35077/H 069

MILIK PERPUSTAKAAN ITS

ITS
Institut
Teknologi
Sepuluh Nopember

RSF
629.836
Fat
P-1
2008

TUGAS AKHIR RF 1483

PERANCANGAN SISTEM PENGENDALIAN KECEPATAN *STEAM TURBINE* DENGAN METODE *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* DI PLTU GRESIK

ARIF FATCHURROHMAN
NRP 2402 100 069

Dosen Pembimbing
Ir. Syamsul Arifin, MT.

JURUSAN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2008

PERPUSTAKAAN ITS

Tgl. Terima	7-8-2008
Terima Dari	H
No. Agenda Prp.	231378

ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT RF-1483

DESIGN OF SPEED STEAM TURBINE CONTROL SYSTEM BASED ON ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM METHOD AT PLTU GRESIK

ARIF FATCHURROHMAN
NRP 2402 100 069

Advisor Lecturers
Ir. Syamsul Arifin, MT.

JURUSAN TEKNIK FISIKA
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2008

PERANCANGAN SISTEM PENGENDALIAN KECEPATAN *STEAM TURBINE* DENGAN METODE *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* DI PLTU GRESIK

TUGAS AKHIR

OLEH:

ARIF FATCHURROHMAN
NRP. 2402 100 069

Surabaya, Agustus 2008
Mengetahui/Menyetujui

Pembimbing,

Ir. Syamsul Arifin, MT.
NIP. 131 835 487

Ketua Jurusan
Teknik Fisika FTI-ITS

Dr. Bambang Lelono W, ST. MT.
NIP. 132 137 895

PERANCANGAN SISTEM PENGENDALIAN KECEPATAN *STEAM TURBINE* DENGAN METODE *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* DI PLTU GRESIK

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Teknik
Pada
Bidang Minat Rekayasa Instrumentasi
Program Studi S-1 Jurusan Teknik Fisika
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember

Oleh :
ARIF FATCHURROHMAN
NRP. 2402 100 069

Disetujui Oleh Tim Penguji Tugas Akhir :

1. Ir. Syamsul Arifin, MT. ..... (Pembimbing)
2. Lizda J. M. ST. MT. ..... (Penguji I)
3. Dr. Ir. Aulia S. A. ..... (Penguji II)
4. Katherin I. ST. MT. ..... (Penguji III)

SURABAYA
Agustus, 2008

PERANCANGAN SISTEM PENGENDALIAN KECEPATAN *STEAM TURBINE* DENGAN METODE *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* DI PLTU GRESIK

Nama Mahasiswa	*: Arif Fatchurrohman*
NRP	*: 2402.100.069*
Jurusan	*: Teknik Fisika FTI-ITS*
Dosen Pembimbing	*: Ir. Syamsul Arifin, MT.*

Abstrak

Sistem pengendalian kecepatan steam turbine adalah salah satu sistem pengendalian utama yang digunakan dalam sistem tenaga listrik karena sangat berpengaruh terhadap kualitas listrik yang dihasilkan khususnya frekuensi. Dasar pengendalian kecepatan steam turbine menggunakan micropocessor kontroler dengan aplikasi software terintegrasi yang dirancang untuk valve turbin uap, di antaranya adalah kendali PID, kemudian dikembangkan ke kendali logika fuzzy dan Adaptive Neuro-Fuzzy Inference System (ANFIS). Penelitian ini menggunakan simulasi MATLAB berbasis data di industri. Pertama-tama dilakukan survei di suatu industri untuk mendapatkan persamaan matematis sistem pengendali kecepatan turbin uap yaitu berupa kontrol PI. Dari kontroler tersebut akan di ambil data acak untuk mendapatkan sejumlah data set input-output yang akan digunakan untuk melakukan pembelajaran kontroler ANFIS. Kontroler ANFIS ini akan disimulasikan dan di uji untuk menggantikan kotroler PI. Dari analisa simulasi yang telah dilakukan, Indeks performansi ANFIS pada uji set point 1 pu sebagai berikut: Rise Time, Tr = 3,927 detik; Settling Time, Ts = 16,0941 detik dan Maximum Overshoot, Mp = 7,4001 %

