

# IMPLEMENTASI ALGORITMA GENETIKA UNTUK MENENTUKAN SOLUSI PADA OPTIMISASI NONLINIER

Hery Andi Sitompul, S.Si, M.Si  
Dosen Kopertis Wilayah I Dpk Fak.Teknik Universitas Darma Agung

## ABSTRACT

Determination of a set solution of nonlinear optimization problem is a complicated task, by that means the an exact method or technique is necessary. Genetic Algorithm is one of the most searching method or technique that has developed in the last few decade. Finding a set solution of nonlinear optimization problem with Genetic Algorithm technique is put into effect by considering parameter of objective function like as chromosome of living organisms, all living organisms whose have the best fitness will be survived and has ever evolution and reproduction an offsprings as a new generation. These generation certainly has have better fitness. By considering that an objective function as a fitness value thus optimum value will be obtained at the moment of specific generation full filled the terminating criteria. Chromosome of population in these generation will be represent a set optimum solution of nonlinier optimization.

**Keywords :** *Optimization, Genetic Algorithm, Objective Function*

## 1. Pendahuluan

Masalah optimasi sering kita lakukan dalam kehidupan sehari – hari untuk mengambil sebuah keputusan dalam memenuhi kebutuhan kita, baik itu secara sadar maupun tidak. Sebagai contoh, bagaimana kita mengambil tindakan untuk mengoptimalkan pengeluaran setiap bulan dengan penghasilan yang sudah tertentu. Dalam bidang rekayasa misalnya, optimasi dibutuhkan untuk mencari solusi termurah dengan segala kendala yang ada. Tetapi pada umumnya optimasi dilakukan secara intuitif tanpa berdasarkan teori optimasi.

Sudah sejak lama masalah optimasi menjadi kajian penelitian para engineering dan ilmuwan untuk mengembangkan teori optimasi. Salah satunya adalah masalah optimasi nonlinier.

Masalah optimasi nonlinier adalah masalah yang sangat kompleks dan susah diprediksi. Sehingga optimasi dengan metode gradien tidak selalu dapat berhasil untuk menyelesaikan permasalahan ini. Hal ini memberikan kesempatan berkembangnya algoritma lain yang tidak berdasarkan metode gradien.

Salah satu algoritma yang diusulkan dalam penelitian beberapa dekade belakangan ini adalah Algoritma Genetika (GA). Algoritma Genetika pertama kali diusulkan oleh Jhon Holland dan rekan – rekannya dari universitas Michigan dalam buku berjudul “*Adaption in Natural and Artificial Systems*” pada tahun 1975.

Algoritma Genetika adalah teknik pencarian yang membutuhkan perhitungan secara komputasi untuk menemukan penyelesaian perkiraan untuk optimisasi dan masalah pencarian.

Algoritma genetika adalah kelas khusus dari algoritma evolusioner dengan menggunakan teknik yang terinspirasi oleh biologi evolusioner seperti warisan, mutasi, seleksi alam dan rekombinasi (atau *crossover*)

Dalam tulisan ini akan diperlihatkan bagaimana menentukan solusi dari permasalahan optimasi nonlinier dengan menggunakan Algoritma Genetika.

## 2. Dasar Teori

### 2.1. Optimasi Nonlinier

Permasalahan yang dihadapi dalam optimasi adalah mencari solusi

yang optimal, optimal dalam arti yang terbaik, yang paling menguntungkan.

Suatu permasalahan optimasi disebut nonlinier jika fungsi tujuan dan kendalanya mempunyai bentuk nonlinier pada salah satu atau keduanya.

