

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Jadwal Mata Pelajaran**

Menurut Jong Jek Siang (2000:22) jadwal merupakan sesuatu yang menjelaskan dimana dan kapan orang – orang dan sumber daya berada pada suatu waktu, Sedangkan penjadwalan merupakan proses cara pembuatan menjadwalkan atau memasukkan dalam jadwal.

Menurut Ana (2006:6) Mata pelajaran adalah suatu pelajaran dimana pelajaran tersebut selalu mempelajari hal-hal dalam kehidupan sehari-hari sehingga dengan memahami pelajaran tersebut kita lebih berhati-hati dalam menjalani kehidupan ini dan tidak menganggap enteng suatu masalah yang kecil padahal masalah yang kecil tersebut dapat mengakibatkan hal-hal yang besar atau suatu cabang pengetahuan yang diajarkan atau diteliti di tingkat perguruan tinggi.

Kebanyakan orang terbiasa melihat rencana pelajaran yang disajikan sebagai tabel hari dalam seminggu dan beberapa waktu sebelumnya. Dapat diperhatikan bahwa setiap hari dipisahkan menjadi segmen waktu. Setiap garis waktu menyertakan daftar mata pelajaran yang diajarkan, serta siapa yang mengajar dan di mana mereka mengajar.

#### **2.2 Algoritma Genetika**

Algoritma genetika adalah metode pencarian yang didasarkan pada seleksi alam dan prinsip genetika. Pada awalnya Algoritma genetika digunakan sebagai algoritma untuk mencari parameter ideal. Namun, seiring perkembangannya, algoritma genetika dapat digunakan untuk berbagai masalah seperti *learning*, peramalan, pemrograman otomatis dan sebagainya.

Dalam proses seleksi alamiah individu secara terus-menerus mengalami perubahan gen untuk menyesuaikan dengan lingkungan hidupnya. individu-individu yang berniali fitness tinggi yang mampu bertahan sedangkan individu bernilai fitness rendah akan mati.

Algoritma genetika pertama kali dikembangkan oleh John Holland dari Universitas Michigan (1975).

Algoritma genetika dibagi atas beberapa bagian diantaranya adalah sebagai berikut (Fadlisyah, Arnawan, Faisal ,2009:3) :

1. Gen

Gen adalah nilai yang menunjukkan unit dasar yang membentuk makna tertentu dalam kromosom, yang merupakan satu kesatuan gen.

2. Kromosm / individu

Kromosom adalah kumpulan gen yang membentuk nilai spesifik dan memberikan solusi potensial untuk masalah.

3. Populasi

Populasi merupakan sekumpulan individu yang akan diproses bersama dalam satu satuan siklus evolusi.

4. *Fitness*

*Fitness* menyatakan seberapa baik nilai dari suatu individu atau solusi yang didapatkan.

5. Seleksi

Seleksi adalah proses untuk mendapatkan calon induk yang baik

6. *Crossover*

*Crossover* merupakan proses pertukaran atau kawin silang gen gen dari dua induk tertentu.

7. Mutasi

Mutasi merupakan proses pergantian sala satu gen yang terpilih dengan nilai tertentu

8. Generasi

Generasi merupakan urutan iterasi dimana beberapa kromosom bergabung

9. *Offspring*

*Offspring* merupakan kromosom baru yang dihasilkan setelah melewati satu generasi

### 2.2.1 Struktur Umum Algoritma Genetika

Beberapa hal yang harus dilakukan dalam Algoritma Genetika adalah

Romauli Statina Sihombing (2014):

1. Mendefinisikan individu, dimana individu menyatakan salah satu solusi yang mungkin dari permasalahan yang diangkat.
2. Mendefinisikan nilai *fitness*, yang merupakan ukuran baik-tidaknya sebuah individu atau baik-tidaknya solusi yang didapatkan.
3. Menentukan proses pembangkitan populasi awal.
4. Menentukan proses seleksi yang akan digunakan.
5. Menentukan proses perkawinan silang (*crossover*) dan mutasi gen yang akan digunakan.

Variabel dan parameter yang digunakan pada algoritma genetika adalah

Romauli Statina Sihombing (2014):

1. Fungsi *fitness* (fungsi tujuan) yang dimiliki oleh masing-masing individu untuk menentukan tingkat kesesuaian individu tersebut dengan kriteria yang ingin dicapai.
2. Populasi jumlah individu yang dilibatkan pada setiap generasi.
3. Probabilitas terjadinya persilangan (*Crossover*) pada suatu generasi.
4. Probabilitas terjadinya mutasi pada setiap individu.
5. Jumlah generasi yang akan dibentuk yang menentukan lama penerapan Algoritma Genetika.

### 2.2.2 Komponen Utama Algoritma Genetika

Ada 7 komponen utama dalam Algoritma Genetika

#### 1. Teknik penyandian

Teknik penyandian disini meliputi penyandian gen dari kromosom. Gen sangat penting dan merupakan bagian dari kromosom. Satu gen biasanya akan mewakili satu variable. Gen dapat di sebut dalam bentuk : string bit, pohon, array, bilangan real, daftar aturan, elemen program, atau representasi lainnya yang dapat di implementasikan untuk operator genetika

## 2. Prosedur inisialisasi

Besar kecilnya populasi tergantung pada masalah yang akan ditangani dan jenis pemberian keturunan yang akan dilakukan. Setelah ukuran populasi tidak ditentukan, maka pada saat itu kromosom harus diinisialisasi dalam populasi. Pengenalan kromosom dapat dilakukan secara acak, namun tetap harus fokus pada ruang regulasi dan batasan masalah yang ada.