Kata kunci: *ANFIS, Steam Turbine, Governor, Setpoint*

DESIGN OF SPEED STEAM TURBINE CONTROL SYSTEM BASED ON ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM METHOD AT PLTU GRESIK

Student Name	***: Arif Fatchurrohman***
NRP	***: 2402 100 069***
Departement	***: Teknik Fisika FTI-ITS***
Advisor	***: Ir. Syamsul Arifin, MT.***

Abstract

Speed steam turbine control system is one of especial control system used in system electric power because influential to electrics quality specially frequency. Basic Speed steam turbine control use micropocessor controller with the integrated application software designed for valve steam turbine, among other things is PID controller, later then developed to fuzzy logic and Adaptive Neuro-Fuzzy Inference System (ANFIS) control. This research use the simulation MATLAB base on data in industry. First of all by survey in industry to get the mathematical equation of speed steam turbine control system in the form of control PI. From the controller will taking random data to get a number of data set the input-output used to training of ANFIS controller. ANFIS controller will be simulated and tested to replace PI controller. From simulation analysis which have been done, the result is shown by Index of ANFIS controller performance at 1 pu set point test, as follows,: Rise Time, Tr = 3,927 second; Settling Time, Ts = 16,0941 second and Maximum Overshoot, Mp = 7,4001 %

Keyword: *ANFIS, Steam Turbine, Governor, setpoint*

KATA PENGANTAR

Segala puji bagi Allah Subhanahuwata'ala yang telah banyak melimpahkan rahmat, taufiq, hidayah, keterbukaan, serta memberi kesabaran dan kesehatan kepada penulis sehingga dapat menyelesaikan tugas akhir yang berjudul:

PERANCANGAN SISTEM PENGENDALIAN KECEPATAN *STEAM TURBINE* DENGAN METODE *ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM* DI PLTU GRESIK

Tugas akhir ini disusun guna memenuhi persyaratan bagi seorang mahasiswa untuk memperoleh gelar Sarjana Teknik pada Jurusan Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Selama menyelesaikan tugas akhir ini penulis telah banyak mendapatkan bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Ibu dan Bapak tercinta, adik tersayang Dewi dan utami, yang telah memberikan dorongan secara moril maupun materil.yang telah memberikan kesempatan untuk menuntut ilmu sejauh ini dan selalu mendoakan agar penulis meraih kesuksesan.
2. Keluarga di Surabaya dan Purbalingga atas do`a dan semangatnya.
3. Bapak Syamsul Arifin selaku pembimbing tugas akhir yang telah memberikan segenap kesabaran meluangkan waktu untuk berdiskusi, bimbingan serta ilmu yang tiada ternilai harganya.
4. Bapak Hendra Cordova selaku dosen wali yang telah memberikan bimbingan serta ilmu nya
5. Bapak dan Ibu dosen Teknik Fisika yang telah banyak memberikan ilmunya sehingga penulis dapat menyelesaikan jenjang kuliah sampai tugas akhir ini.

6. Seluruh staf pegawai Jurusan Teknik Fisika atas semua bentuk kerjasamanya selama masa kuliah.
7. Teman-teman semua angkatan 2002, yang telah berjuang dan belajar bersama di teknik Fisika ini
8. Semua pihak yang tidak dapat kami sebutkan satu persatu yang telah membantu sehingga tugas akhir ini dapat terselesaikan.

Dengan menyadari bahwa segala kesempurnaan adalah hanya milik Allah SWT dan kekurangan hanyalah milik hamba-Nya. Penulis menyadari bahwa tugas akhir ini tidaklah sempurna, tetapi penulis berharap ini dapat memberikan kontribusi yang berarti dan dapat menambah wawasan bagi pembaca dan mahasiswa yang nantinya dapat digunakan sebagai referensi pengerjaan tugas akhir yang akan datang.

