

# Pengantar Softcomputing

**Anto Satriyo Nugroho**

asnugroho@ieee.org

http://asnugroho.net

## **Lisensi Dokumen:**

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di **IlmuKomputer.Com** dapat digunakan, dimodifikasi dan disebarluaskan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari **IlmuKomputer.Com**.

## **Abstrak:**

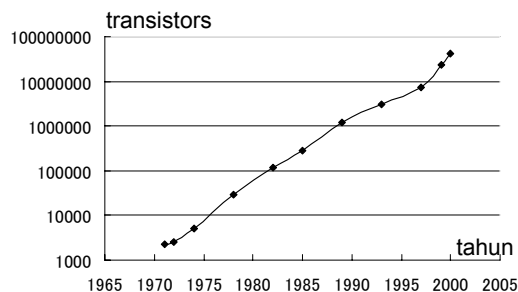
Softcomputing merupakan metode yang dapat mengolah data-data yang bersifat tidak pasti, impresisi dan dapat diimplementasikan dengan biaya yang murah (low-cost solution). Beberapa metode yang termasuk dalam kategori softcomputing misalnya fuzzy logic, artificial neural network, probabilistic reasoning. Softcomputing bukanlah suatu metode yang berjalan sendiri dalam menyelesaikan masalah, melainkan lebih pada kerjasama serasi antara metode-metode di atas, sehingga segi positif tiap metode dapat berkontribusi secara aktif. Sebagian dari materi pada makalah ini disampaikan sebagai pengantar kuliah musim semi 2003: "Special Lecture on Media Science V", pada School of Computer & Cognitive Sciences, Chukyo University, Japan.

**Keywords:** softcomputing, ketidakpastian, low-cost solution, fuzzy, artificial neural network, probabilistic reasoning

## **1. PENDAHULUAN**

Komputer merupakan alat yang mungkin tidak dapat lagi kita lepaskan dari kehidupan sehari-hari. Di masa lampau, komputer selalu diasosiasikan dengan barang yang mahal, super canggih dan dapat menghitung lebih cepat daripada kemampuan manusia. Tapi saat ini ungkapan tersebut sudah banyak mengalami perubahan. Bila disebutkan "komputer", maka yang terbayang saat ini adalah sebuah piranti canggih yang bila dikoneksikan dengan internet, akan membuat sang pengguna dapat berjalan-jalan di dunia virtual yang kaya dengan informasi audio maupun visual. Fenomena ini menunjukkan perubahan fungsi computer dari sekedar "alat hitung" menjadi sebuah piranti yang lebih "manusiawi". Hal ini merupakan buah yang

dipetik dari pesatnya perkembangan teknologi hardware maupun software computer.



Gambar 1. Perkembangan jumlah transistor pada IC dari tahun ke tahun yang menunjukkan kesesuaian dengan Moore's law.

Pada tahun 1965, Gordon Moore telah memberikan prediksi bahwa jumlah transistor pada IC akan selalu berlipat dua setiap 18 bulan [1]. Dengan kata lain, tiap 18 bulan kemampuan komputer akan menjadi duakali lebih cepat. Korelasi ini ditunjukkan pada gambar 1. Ramalan ini secara ajaib masih berlaku hingga saat ini, setidaknya dalam dua dekade terakhir.

Melihat perkembangan pesat dari komputer ini, seringkali timbul fikiran bahwa pada suatu masa, komputer dapat mengatasi berbagai permasalahan. Semua masalah dalam hidup dapat dirumuskan dan dikalkulasikan. Hal ini tergambar juga pada beberapa film non-fiksi. Benarkah demikian ?

Kalau kita cermati, tidak semua masalah yang kita hadapi dapat dibuat rumusan yang pasti dan eksak. Memang benar, bahwa komputer dapat menghitung secara akurat persamaan differensial, perkalian matriks, perhitungan eigen value secara cepat dan akurat. Akan tetapi terdapat juga masalah dalam kehidupan sehari-hari yang tidak dapat dibuat rumus matematika yang jelas untuk menghitung output yang diinginkan. Misalnya:

- bagaimanakah trend dari bursa saham pada tahun depan ?
- bagaimanakah cuaca esok hari, hujan atukah mendung, atukah terang benderang ?
- Kapanakah terjadinya gempa bumi di wilayah Tokai (Jepang Tengah) ?
- Bagaimana mengenali seseorang lewat suaranya ?