## 3. Nilai *Fitness*

Seorang individu dinilai berdasarkan kemampuan tertentu sebagai ukuran kinerjanya. Dalam evolusi alami, individu dengan nilai *fitness* tinggi akan bertahan. Sedangkan individu dengan nilai *fitness* rendah akan mati. Dalam masalah perbaikan, solusi yang akan dicari adalah memaksimalkan fungsi *h* (dikenal dengan masalah maksimisasi) sehingga nilai *fitness* yang digunakan adalah nilai dari fungsi *h*, yaitu

$$f = h \dots\dots\dots (2.1)$$

(di mana *f* adalah nilai *fitness*). Tetapi jika masalahnya adalah meminimalkan fungsi *h* (masalah minimasi), maka fungsi *h* tidak bisa digunakan secara langsung. Hal ini disebabkan adanya aturan bahwa individu yang memiliki nilai *fitness* tinggi dapat bertahan hidup pada generasi berikutnya. Oleh karena itu nilai *fitness* yang bisa digunakan adalah

$$f = 1/h \dots\dots\dots (2.2)$$

yang berarti semakin kecil nilai *h*, semakin besar nilai *f*. Akan tetapi hal ini bisa menjadi masalah jika *h* bisa bernilai 0, yang akan mengakibatkan *f* bisa bernilai tak hingga. Dan untuk mengatasinya, *h* perlu ditambahkan sebuah bilangan yang dianggap kecil [0-1] sehingga nilai *fitness*-nya menjadi:

$$f = \frac{1}{(n+a)} \dots\dots\dots (2.3)$$

(suyanto, 2011, Artificial intelligence)

#### 4. Seleksi

Tujuannya adalah untuk memberikan anggota populasi yang paling cocok dengan peluang reproduksi yang lebih besar. Seleksi akan menentukan individu mana yang akan dipilih untuk rekombinasi dan bagaimana keturunan terbentuk dari individu terpilih tersebut. Langkah pertama adalah mencari nilai fitness. Langkah kedua adalah nilai fitness yang diperoleh digunakan pada tahapan seleksi selanjutnya.

Ada beberapa metode seleksi dari induk (Sri Kusumadewi, 2003) yaitu:

##### 1. *Rank-based fitness*

Populasi diurutkan sesuai dengan nilai objektifnya. Nilai fitness dari setiap individu hanya bergantung pada posisi individu tersebut dalam urutan, dan tidak dipengaruhi oleh nilai objektifnya

##### 2. *Seleksi Roda Roulette (Roulette wheel selection)*

Metode ini Individu-individu dipetakan dalam suatu garis segmen secara berurutan dengan tujuan agar tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran fitnessnya. Sebuah bilangan acak dibangkitkan dan individu yang memiliki segmen dalam kawasan segmen dalam kawasan bilangan acak tersebut akan terseleksi. Proses ini diulang sampai diperoleh sejumlah individu yang diharapkan.

##### 3. *Stochastic universal sampling*

Memiliki nilai bias nol dan penyebaran yang paling sedikit. Individu-individu dipetakan dalam suatu segmen garis secara berurut sehingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran fitnessnya seperti halnya pada seleksi

roda roulette. Kemudian diberikan sejumlah pointer sebanyak individu yang ingin diseleksi pada garis tersebut. Andaikan  $N$  adalah jumlah individu yang akan diseleksi, maka jarak antar pointer adalah  $1/N$ , dan posisi pointer pertama diberikan secara acak pada range  $[1, 1/N]$ .

#### **4. Seleksi Lokal (*Local Selection*)**

Dalam seleksi lokal setiap individu yang berada di dalam konstraint tertentu disebut dengan nama lingkungan lokal. Interaksi antar individu hanya dilakukan dalam kawasan tersebut. Lingkungan tersebut ditetapkan sebagai struktur dimana populasi tersebut terdistribusi. Lingkungan tersebut juga dipandang sebagai sekelompok pasangan-pasangan yang potensial.

#### **5. Seleksi dengan Pemotongan (*Truncation selection*)**

Seleksi ini biasanya digunakan oleh populasi yang jumlahnya sangat besar. Pada metode ini individu diurutkan berdasarkan nilai fitness-nya. Hanya individu yang terbaik saja yang akan diseleksi sebagai induk. Parameter yang digunakan adalah suatu nilai ambang trunc yang mengindikasikan ukuran populasi yang akan diseleksi sebagai induk yang berkisar antara 50% - 10%. Individu yang berada dibawah batas nilai tidak akan menghasilkan keturunan.

#### **6. Seleksi dengan Turnamen (*Turnament selection*)**

Ditetapkan suatu nilai tour untuk individu yang dipilih secara acak dari populasi. Individu terbaik dalam kelompok ini akan dipilih sebagai induk. Parameter yang digunakan adalah ukuran tour yang bernilai antara 2 sampai  $N$  (jumlah individu dalam populasi).



## 5. Pindah silang

Pada proses pindah silang terjadi kombinasi pewarisan gen-gen dari induknya, gen dari kedua induk dapat bercampur untuk menghasilkan susunan kromosom yang baru. Dari proses tersebut akan dihasilkan variasi genetik.