Terdapat beberapa sub kelas dari masalah optimasi, bentuk yang paling sederhana adalah bentuk minimasi tanpa kendala :

$$\min_{x \in \mathbb{R}^n} f(x) \text{ atau } \max_{x \in \mathbb{R}^n} f(x)$$

Dengan fungsi tujuan  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

Bentuk satu level diatas adalah :

$$\min_{x \in \mathbb{R}^n} f(x) \text{ atau } \max_{x \in \mathbb{R}^n} f(x)$$

dengan kendala  $c(x) \geq 0$

Dengan fungsi kendala  $c: \mathbb{R}^n \rightarrow \mathbb{R}^m$ ,  $m \leq n$

Bentuk lebih umum adalah :

$$\min_{X \in \mathbb{R}^n} f = f(X) \quad \text{atau}$$

$$\max_{X \in \mathbb{R}^n} f = f(X)$$

Kendala :  $G_j(X) = 0, j = 1, 2, \dots, m$

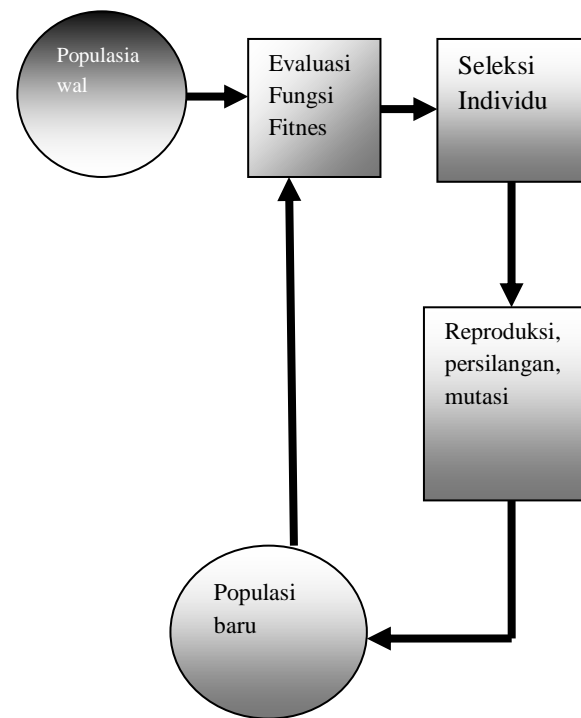
$X = \{x_1, x_2, \dots, x_n\}^t$ , dimana  $m \leq n$

Solusi dari masalah optimasi diatas adalah nilai  $X^* = \{x_1^*, x_2^*, \dots, x_n^*\}^t$  yang memberikan nilai minimum atau maksimum  $f$  dan memenuhi keadaan  $G$ .

## 2.2. Algoritma Genetika

Gagasan dari Algoritma Genetika adalah teori evolusi Darwin, dimana parameter dari suatu fungsi objektif dianggap mempunyai kesamaan dengan makhluk hidup. Untuk mempertahankan kelangsungan hidup, maka satu individu harus mempunyai tingkat kemampuan bertahan (*fitness*) yang tinggi sehingga memenangkan persaingan dengan individu lain, seiring berjalan waktu individu juga mengalami mutasi untuk menyesuaikan diri dengan ekosistem alam, dan melakukan perkawinan antar individu untuk menghasilkan individu baru. Keadaan ini berlangsung terus dari generasi sampai mencapai keadaan tingkat fitness optimum sesuai dengan kriteria fungsi objektif.

Secara umum tahapan kerja Algoritma Genetika dapat dinyatakan dengan :



**Populasi awal.** Populasi awal adalah sekelompok individu yang dibangkitkan secara acak untuk mewakili nilai awal pencarian solusi. Setiap individu dalam generasi awal ini dinyatakan dengan bit – bit string biner yang disebut dengan kromosom.

**Evaluasi fungsi fitness.** Evaluasi fungsi fitness adalah penghitungan nilai fungsi tujuan untuk setiap individu didalam satu kelompok populasi, dan dilakukan terus menerus dalam setiap generasi hingga kriteria pemberhentian terpenuhi.

**Seleksi individu.** Seleksi dilakukan untuk mendapatkan calon induk yang baik. “Induk yang baik akan menghasilkan keturunan yang baik”. Semakin tinggi nilai fitness sebuah individu maka peluang terpilih menjadi indukan semakin besar.

