

# STUDI ANALISA PELATIHAN JARINGAN SYARAF TIRUAN DENGAN DAN TANPA ALGORITMA GENETIKA

**Agustinus Noertjahyana**

Fakultas Teknologi Industri, Jurusan Teknik Informatika – Universitas Kristen Petra  
e-mail: agust@peter.petra.ac.id

**Yulia**

Fakultas Teknologi Industri, Jurusan Teknik Informatika – Universitas Kristen Petra  
e-mail : yulia@petra.ac.id

**ABSTRAK:** Neural network sebagai suatu sistem pengolah informasi yang mempunyai kemiripan dengan jaringan otak manusia, belakangan ini sering digunakan untuk menyelesaikan suatu permasalahan. Neural network memiliki beberapa karakteristik yang ditentukan oleh : arsitektur, algoritma pelatihan serta fungsi aktivasi.

Algoritma genetika adalah suatu metode untuk mendapatkan suatu fungsi yang optimal dengan menggunakan operasi-operasi genetika yang dilakukan pada tiap-tiap individu yang terdapat dalam suatu populasi yang seringkali disebut sebagai kromosom.

Dalam hal ini untuk mendapatkan kemungkinan suatu fungsi bisa lebih optimal adalah dengan menggabungkan antara neural network dengan algoritma genetika. Adapun caranya adalah dengan melakukan proses konversi bias dan bobot pada neural network ke dalam bentuk individu pada algoritma genetika dan demikian sebaliknya. Sehingga nantinya bisa didapatkan suatu kesimpulan antara metode pelatihan neural network dengan dan tanpa menggunakan algoritma genetika.

**Kata kunci:** jaringan saraf tiruan, olgoritma genetika, kecerdasan buatan.

**ABSTRACT:** Neural network as an information processor system which has some similarities with human brain, is lately used to solve general problems. Neural network has several characteristics based on : architecture, learning algorithm, and activation function.

Genetic algorithm is a method to get an optimum function using genetic operations which is done to each individual in a population that is often called as chromosome.

This way, to get a more optimum function is by colliding neural network and genetic algorithm. That is by doing a bias conversion and the weight to the neural network into a kind of individual form of genetic algorithm and the other way around. Thus, a conclusion can be drawn about neural network learning method with or without genetic algorithm.

**Keywords:** neural network, genetic algorithm, artificial intelligence.

## 1. DEFINISI DAN KONSEP DASAR NEURAL NETWORK

Neural network adalah sistem pengolah informasi yang mempunyai karakteristik mirip dengan jaringan syaraf biologis, yaitu jaringan syaraf pada otak manusia. Karakteristik dari neural network ditentukan oleh beberapa hal, yaitu:

1. Arsitektur, merupakan bentuk pola hubungan antara neuron-neuronnya.
2. *Learning algorithm* (algoritma untuk proses belajar), merupakan metode yang digunakan untuk menentukan bobot dari hubungannya.
3. Fungsi aktivasi, merupakan fungsi untuk menghasilkan output.

### 1.1 Arsitektur Jaringan

Pada neural network, neuron-neuron tersusun dalam *layer*. Pengaturan neuron

dalam *layer* dan hubungan-hubungannya disebut dengan arsitektur jaringan. Neural network dapat diklasifikasikan menjadi dua jenis, yaitu *single layer* dan *multilayer*. Dalam jaringan *single layer*, neuron-neuron dapat dikelompokkan menjadi dua bagian, yaitu *input units* (unit-unit input) dan *output units* (unit-unit output). Sedangkan dalam jaringan *multilayer*, selain ada unit-unit input dan output juga terdapat *hidden units* (unit-unit tersembunyi).

### 1.2 Pelatihan Neural Network

Tujuan dari pelatihan neural network adalah untuk mencari bobot-bobot yang terdapat dalam tiap *layer*. Ada dua jenis pelatihan dalam sistem neural network, yaitu proses belajar terawasi (*supervised learning*) dan proses belajar tak terawasi (*unsupervised learning*).

