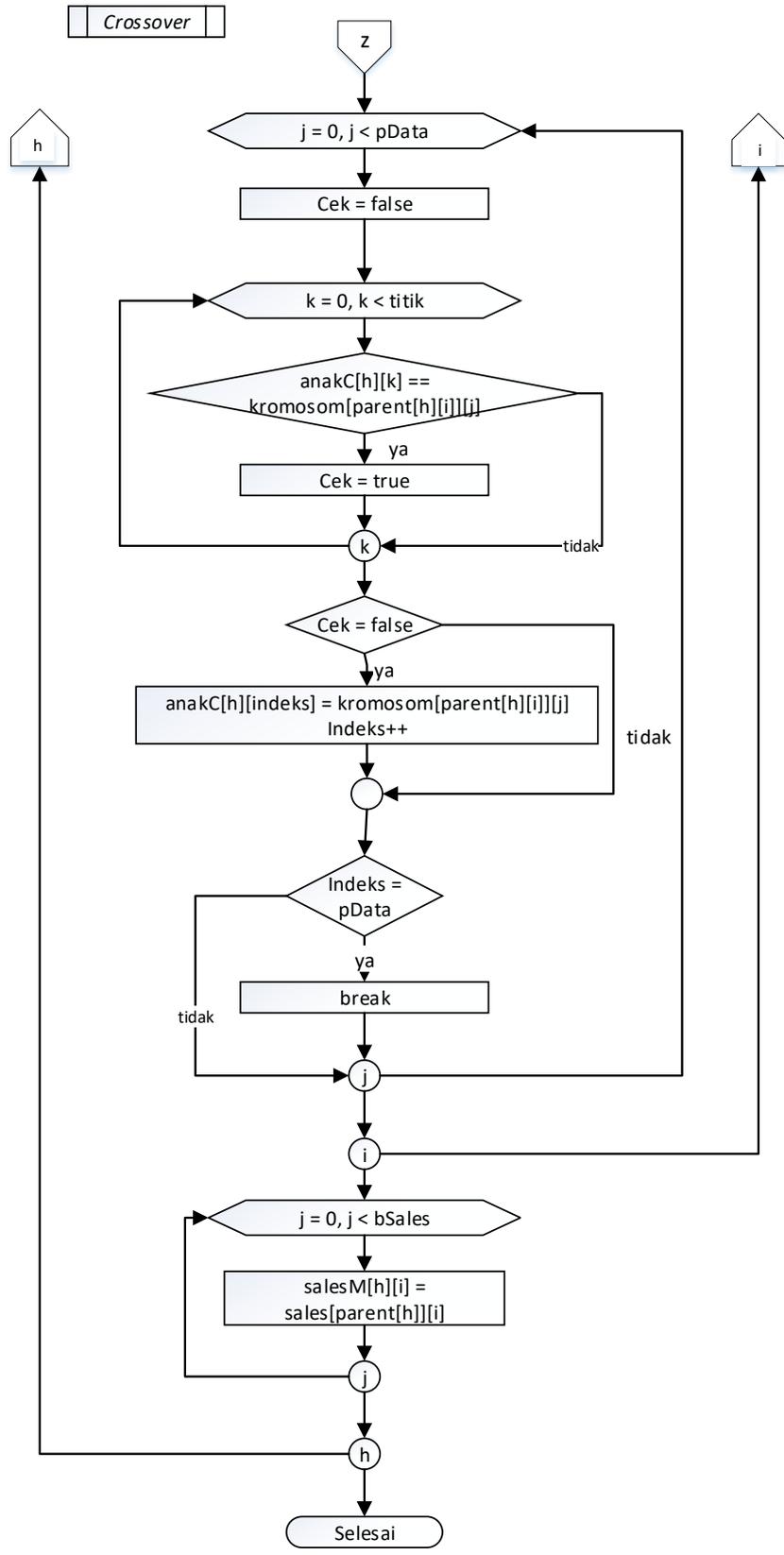
Gambar 4.3 Diagram Alir Reproduksi



Gambar 4.6 Diagram Alir One Cut Point Crossover



Gambar 4.7 Diagram Alir Two Cut Point Crossover (pertama)



Gambar 4.8 Diagram Alir Two Cut Point Crossover (kedua)

4.2.2.2 Mutasi

Tahap reproduksi yang kedua adalah mutasi yaitu proses reproduksi dengan mengubah urutan gen dari individu yang terpilih secara acak. Mutasi memiliki tujuan yaitu untuk meningkatkan jumlah dari keanekaragaman individu yang sebelumnya pada sebuah populasi. Pada penelitian ini metode mutasi yang digunakan adalah metode *exchange mutation* dan *insertion mutation*. Metode mutasi yang digunakan adalah dengan menentukan 2 posisi *point*. Penentuan jumlah offspring pada mutasi adalah dengan *mutation rate* (*mr*). Perhitungan dalam menentukan banyaknya *offspring* dapat diuraikan pada persamaan 4.2.

$$\text{Offspring} = mr \times \text{PopSize} \quad (4.2)$$

$$\text{Offspring} = 0.2 \times 5 = 1$$

Pada Gambar 4.9 akan digambarkan proses *exchange mutation* dengan memilih satu individu secara acak sebagai *parent* dan memilih 2 posisi *point* (individu) secara acak. Pada *exchange mutation*, *point* pertama dan *point* kedua yang telah terpilih akan dilakukan penukaran posisi. Sedangkan, untuk langkah penyelesaian mutai *Insertion* akan digambarkan pada Gambar 4.11.