**Reproduksi, Persilangan, Mutasi.** Reproduksi adalah pembentukan individu baru yang merupakan hasil persilangan dari dua individu yang terseleksi. Mutasi merupakan pertukaran nilai gen pada sebuah individu dengan tujuan fungsi tujuan tidak terjebak pada optimasi lokal **Populasi baru.** Sekelompok individu yang merupakan hasil reproduksi,

persilangan, dan mutasi dari individu – individu yang terpilih pada generasi sebelumnya.

### 3.Sistem Kerja Algoritma Genetika Pada Optimasi Nonlinier

Misalkan fungsi objektif dari masalah optimisasi nonlinier adalah  $f(x,y)$ , dengan batasan  $a \leq x \leq b$  dan  $c \leq y \leq d$ . Solusi dari permasalahan adalah pasangan  $(x,y)$  sehingga setiap individu didefinisikan sebagai pasangan  $(x,y)$  yang dinyatakan dalam representasi string biner.

Representasi variabel keputusan/solusi dilakukan dengan membangkitkan bit biner secara acak yang disebut dengan kromosom. Sebagai contoh akan digunakan 16 bit string untuk mewakili variabel, dari hasil pembangkitan secara acak diperoleh: 1110011011001100011010110000010. 16 bit pertama adalah mewakili variabel  $x$  dan 16 bit terakhir mewakili variabel  $y$ . Selanjutnya bit – bit tersebut akan dikonversi kedalam bilangan ril dengan aturan berikut :

$$x = a + x' \times \frac{b - a}{(2^{16} - 1)}$$

$$y = a + y' \times \frac{d - c}{(2^{16} - 1)}$$

Dengan  $x'$  dan  $y'$  adalah bilangan desimal dari representasi bit biner.

Pembentukan individu awal dilakukan dengan membangkitkan secara acak sejumlah kromosom masing – masing individu dengan jumlah populasi tertentu. Sebagai contoh, dari pembangkitan secara acak kromosom individu dengan 16 bit dan ukuran populasi 10 adalah :

$I_1 = [111100110110011000110101110000010]$

$I_2 = [11000000001100011011110001010101]$

$I_3 = [110000010100001010111010100011000]$

$I_4 = [01001000111100011001110110001100]$

$I_5 = [00110011011111100101110110001010]$

$I_6 = [01000110111110000000011110011000]$

$I_7 = [01101000000111101000011001111010]$

$I_8 = [0001011000010111101011111110000]$

$I_9 = [10110111110111111010110000111111]$

$I_{10} = [0000000010111111100110101111111]$

Evaluasi fungsi fitness dilakukan dengan menghitung nilai fungsi objektif untuk setiap individu kemudian mengurangkannya dengan dengan nilai maksimum dari fungsi objektif.

Setelah nilai fitness dihitung, selanjutnya adalah pembentukan populasi baru dari generasi sekarang dengan cara reproduksi, seleksi, persilangan, dan mutasi.

Reproduksi dilakukan dengan memilih dua individu yang mempunyai nilai fitness paling tinggi. Kedua individu ini akan menghasilkan keturunan untuk generasi berikutnya. Seleksi pemilihan indukan dilakukan berdasarkan probabilitas kumulatif tiap individu. Probabilitas kumulatif dihitung dengan :

1. Hitung jumlah total nilai fitness setiap individu
2. Hitung probabilitas setiap individu :  $P_i = \frac{fitness_i}{totalfitness}$
3. Hitung probabilitas kumulatif setiap individu :  $Q_i = \sum_{k=0}^i P_k$
4. Bangkitkan bilangan acak  $r \in [0,1]$
5. Jika  $Q_{i-1} < r \leq Q_i$  pilih individu ke- $i$  menjadi indukan pertama
6. Ulangi langkah 4 dan 5 untuk mendapatkan indukan kedua
7. Lakukan persilangan indukan satu dan dua untuk mendapatkan 2 keturunan baru.
8. Ulangi langkah 4,5,6, dan 7 sampai diperoleh ukuran populasi yang baru.