Solusi untuk permasalahan jenis kedua ini tidak dapat dirumuskan dengan mudah. Banyak sekali faktor-faktor terlibat, yang mengandung kerancuan, ketidakpastian, kebenaran parsial, dsb. Dalam hal ini, solusi yang diharapkan lebih menitikberatkan pada hal-hal sbb.

- memiliki kemampuan untuk mempelajari trend yang telah ada, dan memprediksi keadaan di masa yad. (*learning ability*)
- kemampuan menganalisa informasi yang disertai oleh *noise*.

- *robust*, sangat unggul
- *low cost solution*
- praktis dan mudah direalisasikan

Salah satu alternatif solusi yang memenuhi kriteria ini adalah *softcomputing*. Dalam bab-bab berikutnya akan dijelaskan lebih lanjut mengenai definisi, metode-metode dalam *softcomputing*, dan contoh aplikasinya.

## 2. DEFINISI SOFTCOMPUTING

Berbagai macam definisi *softcomputing* diberikan oleh para ahli. Salah satu definisinya adalah sebagaimana disampaikan oleh pencetus *softcomputing*, yaitu Prof. Lotfi A. Zadeh, di homepage BISC [2], sbb.

*“Berbeda dengan pendekatan konvensional hardcomputing, softcomputing dapat bekerja dengan baik walaupun terdapat ketidakpastian, ketidakakuratan maupun kebenaran parsial pada data yang diolah. Hal inilah yang melatarbelakangi fenomena dimana kebanyakan metode softcomputing mengambil human-mind sebagai model.”*

Mengapakah *human-mind* merupakan model yang menarik bagi pengembangan *softcomputing* ? Kunci dari pertanyaan ini sebenarnya terletak pada struktur dan fungsi dari otak manusia. Otak manusia merupakan mesin molekuler, yang terdiri dari dua jenis sel: neuron dan glia. Dalam otak kita terdapat sekitar  $10^{11}$  sel neuron, sedangkan sel glia sekitar 3 sampai 4 kali lipatnya. Sel neuron berfungsi sebagai pemroses informasi yang diterima oleh otak. Sel neuron terhubung antara satu dengan yang lain dengan benang-benang panjang. Berat otak manusia saat lahir sekitar 400 gram, sedangkan saat dewasa sekitar 1500 gram. Pertambahan berat ini disebabkan oleh bertambahpanjangnya benang-benang tersebut, disamping pertambahan sel glia. Pertambahan panjang ini berkaitan erat dengan proses pembelajaran yang dialami oleh manusia. Hal ini merupakan ide awal bagi pengembangan metode *softcomputing*: artificial neural network, yang memiliki

kemampuan pembelajaran terhadap informasi yang telah diterima.

Selain kemampuan pembelajaran, otak manusia juga memiliki kemampuan untuk mengambil keputusan walaupun informasi mengandung unsur ketakpastian dan kekurangtegasan, seperti “manis”, “pahit”, “tinggi”, “rendah”, dsb. Hal ini merupakan konsep yang mendasari pengembangan metode fuzzy, yang mencerminkan cara berfikir manusia. Selain neural network dan fuzzy, masih banyak lagi jenis-jenis metode softcomputing, yang ide awalnya bersumber dari otak manusia maupun mekanisme biologi yang terdapat di alam semesta.

### 3. METODE-METODE SOFTCOMPUTING

Mengacu pada definisi yang diberikan oleh Zadeh, metode-metode dalam softcomputing dapat dikategorikan ke dalam tiga kategori besar:

- Fuzzy Logic (FL)
- Neural Network Theory (NN)
- Probabilistic Reasoning (PR)

Metode-metode ini sebenarnya bukanlah sesuatu yang baru diadakan setelah konsep softcomputing dirumuskan. Yang terjadi justru sebaliknya. Metode-metode Fuzzy Logic, Neural Network maupun Probabilistic Reasoning telah ada lebih dahulu. Fuzzy Logic telah berkembang sejak tahun 1965. Konsep-konsep dasar neural network telah digali sejak tahun 1940-an. Probabilistic Reasoning juga bukanlah hal yang baru sama sekali. Karena itu, Zadeh menyebut softcomputing sebagai reinkarnasi dari metode-metode di atas.

