

PERTEMUAN 16

ALGORITMA GENETIKA

A. Tujuan Pembelajaran

Setelah menyelesaikan pada pertemuan ini, mahasiswa mampu menjelaskan Algoritma genetika. Sub materi pada pertemuan ini yaitu:

1. Definisi algoritma genetika
2. Sejarah algoritma genetika
3. Tahapan Proses Algoritma Genetik
4. Struktur umum algoritma genetika
5. Studi Kasus Algoritma Genetik

B. Uraian Materi

1. Definisi Algoritma Genetika

Algoritma Evolusi adalah istilah umum yang dibagi menjadi empat :

- a. *Genetic Algorith*m
- b. *genetic programming*,
- c. *evolution strategies*, dan
- d. *evolutionary programmin*).

Akan Tetapi, *perubahan algoritma* yang terkenal dan sering dijadikan bahan penelitian adalah *genetic algorithm*.

Algoritma genetika (AG) adalah perubahan berkembangnya masa komputerisasi dalam bahasan *artificial intelligence*. Kemunculan algoritma ini gagasan dari teori evolusi Darwin (walaupun nyatanya salah) . seperti evolusi biologi perubahann yang terjadi pada Algoritma genetik konsepnya mirip.

Algoritma genetika adalah teknik searching nilai optimal secara *stochastic* berdasarkan penyeleksian alam. Caranya sangat berbeda dengan algoritma optimasi lainnya, bisa kita lihat dibawah ini .

- a. memanfaatkan hasil pengkodean dari ukuran,bukan ukuran itu sendiri.
- b. Bekerja pada perkumpulan bukan pada sesuatu yang tidak biasa.
- c. Memakai nilai satu-satunya pada fungsi dalam prosesnya. Tidak memakai fungsi luar atau ilmu luar lainnya.

d. memakai fungsi transisi *kemungkinan*.

Algoritma Genetika merupakan algoritma searching menunjukan untuk pencarian solusi dari suatu masalah, bisa menggunakan satu variable bisa juga multivariable. Cara ini meniru kinerja dari genetika alam yaitu untuk menemukan semua susunan gen yang terbaik dalam tubuh makhluk hidup.

Dasar dari Algoritma Genetika itu sendiri adalah teori evolusi Darwin, yang memaparkan prinsip dasar terciptanya banyak macam – macam makhluk hidup yang ada sekarang ini. Makhluk hidup yang bisa beradaptasi lebih baik terhadap lingkungannya akan memiliki kesempatan yang lebih banyak untuk mempertahankan hidup dan berkembangbiak sehingga mempengaruhi jumlah populasi spesies yang bersangkutan.

Dalam perkembangannya, metode Algoritma Genetika banyak digunakan oleh banyak keilmuan. Algoritma ini dipakai untuk menyelesaikan semua kasus yang memiliki solusi lebih dari 2, dimana tidak ada solusi yang terbaik. Sehingga dalam menyelesaikan masalah tersebut akan memerlukan waktu yang tidak sebentar. Setiap solusi Algoritma Genetika diwakili oleh suatu individu atau satu kromosom. Keuntungan Algoritma Genetika adalah sifat cara searching yang lebih optimal, tanpa menggunakan ruang yang besar dalam pencarian, dan lengkap.

2. Sejarah Algoritma Genetik

Asal mula perkembangan dari algoritma genetika (*genetic algorithm*) dimulai tahun 1960. Pada saat itu, I. Rothenberg menulis buku yang mempunyai judul “*Evolution Strategies*” memaparkan mengenai evolusi komputer (*computer evolutionary*) lalu dikembangkan oleh peneliti lain. Algoritma genetika sendiri pertama kali dikembangkan oleh John Holland dan rekannya serta mahasiswanya pada tahun 1970-an di New York Amerika Serikat. terbukti dengan adanya buku yang ditulis oleh Holland yang berjudul “*Adaptation in Natural and Artificial System*” terbit pada tahun 1975. Tujuannya hanya mempelajari fenomena adaptasi yang ada di alam dan di lakukan percobaan secara komputerisasi.

John Koza meneliti suatu program yang berkembang dengan memakai algoritma genetika Tujuh belas tahun setelahnya. Program yang terkenal dengan

metode “*Genetic Programming*” tersebut dibuat menggunakan LISP . Sampai saat ini algoritma genetika ini terus dipakai untuk menyelesaikan permasalahan yang sulit diselesaikan dengan memakai algoritma konvensional.

