

V. MENENTUKAN NILAI MINIMUM DARI SEBUAH FUNGSI OBJEKTIVE DGN MENGGUNAKAN ALGORITMA GENETIKA (GA)

5.1 Pendahuluan

Algoritma genetika baru-baru ini telah menjadi subjek yang sangat menarik dan relatif berkembang terutama karena keunggulannya dalam menyediakan suatu prosedur kerja yang tangguh (robust) dalam memecahkan berbagai persoalan didalam dunia science. Hal ini salah satunya dikarenakan GA didasarkan atas gagasan dari ilmu genetika dan proses seleksi secara alamiah.

Kita akan mengaplikasikan GA ini terutama untuk memecahkan persoalan –persoalan yang terkait dengan optimasi. Pada dasarnya GA ini bekerja dengan suatu populasi awal yang mana biasanya dituliskan dalam bentuk numerik dan disebut sebagai variabel khusus . ukuran dari populasi ini bisa bervariasi dan pada umumnya dikaitkan dengan persoalan yang akan dipecahkan. Keanggotaan dari populasi ini biasanya dinyatakan dalam bentuk bilangan biner : 1 dan 0. Jadi bilangan biner ini juga dapat dikodekan menjadi variabel yang hendak kita uji. Populasi awal ini dibangkitkan oleh sinyal random dan kita bisa menggunakan terminologi genetika untuk mengkarakterisasinya. Masing- masing deretan (string) biner ini adalah menunjukkan populasi dari kromosom dan elemen – elemen binernya adalah menyatakan gen. Sebuah populasi yang baru harus bisa dikembangkan dari populasi awal dan untuk melakukan ini perlu diimplementasikan sebuah proses genetika alamiah yang pada dasarnya ditempuh dalam tiga tingkatan sbb:

1. Reproduksi atas dasar kelayakan (fitness)
2. Crossover
3. Mutation

Sekumpulan kromosom akan dipilih pada tahap reproduksi , berdasarakan seleksi alamiah . dengan demikian keanggotaan dari populasi itu , dipilih berdasarakan beberapa kriteria, salah satunya akan dilakukan Fittest (uji kelayakan).

Selanjutnya akan didefinisikan mengenai Crossover. Sebuah proses aktual akan diaplikasikan dengan menggunakan konsep crossover (tukar silang). Yang dimaksud crossover disini adalah ketika ada dua buah anggota populasi saling bertukar gen. Ada beberapa cara yang digunakan dalam crossover ini , sebagai contoh ada yang disebut titik crossover . titik –titik crossover ini akan dipilih secara acak . crossover sederhana dapatt diilustrasikan sebagai berikut:

Misalnya kita punya titik crossover yang dipilih setelah empat digit :

1111 | 000
1010 | 101 dan setelah dilakukan proses cross over bentuk chromosomnya akan menjadi;

1111 | 101
1010 | 000

dari penerapan konsep crossover ini akan menghasilkan generasi baru (new generation).

Proses terakhir yang harus dilalui adalah mutasi. Disini kita secara acak akan merubah gen-gen tertentu pada kromosom tertentu pula, (dengan kata lain misalnya 0 akan diganti dengan 1 atau sebaliknya). Proses mutasi dalam GA ini umumnya berlangsung sangat jarang, karenanya peluang perubahan didalam ‘string’ ini kita jaga sangat lambat. Sebagai mana yang telah disebutkan dalam konsep dasar tentang GA maka disini akan diaplikasikan GA untuk memecahkan permasalahan yang terkait dengan Optimasi

5.2 Problem Solving Optimasi

Seperti permasalahan yang telah disebutkan sebelumnya, yaitu menentukan minimisasi dari fungsi cost pembuatan kotak suara sebagai berikut:

$$\text{Minimize } F(L) = 0.0078 L^3 + 0.0052 L^2 + 0.004 L^{-1} + 0.0052$$

$$\text{Subject to : } L \leq 1$$

$$V = 1$$

Penerapan Algoritma genetika untuk menyelesaikan ini dinyatakan dengan beberapa langkah sbb;