Dengan rencana khusus, dua individu dipilih sebagai wali. Setelah mendapatkan dua individu wali, selanjutnya ditentukan titik pindah silang secara acak. Kemudian beberapa bagian dari kedua kromosom tersebut ditukar pada titik pindah silang yang dipilih. Titik pindah silang adalah tempat terjadinya pertukaran gen antara kedua wali individu. Pertukaran itu akan menghasilkan dua individu anak. meskipun demikian, operasi pindah silang tidak selamanya berhasil. Peluang keberhasilan operasi pindah silang dinyatakan dengan probabilitas pindah silang atau  $pc$ . Terdapat tiga skema pindah silang yang biasa digunakan (suyanto, 2011, Artificial intelligence) yaitu:

### 1. Pindah silang satu titik (*single – point Crossover*)

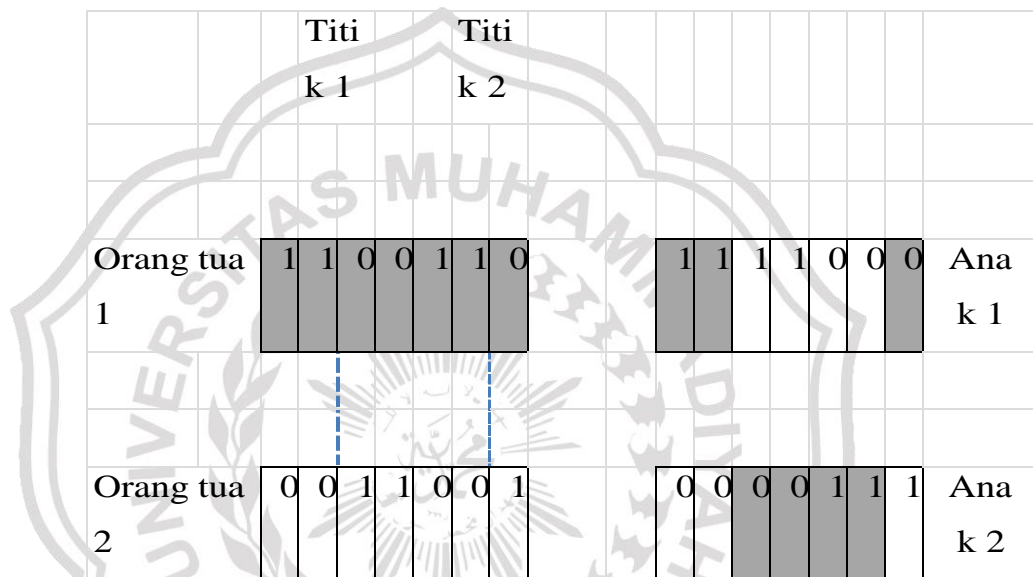
Pindah silang ini merupakan skema yang paling sederhana. Titik pindah silang hanya satu dengan posisis yang dibangkitkan secara acak.

		Titik Potong																	
Orang tua 1		1	1	0	0	1	1	0		0	0	1	1	1	1	0	Anak 1		
Orang tua 2		0	0	1	1	0	0	1		1	1	0	0	0	0	1	Anak 2		

**Gambar 2.1** Contoh pindah silang satu titik

**2. Pindah silang banyak titik ( *multi – point Crossover* )**

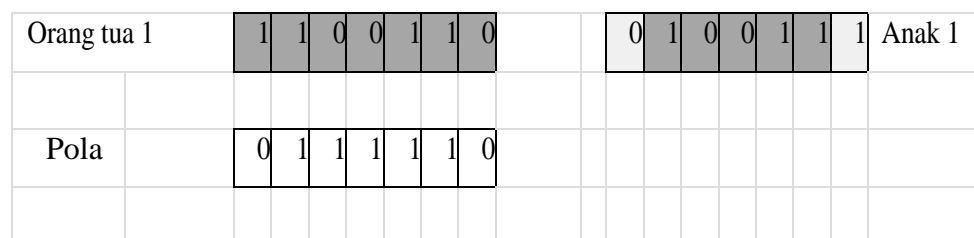
Pada masalah tertentu dimana individu terdiri dari sangat banyak gen ( misalnya 10000 gen), disini kita memerlukan lebih dari satu titik pindah silang. Banyak nya titik pindah silang ini akan berpengaruh pada pola pertukaran gen-gen antar individu orang tua.



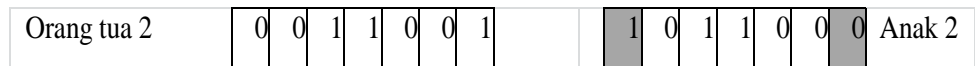
**Gambar 2.2** Contoh pindah silang banyak titik (lebih dari satu). Dalam contoh ini terdapat dua titik pindah silang yang dibangkitkan secara acak

**3. Pindah silang pola seragam ( *Uniform Crossover* )**

Dengan operasi pindah silang pola seragam maka komposisi gen-gen tertentu pada suatu individu dapat sangat dipertahankan. Hal ini akan memudahkan proses pencarian solusi.







**Gambar 2.3** Contoh pindah silang pada pola seragam. Dalam contoh ini pindah silang dilakukan berdasarkan suatu pola tertentu. Pindah silang dilakukan jika pola bernilai 0.