Algoritma Genetika Holland adalah sebuah cara untuk memindahkan suatu populasi kromosom (terdiri dari bit-bit 1 dan 0) ke populasi baru dengan menggunakan “ seleksi alam “ dan operator genetic seperti :

a. *Crossover*

pertukaran bagian kecil dari dua kromosom

b. *Mutasi*

penggantian secara random nilai gen di beberapa lokasi pada kromosom,

c. *Inversion*

Membalikkan runutan beberapa gen yang berurut dalam kromosom.

Dasar teori inilah yang menjadi dasar kebanyakan program yang menggunakan Algoritma Genetika.

3. Tahapan Proses Algoritma Genetik

Algoritma genetika dapat melalui lima tahap yaitu :

a. Pembentukan awal populasi

Langkah awal membuat sebuah populasi untuk gen- gen yang banyak dan terbaik.

b. Mencari *Fitness Cost*

Individu atau gen yang sudah dibentuk dicari *fitness cost*-nya sebagai nilai komparasi antara individu satu dengan individu yang satunya.

c. Pengurutan (*Sorting*)

Individu yang ada pada populasi di urutkan dari *fitness cost*nya. Untuk mendapatkan solusi terbaik.

d. Proses Regenerasi

Proses ini terdapat dua metode yaitu :

1) *Elitism Method*

Semua individu yang hanya mengalami proses regenerasi

2) *Non Elitms*

Seluruh individu dilibatkan pada metode ini .

e. Tahapan looping

Setelah selesainya proses regenerasi, maka akan dilakukan looping. Looping ini di ulang sampai sejumlah generasi yang ditargetkan.

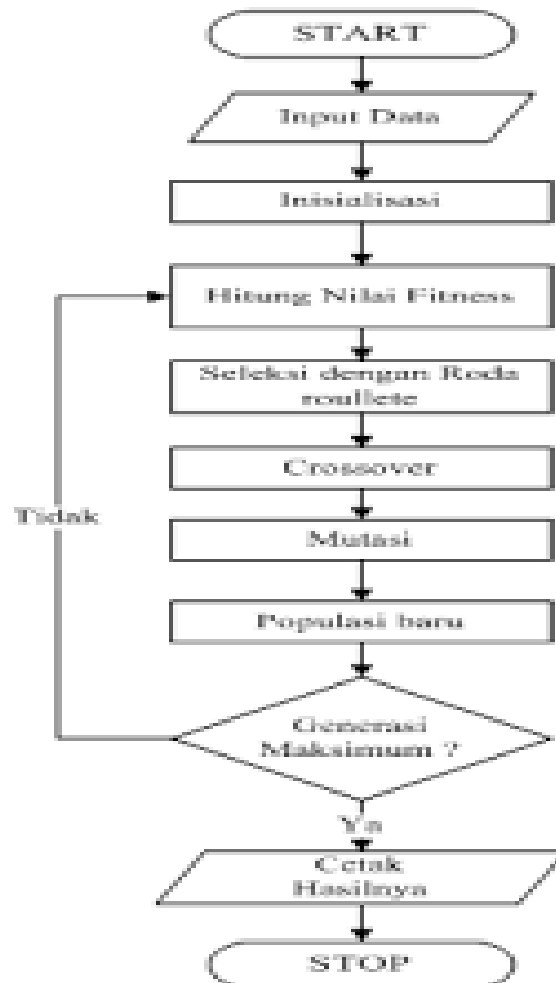
4. Kelebihan Algoritma Genetik

Kelebihan - kelebihan dari Algoritma Genetika adalah :

- a. Variable distrik atau kontinu yang dioptimalkan,
- b. Tidak membutuhkan informasi derivatif,
- c. Berbarengan pencarian dari sebuah sampel yang luas pada permukaan biaya,
- d. Erat kaitannya dengan sekumpulan besar variabel,
- e. Bagus untuk komputer paralel,
- f. Mengoptimalkan permukaan variable dengan biaya yang sangat kompleks (GA bias melompat dari minimum lokal),
- g. Menentukan daftar variabel yang optimal, tidak hanya solusi tunggal,
- h. Variabel dapat disandikan sehingga optimasi bisa dilakukan dengan mengkodekan variabel, dan
- i. Bekerja dengan data numerik yang dihasilkan, data eksperimen, atau analitis fungsi.

