

PROTOTYPE PERANGKAT LUNAK SISTEM KENDALI ADAPTIF MENGGUNAKAN ALGORITMA PEMBELAJARAN PERCEPTRON

Oleh
Masduki Zakaria
Staf Pengajar FT UNY

Abstract

This aims of this research is to build and to make performance test a software prototype of adaptive control system with perceptron learning algorithm.

Needs analysis is performed to find an adaptive control system specifications are required, the result is a blue print software, after a modification of the learning algorithms are tailored to the programming language, the result is a flow chart design. Software implementation is based on the flow chart that was created by using a structured programming language. The performance of the software test which has been developed by research and development approach

The results showed that the group data in the analytical calculations with research data from a running program at every step in iteration, there was no significant difference between the description of data in two research groups.

Keywords: Adaptive control system, Perceptron.

PENDAHULUAN

Secara esensial sistem kendali adaptif pada hakekatnya merupakan suatu sistem yang digunakan untuk mengendalikan *plant* secara adaptif. Pendekatan yang digunakan dalam sistem kendali adaptif antara lain menggunakan: *adaptive model control*, *adaptive inverse control*, dan *adaptive reference control*. Ketiga model tersebut masing-masing dapat dikombinasikan dengan ragam algoritma, antara lain dengan algoritma LMS (*least-mean-square*) (Widrow Bernard, Stearns Samuel D., 1985: 271-298).

Penggunaan algoritma jaringan syaraf tiruan telah banyak dilakukan oleh para peneliti, antara lain: Sridar R. dan Sin Yong-Chul (1996) telah mengimplementasikan jaringan syaraf tiruan dalam VLSI yang ber-

basis pada model jaringan memori asosiatif dengan arsitektur SIMD (*Single Instruction Multiple Data*), metode pembelajaran yang digunakan adalah perambatan mundur. Guyete L, Tuazon J.O, dan Hamidian K. (1996) mengimplementasikan jaringan syaraf tiruan dengan menggunakan chip memori dengan model jaringan perceptron, kajian yang dihasilkan adalah model perceptron tidak menghabiskan banyak waktu pembelajaran yang diperlukan serta implementasi dapat dicapai dengan arsitektur register dan memori.

Donald James dan Akers Lex. A. (1996) merancang jaringan syaraf tiruan dengan pemrosesan chip analog yang digunakan sebagai pengkode pulsa untuk komunikasi yang menggunakan jaringan umpan maju, menggunakan aturan pembelajaran dengan

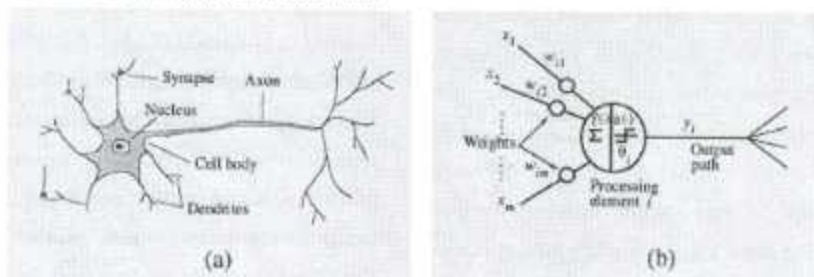
algoritma Hebbian yang ditambah dengan mekanisme hambatan secara lateral. Sedangkan Malhota Raj P. dan Siferd Raymond (1996) mengimplementasikan jaringan syaraf tiruan pada tingkat fabrikasi dengan teknologi CMOS.

Penelitian ini bertujuan untuk (a) membangun prototipe perangkat lunak sistem kendali adaptif menggunakan jaringan syaraf tiruan dengan algoritma pembelajaran Perceptron berbantuan perangkat lunak Pascal, dan (b) menguji kinerja prototipe perangkat lunak sistem kendali adaptif menggunakan jaringan syaraf tiruan dengan algoritma pembelajaran perceptron.

Karakteristik jaringan syaraf tiruan ditentukan oleh faktor-faktor: (a) pola interkoneksi antar neuron, (b) cara pembentukan

nilai bobot jaringan, dan (c) fungsi aktivasi pada neuron.