Sedangkan persilangan dilakukan dengan menukar gen indukan satu dengan indukan dua sesuai dengan probabilitas persilangan. Misalkan yang terpilih sebagai indukan adalah  $I_1$  dan  $I_2$ ,

probabilitas persilangan adalah 25%, maka keturunan yang baru adalah :

$I_1 = [1\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1]$

$I_2 = [1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 0]$

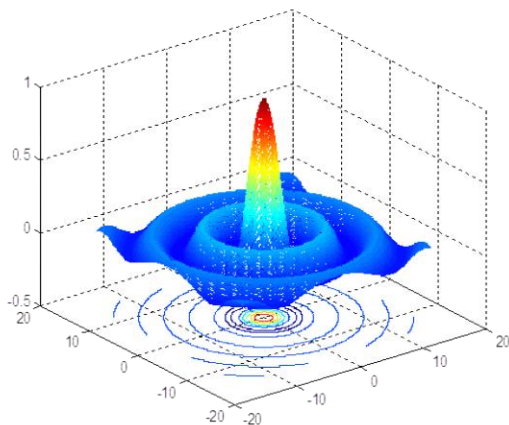
Selanjutnya mutasi dilakukan dengan menukar gen 1 menjadi 0 atau 0 menjadi 1 pada posisi yang dipilih secara acak, yaitu dengan cara membangkitkan bilangan bulat dari 1 sampai total ukuran gen. Misal untuk ukuran populasi 10, jumlah bit 32 maka total adalah 320, andaikan mutasi rate adalah 0,01. Maka jumlah gen yang mengalami mutasi adalah  $320 \times 0,01 = 3,2 \approx 3$ . Bangkitkan bilangan bulat antara 1 sampai 320 sebanyak 3 bilangan, misalnya adalah 10,36,dan 67. Gen ke-10 pada individu 1, gen ke-4 individu 2, dan gen ke-3 individu 3 diganti dari 0 menjadi 1 atau sebaliknya.

#### 4.Hasil dan Pembahasan

##### 4.1. Optimisasi Nonlinier Tanpa Kendala

Untuk memperlihatkan Algoritma genetika dapat bekerja pada masalah optimasi, berikut ini akan diberikan contoh penggunaannya. Misalkan akan dicari nilai maksimum dari  $f(x,y) = \frac{\sin\sqrt{x^2+y^2}}{\sqrt{x^2+y^2}}$ , untuk batasan  $-20 \leq x \leq 20, -20 \leq y \leq 20$

Berikut adalah plot dari fungsi diatas, terlihat bahwa nilai maksimum mendekati 1.



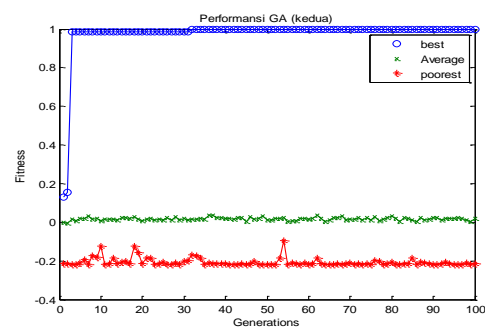
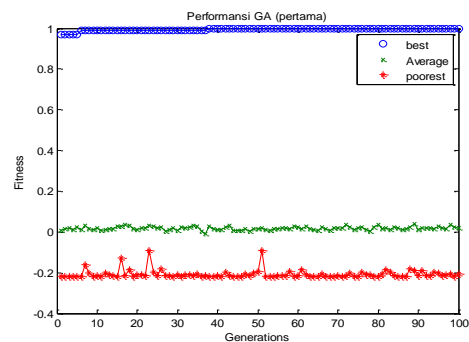
Kasus diatas dapat dapat diformulasikan dengan:

$$\max_{x,y \in \mathbb{R}} \left( \frac{\sin\sqrt{x^2+y^2}}{\sqrt{x^2+y^2}} \right), \text{ dengan } -20 \leq x \leq 20, -20 \leq y \leq 20$$