## 6. Mutasi

Mutasi diperlukan untuk mengembalikan informasi bit yang hilang akibat crossover. Mutasi ditetapkan dengan probabilitas yang sangat kecil. Jika mutasi dilakukan terlalu sering, maka akan menghasilkan individu yang lemah karena konfigurasi gen pada individu yang unggul akan rusak. Dilihat dari bagian yang termutasi, proses mutasi dapat dibedakan menjadi tiga bagian (suyanto, 2011, Artificial intelligence) yaitu:

1. Mutasi pada tingkat kromosom semua gen dalam kromosom berubah.



**Gambar 2.4** Contoh mutase pada tingkat kromosom. Semua gen pada kromosom berubah. Pada contoh ini, yang terjadi pada gen bernilai 0 berubah menjadi 1 dan gen yang tadinya bernilai 1 berubah menjadi 0

2. Mutasi pada tingkat gen: semua bit dalam satu gen akan berubah. Misal gen 2 yang mengalami mutasi.



**Gamabar 2.5** Contoh mutase pada tingkat gen. semua bit dalam suatu gen berubah

3. Mutasi pada tingkat hanya satu bit yang akan berubah



**Gambar 2.6** Contoh mutase tingkat bit. Hanya satu bit yang akan berubah

## 7. Parameter Kontrol

Parameter kontrol dalam algoritma genetika diperlukan untuk mengendalikan para operator seleksi. Pemilihan parameter genetika dalam mengurus masalah ditentukan oleh kinerja algoritma genetika dalam memecahkan masalah. Ada dua parameter dasar dari algoritma genetika, yaitu probabilitas crossover ( $P_c$ ) dan probabilitas mutasi ( $P_m$ )

### 1. Probabilitas *Crossover*

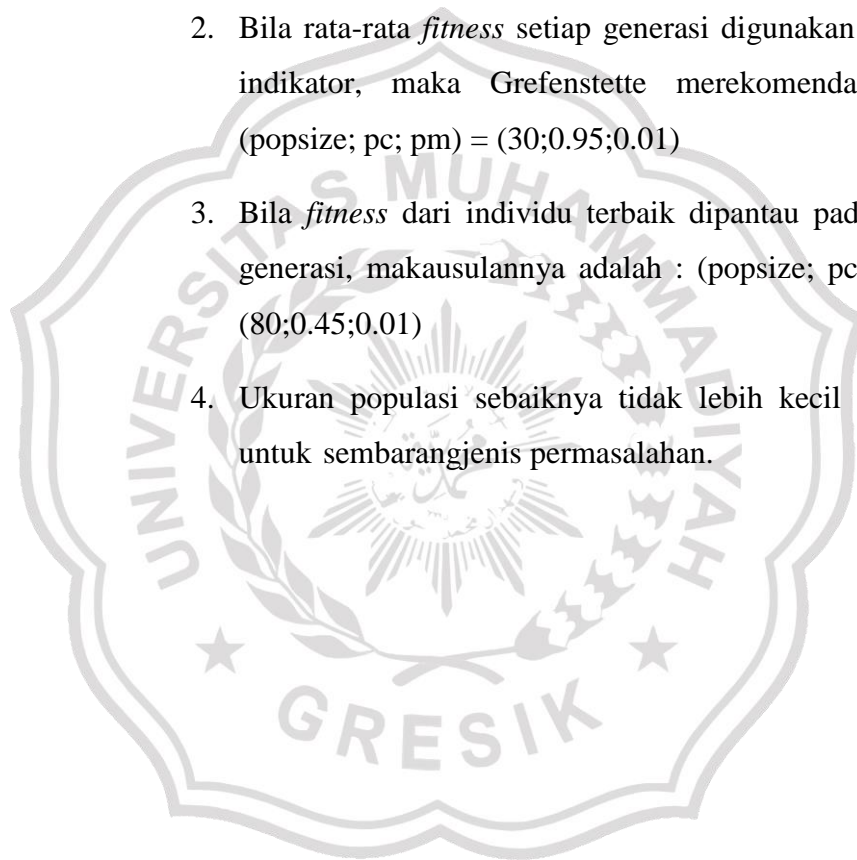
Probabilitas crossover menentukan seberapa sering proses crossover akan terjadi antara dua kromosom orang tua. Jika tidak terjadi crossover, salah satu orang tua dipilih secara acak dengan probabilitas yang sama dan disalin menjadi anak. Jika terjadi crossover, keturunan dibuat dari bagian-bagian kromosom orang tua. Jika probabilitas crossover adalah 100% maka menghasilkan keseluruhan keturunan baru yang dibuat dengan crossover. Namun sebaliknya jika probabilitas crossover 0% maka seluruh generasi baru dihasilkan dari salinan kromosom populasi lama yang belum tentu menghasilkan populasi yang sama dengan populasi sebelumnya,