Asumsi pemodelan matematik yang dikemukakan (Fausett Laurenc, 1994) sehubungan dengan pemodelan neuron biologi ke dalam neuron tiruan adalah: (a) Pemrosesan informasi terletak pada elemen pemroses yang lazim disebut neuron, (b) sinyal antar neuron dilewatkan melalui jalinan interkoneksi, (c) setiap jalinan interkoneksi mempunyai bobot, (d) setiap neuron merupakan fungsi aktivasi untuk menentukan keluarannya. Gambar 1(a) dan 1(b) diperlihatkan korespondensi neuron biologi dengan neuron tiruan.



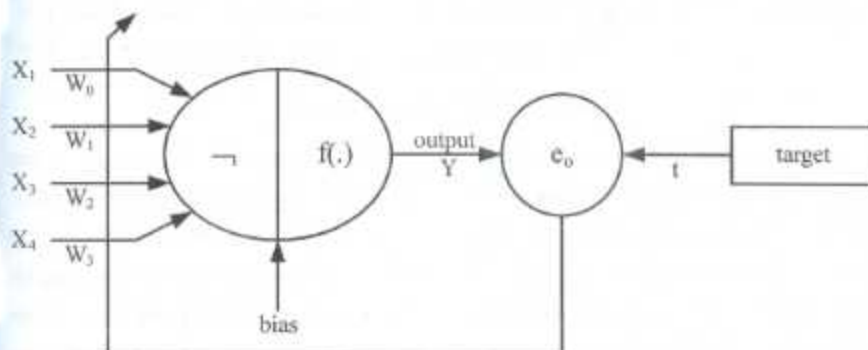
Gambar 1 (a) Neuron Biologi (b) Neuron Tiruan
(Lin Chin-Teng, Lee C.S. George, 1996: 206)

Arsitektur jaringan syaraf tiruan lapis tunggal dengan algoritma pembelajaran perceptron ditunjukkan pada gambar 2. Algoritma pembelajaran perceptron ini akan menerima masukan Variabel-variabel $X_1, X_2,$

$X_3,$ sampai dengan X_i yang masing-masing secara penuh terhubung ke dalam neuron. Proses ini akan terjadi proses perkalian antara X_i dan bobot W_i dan penjumlahan antara hasil perkalian $X_i W_i$ dengan nilai bias

b. Fungsi aktivasi $f(\cdot)$ akan menentukan nilai Y , yang selanjutnya akan dibandingkan

dengan nilai target t pada masing-masing langkah.



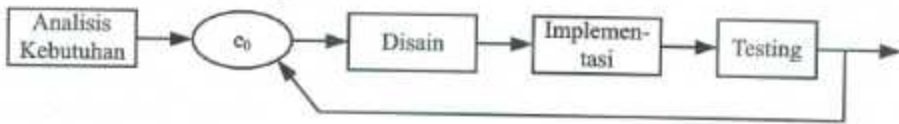
Gambar 2. Arsitektur Jaringan Syaraf Tiruan Lapis Tunggal dengan Algoritma Perceptron

Jika terjadi perbedaan nilai antara nilai Y dengan target t , maka akan terjadi nilai galat $e_0 \neq 0$ yang selanjutnya dilakukan proses perbaruan penghitungan nilai bobot W_i dan nilai bias b . Jika tidak terjadi perbedaan antara nilai Y dan target t , atau nilai galat $e_0 = 0$, maka mengakibatkan tidak terjadinya perubahan pada nilai bobot dan nilai bias. Oleh karena itu pada algoritma pembelajaran perceptron secara adaptif akan mencari nilai bobot dan nilai bias sehingga didapatkan keluaran Y sama dengan nilai target t , dengan diperolehnya $Y = t$, maka proses komputasi dalam jaringan akan berhenti dengan sendirinya.

METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Research and Development* dan model "siklus air terjun" (Pressman, 1997) yang dimodifikasi. Modifikasi aktivitas pada rekayasa perangkat lunak dapat dilakukan dengan mempertimbangkan aspek karakteristik produk perangkat lunak (Suyanto Asep Herman, 2005: 2) yang meliputi: (a) *maintainability*, (b) *dependability*, (c) *Efficiency*, dan (d) *Usability*.

Tahapan yang dilakukan sehubungan dengan pembuatan prototipe perangkat lunak sistem kendali adaptif menggunakan jaringan syaraf tiruan dengan algoritma pembelajaran perceptron ditunjukkan gambar 3.