Lebih lanjut lagi, dalam konsep softcomputing, ketiga jenis metode ini ibarat pilar, saling mendukung dan bekerjasama dalam memecahkan suatu permasalahan. Keunggulan yang diperoleh dari kerjasama metode-metode itu lebih ditekankan daripada keunggulan individual salah satu daripadanya. Kekurangan satu metode akan ditutup dengan kelebihan metode yang lain. Keunggulan satu metode disumbangkan, sehingga segi-segi

positif dari metode yang ada tersebut dapat dimanfaatkan secara optimal.

Berikut diuraikan konsep dan gambaran mengenai masing-masing pilar dalam softcomputing.

#### a. Fuzzy Logic (FL)

Fuzzy merupakan representasi suatu pengetahuan yang dikonstruksikan dengan *if-then rules*. Karakteristik dari metode ini adalah [3]

- pemecahan masalah dilakukan dengan menjelaskan sistem bukan lewat angka-angka, melainkan secara linguistik, atau variable-variable yang mengandung ketakpastian/ketidaktegasan.
- Pemakaian *if-then rules* untuk menjelaskan kaitan antara satu variable dengan yang lain.
- Menjelaskan sistem memakai algoritma fuzzy

Berawal dari paper-paper Zadeh di tahun 1965 mengenai fuzzy-sets, ilmu ini berkembang pesat, dan mulai menemukan aplikasinya di bidang control pada tahun 1974. Pada saat itu, Mamdani memperkenalkan aplikasi fuzzy sebagai alat kontrol *steam-engine*. Hal ini merupakan momentum penting, sebagai awal bagi teknologi fuzzy untuk menemukan ladang aplikasi di dunia industri. Fuzzy memiliki kelebihan-kelebihan, diantaranya [3]

1. Dapat mengekspresikan konsep yang sulit untuk dirumuskan, seperti misalnya “suhu ruangan yang nyaman”
2. Pemakaian *membership-function* memungkinkan fuzzy untuk melakukan observasi obyektif terhadap nilai-nilai yang subyektif. Selanjutnya *membership-function* ini dapat dikombinasikan untuk membuat pengungkapan konsep yang lebih jelas.
3. Penerapan logika dalam pengambilan keputusan

Dewasa ini, fuzzy merupakan salah satu metode memiliki aplikasi luas di bidang kontrol. Hal ini disebabkan a.l. [3]

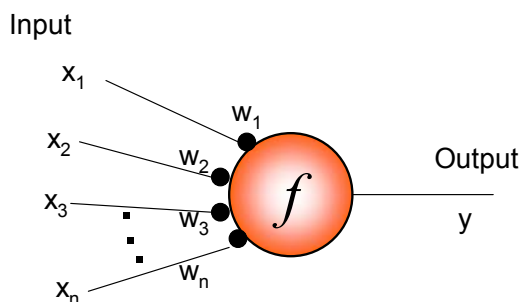
1. kontrol memiliki potensi aplikasi yang sangat luas dan dibutuhkan di berbagai bidang
2. kuantitas suatu materi dalam sistem kontrol sangat jelas, dan dapat diekspresikan dengan istilah-istilah yang *fuzzy* seperti “besar”, “banyak”
3. aturan dalam kontrol mudah untuk didefinisikan memakai kata-kata. Misalnya “jika suhu dalam ruangan terlalu dingin, naikan suhu penghangat”
4. perkembangan teori fuzzy sangat pesat, sehingga batas-batasnya dapat dirumuskan dengan jelas.

### b. Neural Networks (NN)

Neural Networks (Jaringan Saraf Tiruan) menurut Haykin [4] didefinisikan sebagai berikut :

*“Sebuah neural network (JST: Jaringan Saraf Tiruan) adalah prosesor yang terdistribusi paralel, terbuat dari unit-unit yang sederhana, dan memiliki kemampuan untuk menyimpan pengetahuan yang diperoleh secara eksperimental dan siap pakai untuk berbagai tujuan. Neural network ini meniru otak manusia dari sudut : 1) Pengetahuan diperoleh oleh network dari lingkungan, melalui suatu proses pembelajaran. 2) Kekuatan koneksi antar unit yang disebut synaptic weights, berfungsi untuk menyimpan pengetahuan yang telah diperoleh oleh jaringan tersebut.”*

Pada tahun 1943, Mc.Culloch dan Pitts memperkenalkan model matematika yang merupakan penyederhanaan dari struktur sel saraf yang sebenarnya (lihat gambar 1).