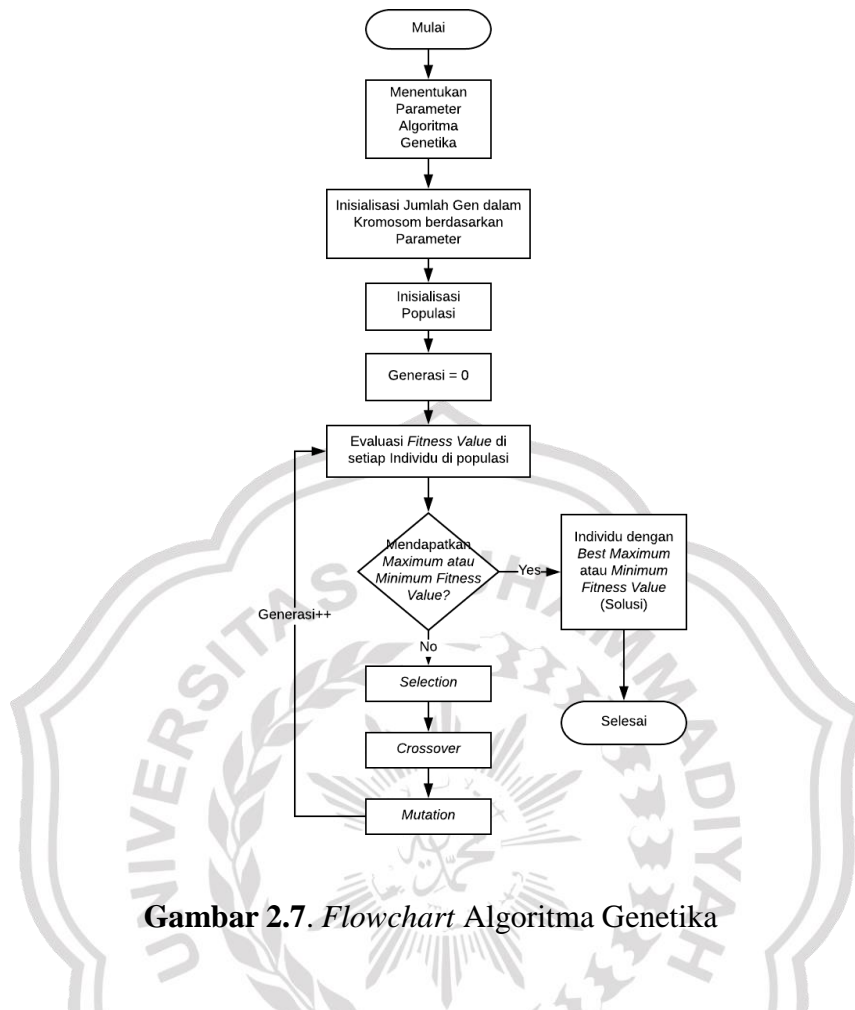
### 2. Probabilitas mutasi ( $p_m$ )

Probabilitas mutasi menjelaskan seberapa sering bagian-bagian kromosom akan dimutasikan. Jika tidak ada mutasi, keturunan diambil atau disalin langsung setelah crossover tanpa perubahan. Jika mutasi dilakukan, maka bagian-bagian tertentu kromosom diubah. jika probabilitas mutasinya 100%, keseluruhan kromosom diubah. Jika probabilitas mutasi 0%, maka tidak ada yang diubah. probabilitas mutasi dalam ruang algoritma genetika seharusnya diberi nilai yang kecil.

Parameter adalah parameter kontrol algoritma genetika, yaitu ukuran populasi (*popsi*), peluang *Crossover* (*pc*) dan peluang mutasi (*pm*). Rekomendasi untuk menentukan nilai parameter (Sri kusumadewi, 2003) :

1. Untuk permasalahan yang memiliki kawasan solusi cukup besar, De Jong merekomendasikan nilai parameter: (*popsi*; *pc*; *pm*) =(50;0.6;0.001)
2. Bila rata-rata *fitness* setiap generasi digunakan sebagai indikator, maka Grefenstette merekomendasikan : (*popsi*; *pc*; *pm*) = (30;0.95;0.01)
3. Bila *fitness* dari individu terbaik dipantau pada setiap generasi, maka usulannya adalah : (*popsi*; *pc*; *pm*) = (80;0.45;0.01)
4. Ukuran populasi sebaiknya tidak lebih kecil dari 30, untuk sembarang jenis permasalahan.





**Gambar 2.7.** *Flowchart* Algoritma Genetika

### Contoh Penyelesaian

Persoalan TSP yang diselesaikan dengan menggunakan algoritma genetika. Terdapat 5 kota yang akan dilalui oleh seorang pedagang keliling, misalnya Kota A,B,C,D,E. Perjalanan dimulai dari kota A dan berakhir di kota A.

Persoalan TSP akan diselesaikan dengan menggunakan algoritma genetika. Penghentian Kriteria ditentukan apabila dalam beberapa generasi berturut-turut diperoleh nilai fitness yang terendah tidak berubah. Penentuan nilai fitness yang terendah adalah syarat karena nilai tersebut yang merepresentasikan jarak terdekat yang dicari dalam masalah TSP ini. Ada 4 kota yang akan menjadi gen dalam

kromosom yaitu kota-kota selain kota asal.