Gambar 3. Tata Urutan Penelitian

Analisis kebutuhan, melakukan aktivitas antara lain: persyaratan yang diperlukan pada sistem kendali adaptif, algoritma yang digunakan, serta keterpaduan antara sistem dengan algoritma; produk dari aktivitas analisis kebutuhan adalah spesifikasi perangkat lunak sistem.

Disain melakukan aktivitas yang membuat cetak biru perangkat lunak sistem berdasarkan spesifikasi yang telah ditentukan, produk yang dihasilkan adalah berupa diagram alir perangkat lunak sistem dengan menggunakan algoritma pembelajaran

perceptron yang telah disesuaikan dengan kaidah dalam bahasa pemrograman Pascal.

Tahapan implementasi, aktivitas yang dikerjakan adalah mendiskripsikan diagram alir yang telah didisain kedalam bahasa pemrograman Pascal, sehingga produk yang dihasilkan adalah prototipe perangkat lunak sistem, dengan menggunakan bahasa pemrograman Pascal, yang sesuai dengan analisis kebutuhan.

Tabel 1. Aktivitas Penelitian

Tahap	Analisis Kebutuhan	Disain	Implementasi	Testing	Umpun Balik [e ₀]
(1)	(2)	(3)	(4)	(5)	(6)
Proses	<ul style="list-style-type: none"> - Persyaratan sistem - Algoritma yang digunakan - Integrasi sistem 	<ul style="list-style-type: none"> - Merencanakan cetak biru 	<ul style="list-style-type: none"> - Deskripsi diagram alir ke dalam bahasa pemrograman 	<ul style="list-style-type: none"> - Uji kinerja 	<ul style="list-style-type: none"> - Hasil dibandingkan dengan spesifikasi
Hasil	<ul style="list-style-type: none"> - Spesifikasi prototipe 	<ul style="list-style-type: none"> - Algoritma dan Diagram alir 	<ul style="list-style-type: none"> - Prototipe perangkat lunak 	<ul style="list-style-type: none"> - Data pengamatan setiap tahapan iterasi 	<ul style="list-style-type: none"> - Jika e₀ ≠ 0 Hasil ≠ Spesifikasi Cek proses setiap tahap. - Jika e₀ = 0 Hasil = Spesifikasi

Tahapan akhir dari serangkaian proses pada gambar 3 adalah testing, suatu tahap untuk menguji perangkat lunak sistem

kendali adaptif dengan spesifikasi yang ditentukan, keluaran dari langkah ini berupa data hasil proses iterasi setiap langkah, serta

diumpunbalikkan ke e_o . Selanjutnya jika $e_o \neq 0$ maka perlu peninjauan ulang disain dan implementasi sistem; sedangkan jika $e_o = 0$ maka terdapat kesamaan antara analisis kebutuhan dengan hasil akhir dari sistem yang dibangun.

Algoritma pembelajaran perceptron untuk lapis tunggal mengikuti langkah-langkah seperti ditunjukkan pada gambar 4 (Fausett L., 1994: 61). Fungsi aktivasi merupakan pemetaan neuron pada kawasan

tak tentu menjadi jangkauan tertentu. Salah satu fungsi aktivasi tersebut adalah fungsi bipolar, dimana $f(x)$ akan bernilai 1, 0, atau -1, tergantung nilai dari hasil perkalian antara bobot dan bias. Secara umum ditunjukkan persamaan 1 berikut.

$$f(x) = \begin{cases} 1, & \text{jika } y_{in} > \theta \\ 0, & \text{jika } -\theta \leq y_{in} \leq \theta \\ -1, & \text{jika } y_{in} < -\theta \end{cases} \dots\dots\dots (1)$$

Langkah 0 : Inisialisasi awal;
bobot $w = 0$, bias $b = 0$, dan laju pelatihan $\alpha = 0 - 1$.

Langkah 1 : Bila syarat berhenti masih salah, kerjakan langkah 2 sampai dengan langkah 6.

Langkah 2 : Untuk setiap pasangan pelatihan kerjakan langkah 3 - 5

Langkah 3 : Menetapkan aktivasi unit keluaran, menggunakan fungsi aktivasi ambang seperti pada persamaan 1.