5. Struktur Umum Algoritma Genetik

Ilustrasi diagram alir struktur algoritma genetik:

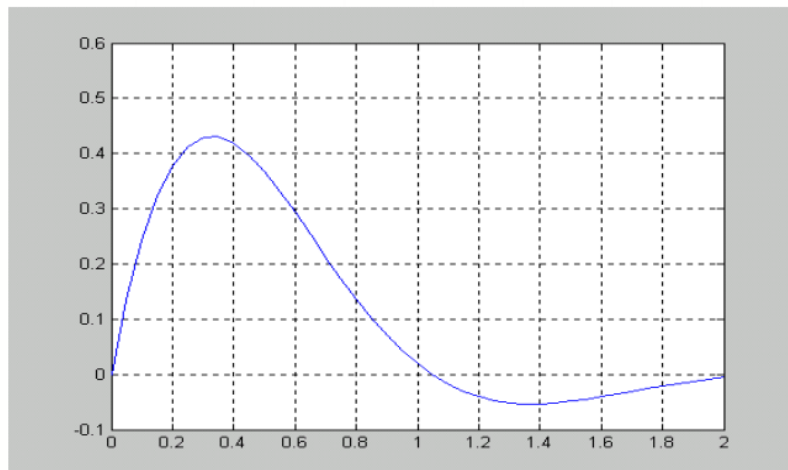


Gambar 16.1 Struktur umum algoritma genetika

Inisialisasi populasi pertama atau bisa disebut penginputan populasi, sebanyak kromosom yang diperlukan akan diinisialisasi secara random. Kemudian dihitung nilai fitness dan dilanjutkan dengan dilakukan seleksi dengan menggunakan metode *rodaroulette*, *tournament* atau ranking. Lalu di kawin silang (*crossover*) dan di lakukan mutasi. Setelah melalui banyak generasi maka algoritma ini akan berhenti sebanyak generasi yang diperlukan.

6. Contoh studi kasus

- a. Fungsi $F(x) = e^{-2x} \cdot \sin(3x)$ menggunakan Algoritma genetik dalam mencari nilai maks. Fungsi di atas bila digambarkan akan menjadi:

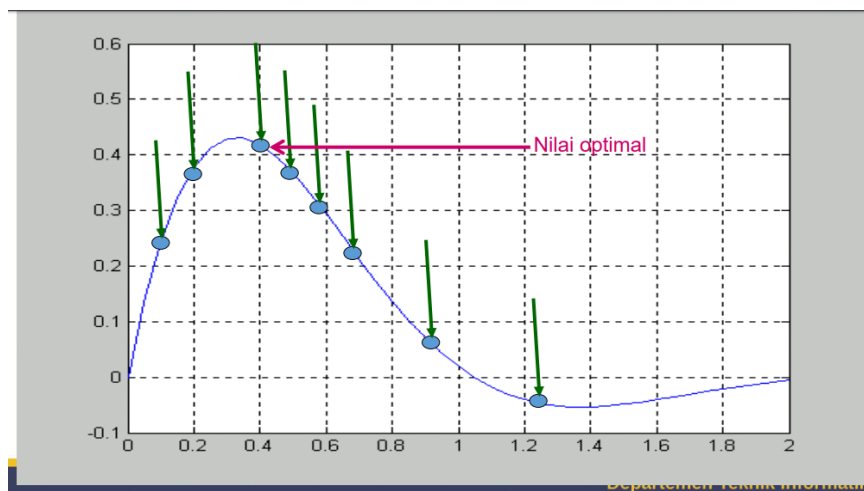


Gambar 16.2 Algoritma genetik dalam mencari nilai maks

Bisa dilihat dari gambar bahwa hasil ada pada nilai $0 < x < 1$ Jadi jika menggunakan 8 bit biner didefinisikan :

00000000 \longrightarrow 0

11111111 \longrightarrow 1



Gambar 16.3 processing Dalam pencarian dan Optimasion

Processing Algoritma Genetika Dalam pencarian dan Optimasion

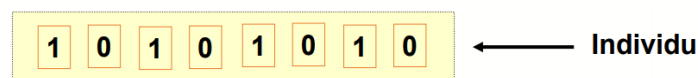
Definisi beberapa individu dinyatakan dalam 8 gen biner, dengan batas 0 sampai dengan 1, berarti 1 bit setara dengan 2^{-8} .