Dari hasil eksekusi Algoritma Genetika dengan pemrograman Matlab untuk kasus diatas adalah :

Tabel 1. Hasil Algoritma Genetika pada eksperimen 1(2 kali running)

Item	GA (Pertama)	GA (kedua)
Iterasi	100	100
Cpu time (s)	6.21875	5.65625
Optimum	x=0.114443	x=0.069886
	y=0.041810	y=-0.094911
Nilai maksimum	0.997528	0.997686



Pada eksperimen pertama ini, jumlah populasi yang digunakan adalah 120, rasio/tingkat persilangan adalah 0,9. Tingkat/rasio mutasi adalah 0,1 dan jumlah gen untuk masing - masing variabel adalah 16.

Dari tabel terlihat ada perbedaan yang sangat kecil dari hasil running pertama dan kedua, walaupun operator yang digunakan adalah sama. Hal ini disebabkan oleh faktor acak pada gen

yang dibangkitkan, dan akan berbeda setiap program dijalankan, dan juga nilai fungsi akan sama untuk beberapa pasangan (x,y) yang berbeda seperti yang terlihat pada gambar grafik fungsi objektif. Tetapi secara umum tidak memberikan perbedaan yang signifikan. Dari grafik performansi GA (Genetic Algorithm) pertama dan kedua terlihat bahwa algoritma ini cukup cepat mencapai konvergen ke nilai optimum.

#### 4.2. Optimisasi Nonlinier Dengan Kendala

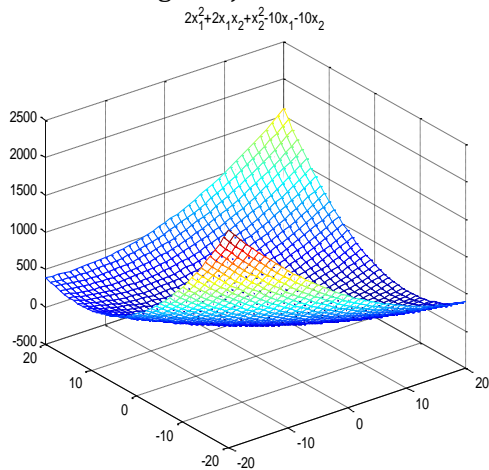
Berikut ini adalah contoh penggunaan Algoritma Genetika untuk optimisasi nonlinier dengan kendala.

$$\min_{x_1, x_2 \in \mathbb{R}} f(x_1, x_2) = 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2$$

Terhadap kendala :

$$\begin{aligned} x_1^2 + x_2^2 - 5 &\leq 0 \\ 3x_1 + x_2 - 6 &\leq 0 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

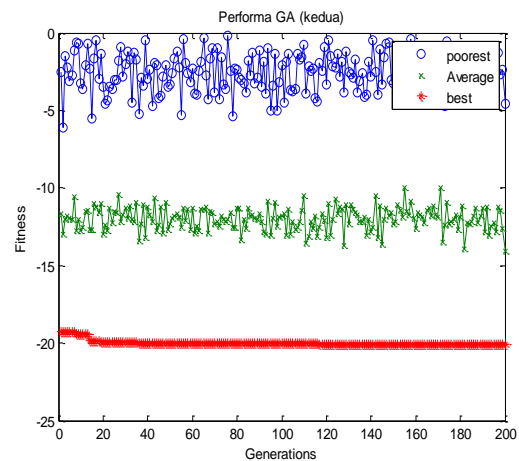
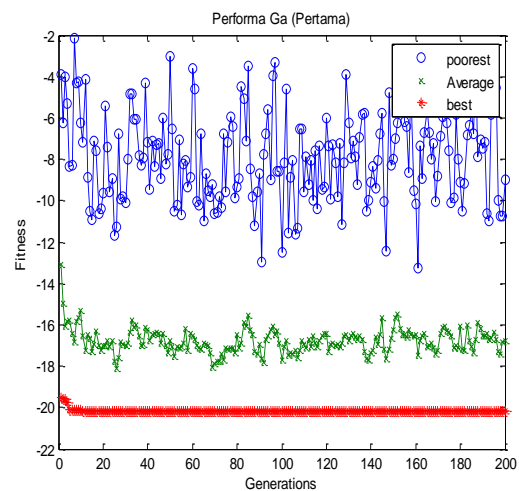
Grafik dari fungsi objektif diatas adalah :