Gambar 2 McCulloch & Pitts neuron model

Gambar 2 memperlihatkan bahwa sebuah neuron memiliki tiga komponen:

- synapse  $(w_1, w_2, \dots, w_n)^T$
- alat penambah (adder)
- fungsi aktivasi  $f$

Korelasi antara ketiga komponen ini dirumuskan pada persamaan (1).

$$y = f\left(\sum_{i=1}^n x_i \times w_i\right) \quad (1)$$

Signal  $\mathbf{x}$  berupa vektor berdimensi  $n$   $(x_1, x_2, \dots, x_n)^T$  akan mengalami penguatan oleh synapse  $\mathbf{w}$   $(w_1, w_2, \dots, w_n)^T$ . Selanjutnya akumulasi dari penguatan tersebut akan mengalami transformasi oleh fungsi aktivasi  $f$ . Fungsi  $f$  ini akan memonitor, bila akumulasi penguatan signal itu telah melebihi batas tertentu, maka sel neuron yang semula berada dalam kondisi “0”, akan mengeluarkan signal “1”. Berdasarkan nilai output tersebut ( $\Rightarrow y$ ), sebuah neuron dapat berada dalam dua status: “0” atau “1”. Neuron disebut dalam kondisi *firing* bila menghasilkan output bernilai “1”.

Sebuah neural network dapat dianalisa dari dua sisi:

- bagaimana neuron-neuron tersebut dirangkaikan dalam suatu jaringan (arsitektur)
- bagaimana jaringan tersebut dilatih agar memberikan output sesuai dengan yang dikehendaki (algoritma pembelajaran). Algoritma pembelajaran ini menentukan cara bagaimana nilai penguatan yang optimal diperoleh secara otomatis.

Berdasarkan arsitekturnya, neural network dapat dikategorikan, antara lain, *single-layer neural network, multilayer neural network, recurrent neural network* dsb. Berbagai algoritma pembelajaran antara lain Hebb’s law, Delta rule, Backpropagation algorithm, Self Organizing Feature Map, dsb.

Berawal dari diperkenalkannya model matematika neuron oleh McCulloch & Pitts, penelitian di bidang neural network berkembang cukup pesat, dan mencapai

puncak keemasan pertama pada era tahun 60, dan puncak kedua pada pertengahan tahun 80-an. Penelitian dalam bidang ini, dapat dibagi dalam tiga kategori:

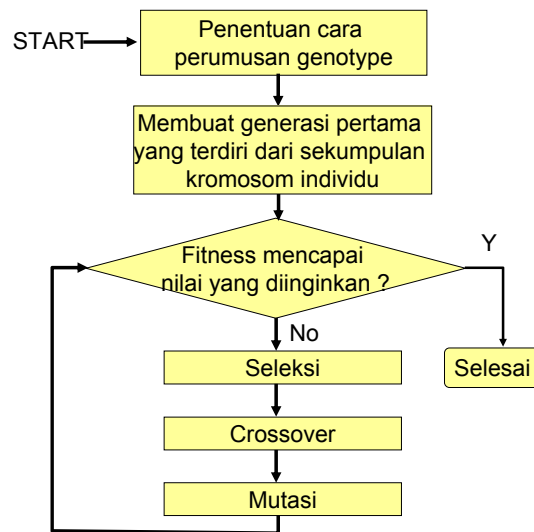
1. Riset untuk meneliti proses informasi yang terjadi pada otak dan jaringan saraf. Tema ini merupakan porsi penelitian para ahli medis dan neuroscientist.
2. Penelitian teoritis untuk mendalami konsep dasar proses informasi pada otak. Kategori ini memerlukan ketajaman analisa matematika untuk menggali dasar-dasar teori dari proses tersebut.
3. Penelitian yang bertujuan memanfaatkan teori-teori yang telah ada untuk aplikasi. Dalam hal ini, perlu sekali memperhatikan tingkat akurasi sistem, dan menekan biaya serendah mungkin (*low cost solution*).

Dewasa ini, neural network telah diaplikasikan di berbagai bidang. Hal ini dikarenakan neural network memiliki kelebihan-kelebihan sbb.