Sebagai contoh:

$$1000\ 1001 = (1\ 2\ 8,0 + 8,0 + 1,0) / 2\ 5\ 6,0 = 0.5\ 35\ 2$$

$$0011\ 0100 = (4,0 + 1\ 6,0 + 32,0) / 2\ 56,0 = 0.2031$$

$$0101\ 0010 = (2,0 + 16,0 + 64,0) / 256,0 = 0.3203$$



Gambar 16.4 individu pada processing Algoritma Genetika

Fungsi Fitness : fungsi $f(x)$, karena nilai maks yang akan di cari.

Membangkitkan Populasi Awal

Individu maksimum

contohnya ada 10 individu dalam 1 populasi, maka 10 individu dibangkitkan dengan 8 gen biner yang akan dibangkitkan secara random.

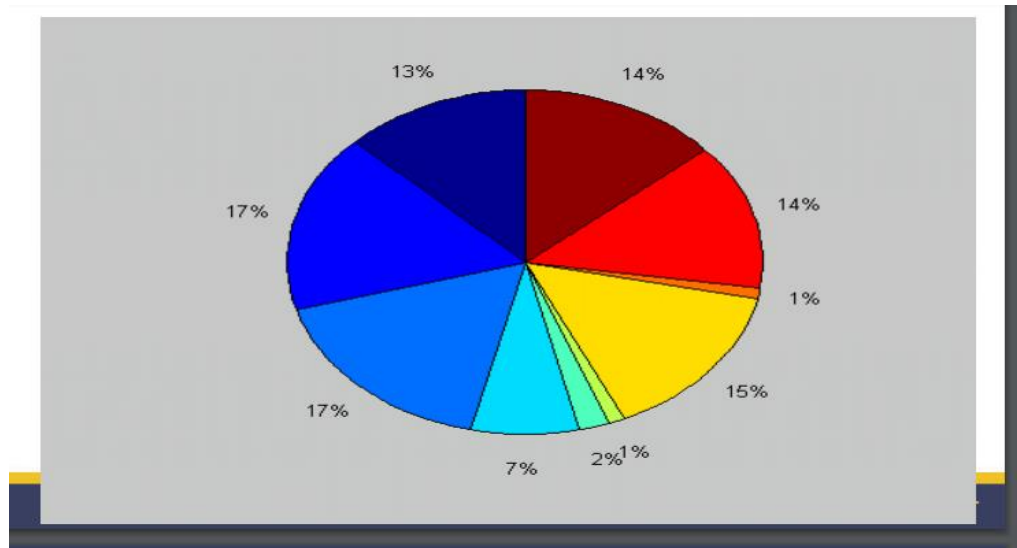
10010000 -- 0.56250 -- 0.32244	
01001110 -- 0.30469 -- 0.43060	→ Individu maksimum
01100110 -- 0.39844 -- 0.41933	
10111101 -- 0.73828 -- 0.18266	
11101000 -- 0.90625 -- 0.06699	
11110010 -- 0.94531 -- 0.04543	
00110011 -- 0.19922 -- 0.37778	
11111100 -- 0.98438 -- 0.02616	
10000111 -- 0.52734 -- 0.34828	
10001011 -- 0.54297 -- 0.33702	

Gambar 16.5 Proses seleksi individu maksimum

Seleksi

Seleksi adalah proses pemilihan calon induk, metode metode yang bisa digunakan dalam proses seleksi ini antara lain:

- Mesin Roulette (Roulette Wheel),
- Competition dan Tournament.



Gambar 16.6 Competition dan Tournament

(Perkawinan Silang) adalah proses kombinasi antara 2 individu untuk memperoleh banyak individu baru dengan nilai fitness lebih baik seperti yang diharapkan. Pasangan yang mengikuti metode perkawinan silang ditentukan dengan nilai kemungkinan, jadi tidak semua pasangan yang menggunakan metode ini.

		Fitness
0 0 1 1 1 0 0 1	-- 0.22266 ← induk 1	0.3968
1 0 0 1 1 0 1 0	-- 0.60156 ← induk 2	0.2921
0 0 1 1 1 0 1 1	-- 0.23050 ← anak 1	0.4022
1 0 0 1 1 0 0 0	-- 0.59382 ← anak 2	0.2982

Gambar 16.7 Perkawinan silang 2 individu

Pemprosesan perubahan gen dengan nilai inversi diri sendiri disebut dengan mutasi gen, gen 0 menjadi 1 dan gen 1 jadi 0.