	Kromosom																	Sales				
p3	16	45	...	49	27	10	14	2	33	...	48	25	47	41	15	13	...	30	3	18	15	17
p3	16	45	...	49	27	41	14	2	33	...	48	25	47	10	15	13	...	30	3	18	15	17

↑
point 1
↑
point 2

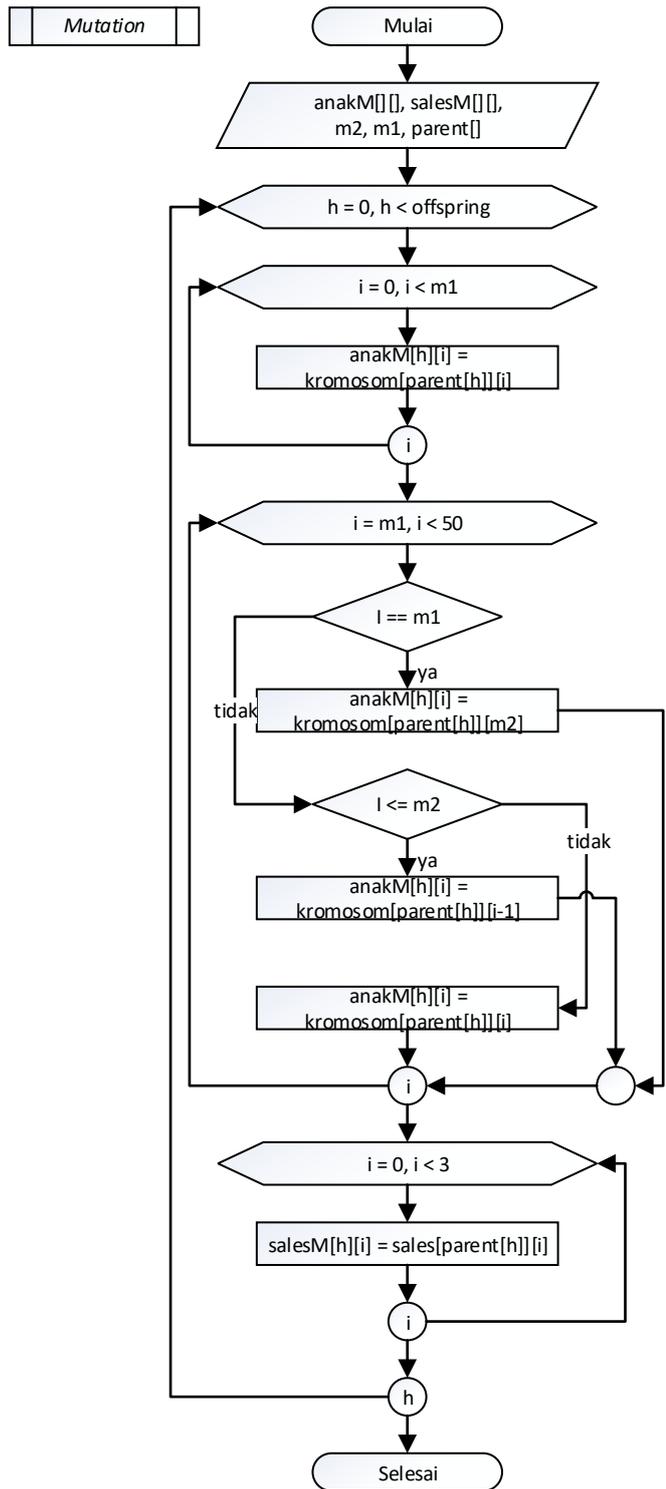
Gambar 4.9 Proses Exchange Mutation

Pada Gambar 4.10 akan digambarkan proses dari *insertion mutation*, prosesnya sama dengan *exchange mutation*, yaitu dengan memilih satu parent secara acak, kemudian memilih dua posisi *point* (individu) secara acak. *Point* pertama digunakan sebagai titik *insert* sedangkan pada *point* kedua digunakan sebagai titik yang akan dimasukkan pada *point* 1. Setelah itu dari *point* pertama sampai dengan sebelum *point* kedua dilakukan pergeseran. Sedangkan, untuk langkah penyelesaian mutai *Exchange* akan digambarkan pada Gambar 4.12.