Dalam proses belajar yang terawasi, seolah-olah ada “guru” yang mengajarkan neural network. Cara pelatihan neural network ini adalah dengan memberikan data-data yang disebut *training data* atau *training vectors*, yang terdiri dari pasangan input dan output yang diinginkan dan kemudian diberikan ke neural network sehingga neural network dapat memodifikasi bobot-bobot yang ada untuk mencoba mencari kesamaan antara hasil output yang dihasilkan oleh neural network dengan hasil output yang diinginkan. Setelah proses pelatihan selesai, neural network kemudian diberi suatu nilai input dan akan menghasilkan suatu hasil output.

Dalam proses belajar yang tak terawasi, tidak ada “guru” yang mengajarkan. Neural network hanya diberi data input (*input vectors*) dan tidak dilengkapi dengan suatu hasil output yang diinginkan. Neural network akan memodifikasi bobot sehingga untuk input yang hampir sama, output yang dihasilkan sama.

### 1.3 Fungsi Aktivasi

Berikut ini adalah fungsi aktivasi yang biasa digunakan:

- Fungsi Identitas

$$f(x) = x \tag{1}$$

- Fungsi Sigmoid Biner

$$f(x) = \frac{1}{1 + \exp(-sx)} \tag{2}$$

$$f'(x) = sf(x)[1 - f(x)] \tag{3}$$

dimana  $\sigma$  adalah parameter kecuraman fungsi. Fungsi sigmoid biner mempunyai nilai dengan interval antara 0 dan 1.

- Fungsi Sigmoid Bipolar

$$g(x) = 2f(x) - 1 = \frac{2}{1 + \exp(-sx)} - 1$$

$$= \frac{1 - \exp(-sx)}{1 + \exp(-sx)} \tag{4}$$

$$g'(x) = \frac{s}{2}[1 + g(x)][1 - g(x)] \tag{5}$$

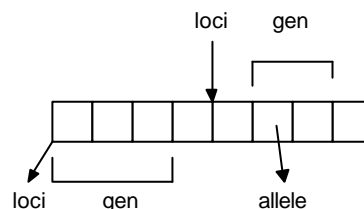
dimana  $\sigma$  adalah parameter kecuraman fungsi. Fungsi sigmoid bipolar mempunyai interval nilai antara -1 dan 1.

### 1.4 Pelatihan Neural Network

Selain untuk meminimalkan error pada output yang dihasilkan oleh jaringan, tujuan lain dari pelatihan neural network adalah memperoleh keseimbangan antara kemampuan untuk merespon *input pattern* yang digunakan pada proses pelatihan secara benar, hal ini dikenal dengan *memorization* dan kemampuan untuk memberikan respon yang masuk akal (baik) terhadap input yang mirip, tetapi tidak sama persis dengan yang digunakan pada saat training (*generalization*).

## 2. DEFINISI DAN KONSEP ALGORITMA GENETIK

Individu-individu yang ada dalam populasi seringkali disebut dengan kromosom (string) yang mempunyai panjang yang sama. Setiap kromosom terdiri dari gen-gen yang tersusun secara linier. Setiap gen mempunyai sebuah nilai yang disebut dengan allele dan berada pada posisi tertentu dalam kromosom yang disebut dengan loci/locus. Allele tersebut dapat berupa angka biner (0/1) atau bertipe floating point tergantung dari bentuk representasi genetik yang digunakan.

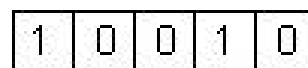


Gambar 1. Bagian-bagian dari Individu pada Algoritma Genetik

### 2.1 Langkah-langkah Persiapan pada Algoritma Genetik

Langkah persiapan yang harus dilaksanakan adalah sebagai berikut :

1. Menentukan bentuk representasi genetik. Representasi gennya bisa berupa:
  - a. Representasi Bit: gen dalam kromosom hanya dapat bernilai 0 atau 1.