Contoh Hasil Algoritma Genetika

Generasi ke 1 :

```
10100111 -- 0.65234 -- 0.25127
01000110 -- 0.27344 -- 0.42328
01001110 -- 0.04297 -- 0.43060
01110110 -- 0.46094 -- 0.39076
10111001 -- 0.72266 -- 0.19488
10001111 -- 0.55859 -- 0.32540
10001000 -- 0.53125 -- 0.34550
10010011 -- 0.57422 -- 0.31348
00111011 -- 0.23047 -- 0.40214
10000011 -- 0.51172 -- 0.35913
```

Generasi ke 2 :

```
10000000 -- 0.50000 -- 0.36696
10001010 -- 0.53906 -- 0.33987
01001110 -- 0.04297 -- 0.43060
10010111 -- 0.58984 -- 0.30132
10000011 -- 0.51172 -- 0.35913
10000011 -- 0.51172 -- 0.35913
01001110 -- 0.30469 -- 0.43060
10001000 -- 0.53125 -- 0.34550
10111101 -- 0.73828 -- 0.18266
01000010 -- 0.25781 -- 0.41715
```

Gambar 16.8 contoh Hasil Algoritma Genetika 1

Contoh Hasil Algoritma Genetika

Generasi ke 3 :

```
01001110 -- 0.30469 -- 0.43060
01001110 -- 0.30469 -- 0.43060
10000011 -- 0.51172 -- 0.35913
10001000 -- 0.53125 -- 0.34550
11001110 -- 0.80469 -- 0.13301
01001110 -- 0.04297 -- 0.43060
10001010 -- 0.53906 -- 0.33987
10000011 -- 0.51172 -- 0.35913
01001110 -- 0.30469 -- 0.43060
10000011 -- 0.51172 -- 0.35913
```

Generasi ke 4 :

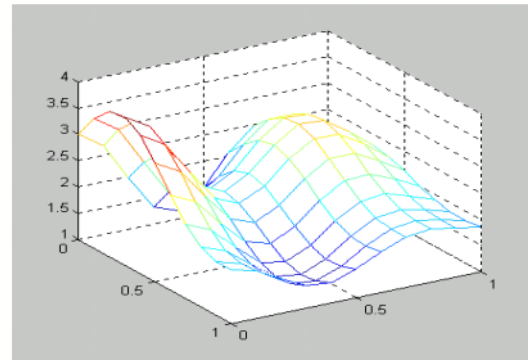
```
00001110 -- 0.05469 -- 0.14641
11001000 -- 0.78125 -- 0.15005
01001110 -- 0.30469 -- 0.43060
10000011 -- 0.51172 -- 0.35913
01001110 -- 0.30469 -- 0.43060
01001110 -- 0.30469 -- 0.43060
01000011 -- 0.26172 -- 0.41885
10001110 -- 0.55469 -- 0.32833
00001110 -- 0.05469 -- 0.14641
01001110 -- 0.78906 -- 0.43060
```

Gambar 16.9 contoh Hasil Algoritma Genetika 2

Algoritma Genetika Untuk Menentukan Nilai Maksimal Fungsi 2 variabel bebas

$$f(x, y) = 2 + e^{-(x^2 + y^2)} \{ \sin(4x) + \cos(8y) \}$$

- Penyelesaian berupa pasangan nilai (x,y), sehingga individu didefinisikan sebagai pasangan (x,y).
- Dalam hal ini digunakan gen float untuk penyederhanaan sistem, karena gen biner akan menyebabkan besarnya ukuran kromosom.
- Fungsi fitness adalah fungsi f(x,y).



Gambar 16.10 Menentukan nilai maksimal fungsi 2 variabel

Populasi awal dapat dibangkitkan dengan membangkitkan sejumlah pasangan (x,y) secara acak.

x	y	fitness
0.66682	0.98088	2.02703
0.68314	0.29875	1.86538
0.87539	0.15460	2.00401
0.50769	0.93574	2.15746
0.46789	0.67386	1.45168
0.24484	0.42791	2.21924
0.56603	0.83281	1.90247
0.76072	0.17132	1.99177
0.34517	0.44064	1.86459
0.44755	0.75244	1.77778
0.51579	0.61550	1.23893
0.15734	0.25417	3.55771