Langkah 4 : Menghitung tanggapan unit keluaran, dengan menjumlahkan sinyal masukan berbobot.
 $y_{in} = b_o + \sum x_i w_i$ (2)
 dan keluarannya dikenakan fungsi aktivasi $Y = f(y_{in})$

$$Y = \begin{cases} 1, & \text{jika } y_{in} > \theta \\ 0, & \text{jika } -\theta \leq y_{in} \leq \theta \\ -1, & \text{jika } y_{in} < -\theta \end{cases} \dots\dots\dots (3)$$

Langkah 5 : Memperbarui bobot dan bias bila terjadi galat.
 Bila $Y \neq t$:
 $1. w_i(\text{baru}) = w_i(\text{lama}) + \alpha t x_i$ (4)
 $b(\text{baru}) = b(\text{lama}) + \alpha t$ (5)
 Bila $Y = t$:
 $w_i(\text{baru}) = w_i(\text{lama})$ (6)
 $b(\text{baru}) = b(\text{lama})$ (7)

Langkah 6 : Syarat berhenti, bila tidak ada bobot yang berubah pada langkah 2. Bila kondisi tersebut tidak terpenuhi, maka lanjutkan.

Gambar 4. Algoritma Pembelajaran Perceptron

HASIL PENELITIAN

Hasil penelitian terdiri atas: (a) diagram alir yang disesuaikan dengan bahasa pemrograman Pascal yang ditunjukkan pada gambar 5, (b) data penelitian yang menunjukkan hubungan antara masukan, proses, dan output, yang ditunjukkan pada tabel 1 dan 2, dan (c) *listing* program yang secara lengkap ditunjukkan pada lampiran.

Diagram alir algoritma pembelajaran perceptron dikerjakan sebagai realisasi dari tahapan mula untuk membangun prototipe

$$y_{in} = w_1(0) * x_1(1) + w_2(0) * x_2(1) + b(0) = 0 * 1 + 0 * 1 + 0 = 0$$

Oleh karena nilai masukan berbobot didapatkan $y_{in} = 0$, maka nilai Y di

$$Y = f(y_{in}) = \begin{cases} 1, & y_{in} > 0,2 \\ 0, & 0,2 \geq y_{in} \geq -0,2 \\ -1, & y_{in} < -0,2 \end{cases}$$

Jika nilai $Y = 0$, dan jika nilai target $t(1) = 1$; maka perlu diuji hubungan antara Y dengan $t(n)$. Berpijak pada algoritma pembelajaran perceptron langkah ke-5 pada persamaan 4 sampai dengan persamaan 7, maka nilai Y perlu diuji kesamaan dan ketidaksamaannya dengan $t(n)$. Jika $Y \neq t(n)$,

perangkat lunak menggunakan Arsitektur jaringan syaraf tiruan lapis tunggal dengan algoritma pembelajaran perceptron.

Pengambilan data penelitian dilakukan dengan terlebih dahulu menentukan nilai awal (inisialisasi) antara lain: laju pelatihan, nilai bias, penentuan bobot awal, dan fungsi aktivasi yang digunakan.

Secara analitik penentuan nilai y_{in} berdasarkan persamaan 2, nilai y_{in} untuk data ke-0 didapatkan nilai masukan berbobot:

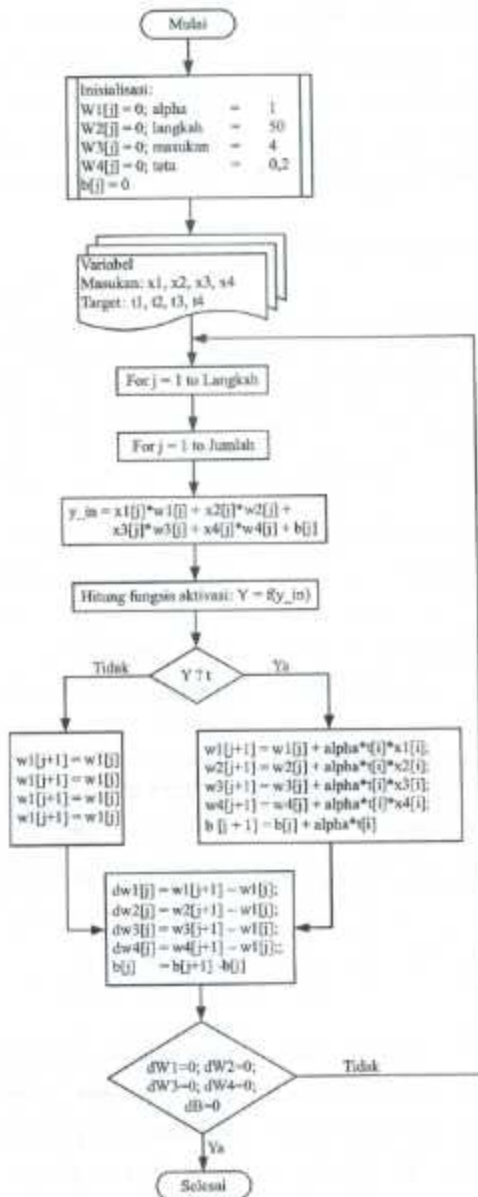
didapatkan dengan memasukkan nilai y_{in} pada persamaan 3, dengan demikian nilai