Gambar 2. Contoh Individu dengan Representasi Bit

- b. Representasi Floating Point : allele dari setiap gen bertipe floating point.

0.576	0.065	1.000
-------	-------	-------

**Gambar 3. Contoh Individu dengan Representasi Floating Point**

- c. Representasi Integer : allele dari setiap gen bertipe integer.

4	2	5	1	3
---	---	---	---	---

**Gambar 4. Contoh Individu dengan Representasi Integer**

- Menentukan suatu cara untuk menciptakan populasi awal.
- Menentukan fungsi fitness (*Fitness Function*). Fungsi fitness akan menentukan keoptimuman suatu individu dan nilai yang dihasilkan oleh fungsi tersebut disebut dengan nilai fitness.
- Menentukan operasi-operasi genetik yang akan digunakan.
- Menentukan parameter-parameter pengendali jalannya algoritma genetik, yaitu:
  - Ukuran populasi ( $L$ ), yaitu banyaknya individu yang terdapat dalam populasi.
  - Jumlah maksimum generasi ( $Gen$ ), yaitu jumlah maksimum iterasi yang akan dijalankan pada algoritma genetik.
  - Probabilitas crossover ( $P_c$ ), probabilitas mutasi ( $P_m$ ), dan probabilitas reproduksi ( $P_r$ ) yang masing-masing menentukan besarnya kemungkinan individu untuk melakukan operasi crossover, mutasi dan reproduksi.
- Menentukan suatu kriteria untuk menghentikan jalannya algoritma, yaitu:
  - Apabila generasi saat ini telah mencapai jumlah maksimum generasi.
  - Apabila solusi yang paling optimal telah ditemukan.
- Menentukan individu terbaik yang terdapat dalam populasi pada saat kriteria pemberhentian jalannya algoritma terpenuhi.

## 2.2 Operasi-operasi Genetik pada Algoritma Genetik

Sebelum melakukan operasi genetik, terlebih dahulu harus melakukan proses

seleksi untuk menentukan individu yang berhak melakukan operasi genetik.

Adapun metode seleksi yang sering digunakan adalah Roulette Wheel dan membutuhkan langkah-langkah persiapan sebagai berikut :

- Menghitung nilai fitness ( $eval(v_i)$ ) setiap individu  $v_i$  ( $i=1,2,\dots$ , ukuran populasi).
- Menjumlahkan nilai fitness seluruh individu yang terdapat pada populasi. Rumus:

$$F = \sum_{i=1}^{\text{ukuranpopulasi}} eval(v_i) \quad (6)$$

- Menghitung probabilitas terpilihnya setiap individu ( $p_i$ ) dengan cara membagi nilai fitness masing-masing individu dengan total nilai fitness populasi. Rumus:

$$p_i = eval(v_i) / F \quad (7)$$

- Menghitung probabilitas kumulatif untuk setiap individu dalam populasi ( $q_i$ ). Rumus:

$$q_i = \sum_{j=1}^i p_j \quad (8)$$

Setelah itu, maka proses seleksi siap untuk dijalankan dengan cara:

- Membangkitkan bilangan acak  $r$  bertipe real yang terletak dalam range  $[0..1]$ .
- Apabila  $r < q_1$  maka individu yang terpilih adalah individu ke-1 ( $v_1$ ). Jika tidak, maka individu yang terpilih adalah  $v_i$  ( $2 \leq i \leq$  ukuran populasi), dimana:  $q_{i-1} < r \leq q_i$ .