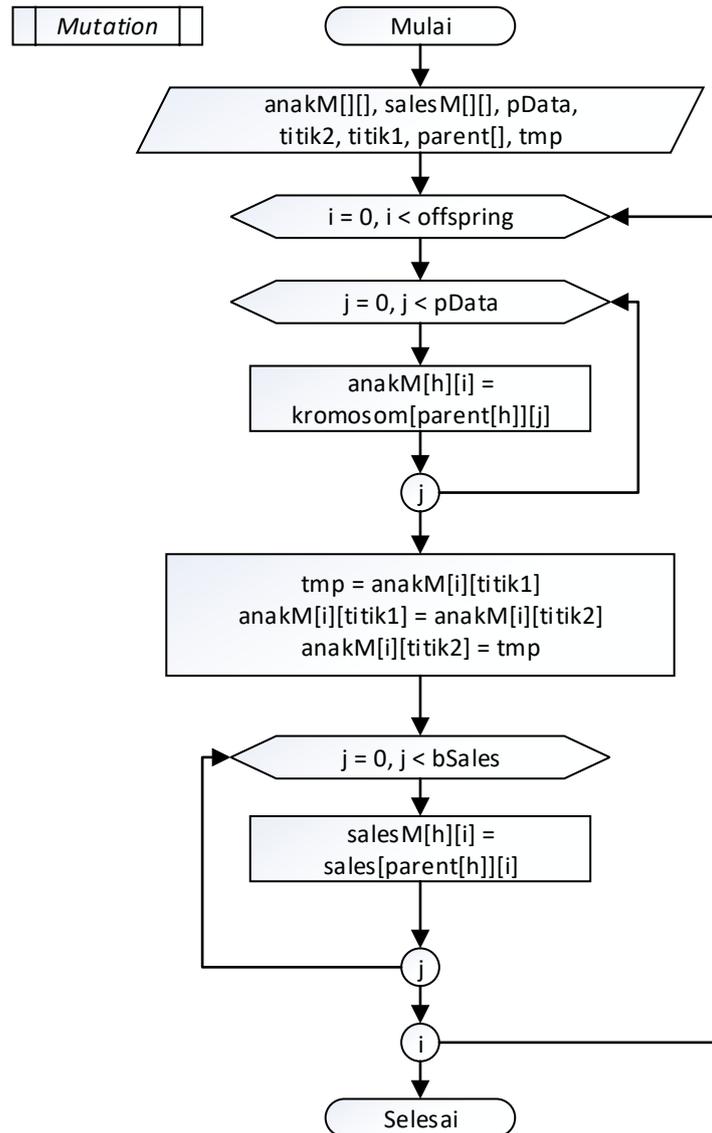
	Kromosom																	Sales				
p3	16	45	...	49	27	10	14	2	33	...	48	25	47	41	15	13	...	30	3	18	15	17
p3	16	45	...	49	27	41	10	14	2	...	44	48	25	47	15	13	...	30	3	18	15	17

↑
point 1
↑
point 2

Gambar 4.10 Proses Insertion Mutation



Gambar 4.11 Diagram Alir Insertion Mutation

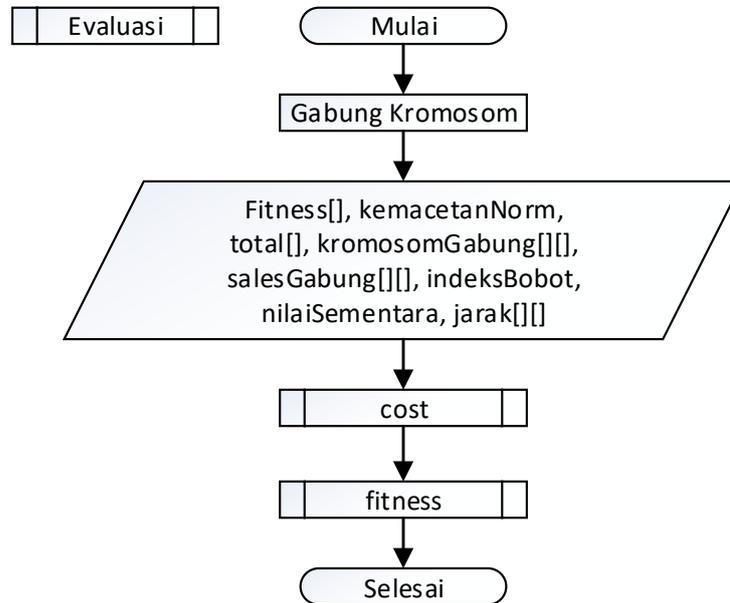


Gambar 4.12 Diagram Alir *Exchange Mutation*

4.2.3 Evaluasi

Evaluasi merupakan sebuah proses dimana sistem akan menentukan nilai bobot dari setiap kromosom yang sudah dibentuk. Nilai bobot tersebut biasa disebut dengan nilai *fitness*. *Fitness* tersebut akan membantu proses pemilihan kromosom baru pada proses seleksi. Kromosom dengan nilai terkecil akan dipertahankan sedangkan nilai *fitness* yang terbesar akan dieleminasi. Proses perhitungan *fitness* sendiri ada dua yaitu perhitungan *fitness* untuk *Multiple Travelling Salesman Problem* (pada MTSP data yang digunakan untuk melakukan perhitungan *fitness* menggunakan data jarak antar daerah masing masing jarak akan dikalikan dengan data tingkat kemacetan sebagai pinalty) dan *fitness* untuk *Multiple Travelling Salesman Problem Time Window* (Data yang digunakan pada MTSP-TW untuk melakukan perhitungan *fitness* menggunakan data waktu

tempuh antar daerah yang akan dikalikan dengan data tingkat kemacetan). Langkah-langkah proses evaluasi akan dijelaskan pada Gambar 4.13.