**a. Inisialisasi**

Misalkan kita menggunakan 6 buah populasi dalam satu generasi, yaitu

$$\text{Kromosom}[1] = [B D E C]$$

$$\text{Kromosom}[2] = [D B E C]$$

$$\text{Kromosom}[3] = [C B D E]$$

$$\text{Kromosom}[4] = [E B C D]$$

$$\text{Kromosom}[5] = [E C B D]$$

$$\text{Kromosom}[6] = [C D E B]$$

**b. Evaluasi Kromosom**

Kita akan menghitung nilai *fitness* dari tiap kromosom yang telah dibangkitkan:

$$\begin{aligned} \text{Fitness [1]} &= AB+BD+DE+EC+CA \\ &= 7 + 2 + 6 + 3 + 5 = 23 \end{aligned}$$

$$\begin{aligned} \text{Fitness [2]} &= AD+DB+BE+EC+CA \\ &= 9 + 2 + 8 + 3 + 5 = 27 \end{aligned}$$

$$\begin{aligned} \text{Fitness [3]} &= AC+CB+BD+DE+EA \\ &= 5 + 7 + 2 + 6 + 9 = 29 \end{aligned}$$

$$\begin{aligned} \text{Fitness [4]} &= AE+EB+BC+CD+DA \\ &= 9 + 8 + 7 + 4 + 9 = 37 \end{aligned}$$

$$\begin{aligned} \text{Fitness [5]} &= AE+EC+CB+BD+DA \\ &= 9 + 3 + 7 + 2 + 9 = 30 \end{aligned}$$

$$\text{Fitness [6]} = AC+CD+DE+EB+BA$$

$$= 5 + 4 + 6 + 8 + 7 = 30$$

### c. Seleksi Kromosom

Karena pada persoalan TSP yang diinginkan adalah kromosom dengan *fitness* yang lebih kecil akan mempunyai probabilitas untuk terpilih kembali lebih besar maka digunakan inverse.

$$Q[i] = 1/\text{Fitness } [i]$$

$$Q[1] = 1/23 = 0.043$$

$$Q[2] = 1/27 = 0.037$$

$$Q[3] = 1/29 = 0.034$$

$$Q[4] = 1/37 = 0.027$$

$$Q[5] = 1/30 = 0.033$$

$$Q[6] = 1/30 = 0.033$$

$$\text{Total} = 0,043 + 0,037 + 0,034 + 0,027 + 0,033 + 0,033 = 0,207$$

Untuk mencari probabilitas kita menggunakan rumus berikut :

$$P[i] = Q[i] / \text{total}$$

$$P[1] = 0.043 / 0.207 = 0.043$$

$$P[2] = 0.037 / 0.207 = 0.037$$

$$P[3] = 0.034 / 0.207 = 0.034$$

$$P[4] = 0.027 / 0.207 = 0.027$$

$$P[5] = 0.033 / 0.207 = 0.033$$

$$P[6] = 0.033 / 0.207 = 0.033$$



Dari probabilitas di atas terlihat bahwa kromosom ke-1 mempunyai *fitness* paling kecil, mempunyai probabilitas untuk terpilih pada generasi selanjutnya lebih besa dari pada kromosom lainnya. Proses seleksi kita menggunakan roulette- wheel, untuk mencari nilai komulatif dari probabilitasnya.

$$C[1]= 0,028$$

$$C[2]= 0,028+0,179 = 0,387$$

$$C[3]= 0,387+0,164 = 0,551$$

$$C[4]= 0,551+0,130 = 0,681$$

$$C[5]= 0,681+0,159 = 0,840$$

$$C[6]= 0,840+0,159 = 1$$

Proses roulette-wheel adalah untuk membangkitkan nilai acak R antara 0-1. Jika  $R[k]<C[k]$  maka kromosom ke-k akan menjadi induk, selain itu pilih kromosom ke-k sebagai induk dengan syarat  $C[k-1] < R[k] < C[k]$ . Lalu akan di putar roulade-wheel sebanyak jumlah kromosom yaitu 6 kali (membangkitkan bilangan acak R).

$$R[1]= 0,314$$

$$R[2]= 0,111$$

$$R[3]= 0,342$$

$$R[4]= 0,743$$

$$R[5]= 0,521$$

$$R[6]= 0,411$$

Setelah itu, populasi baru akan terbentuk yaitu:

$$\text{Kromosom}[1]= [2] = [D B E C]$$

Kromosom[2]= [1] = [B D E C]

Kromosom[3]= [3] = [C B D E]

Kromosom[4]= [5] = [E C B D]

Kromosom[5]= [4] = [E B C D]

Kromosom[6]= [6] = [C D E B]

**d. Crossover (pindah silang)**

Pindah silang pada TSP dapat diimplementasikan dengan skema order Crossover. Pada skema ini, satu bagian kromosom dipertukarkan dengan tetap menjaga urutan kota yang bukan merupakan bagian dari kromosom tersebut. Kromosom yang digunakan sebagai induk dipilih secara acak dan jumlah kromosom yang diCrossover dipengaruhi oleh parameter Crossover probability ( $pc$ ). Misal kita tentukan  $pc = 25\%$ , maka diharapkan dalam 1 generasi ada 50% (3 kromosom) dari populasi mengalami Crossover. Pertama kita bangkitkan bilangan acak  $R$  sebanyak jumlah populasi yaitu 6 kali.