Setelah proses seleksi dilakukan maka operasi genetik dapat dilakukan. Adapun operasi-operasi genetik ini adalah sebagai berikut:

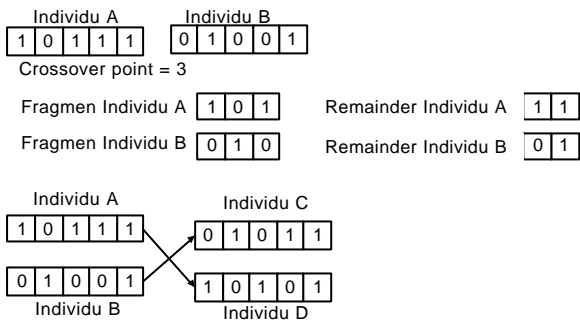
### 1. Operasi Crossover

Penentuan individu yang berhak melakukan operasi crossover tersebut tergantung pada probabilitas crossover ( $P_c$ ) yang telah ditentukan. Langkahnya:

- Membangkitkan bilangan acak  $r$  bertipe real dalam range  $[0..1]$ .
- Jika  $r < P_c$  maka individu tersebut terpilih untuk melakukan crossover.

Metode-metode crossover yang digunakan adalah sebagai berikut:

- One Point Crossover



Gambar 5. One Point Crossover

b. Arithmetic Crossover

$$X' = \lambda X + (1 - \lambda)Y \quad (9)$$

$$Y' = (1 - \lambda)X + \lambda Y \quad (10)$$

dimana: X dan Y merupakan individu yang akan dikawinkan.

X' dan Y' adalah individu baru yang dihasilkan.

$$\lambda \in (0,1).$$

2. Operasi Mutasi

Individu yang berhak melakukan operasi ini tergantung pada probabilitas mutasinya ( $P_m$ ). Langkahnya adalah sebagai berikut:

- a. Membangkitkan bilangan acak  $r$  bertipe real dalam range  $[0..1]$ .
- b. Jika  $r < P_m$  maka individu tersebut terpilih untuk melakukan mutasi.

Metode yang digunakan dalam operasi mutasi adalah:

- Uniform Mutation

Misalkan  $a_i$  dan  $b_i$  adalah batas bawah dan batas atas untuk setiap variabel  $x_i$  (atau  $y_i$ ) maka uniform mutation secara random memilih suatu variabel,  $j$ , dari parent X (atau Y), dan mengubahnya menjadi suatu nomor random tetap  $U(a_i; b_i)$  dengan batasan dari  $a_i$  sampai  $b_i$ :

$$x'_i = \begin{cases} U(a_i, b_i), & \text{jika } i = j \\ x_i, & \text{jika } i \neq j \end{cases} \quad (11)$$

3. Operasi Reproduksi

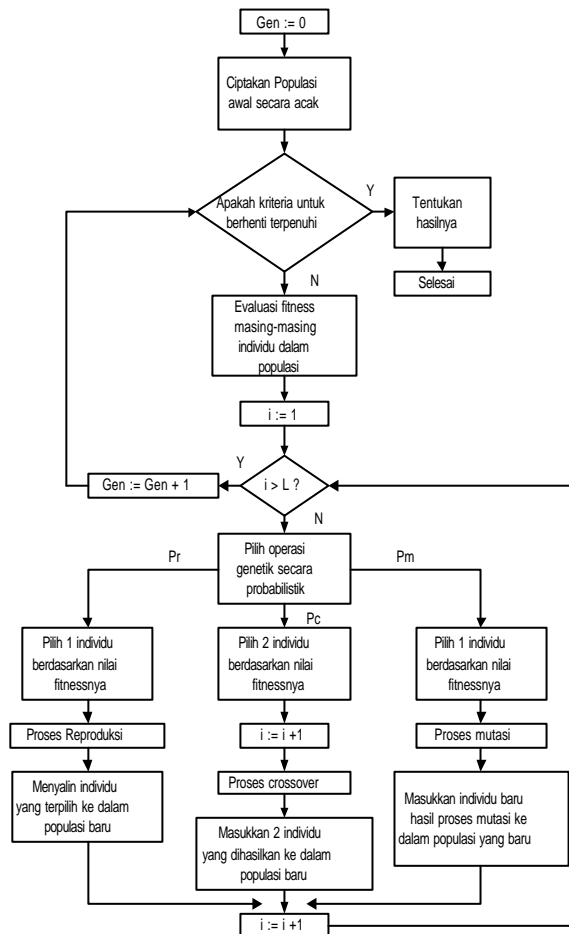
Operasi reproduksi ini pada umumnya akan dilakukan apabila suatu individu yang terpilih tidak berhak melakukan operasi crossover dan mutasi.