Dari hasil eksekusi Genetika Algoritma dengan pemrograman Matlab diperoleh : Tabel 2. Hasil Algoritma Genetika pada eksperimen 2 (2 kali running)

Item	GA (Pertama)	GA (kedua)
Iterasi	200	200
Ukuran populasi	50	30
Rasio	0.1	0.7

mutasi		
Cpu time (s)	8.18750	5.81250
Optimum	$x_1=1.099933$ $x_2=2.000000$	$x_1=1.096072$ $x_2=1.972595$
Nilai minimum	-20.179893	-20.068579



Dari kedua grafik performansi GA (Genetic Algorithm) dengan operator GA yang berbeda, dapat dilihat bahwa pada eksekusi pertama proses iterasi sudah mendekati konvergen pada iterasi ke-11, pada eksekusi kedua iterasi mendekati konvergen pada iterasi ke-14.

## 5. Kesimpulan dan Saran

### 5.1. Kesimpulan

Dari pembahasan dan setelah mengaplikasikan Algoritma Genetika pada optimisasi nonlinier, dapat diambil kesimpulan :

1. Algoritma Genetika sangat baik dalam mencari solusi optimum pada masalah optimasi nonlinier baik dengan kendala maupun tanpa kendala.
2. Secara matematis Algoritma Genetika tidaklah serumit metode gradien, tetapi secara komputasi algoritma ini cukup rumit.
3. Karena faktor acak dalam algoritma ini, maka setiap kali menjalankan programnya akan diperoleh hasil yang tidak sama, walaupun perbedaannya dapat dikatakan tidak signifikan.

## 5.2. Saran

Berdasarkan pengalaman dalam kasus yang disajikan dalam jurnal ini dan kasus lain yang pernah dicoba, saran yang dapat diberikan adalah:

1. Sebaiknya ukuran populasi jangan terlalu kecil, populasi  $\geq 30$ . Tetapi jangan juga terlalu besar karena akan memakan waktu komputasi.
2. Bagi orang yang baru pertama kali menggunakan dan membuat program untuk algoritma ini, sebaiknya dibandingkan dengan hasil metode lain dan program dijalankan secara berulang ulang dengan mengganti nilai operator genetika seperti : rasio mutasi, rasio persilangan, ukuran populasi, dan jumlah generasi, jika hasil yang diperoleh tidak berbeda secara signifikan pada setiap kali program dijalankan maka nilai optimum dapat dikatakan sudah tercapai.

## Daftar Pustaka

1. David G. Luenberger, Yinyu Ye, *Linear and Nonlinear Programming* Third Edition, USA : Springer, 2007.
2. Igor Griva, Stephen G. Nash, Ariela Sofer, *Linear and Nonlinear Programming, Sec.Edition.* Philadelphia : SIAM, 2008.
3. Johannes Jahn, *Introduction to the Theory of Nonlinear Optimization, Third Edition,* Berlin : Springer-Verlag, 2007
4. Jorge Nocedal, Stephen J. Wright, *Numerical Optimization, Sec. Edition, USA : Springer, 2000.*
5. Michael Bartholomew-Biggs, *Nonlinear Optimization with Engineering Applications, Vol.19,* New York : Springer, 2008.
6. Paul Charbonneau, *An Introduction to Genetic Algorithms for Numerical Optimization,* Boulder, Colorado : National Centre for Atmospheric Research, 2002.
7. Randy L. Haupt, Sue Ellen Haupt, *Practical Genetic Algorithms, Sec.Edition,* New York : John Wiley & Sons Inc, 2004
8. Singiresu S. Rao, *Engineering Optimization, Theory and Practice, Third Edition,* New York : John Wiley & Sons.Inc, 1996.