Gambar 4.13 Diagram Alir Evaluasi

Pada penelitian ini, proses evaluasi atau perhitungan nilai *fitness* dilakukan dengan melakukan pengurangan antara nilai konstanta (*c*) dengan nilai *cost*. Nilai konstanta adalah nilai maksimal yang mungkin terjari (Widodo, AW & Mahmudy, WF 2010). Sedangkan untuk nilai *cost* adalah nilai hasil persamaan pada setiap metode untuk setiap individu. Sehingga dari penjelasan tersebut didapatkan sebuah rumus perhitungan pada persamaan 4.3.

$$fitness = \sum_{n=0} c - cost \quad (4.3)$$

- c* = nilai Konstanta
- cost* = nilai cost untuk rute tiap *sales*
- n* = indeks *sales*

Langkah langkah perhitungan untuk nilai *cost* dapat dilihat pada Gambar 4.14 dan Gambar 4.15, sedangkan untuk langkah langkah perhitungan untuk nilai *fitness* dapat dilihat pada Gambar 4.16.

4.2.3.1 Nilai *Fitness Multiple Travelling Salesman Problem Time Window*

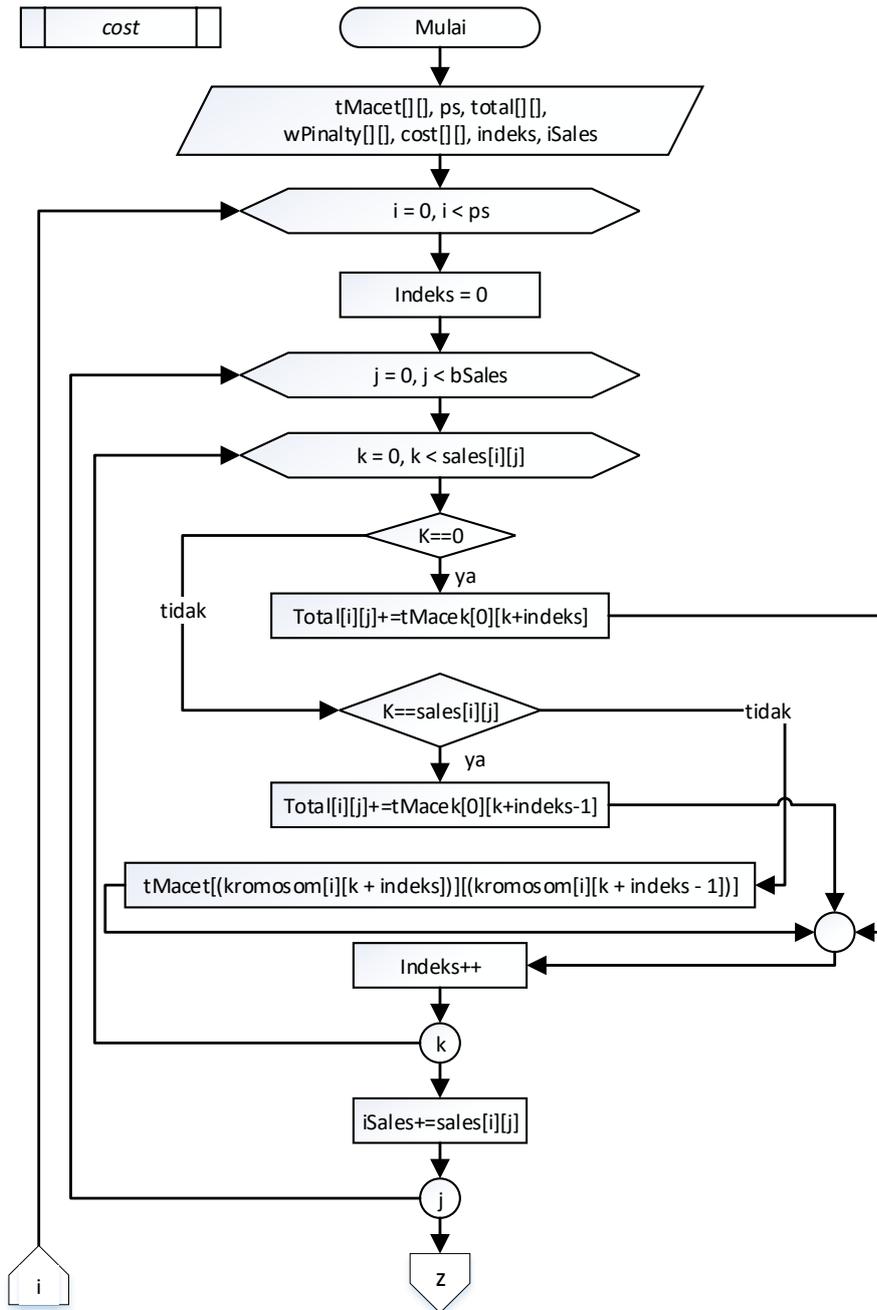
Fitness pada algoritme genetika merupakan sebuah nilai yang berfungsi sebagai alternatif calon solusi pada suatu permasalahan. Nilai *fitness* merupakan perwakilan dari setiap individu-individu yang digunakan untuk menentukan kesesuaian dengan tujuan ataupun kriteria dari masalah yang ingin diselesaikan (Suprayogi & Mahmudy, 2015). Tujuan dari nilai *fitness* adalah untuk mengetahui individu-individu yang paling baik dan cocok digunakan sebagai sebuah solusi.