2.3 Cara Kerja Algoritma Genetik

Adapun cara kerja algoritma genetik secara umum adalah :

1. Menciptakan sebuah populasi awal secara acak.
2. Mengevaluasi setiap individu yang ada dalam populasi.
3. Menghasilkan populasi yang baru dengan menggunakan operasi genetik.
4. Menentukan hasil akhir pada saat kriteria pemberhentian.

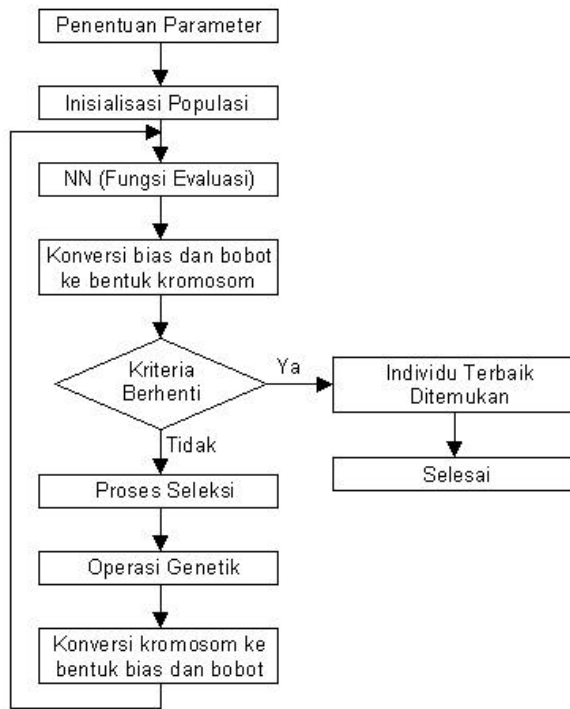
Cara kerja algoritma genetik digambarkan pada gambar 6. Pada gambar tersebut, index  $i$  menyatakan sebuah individu yang ada dalam populasi ( $i = 1,2,3,\dots$ ,ukuran populasi) dan variabel  $L$  menyatakan ukuran populasi. Sedangkan variabel  $GEN$  menyimpan banyaknya generasi yang telah dijalankan



Gambar 6. Flowchart dari Algoritma Genetik

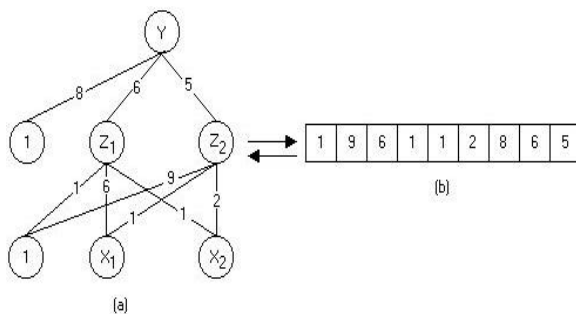
3. NEURAL NETWORK DENGAN MENGGUNAKAN ALGORITMA GENETIK

Diagram flow penerapan algoritma genetik pada neural network dapat dilihat pada gambar 7.