Surabaya, Juli 2008

Penulis

DAFTAR ISI

		Hal
Halaman Judul		i
Lembar Pengesahan		ii
Abstrak		iii
Kata Pengantar		iv
Daftar Isi		vi
Daftar Gambar		viii
Daftar Tabel		x
Daftar Simbol		xi
BAB I	PENDAHULUAN	1
	1.1 Latar Belakang	1
	1.2 Permasalahan	2
	1.3 Batasan Masalah	3
	1.4 Tujuan dan manfaat penelitian	3
	1.5 Metodologi Penelitian	3
	1.6 Sistematika Laporan	4
BAB II	TEORI PENUNJANG	7
	2.1 Sistem Tenaga Listrik	7
	2.2 Logika Fuzzy	13
	2.3 Struktur Dasar Sistem Inferensi Logika Fuzzy	18
	2.4 Adaptive Neuro-Fuzzy Inference System	21
BAB III	PERANCANGAN DAN PEMODELAN	29
	3.1 Pemodelan Matematis	29
	3.3 Perancangan Kontroler ANFIS	35
BAB IV	ANALISA DATA DAN PEMBAHASAN	
	4.1 Analisa Data	46
	4.2 Pengujian Tracking	56

Hal

4.3 Perbandingan Sistem Pengendalian 60

BAB V KESIMPULAN DAN SARAN 63
5.1 Kesimpulan 63
5.2 Saran 63

Daftar Pustaka 64
Lampiran A DATA TEKNIK PERALATAN
Lampiran B LISTING PROGRAM

DAFTAR GAMBAR

		Hal
Gambar 2.1	Sistem pengendalian tenaga listrik	7
Gambar 2.2	Turbin	8
Gambar 2.3	sistem pengatur governor	9
Gambar 2.4	Model governor *Isochronous*	10
Gambar 2.5	Model governor *load reference*	11
Gambar 2.6	Himpunan fuzzy dan fungsi keanggotaannya	14
Gambar 2.7	Fungsi keanggotaan S	16
Gambar 2.8	fungsi keanggotaan segitiga	17
Gambar 2.9	fungsi keanggotaan gaussian	17
Gambar 2.10	fungsi keanggotaan *bell gaussian*	18
Gambar 2.11	Struktur Dasar Sistem Inferensi Fuzzy	18
Gambar 2.12	Sugeno-type fuzzy inference system	22
Gambar 2.13	Arsitektur ANFIS	23
Gambar 2.14	Arsitektur ANFIS model fuzzy sugeno	24
Gambar 3.1	skema Sistem Pengendalian Kecepatan *Steam Turbine*	29
Gambar 3.2	Pemodelan Turbin Uap	29
Gambar 3.3	Fungsi Transfer *Steam Turbine*	30
Gambar 3.4	Fungsi Transfer *Steam Chest*	31
Gambar 3.5	Model Generator	31
Gambar 3.6	Model Generator dan Beban	32
Gambar 3.7	Fungsi Transfer Control Valve	32
Gambar 3.8	Fungsi Transfer Power Transduser	33
Gambar 3.9	Fungsi Transfer Speed Drop	33
Gambar 3.10	Sistem Pengendalian Kecepatan *Steam Turbine*	34
Gambar 3.11	Diagram Blok Sistem Pengendalian Kecepatan *Steam Turbine*	34
Gambar 3.12	Fungsi Transfer Sistem Pengendalian Kecepatan *Steam Turbine*	35
Gambar 3.13	Skema dasar identifikasi sistem	36

		Hal
Gambar 3.14	Skema ANFIS 2 masukan dengan 3 fungsi keanggotaan	36
Gambar 3.15	Flowchart ANFIS	37
Gambar 3.16	Skema dasar ANFIS sebagai kontroler	43
Gambar 4.1	Simulasi Sistem Kontroler PI	47
Gambar 4.2	Respon Sistem Kontroler PI	47
Gambar 4.3	Simulasi Sistem pengambilan data input-output	48
Gambar 4.4	Data Sampel dan Data Target	49
Gambar 4.5	Fungsi Keanggotaan Awal dan Akhir	49
Gambar 4.6	RMSE Proses *Training*	51
Gambar 4.7	Data Target-Output Proses *training*	51
Gambar 4.8	Blok Diagram Sistem Pengendalian Kecepatan *Steam Turbine* dengan ANFIS	52
Gambar 4.9	Respon sistem pada pengujian set point 1 pu tanpa beban	53
Gambar 4.10	Respon sistem pada pengujian set point 1 pu	54
Gambar 4.11	Respon sistem pada pengujian set point 1,1 pu	54
Gambar 4.12	Respon sistem pada pengujian set point -0,9 pu	55
Gambar 4.13	Respon sistem pada pengujian tracking set point	56
Gambar 4.14	Respon sistem pada pengujian tracking beban	58
Gambar 4.15	Respon sistem pada pengujian gabungan	59
Gambar 4.16	Perbandingan Respon sistem kontroler ANFIS dan PI	60