1. Dapat memecahkan problema non-linear yang umum dijumpai di aplikasi
2. Kemampuan memberikan jawaban terhadap pattern yang belum pernah dipelajari (*generalization*)
3. Dapat secara otomatis mempelajari data numerik yang diajarkan pada jaringan tersebut

### c. Probabilistic Reasoning (PR) dan Genetic Algorithm (GA)

Reasoning berarti mengambil suatu keputusan atas suatu alasan atau sebab tertentu. Dua jenis reasoning adalah logical reasoning dan probabilistic reasoning. Salah satu kelebihan probabilistic reasoning dibandingkan logical reasoning terletak pada kemampuan untuk mengambil keputusan yang rasional, walaupun informasi yang diolah kurang lengkap atau mengandung unsur ketidakpastian. Termasuk dalam kategori PR antara lain teori Chaos, Belief Networks, Genetic Algorithm. Diskusi dalam makalah ini difokuskan pada salah satu metode dalam PR, yaitu Genetic Algorithm (GA).



Gambar 3 Urutan proses pada GA

Dasar-dasar GA digali oleh John Holland pada pertengahan tahun 70-an. GA adalah metode komputasi yang meniru proses evolusi dan seleksi alam. Metode ini sering dimanfaatkan untuk mencari nilai optimal suatu fungsi/permasalahan.

Gambar 3 menunjukkan urutan tahapan dalam GA. Untuk mencari nilai optimal tersebut, pertama-tama parameter-parameter permasalahan ditransfer kedalam bentuk genetik sebuah kromosom individu yang disebut *genotype*. Kromosom ini terdiri dari sederetan *string* (misalnya angka "0" dan "1") yang merupakan analogi dari rantai DNA: A, T, G dan C yang sebenarnya, pada tubuh makhluk hidup. Selanjutnya suatu populasi yang terdiri dari ribuan kromosom individu ini mengalami proses seleksi, *crossover* (persilangan) dan mutasi yang meniru proses biologi yang terjadi di alam. Operasi ini diulang-ulang, dari satu generasi ke generasi berikutnya. Kualitas suatu individu ditunjukkan oleh nilai fitness, yang diukur dengan suatu kriteria yang mencerminkan sejauh mana kromosom individu tersebut mendekati nilai optimal yang diinginkan. Kriteria ini menjadi alat kontrol bagi proses evolusi, agar kondisi fitness generasi yang mendatang lebih baik daripada generasi-generasi sebelumnya. Setelah melewati ratusan atau mungkin ribuan generasi, proses evolusi ini akan menghasilkan individu-individu dengan nilai

fitness yang tinggi. Hal ini mencerminkan diperolehnya jawaban yang merupakan pendekatan terhadap nilai optimal yang diinginkan.

Beberapa keunggulan yang dimiliki oleh GA adalah sbb. [3]

1. GA memiliki kemampuan untuk mencari nilai optimal secara paralel, melalui proses kerjasama antara berbagai unit yang disebut kromosom individu.
2. GA tidak memerlukan perhitungan matematika yang rumit seperti differensial yang diperlukan oleh algoritma optimisasi yang lain.

Namun demikian GA memiliki juga kelemahan dan keterbatasan.

1. Tidak memiliki rumusan yang pasti, bagaimana mentransfer parameter permasalahan ke dalam kode genetik. Dengan kata lain, hal ini memerlukan pengalaman dan wawasan dari desainer.
2. Banyak parameter yang perlu diset secara baik agar proses evolusi dalam GA berjalan sesuai dengan yang diharapkan.
3. Penentuan rumus menghitung fitness merupakan hal yang sangat penting dan mempengaruhi proses evolusi pada GA. Sayangnya tidak ada prosedur yang baku bagaimana menentukan rumus tsb. Dalam hal ini pengalaman dari desainer memegang peranan penting.

Terlepas dari kendala yang ada, GA merupakan alternatif solusi yang dikenal cukup handal dalam berbagai masalah optimisasi.

#### **4. RISET DAN APLIKASI SOFTCOMPUTING**

Dewasa ini penelitian di bidang softcomputing berkembang dengan pesat dan aplikasinya dapat ditemukan di berbagai bidang. Hal ini disebabkan softcomputing menawarkan solusi yang sangat sesuai dengan karakteristik informasi pada *real-life domain* yang senantiasa diikuti dengan faktor

impresisi, ketidakpastian, dan memerlukan kemampuan pembelajaran.