Departemen Teknik Informatika & Komputer

16.11 Populasi awal pasangan secara acak

Hasil Algoritma Genetika

Generasi 1:

x	y	fitness
0.49263	0.67386	1.41828
0.73599	0.17132	1.99971
0.46789	0.21433	2.53206
0.26464	0.17132	3.33107
0.56603	0.83281	1.90247
0.56603	0.83281	1.90247
0.70012	0.17132	2.02239
0.21795	0.25417	3.37522
0.15734	0.25417	3.55771
0.42702	0.25417	2.59217
0.17084	0.25417	3.52082
0.74723	0.17132	1.99537



Generasi 2:

x	y	fitness
0.32399	0.21433	3.08441
0.40854	0.17132	2.75335
0.21732	0.25417	3.37731
0.15797	0.25417	3.55604
0.56603	0.30491	2.00455
0.17084	0.78207	2.39452
0.17084	0.19691	3.65879
0.73599	0.22859	1.99249
0.35642	0.25417	2.85825
0.91416	0.83281	1.89165
0.66069	0.17132	2.06394
0.15734	0.25417	3.55771

Gambar 16.12 Hasil algoritma genetika generasi 1 dan 2

Hasil Algoritma Genetika

Generasi 3:

x	y	fitness
0.40784	0.22363	2.73759
0.17084	0.72976	2.19665
0.85135	0.29176	1.89196
0.56603	0.26732	2.13788
0.40854	0.17132	2.75335
0.17084	0.78207	2.39452
0.15734	0.25417	3.55771
0.15797	0.25417	3.55604
0.17084	0.19691	3.65879
0.76096	0.83281	1.82495
0.40854	0.17132	2.75335
0.40854	0.17132	2.75335



Generasi 4:

x	y	fitness
0.27182	0.14624	3.25968
0.17084	0.19691	3.65879
0.15765	0.32497	3.16152
0.16415	0.29516	3.33433
0.37024	0.25417	2.80481
0.27987	0.17132	3.27160
0.26467	0.17335	3.33281
0.17084	0.19488	3.66005
0.17415	0.17132	3.64583
0.17714	0.36596	2.82353
0.17182	0.36596	2.83666
0.17947	0.17132	3.62975

Gambar 16.13 Hasil algoritma genetika generasi 3 dan 4

Hasil Algoritma Genetika

Generasi 5:

x	y	fitness
0.17164	0.19047	3.65955
0.17084	0.17584	3.65923
0.16415	0.21711	3.65133
0.64027	0.24422	2.05715
0.96166	0.19499	2.09101
0.17155	0.42238	2.43739
0.17084	0.19488	3.66005
0.17415	0.49995	2.00790
0.17084	0.19488	3.66005
0.37024	0.25933	2.78706
0.16415	0.17137	3.67512
0.17164	0.22974	3.60074



Generasi 6:

x	y	fitness
0.17129	0.69742	2.07593
0.17164	0.45035	2.26035
0.21598	0.22974	3.46123
0.16415	0.17137	3.67512
0.30561	0.21997	3.14860
0.28211	0.20465	3.26279
0.28211	0.22974	3.21996
0.28211	0.22974	3.21996
0.16415	0.19227	3.68061
0.17129	0.17188	3.65484
0.32762	0.22974	3.04113
0.28211	0.22974	3.21996

Gambar 16.14 Hasil algoritma genetika generasi 5 dan 6

C. Soal Latihan/Tugas

1. Apa yang dimaksud algoritma genetika jelaskan disertai sejarahnya?
2. Sebutkan 5 contoh pengaplikasian algoritma genetika?
3. Carilah permasalahan lain yang dapat diselesaikan dengan Algoritma Genetika!
4. Implementasikan program untuk salah satu permasalahan di atas!

D. Referensi

- Kusumadewi, Sri. 2003. Artificial Intelligence – Teknik dan Aplikasinya. Yogyakarta : Graha Ilmu.
- Basuki, Achmad. 2003. Algoritma Genetika : Suatu Alternatif Penyelesaian Permasalahan Searching, Optimasi dan Machine Learning. Surabaya : Politeknik Elektronika Negeri Surabaya PENS – ITS.
- Martiana, Barakbah, Yuliana Setiowati. 2014. Modul Ajar Kecerdasan Buatan. Surabaya : Politeknik Elektronika Negeri Surabaya PENS – ITS.