DAFTAR TABEL

		Hal
Tabel 2.1	Proses Pembelajaran *Hybrid* ANFIS	25
Tabel 4.1	Nilai parameter Premis awal dan akhir Training	50
Tabel 4.2	Nilai parameter Konsekuen	50
Tabel 4.3	Perbandingan performansi pengendalian kecepatan turbin	61

DAFTAR SIMBOL

f = frekuensi (Hz)
ω = Kecepatan Turbine (per unit, pu ; 1 pu speed = 3000 rpm)
$\dot{m}$ = Laju aliran uap (Kg/dt)
h = *Enthalpi* (J/Kg)
V = Volume Penyimpanan uap (m^3)
Pm = Daya mekanik Turbin (watt, w)
Pe = Daya elektrik generator (watt, w)
K = Penguatan turbin
H = Momen inersia (m^2 Kg)

BAB I
PENDAHULUAN

1.1 Latar Belakang

Pada pembangkitan tenaga listrik, turbin atau penggerak mula (*prime mover*) memutar generator untuk menghasilkan output tenaga listrik. Faktor yang mempengaruhi output tenaga listrik meliputi frekuensi, tegangan dan daya. Oleh sebab itu, kestabilan frekuensi dan tegangan harus dikontrol dengan baik. Stabilitas frekuensi pada generator dipengaruhi kecepatan putar turbin. Pengendalian kecepatan turbin uap dilakukan agar diperoleh kualitas dan kuantitas produksi listrik yang dihasilkan tetap terjaga.

Dalam penyediaan tenaga listrik untuk konsumen, frekuensi harus dijaga agar tetap konstan. Penyimpangan frekuensi dari nominal harus selalu dalam batas toleransi yang diperbolehkan (Marsudi, 2006). Daya aktif mempunyai hubungan erat dengan nilai frekuensi dalam sistem, sehingga pengendalian frekuensi dalam sistem harus dilakukan dengan melakukan pengendalian penyediaan daya aktif dalam sistem. Pengendalian tersebut dilakukan dengan pengendalian besarnya kopel mekanis yang berhubungan dengan pengendalian frekuensi sekaligus kecepatan turbin yang diperlukan untuk memutar generator, hal ini berarti melakukan pengendalian pemberian uap pada turbin uap atau pengendalian pemberian bahan bakar pada turbin gas dan mesin *diesel*.

Kestabilan dinamik dalam sistem daya listrik ditentukan oleh kemampuan berbagai komponen pembangkit dalam pemberian transfer respons terhadap perubahan beban yang terjadi secara tiba-tiba dan periodik sehingga dapat mempengaruhi kestabilan dinamik sistem (Robandi, 2006). Karena permintaan daya yang berkembang dan perubahan beban yang yang bervariasi setiap waktu, kemampuan pembangkit untuk merespons kenaikan output secara cepat dari nol hingga beban

penuh merupakan hal yang penting. Kecepatan governor pada setiap unit pembangkit memberikan kecepatan utama sebagai fungsi kontrol yang sangat diperlukan dalam sistem tenaga. Oleh sebab itu, frekuensi bukan merupakan besaran konstan, tetapi merupakan besaran yang terus-menerus berubah sesuai perubahan beban sehingga diperlukan sistem pengendalian kecepatan turbin

Beberapa metode telah digunakan untuk mengendalikan suatu sistem elektro-hidrolik, diantaranya adalah kendali PID dan kendali logika fuzzy (Wahyunggoro, 2005). Kelemahan sistem kendali logika fuzzy adalah penggunaan metode coba-coba dalam proses optimasinya. Salah satu cara mengatasi kelemahan tersebut adalah dikembangkannya sistem kendali *Adaptive Neuro Fuzzy Inference System* (ANFIS). Untuk mendukung tercapainya performansi optimal sistem, diperlukan suatu sistem kontrol kecepatan turbin yang mampu beradaptasi secara cepat terhadap perubahan-perubahan parameter proses yang terjadi yaitu menggunakan metode *ANFIS.*

ANFIS adalah sistem inferensi fuzzy yang diimplementasikan dalam jaringan adaptif (Jang, 1994). Parameter pada *ANFIS* adalah fungsi keanggotaan premis dan konsekuensi. Pembelajaran ANFIS adalah pengubahan parameter fungsi keanggotaan masukan dan keluaran. Pembelajaran ANFIS dapat menggunakan algoritma perambatan balik atau algoritma *hybrid.* Algoritma *hybrid* adalah gabungan antara algoritma perambatan balik untuk menentukan parameter konsekuensi dan metode kuadrat terkecil (*Least Squares Estimate*) untuk memperbaharui bobot premis.