1. Membangkitkan ‘string awal’ sebagai populasi awal, yang mana jumlah bit pada masing –masing string ini menunjukkan panjang string, bila untuk populasi awal kita membangkitkan 5 kromosom dan masing-masing kromosom terdiri dari 6 gen. Dengan menggunakan command sbb:

chroms = genbin(6,5) didapatkan fungsinya:

```
chroms=genbin(6,5)
```

```
chroms =
```

```
1 0 1 0 1 1  
0 0 1 0 1 1  
1 1 0 0 1 1  
0 1 0 0 1 0  
0 0 0 1 0 0
```

2. Mengkonversi fungsi yang dibangkitkan oleh sinyal random tersebut diatas, menjadi bilangan real yang terletak antara $L = 0$ sampai $L = 1$ m
caranya dengan command sbb:

```
for i=1:5, rval(i)=binvreal(chroms(i,:),0,1); end;  
>> rval
```

```
rval =
```

```
0.6825 0.1746 0.8095 0.2857 0.0635
```

3. melakukan langkah reproduksi dengan pemilihan chromosom yang dikehendaki.
Dengan menggunakan Command:

```
matepool=selectga('f805',chroms,0,1)
```

```
matepool =
```

```
1 0 1 0 1 1  
1 0 1 0 1 1  
1 0 1 0 1 1  
1 0 1 0 1 1  
1 0 1 0 1 1
```

dan ini dinyatakan sebagai populasi baru dari kromosom sebelumnya yang dibangkitkan oleh sinyal random.

- 4..Melakukan uji kelayakan (fitness) terhadap populasi baru
dengan command:

```
fitness('f805',matepool,0,1)
```

```
ans =
```

```
0.0162 0.0162 0.0162 0.0162 0.0162
```

dan ini menunjukkan bahwa untuk nilai L antara 0 sampai 1 m , tingkat kelayakannya sama

5. Melakukan Crossover (pertukaran gen) untuk masing masing Kromosom.

Bila prosentase pertukaran gen ini dibatasi hanya sekitar 60 %, maka dengan command berikut akan diperoleh gen baru dari gen yang telah dipilih sebelumnya;

```
newgen=matesome(matepool,.6)
```

chrom1 =

```
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
```

newgen =

```
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
```

6. Melakukan mutasi terhadap sebagian gen, sebelum dilakukan pengulangan terhadap langkah –langkah sebelumnya

Jika nilai mu (mutation value) yang diberikan adalah sangat kecil, maka dengan command:

```
mutate(newgen,.005)
```

ans =

```
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
0 1 1 0 1 1
```

ini adalah kromosom yang sebelumnya, dan dipanggil kembali, dan inilah yang disebut sebagai gen baru. Kejadian ini akan terus diulang hingga didapatkan kromosom dengan gen yang memenuhi kelayakan maksimum, nah kromosom

ini yang nantinya merepresentasikan dari nilai L yang sebelumnya dinyatakan dalam 'string', dalam hal ini adalah kromosom

7. Menentukan nilai L optimum dan minimum Fungsi Costnya dengan menggunakan fungsi *optga* (*optimization genetic Algorithm*).

Dalam menggunakan fungsi Optga ini, dipersyaratkan untuk memasukkan beberapa persyaratan, seperti:

- a. Range L = [0 1]
- b. Panjang bit Kromosom = 6 bit
- c. Populasi awal = 10
- d. Pengulangan proses 100 generasi
- e. Probability mutasi = 0.001
- f. Crossover = 0.5

Dengan menggunakan command :

```
[x,f]=optga('f805',[0 1],8,10,20,.005,.6)
```

dan hasilnya adalah:

L= 0.6157	(Lebar Kotak suara dalam meter)
------------------	-----------------------------------

f (L) =0.0158	(fungsi cost minimumnya untuk L tersebut diatas)
----------------------	---