$R[1] = 0,451$

$R[2] = 0,211$

$R[3] = 0,302$

$R[4] = 0,877$

$R[5] = 0,771$

$R[6] = 0,131$

Kromosom ke- $k$  yang dipilih sebagai induk jika  $R[k] < pc$ . Maka yang akan dijadikan induk adalah kromosom[2], kromosom[3], dan kromosom[6]. Setelah memilih induk, proses selanjutnya adalah menentukan posisi Crossover. Hal ini dilakukan dengan membangkitkan bilangan acak antara 1 sampai dengan panjang

kromosom-1. Dalam kasus TSP ini bilangan acaknya adalah antara 1-3. Misalnya bilangan acak didapat 1, maka gen yang ke-1 pada kromosom induk pertama diambil dan kemudian ditukar dengan gen pada kromosom induk kedua yang belum ada pada induk pertama dengan tetap memperhatikan urutannya. Bilangan acak untuk 3 kromosom induk yang akan di-Crossover:

$$C[2]= 2$$

$$C[3]= 1$$

$$C[6]= 2$$

Proses *Crossover*.

$$\text{Kromosom}[2]= \text{Kromosom}[2] \times \times \text{Kromosom}[3]$$

$$= [B D E C] \times \times [C B D E]$$

$$= [B D C E]$$

$$\text{Kromosom}[3]= \text{Kromosom}[3] \times \times \text{Kromosom}[6]$$

$$= [C B D E] \times \times [C D E B]$$

$$= [C D E B]$$

$$\text{Kromosom}[6]= \text{Kromosom}[6] \times \times \text{Kromosom}[2]$$

$$= [C D E B] \times \times [B D E C]$$

$$= [C D B E]$$

Populasi setelah di-Crossover:

$$\text{Kromosom}[1]= [D B E C]$$

$$\text{Kromosom}[2]= [B D C E]$$

$$\text{Kromosom}[3]= [C D E B]$$

$$\text{Kromosom}[4]= [E C B D]$$

$$\text{Kromosom}[5]= [E B C D]$$

Kromosom[6]= [C D B E]

**e. Mutasi**

Dala kasus TSP ini skema mutasi yang digunakan adalah swapping mutation. Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh parameter mutation rate (pm). Proses mutasi diselesaikan dengan cara menukar gen yang dipilih secara acak dengan gen sesudahnya. Jika gen tersebut berada di akhir kromosom, maka ditukar dengan gen yang pertama. Pertama-tama kita hitung dulu panjang total gen yang ada pada satu populasi:

Panjang total gen = jumlah gen dalam 1 kromosom \* jumlah Kromosom (3)

$$= 4 * 6 = 24$$

Untuk memilih posisi gen yang mengalami mutasi dilakukan dengan cara membangkitkan bilangan acak antara 1 – panjang total gen yaitu 1-24. Misal kita tentukan pmm = 20% Maka jumlah gen yang akan di mutase adalah =  $0,2 * 24 = 4,8 = 5$

5 buah posisi gen yang akan dimutasi, setelah diacak adalah posisi 3, 7, 10, 20, 24.

Proses mutase :

Kromosom[1]= [D B C E]

Kromosom[2]= [B D E C]

Kromosom[3]= [C E D B]

Kromosom[4]= [E C B D]

Kromosom[5]= [D B CE]

Kromosom[6]= [E D B C]

Proses algoritma genetika untuk 1 generasi telah selesai. Maka nilai *fitness* setelah 1 generasi adalah :

$$\begin{aligned} \text{Fitness}[1] &= AD+DB+BC+CE+EA \\ &= 9 + 2 + 7 + 3 + 9 = 30 \end{aligned}$$

$$\begin{aligned} \text{Fitness}[2] &= AB+BD+DE+EC+CA \\ &= 7 + 2 + 6 + 3 + 5 = 23 \end{aligned}$$

$$\begin{aligned} \text{Fitness}[3] &= AC+CE+ED+DB+BA \\ &= 5 + 3 + 6 + 2 + 7 = 23 \end{aligned}$$

$$\begin{aligned} \text{Fitness}[4] &= AE+EC+CB+BD+DA \\ &= 9 + 3 + 7 + 2 + 9 = 30 \end{aligned}$$

$$\begin{aligned} \text{Fitness}[5] &= AD+DB+BC+CE+EA \\ &= 9 + 2 + 7 + 3 + 9 = 30 \end{aligned}$$

$$\begin{aligned} \text{Fitness}[6] &= AE+ED+DB+BC+CA \\ &= 9 + 6 + 2 + 7 + 5 = 29 \end{aligned}$$

Sebelumnya telah ditentukan kondisi berhenti, yaitu jika dalam beberapa generasi berturut-turut diperoleh nilai *fitness* yang terendah tidak berubah. Pada 1 generasi telah terlihat bahwa terdapat nilai *fitness* terkecil yang tidak berubah. Jika perhitungan dilanjutkan hingga ke generasi ke-N maka nilai *fitness* yang terendah tetap tidak akan berubah. Walaupun perhitungan cukup dijabarkan hingga generasi ke-1 saja namun solusi yang mendekati optimal telah didapatkan. Oleh karena itu, terbukti bahwa algoritma genetika dapat menyelesaikan persoalan TSP

### 2.3 Penelitian Sebelumnya

Beberapa artikel digunakan sebagai acuan pembelajaran, berikut artikale yang digunakan sebagai wacanan antara lain :