Pada nilai *fitness* yang digunakan untuk *Multiple Travelling Salesman Problem Time Window* menggunakan persamaan 4.3. Nilai konstanta yang digunakan adalah 600. Nilai tersebut dipilih sebagai *role* dari *Time Window* sebagai waktu jam kerja (jam keberangkatan *sales*) yang dimiliki oleh *sales* yaitu 10 jam (berangkat jam 09.00 – 19.00). Nilai konstanta tersebut didapatkan dengan merubah jam kerja *sales* ke dalam bentuk menit. Sedangkan pada proses perhitungan MTSP-TW perhitungan nilai *cost* untuk *Multiple Travelling Salesman Problem Time Window* pada penelitian ini menggunakan data waktu tempuh antar tempat atau jarak antar alamat pengiriman barang yang didapatkan langsung dari J&T Express yang dikalikan dengan nilai tingkat kemacetan sebagai *pinalty*. Proses perhitungan *fitness* selalu berbeda-beda tergantung jenis permasalahan yang ingin diselesaikan. Nilai *fitness* didapatkan dari hasil pembentukan kromosom pada proses inialisasi populasi awal hingga proses reproduksi. Perhitungan jumlah waktu tempuh yang telah dikalikan dengan *pinalty* untuk proses pengiriman barang pada setiap *sales* adalah dari kromosom atau urutan rute yang telah ada sebelumnya, kemudian hasil tersebut akan dijumlahkan dengan *sales* yang lain. Sehingga dari penjelasan tersebut didapatkan sebuah rumus perhitungan pada persamaan 4.4.

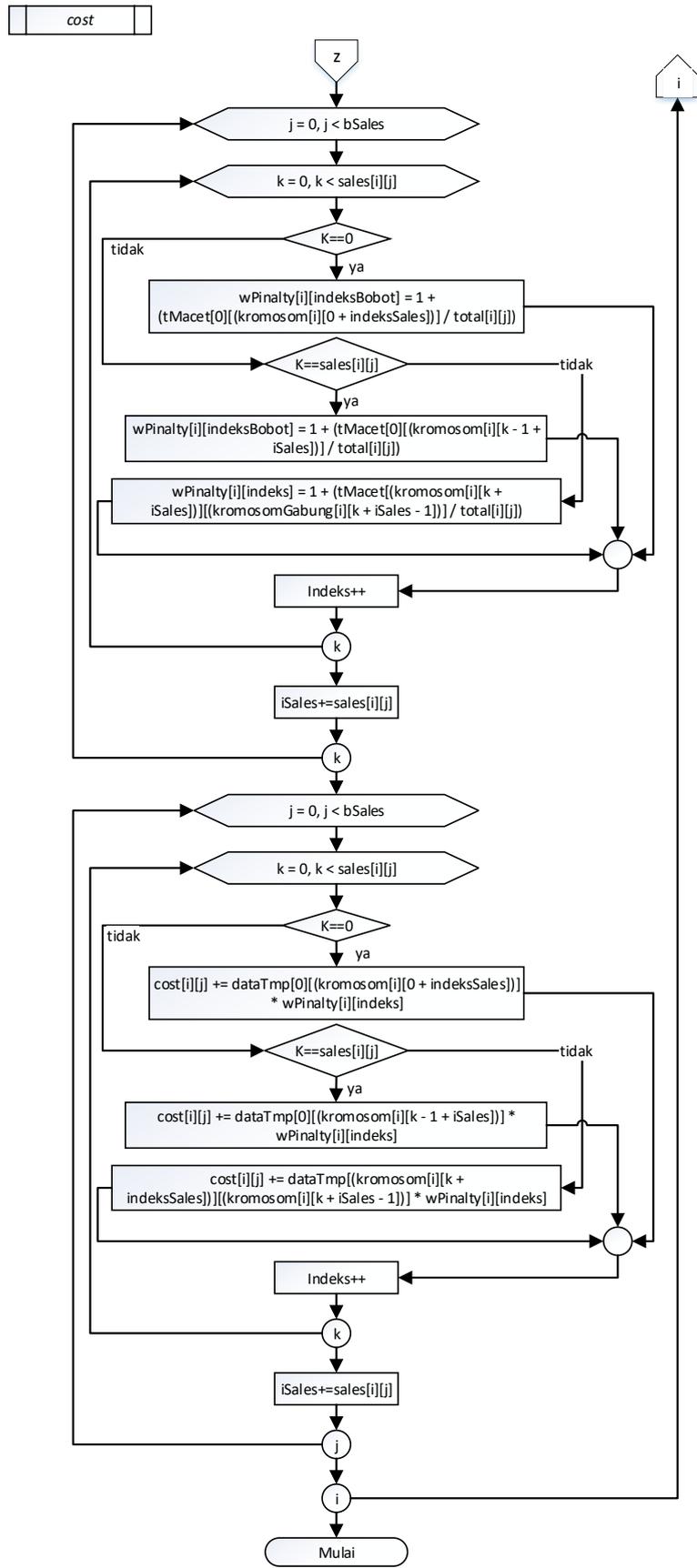
$$cost = \sum_{k=0}^n (X_k * (1 + W'_k)) \quad (4.4)$$

$$W'_k = \frac{W_k}{\sum_{k=0}^n W_k} \quad (4.5)$$

- X = Nilai waktu tempuh antar daerah
- W' = Nilai kemacetan normalisasi
- W = Nilai tingkat kemacetan
- k = indeks tingkat kemacetan