Berawal dari pemakaiannya untuk alat kontrol mesin uap, dewasa ini fuzzy memiliki wilayah aplikasi yang luas terutama dalam bidang kontrol, robotika, pattern recognition, sistem cerdas, dll. Selain paper dan artikel yang dipublikasikan oleh berbagai journal ilmiah, diskusi mengenai fuzzy ini dapat diikuti di berbagai milis, maupun newsgroup seperti *comp.ai.fuzzy*. Arsip posting newsgroup ini dapat dibaca lewat <http://groups.google.com/> dengan mengetikkan "comp.ai.fuzzy" pada kotak pencarian.

Neural Network memiliki aplikasi yang sangat luas di bidang pattern recognition, seperti voice recognition, character recognition maupun aplikasi-aplikasi pada bidang ekonomi, bisnis dan bioteknologi. Salah satu forum diskusi virtual (newsgroup) yang terkenal di bidang ini adalah *comp.ai.neural-nets* yang arsipnya dapat dilihat di google, dengan cara sama sebagaimana uraian sebelumnya. Kumpulan dari materi diskusi yang berlangsung sejak 1994 di forum ini, kemudian dirangkumkan oleh Warren Sale sebagai FAQ (Frequently Asked Questions) yang dapat dilihat arsipnya di

<ftp://ftp.sas.com/pub/neural/FAQ.html>

Aplikasi Genetic Algorithm dapat ditemukan di problem optimisasi seperti *job-scheduling*, optimisasi rute mobil, penentuan kandidat gen yang memiliki potensi kedokteran dan farmasi pada bidang bioinformatika. Salah satu forum diskusi yang cukup hangat mengenai riset di bidang GA adalah *comp.ai.genetic* (newsgroup). Arsip diskusi pada forum ini dapat dicari dengan cara sebagaimana diuraikan di atas. Selain itu, informasi mengenai Genetic Algorithm ini dapat juga dilihat pada situs

<http://www.aic.nrl.navy.mil/galist/>

Salah satu journal yang didedikasikan untuk mempublikasikan penelitian di bidang softcomputing di antaranya adalah "Soft

Computing - A Fusion of Foundations, Methodologies and Applications” terbitan Springer-Verlag Heidelberg. Journal lain adalah Applied Soft Computing yang diterbitkan oleh Elsevier. Disamping itu, paper mengenai teori dan aplikasi softcomputing dapat dibaca pada journal-journal IEEE, IEICE, maupun proseding seminar-seminar internasional dalam bidang tsb.

Ciri khas dari softcomputing adalah penekanan pada partnership atau kerjasama yang saling menguntungkan dari berbagai metode yang ada. Tiap metode memiliki segi positif yang dapat disumbangkan secara komplementer, menutupi kekurangan dari metode yang lain. Contoh populer dari kerjasama komplementer ini adalah sistem *neurofuzzy*. Aplikasi dari kombinasi kedua metode ini dapat dimanfaatkan untuk sistem control pada AC, mesin cuci dan berbagai macam alat kebutuhan sehari-hari konsumen. Kombinasi lain dapat ditemukan pada sistem *neuroGA*, dimana GA dimanfaatkan untuk menentukan struktur yang optimal dari suatu neural network.

## 5. PENUTUP

Makalah ini membahas garis besar konsep-konsep dalam softcomputing dan karakteristik masing-masing metode. Sebagai suatu solusi, softcomputing memiliki kelebihan dalam hal kemampuan mengolah informasi mengandung unsur ketakpastian (*uncertainty*), kebenaran parsial, atau pada masalah yang memerlukan proses pembelajaran terhadap *trend* yang dialami sebelumnya. Softcomputing lebih menekankan pada partnership antara metode-metodenya, sehingga kelebihan metode yang satu akan menutup kelemahan dari metode yang lain. Faktor-faktor inilah yang menyebabkan softcomputing menjadi suatu alternatif yang menjanjikan untuk aplikasi yang luas di berbagai bidang.

## 6. REFERENSI

1. Moore's Law dan Intel:  
<http://www.intel.com/research/silicon/m>

2. Homepage Berkeley Initiative in SoftComputing (BISC)  
<http://www-bisc.cs.berkeley.edu/bisc/bisc.memo.html>
3. Masafumi Hagiwara, "Neuro-Fuzzy-GA", Sangyotosho, cetakan ke-9, 2003 (bahasa Jepang).
4. Simon Haykin, Neural Networks: A Comprehensive Foundation, Pearson Education, cetakan ke-3 di India, 1999.