1. Penelitian yang dilakukan Yesri Elva (2019) pada penelitian ini Proses pembuatan jadwal mata pelajaran sekolah dasar dapat dilakukan dengan menggunakan Algoritma Genetika. Melalui langkah-langkah yang digunakan dalam Algoritma Genetika yaitu menentukan batasan penjadwalan, merepresentasikan nilai kromosom, menginisialisasi populasi awal, melakukan seleksi, crossover, mutasi, hingga mencapai kondisi selesai. Output dari sistem penjadwalan pada penelitian ini berupa jadwal mata pelajaran sekolah dasar yang tidak bentrok dan dapat membantu pihak sekolah dalam penentuan jadwal mengajar pada SMKN 3 Pariaman
2. Penelitian yang dilakukan Delpiah Wahyuningsih dan Ellya Helmud (2020) Jadwal yang masih dilakukan secara manual pada MTS Negeri 1 Pangkalpinang ini membutuhkan waktu untuk mengatur slot guru, kelas, mata pelajaran, dan waktu dimana pada MTS jam guru telah ditentukan oleh departemen agama sehingga membutuhkan waktu yang cukup lama untuk memproses terbentuknya jadwal. Penelitian ini bertujuan untuk memanfaatkan algoritma genetika dalam mengoptimalkan pembuatan jadwal dengan waktu yang singkat. Algoritma genetika merupakan algoritma yang efektif dalam mengatasi pembuatan jadwal. Hasil dari pengujian data dilakukan dengan 15, 20, 25 dan 30 mata pelajaran. Pengujian dengan 15 mata pelajaran membutuhkan waktu 19,56 detik untuk terbentuknya jadwal dan tidak ada jadwal yang bentrok sedangkan dengan 20 data mata pelejaran waktu untuk memproses terbentuknya jadwal membutuhkan waktu 42,15 detik, 25 data dengan waktu 94,07 detik dan 30 data dengan waktu 471,60. Rata-rata waktu yang dibutuhkan untuk memproses data tersebut



yaitu 156,845 detik.

3. Penelitian yang dilakukan Edi Saputra, Guidio L, Ginting, ikwan Lubis (2019) permasalahan umum yang selalu menjadi beban bagi bidang kurikulum. Masalah yang sering muncul adalah jam mengajar yang sama oleh seorang guru dalam ruangan yang berbeda. Hal ini sangat mengganggu waktu belajar siswa. Sehingga di butuhkan metode optimasi untuk menyelesaikan masalah ini. Salah satu metode optimasi untuk permasalahan ini adalah melalui metode Algoritma Genetika. Penerapan metode Algoritma Genetika untuk Penjadwalan Mata Pelajaran dikembangkan lebih lanjut ke dalam suatu sistem terkomputerisasi yang dapat mengganti cara manual penjadwalan mata pelajaran dan dapat menghasilkan jadwal yang lebih akurat dengan aturan yang ada dalam waktu yang lebih singkat. Dari penelitian ini di harapkan terciptanya suatu aplikasi penjadwalan yang optimal dan dapat di gunakan untuk menyelesaikan masalah penjadwalan kegiatan belajar mengajar pada SMP N 1 Peunaron.
4. Penelitian sebelumnya dilakukan Hendri Ardiansyah dan Mochammad Bagoes Satria Junianto (2022) Penjadwalan mata pelajaran di sekolah sering kali memiliki masalah terhadap jalannya kegiatan belajar mengajar. Terdapat berbagai macam variasi dalam permasalahan tersebut, dikarenakan perbedaan kebijakan yang berbeda-beda dalam penjadwalan, di SD Lazuardi Global Compassionate School Kota Depok pembuatan jadwal dilakukan dengan cara membuat gambaran kasar jadwal kemudian dipindahkan ke aplikasi google sheet, penjadwalan dengan cara ini masih sering terjadi ketidaksesuaian penjadwalan yang dibuat dan juga memerlukan waktu yang cukup lama. Hal ini menyebabkan penjadwalan yang optimal sulit didapatkan. Penelitian ini membahas

penerapan algoritma genetika untuk penjadwalan mata pelajaran di SD Lazuardi Global Compassionate School. Algoritma genetika adalah algoritma yang dapat memecahkan masalah dalam penjadwalan mata pelajaran dan memberikan solusi untuk mendapatkan hasil penjadwalan yang optimal. Berdasarkan hasil pengujian yang dilakukan, parameter yang digunakan adalah jumlah populasi, jumlah generasi, persilangan dan mutasi bahwa algoritma genetika dapat bekerja dengan baik dan menyelesaikan masalah dengan menghasilkan penjadwalan yang optimal dan mempersingkat waktu pembuatan jadwal sehingga pembuatan jadwal menjadi efektif dan efisien.

5. Penelitian sebelumnya yang dilakukan Romauli Statina Sihombing (2014) Penempatan Buku di dalam rak, di perpustakaan sekolah, harus bisa seoptimal mungkin untuk mempermudah dalam hal mencari buku. Algoritma genetika sebagai metode pencarian solusi yang berpatokan pada seleksi alam mendapatkan sebuah individu dengan susunan gen-gen terbaik, mampu memberikan solusi bagaimana pola penyusunan barang dioptimalkan melalui proses iterasi sampai beberapa generasi dengan operatornya yaitu, pembentukan populasi awal, crossover( pindah silang), dan mutase.
6. Penelitian sebelumnya dilakukan Aulia Fitrah, Achmad Zaky (2015), Fitrasani Persoalan pedagang keliling (*Travelling Salesperson Problem-TSP*) merupakan persoalan optimasi untuk mencari jalan terpendek bagi pedagang keliling yang ingin berkunjung ke beberapa kota, dan kembali ke kota asal keberangkatan. Dengan metode algoritma genetika dapat membantu menemukan solusi untuk menentukan jalan terpendek yang melalui kota lainnya hanya sekali dan kembali ke kota asal keberangkatan.