Oleh karena itu, Pada penelitian ini akan dirancang dan disimulasikan sistem pengendalian kecepatan *steam turbine* dengan sistem kendali metode *ANFIS* yang diharapkan dapat meningkatkan performansi optimal sistem.

1.2 Permasalahan

Permasalahan dalam penelitian adalah sebagai berikut:

- Bagaimana membuat pemodelan komponen governor, turbin uap, sistem inersia rotor-beban dan *speed droop governor* sebagai sistem pengandalian kecepatan *steam turbine*.
- Bagaimana merancang sistem pengendalian kecepatan turbin uap metode ANFIS untuk dapat meningkatkan performansi optimal sistem dan mensimulasikan pada matlab.

1.3 Batasan Masalah

Dalam tugas akhir ini batasan masalah yang telah ditetapkan, antara lain :

- *Steam Turbine* yang kendalikan adalah jenis sistem *non-reheat turbine* kapasitas 100 MW *single area system* yang ada di PT. PLTU Gresik Unit 1.
- Variabel yang dikontrol adalah kecepatan turbin dengan variabel yang dimanipulasi adalah laju aliran *steam*
- ANFIS menggunakan model TSK (Takagi, Sugeno dan Kang), proses pembelajaran *Inverse* dan model fungsi keanggotaan *gaussian*.

1.4 Tujuan Dan Manfaat Penelitian

Tujuan dan manfaat dari penelitian ini adalah Merancang sistem pengendalian kecepatan turbin uap dengan metode ANFIS untuk memperbaiki kinerja sistem kontrol kecepatan *steam turbine*, untuk kemudian mensimulasikan unjuk kerja sistem pengendalian tersebut dan membandingkan metode pengendalian kecepatan yang ada di PLTU GRESIK dengan metode *Adaptive Neuro-Fuzzy Inference System* (ANFIS) yang telah dibuat.

1.5 Metodologi Penelitian

Metodologi penelitian dalam pengerjaan Tugas akhir ini adalah sebagai berikut :

A. Studi literatur dan data spesifikasi model
Studi tentang pemahaman *plant*, proses *plant*, *software* dan algoritma kontroler. Pengambilan data spesifikasi model digunakan sebagai referensi untuk memodelkan *plant*.

B. Memodelkan plant
Memodelkan komponen sistem governor, *steam turbine*, dan generator sebagai sistem pengendalian kecepatan *steam turbine*.

C. Pengujian model
Plant yang sudah dimodelkan disimulasikan menggunakan *software* Matlab 7.04.

D. Simulasi pengujian performansi struktur pengendali
Melakukan pengujian Sistem Kontrol ANFIS pada sistem pengendalian kecepatan *steam turbine* dengan menggunakan *software* Matlab 7.04.

E. Analisa hasil pengujian
Hasil pengujian dianalisa kelebihan dan kekurangan sistem pengendalian.

F. Penarikan kesimpulan dari hasil pengujian.

G. Penulisan laporan

1.6 Sistematika Laporan

Sistematika laporan yang digunakan dalam penyusunan laporan tugas akhir ini adalah sebagai berikut :

Bab I Pendahuluan

Berisi tentang latar belakang, permasalahan, batasan masalah, tujuan dan manfaat, metodologi penelitian, dan sistematika laporan.

Bab II Teori Penunjang

Berisi tentang dasar sistem tenaga listrik, Fuzzy, Dasar Sistem Inferensi Fuzzy, *Adaptive Neuro-Fuzzy*

Inference System (ANFIS) dan algoritma pembelajaran ANFIS.

Bab III Pemodelan Sistem Dan Perancangan Sistem Pengendalian.