**Gambar 7. Diagram Flow Pelatihan Neural Network Dengan Algoritma Genetik**

Akan tetapi yang menjadi masalah adalah bagaimana melakukan proses konversi bias dan bobot pada neural network ke dalam bentuk individu pada algoritma genetik dan demikian sebaliknya. Solusi yang dicapai adalah menggabungkan bias dan bobot yang ada pada neural network menjadi suatu string yang mewakili individu pada algoritma genetik dan untuk mengkonversikannya kembali ke dalam bentuk bias dan bobot, bisa dilakukan proses pemisahan gen-gen yang ada pada suatu individu pada algoritma genetik menjadi bias dan bobot pada jaringan syaraf buatan. Untuk lebih jelasnya, dapat dilihat pada gambar 8.



**Gambar 8. Konversi Bobot dan Bias menjadi Individu**

(a) Neural Network,  
(b) Individu pada Algoritma Genetik

#### 4. KESIMPULAN

Berdasarkan hasil penelitian, analisa, realisasi dan uji coba perangkat lunak maka dapat ditarik kesimpulan sebagai berikut:

1. Sangat sulit untuk melakukan perbandingan antara metode pelatihan neural network dengan dan tanpa algoritma genetik karena metode-metode tersebut mempunyai parameter yang berbeda.
2. Dari beberapa percobaan yang telah dilakukan, dapat diketahui bahwa algoritma genetik tidak melakukan perbaikan terhadap *neural network*. Memang, kalau *backpropagation* tersebut ditambah dengan iterasi milik algoritma genetik akan menghasilkan *output* yang lebih optimal tetapi membutuhkan waktu yang lebih tinggi. Padahal, seandainya jika waktu tersebut diberikan kepada metode *backpropagation* maka akan dihasilkan suatu *output* yang jauh lebih optimal lagi. Jadi sebenarnya, algoritma genetik tidak melakukan perbaikan sama sekali.
3. Hasil pelatihan *neural network* baik dengan dan tanpa algoritma genetik sangat tergantung pada proses inisialisasi yang diambil dari nilai acak (random) sehingga hasilnya akan berlainan setiap kali melakukan pelatihan.
4. Keakuratan data sangat tergantung pada nilai *error* yang dihasilkan pada saat pelatihan. Semakin kecil nilai *error* yang dihasilkan maka solusi yang dihasilkan akan semakin akurat dan untuk mendapatkan nilai *error* yang semakin kecil diperlukan jumlah iterasi yang banyak pula. Dalam hal ini, jumlah iterasi pada pelatihan neural network yang menggunakan algoritma genetik adalah jumlah generasi.
5. Semakin besar populasi dan semakin banyak jumlah generasi maka waktu yang diperlukan untuk proses pelatihan akan semakin lama. Tetapi, lebih baik menggunakan jumlah ukuran populasi yang lebih sedikit dan dikompensasikan dengan jumlah generasi yang lebih banyak.

---

**DAFTAR PUSTAKA**

1. Beale, R and Jackson, T. *Neural Computing: An Introduction*. London: IOP Publishing, 1992.
2. Yohanes, et al. *Dasar-dasar Jaringan Syaraf Tiruan*. Bandung: Program Pasca Sarjana Sub Program Elektroteknik Bidang Mikroelektronika ITB:1993.
3. Freeman, James.A. *Neural Network Algorithms, Applications, and Programming Techniques*, Addison-Wesley Publishing Company, Inc., 1991.
4. Fausett, Laurene. *Fundamentals Of Neural Network*. Englewood Cliffs, New Jersey: Prentice-Hall.Inc., 1994.
5. Wong, F. *Neural Networks and genetic algorithms for economic forecasting*. Technical report, Institute of Systems Science, National University of Singapore, 1992.
6. *Foundations of Genetic Algorithms*, Rawlins, G. J. E., San Mateo, CA, Morgan Kaufmann, 1991.
7. Cohoon, J. P., Martin, W. N. and Richards, D. S., *A Multi-population Genetic Algorithm for Solving the K-Partition Problem on Hyper-cubes*,