maka perlu dilakukan pembaharuan nilai bobot dan bias,

$$\begin{aligned} w_1(1) &= w_1(0) + \alpha * t(1) * x_1(1) = 0 + 1 * 1 * 1 = 1; \\ w_2(1) &= w_2(0) + \alpha * t(1) * x_2(1) = 0 + 1 * 1 * 1 = 1; \\ b(1) &= b(0) + \alpha * t(1) = 0 + 1 * 1 = 1; \end{aligned}$$

serta menghitung selisih antara bobot dan bias pada langkah yang ke-n.

$$\begin{aligned} \Delta w_1(1) &= w_1(1) - w_1(0) = 1 - 0 = 1; \\ \Delta w_2(1) &= w_2(1) - w_2(0) = 1 - 0 = 1; \\ \Delta b(1) &= b(1) - b(0) = 1 - 0 = 1; \end{aligned}$$



Gambar 5. Diagram alir sistem kendali adaptif menggunakan algoritma pembelajaran perceptron

Data hasil penelitian secara lengkap ditabulasi dalam bentuk kuantitatif dan hasilnya digunakan untuk mencocokkan antara perhitungan berdasarkan persamaan (1) sampai dengan persamaan (6) dengan hasil

yang didapatkan dari *running program* yang didasarkan atas diagram alir pada gambar 5. *Listing program* secara ditunjukkan pada lampiran. Secara kuantitatif data pada langkah ke-1 ditunjukkan pada tabel 2.

Tabel 2. Ringkasan hasil pada langkah ke-1

	x_1	x_2	y_{in}	Y	t	Δw_1	Δw_2	Δb	Bobot		
									w_1	w_2	b
									(10)	(11)	(12)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
(1)	1	1	0	0	1	1	1	1	1	1	1
(2)	1	0	2	1	-1	-1	0	-1	0	1	0
(3)	0	1	1	1	-1	0	-1	-1	0	0	-1
(4)	0	0	-1	-1	-1	0	0	0	0	0	-1

Melihat hasil pada tabel 1 kolom 10, 11, dan 12 terlihat bahwa nilai w_1 , w_2 , dan b masih berubah-ubah, oleh karena itu dilakukan proses pengulangan perhitungan sampai didapatkan nilai w_1 , w_2 , dan b dalam

keadaan konstan, dengan cara iterasi proses perhitungan yang ditunjukkan pada diagram alir. Hasil analitik pada langkah ke-10 didapatkan nilai w_1 , w_2 , dan b yang konstan seperti ditunjukkan pada tabel 3.

Tabel 3 Ringkasan hasil pada langkah ke-10

	x_1	x_2	y_{in}	Y	t	Δw_1	Δw_2	Δb	Bobot		
									w_1	w_2	b
									(10)	(11)	(12)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
(1)	1	1	1	1	1	0	0	0	2	3	-4
(2)	1	0	-2	-1	-1	0	0	0	2	3	-4
(3)	0	1	-1	-1	-1	0	0	0	2	3	-4
(4)	0	0	-4	-1	-1	0	0	0	2	3	-4

Secara kuantitatif terdapat dua kelompok data penelitian, yaitu kelompok kontrol merupakan representasi dari hasil perhitungan secara analitik dari masing-masing langkah pada algoritma pembelajaran perceptron yang ditunjukkan pada gambar 5

yang hasil akhir ditunjukkan pada tabel 2. Serta kelompok eksperimen merupakan representasi data dari produk uji perangkat lunak sistem. Sehingga dari kedua variasi data tersebut dapat diketahui tingkat keberhasilan dan ketidakberhasilan dari hasil

prototipe perangkat lunak sistem yang dibangun. "