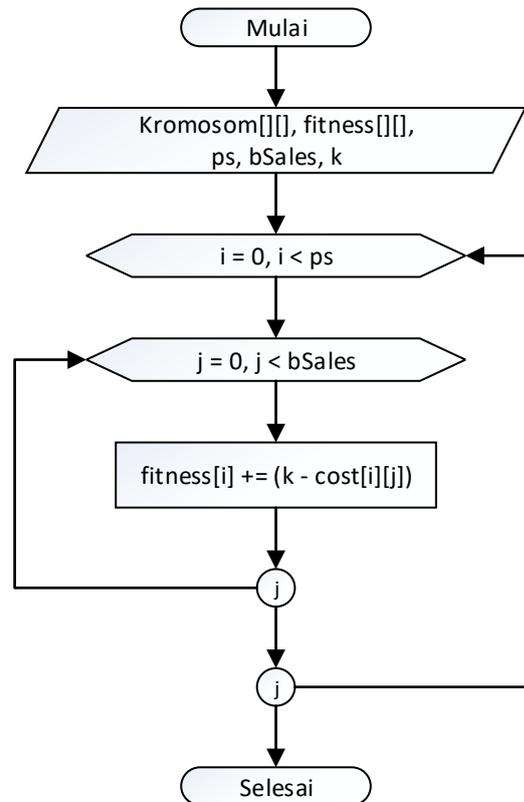


Gambar 4.14 Diagram Cost (pertama)



Gambar 4.15 Diagram Cost (kedua)