Dalam bab ini akan dijelaskan tentang sistem pengendalian kecepatan *steam turbine*, penurunan model matematis dari *plant* dan perancangan sistem ANFIS *controller*.

Bab IV HASIL DAN ANALISA

Pada bab ini berisi analisa hasil simulasi berdasarkan uji performansi baik secara kualitatif maupun kuantitatif, pengujian tracking dan perbandingan sistem pengendalian Kontroler PI dan ANFIS .

Bab V KESIMPULAN DAN SARAN

Berisi tentang hasil yang diperoleh dari analisa sistem, analisa data, dan saran.

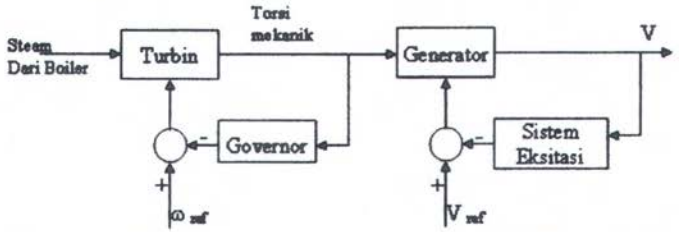
----------(HALAMAN INI SENGAJA DIKOSONGKAN)----------

BAB II
TEORI PENUNJANG

Dalam sistem tenaga listrik, pengendalian frekuensi dan beban merupakan persoalan utama karena berhubungan langsung dengan kualitas listrik yang dihasilkan untuk pelanggan. Permintaan beban yang sering berubah-ubah menimbulkan perubahan frekuensi yang dapat menurunkan kualitas listrik yang dihasilkan. Untuk mempertahankan frekuensi dalam batas toleransi yang diperbolehkan, penyediaan dan pembangkitan daya aktif dalam sistem harus selalu disesuaikan dengan kebutuhan pelanggan atas daya aktif.

2.1 Sistem Tenaga Listrik

Dalam sistem tenaga listrik terdapat dua sistem pengedalian yang utama, yaitu sistem pengendalian kecepatan turbin yang berhubungan langsung dengan pengendalian frekuensi listrik yang dihasilkan dan sistem pengendalian tegangan atau Sistem eksitasi merupakan sistem pengendalian yang berfungsi untuk mengontrol tegangan, bila terjadi perubahan pada tegangan maka sistem ini berfungsi sebagai umpan balik untuk mengembalikan tegangan kerja ke nilai semula.

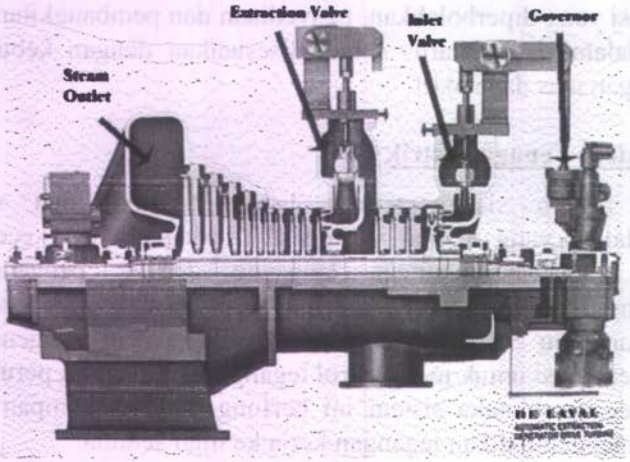


Gambar 2.1 Sistem pengendalian tenaga listrik (Kundur, 1994)

2.1.1 Turbin

Turbin merupakan mesin yang berputar yang digerakan oleh aliran fluida untuk menggerakan suatu peralatan lain. Turbin yang paling sederhana terdiri dari satu bagian yang bergerak

(rotor) berupa poros dengan sudu-sudu yang menempel. Untuk turbin yang komplek, mempunyai susunan sudu bergerak berselang-seling dengan sudu tetapnya. Aliran fluida berupa *steam* melakukan aksi terhadap sudu (*blades)*, atau sudu melakukan reaksi terhadap aliran fluida, sehingga terjadi konversi energi thermal menjadi energi mekanis berotasi yang menyebabkan sudu-sudu berputar dan memberikan energi ke rotor. Perputaran rotor ini yang akan menggerakan generator yang akan mengubah energi mekanik menjadi energi listrik.