Data penelitian yang dihasilkan tidak akan berhenti manakala dalam satu langkah nilai bobot dan bias yang masih berubah-ubah, dengan catatan masih dalam jangkauan pelatihan, walaupun kadang diperoleh nilai yang sama antara keluaran Y dengan nilai target t . Akan tetapi ketika dalam satu langkah tidak terjadi perubahan nilai bobot dan bias, maka nilai Y sama dengan nilai target t . Sehingga keluaran sistem tetap akan stabil. Hal ini direpresentasikan dengan $Y = t$, walupun nilai masukan pada neuron X_1, X_2, X_3 , sampai dengan X_j berubah-ubah.

Jumlah langkah dalam proses iterasi pada prototipe perangkat lunak sistem yang dibangun akan menentukan proses menuju nilai bobot dan bias yang stabil, disamping itu nilai pelatihan α juga ikut menentukan kecepatan proses iterasi pada setiap langkah untuk menuju nilai bobot W , dan bias b yang tetap.

Diskripsi data antara 2 kelompok penelitian, yaitu kelompok hasil penghitungan secara analitik dengan data hasil *running program* menunjukkan kedua data yang dihasilkan mempunyai nilai yang sama dalam setiap langkah pada masing-masing iterasi.

KESIMPULAN

Prototipe perangkat lunak sistem kendali adaptif menggunakan algoritma pembelajaran perceptron dapat dibangun melalui beberapa tahapan, yaitu: analisis kebutuhan, penentuan topologi jaringan, memodifikasi algoritma pembelajaran perceptron dengan algoritma dalam bahasa pemrograman, mengimplementasikan ke dalam bahasa pemrograman pascal, menguji dan melakukan koreksi prototipe perangkat lunak yang dihasilkan dengan berpedoman pada analisis kebutuhan.

Unjuk kerja sistem yang dibangun yang direpresentasikan dari data hasil *running program* sesuai dengan data hasil penghitungan secara analitik.

PUSTAKA

- Donald James, Aker Lex A., 1996, *A Neural Processing Node with On-chip Learning*, IEEE Technology Update Series.
- Fausett Laurene, 1994, *Fundamentals of Neural Networks, Architecture, Algorithms, and Application*, Prentice-Hall Inc. Englewood Cliffs New Jersey.
- Gallant Stephan, 1990, *Perceptron-Based Learning Algorithms*, IEEE Transactions on Neural Network Vol. 1 No. 2 Juni 1990.
- Horn L. Wayne, 1995, *Structured Programming in Turbo Pascal*, Prentice-Hall Englewood Cliffs New Jersey.

- Lin Chin-Teng, Lee C.S. George, 1996, *Neural Fuzzy System A Neuro Fuzzy Synergism to Intelligent Systems*, Prentice-Hall Inc, Singapore
- Phillips Charles L., Harbor Royce D., 1996, *Feedbaack Control Systems*, Singapore: Prentice-Hall.
- Pressman, Roger S, 1997, *Software Engineering: A Practitioner's Approach*, McGraw-Hill Companies, Inc.
- Simpson Patrick K., 1996, *Foundation of Neural Networks*, IEEE Technology Update Series.
- Sridar Ramahngam, Sin Yong-Chul, 1996, *VLSI Neural Network Architectures*, IEEE Update Series.
- Suyanto Asep Herman, 2005, *Review Metodologi Pengembangan Perangkat Lunak*, <http://www.asep-hs.web.ugm.ac.id>.
- Tuazon J.O. dkk., 1996, *A Digital Neural Network and Its Application*, IEEE Update Series.
- Wang Zenfeng, dkk., 1996, *Single-Chip Realization of Fuzzy Logic Controller with Neural Network Structure (NNFLC)*, IEEE Technology Update Series.
- Weisman Omri, Pollack Ziv., 1995, *The Perceptron*, <http://www.c.s.bgu.il/~omri/Perceptron> (Juli 2000)
- Widrow Bernard, Stearns Samuel D., 1985, *Adaptive Signal Processing*, Englewood Clif NJ: Prentice-Hall.