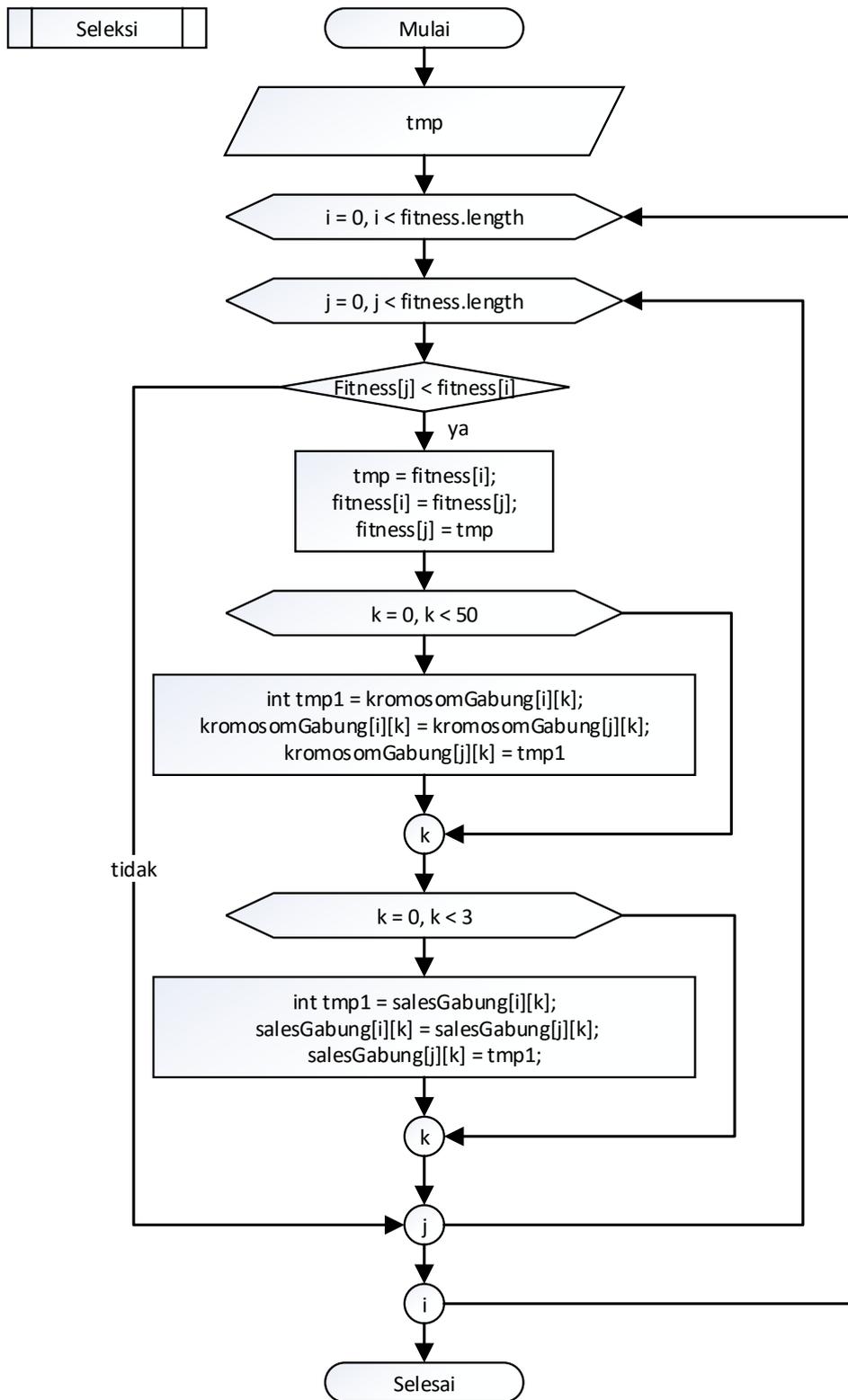
fitness



Gambar 4.16 Diagram *Fitness*

4.2.4 Seleksi

Tahap algoritme genetika selanjutnya adalah proses seleksi, dimana termasuk dalam tahapan terakhir pada perhitungan algoritme genetika. Seleksi adalah proses untuk memilih individu baru sesuai jumlah *popSize* yang telah ditentukan. Pemilihan individu baru dapat dilakukan dengan menggunakan beberapa metode, salah satunya adalah metode elitism. Pada metode elitism dilakukan pengumpulan individu-individu terbaik yaitu didapatkan dari *parent* maupun *offspring* (keturunan) dalam satu penampungan yang biasa disebut populasi. Selanjutnya pemilihan dilakukan sebanyak *popSize* yang ditentukan untuk dipertahankan hidup sebagai solusi dan dapat melanjutkan generasi selanjutnya (Widodo & Mahmudy, 2010). Langkah-langkah proses seleksi dijelaskan lebih lanjut pada Gambar 4.17 Diagram Alir Seleksi.



Gambar 4.17 Diagram Alir Seleksi

4.3 Perancangan Pengujian

Pengujian merupakan hal terpenting yang bertujuan untuk menemukan hasil serta kekurangan atau kelemahan dari penelitian. Pengujian bermaksud untuk mendapatkan sebuah kesimpulan maupun saran untuk penelitian selanjutnya. Pengujian yang dilakukan yaitu pengujian untuk menentukan jenis kombinasi *crossover* dengan *mutation* pada proses reproduksi algoritme genetika. Pengujian kedua dilakukan untuk melihat hasil rute dari permasalahan *Multiple Travelling Salesman Problem Time Window* (Pengujian MTSP-TW). Pengujian yang ketiga dilakukan untuk membandingkan hasil dari *Multiple Travelling Salesman Problem* dengan *Multiple Travelling Salesman Problem Time Window* untuk menemukan metode mana yang lebih baik dalam menghasilkan rute pengiriman barang dengan beberapa syarat.

4.3.1 Pengujian *Crossover* dan *Mutation*

Pengujian reproduksi ini dilakukan dengan melakukan kombinasi pada dua jenis *crossover* dan dua jenis *mutation*. Jenis *crossover* yang dipilih adalah *crossove one cut point* dan *two cut point crossover*. Sedangkan untuk *mutation* sendiri yang digunakan adalah *insertion* dan *exchange*. Untuk ukuran populasi dan generasi maksimum menggunakan nilai sebesar 10 dan 1000, dan untuk nilai *crossover rate* dan *mutation rate* adalah sebesar 0,2 dan 0,1 (Widodo, AW & Mahmudy, WF 2010). Selain itu pengujian akan dilakukan sebanyak 10 kali percobaan dan untuk simbol yang akan digunakan dalam percobaan adalah:

1. C1: *one cut point crossover*.
2. C2: *two cut point crossover*.
3. M1: *exchange mutation*.
4. M2: *insertion mutation*.
5. cr: *crossover rate*.
6. mr: *mutation rate*.

Tabel pengujian *crossover* dan *mutation* yang digunakan dapat dilihat pada Tabel 4.2 .

Tabel 4.2 Tabel Pengujian *Crossover* dan *Mutation*

Crossover Mutation	Nilai <i>fitness</i> terbaik pada percobaan ke-										Nilai Rata-rata
	1	2	3	4	5	6	7	8	9	10	
C1 & M1											
C1 & M2											
C2 & M1											
C2 & M2											

4.3.2 Pengujian *Multiple Travelling Salesman Problem Time Window*

Pengujian *Multiple Travelling Salesman Problem Time Window* dilakukan untuk menguji hasil rute dari sistem MTSP-TW bisa menghasilkan rute dengan batasan waktu kerja *sales*. Pengujian dilakukan dengan cara melihat rata-rata waktu tempuh rute yang didapatkan oleh *sales*. Selain itu, pengujian dilakukan

dengan melihat hasil waktu tempuh dari setiap *sales* yang ada. Pengujian tersebut direpresentasikan dalam bentuk Tabel, dan dapat dilihat pada Tabel 4.3.

Tabel 4.3 Tabel Pengujian *Multiple Travelling Salesman Problem Time Window*

Percobaan ke-	Sales ke-					Total waktu	Rata-rata Waktu
	1	2	3	4	5		
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							

4.3.3 Pengujian Validasi

Pengujian Validasi dilakukan dengan membandingkan hasil rute dari *Multiple Travelling Salesman Problem Time Window* dengan hasil rute yang didapat dari perusahaan. Hasil rute *Multiple Travelling Salesman Problem Time Window* didapatkan dari hasil dari pengujian yang dilakukan sebelumnya. Sedangkan untuk hasil rute dari perusahaan, didapatkan dengan memberikan studi kasus pengiriman sebanyak 100 alamat yang harus diselesaikan oleh 3 kurir. Studi kasus tersebut diberikan dengan cara memberikan kuisisioner kepada 5 kurir, dimana setiap kurir akan diminta untuk membuat 10 rute, dimana antar rute yang ada tidak boleh sama, dan pembagian jumlah rute untuk tiap *sales* diberikan kebebasan kepada masing-masing kurir.

Pengujian tersebut akan digambarkan dalam bentuk Tabel untuk hasil kuisisioner masing-masing kurir. Akan terdapat tiga jenis kolom pada pengujian ini. Kolom pertama akan berisi nomor percobaan yang dilakukan, kolom kedua berisi waktu tempuh dari rute yang didapat tiap *sales*, sedangkan untuk kolom ke tiga berisi status waktu tempuh yang didapat semua *sales*. Pada kolom ketiga ketiga rute yang didapat *sales* dianggap gagal maka akan mencetak simbol "X" sedangkan ketika rute yang didapat *sales* dianggap berhasil akan memunculkan simbol "O". Tabel tersebut dapat dilihat pada Tabel 4.4.

Tabel 4.4 Tabel Pengujian Rute Perusahaan

Percobaan ke-	Sales ke-					Status Sales
	1	2	3	4	5	
1						
2						
3						
4						
5						
6						
7						
8						
9						
10						

Sedangkan untuk bentuk kuisisioner akan diberikan dua jenis. Kuisisioner pertama akan digambarkan pada Gambar 4.18 dan untuk Kuisisioner kedua akan digambarkan pada Gambar 4.19. Kuisisioner pertama akan berisikan informasi alamat pengiriman, dan aturan yang ada. Kuisisioner ke dua akan berisikan kolom-kolom untuk mengisikan 10 rute dengan masing-masing rute memiliki 3 kurir. Setiap kurir akan diberikan 45 kolom, guna memudahkan kurir J&T dalam melakukan pengisian rute.

DATA DAN ATURAN PENGISIAN KUISISIONER	
Data Alamat	
Aturan Pemilihan Rute	

Gambar 4.18 Kuisisioner 1

<u>PENGISIAN RUTE</u>													
<u>Rute 1</u>													
<u>Kurir 1</u>													
<u>Kurir 2</u>													
<u>Kurir 3</u>													

Gambar 4.19 Kuisisioner 2

Pada Gambar 4.18 terdapat aturan pemilihan rute, aturan-aturan yang akan diberikan adalah:

1. Semua alamat harus masuk kedalam rute pengiriman.
2. Satu alamat hanya bisa dikirim oleh satu kurir.
3. Jumlah pengiriman setiap kurir bebas, akan tetapi setiap alamat harus masuk kedalam 1 rute (aturan nomor 1).
4. Setiap rute tidak boleh sama.
5. Tuliskan nomor urut alamat pada rute.

4.4 Perancangan *User Interface* (UI)

User Interface adalah suatu perantara yang digunakan untuk menghubungkan user dengan sistem yang telah dibuat. Pada sistem yang dibuat, ada beberapa tampilan yaitu tampilan awal, tampilan hasil uji dan tampilan diagram. Sehingga dari tampilan yang ada user bisa memodifikasi beberapa data yang dibutuhkan oleh sistem (terlebih data yang dibutuhkan untuk menyelesaikan Algoritme Genetika) agar mendapatkan hasil yang lebih baik.

4.4.1 Tampilan Awal

Pada tampilan awal ini, user bias melakukan input beberapa data. Data tersebut nantinya akan diproses untuk melakukan proses dari Algoritme Genetika sendiri. Seperti jumlah popsize, nilai *Crossover Rate*, nilai *Mutation Rate*, dan juga banyaknya *sales*. Inputan tersebut nantinya bisa mempengaruhi dari hasil dari proses Reproduksi pada Algoritme Genetika dalam menghasilkan kromosom baru untuk mendapatkan kromosom paling optimum. Selain itu pada tampilan awal juga tersedia *Button* yang berfungsi untuk melihat data, melakukan proses perhitungan, dan melihat hasil nilai *fitness* tiap iterasi. Representasi dari tampilan awal bisa dilihat pada Gambar 4.20

Judul	
Data Alamat Pengiriman	<input type="button" value="Buton tampilan data"/> <input type="text"/>
	<input type="button" value="Masukan CR"/> <input type="text"/>
	<input type="button" value="Masukan MR"/> <input type="text"/>
	<input type="button" value="Masukan PopBab"/> <input type="text"/>
	<input type="button" value="Masukan Banyak Sales"/> <input type="text"/>
	<input type="button" value="Proses"/>
	<input type="button" value="LIHAT Hasil"/>

Gambar 4.20 Tampilan Awal

4.4.2 Tampilan Hasil

Tampilan hasil akan menunjukkan hasil dari proses perhitungan *Multiple Travelling Salesman Problem* dan *Multiple Travelling Salesman Problem Time Window*, dan juga tampilan hasil pengujian Reproduksi pada ombinasi jenis *crossover* dan *mutation*. Hasil yang ditunjukkan adalah hasil perhitungan nilai *fitness* terbaik untuk setiap iterasi yang telah dilakukan.

Judul		
Nama Kolom	Nama Kolom	Nama Kolom

Gambar 4.21 Tampilan Hasil