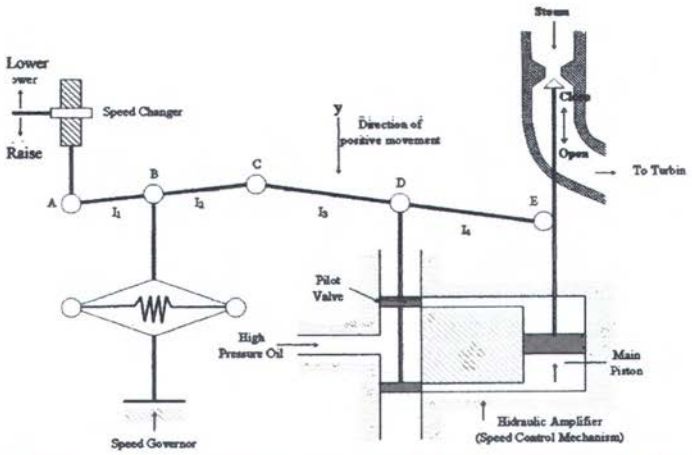


Gambar 2.2 Turbin (Kundur, 1994)

2.1.2 Governor

Governor merupakan komponen yang mengatur kecepatan masuknya aliran bahan bakar berupa uap kedalam turbin dengan cara mengatur pergerakan katup utama uap yang masuk ke turbin governor berfungsi untuk menyesuaikan nilai torsi mekanik (yang memutar rotor) terhadap torsi elektrik bila terjadi perubahan pada torsi beban (torsi elektrik) dan sebagai umpan balik untuk mendapatkan keseimbangan torsi.

Gambar governor mekanik atau konvensional secara skematik dapat dilihat pada gambar :



Gambar 2.3 Sistem pengatur governor mekanik (konvensional) (Marsudi, 2006)

Sistem governor konvensional ini terdiri dari komponen-komponen :

A. Fly ball (bola berputar)
   Komponen ini merupakan yang paling utama dalam sistem, merupakan alat yang sensitif terhadap perubahan putaran atau kecepatan (frekuensi)

B. Penguat Hidrolik
   Terdiri dari katup kemudi dan piston utama, yang berfungsi mengatur pembukaan atau penutupan uap ke turbin.

C. Mekanisme Sambungan
   A-B-C merupakan sambungan batang.
   C-D-E merupakan sambungan batang yang lain , berpusat di D. Mekanisme sambungan ini akan mengatur pergerakan katup yang sebanding dengan putaran.

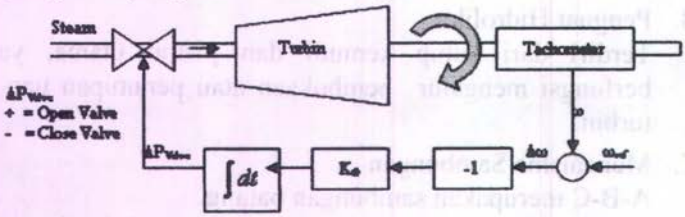
D. Pengubah kecepatan
   merupakan komponen yang memberikan setting daya keluaran turbin.

Sistem pada gambar pada saat awal operasi dapat diasumsikan bahwa:

- Sistem beroperasi pada keadaan tunak (*steady state*), sambungan mekanik dan katup tertutup
- Katup uap terbuka dengan magnitude tertentu.
- Turbin berputar dengan kecepatan tetap, daya keluaran turbin sebanding dengan beban generator.

Governor merupakan sistem pengendalian untuk melakukan pembuka katup uap sehingga aliran uap sesuai dengan beban yang ada pada turbin. Governor digunakan sebagai 'interface' antara turbin penggerak dan generator. Pada saat sekarang ini sebagian besar governor konvensional sudah digunakan lagi., digantikan oleh governor digital.

Governor dibedakan menjadi jenis operasi, yaitu *isochronous* dan *load reference*. Pada mode *isochronous*, governor sudah memiliki "*setting point*" *Pmech* (daya mekanik) yang besarnya sesuai dengan rating generator atau menurut kebutuhan (EEL-6266, 2004). Governor akan mengimbangi kecepatan putar dari turbin sehingga perubahan beban akan mengakibatkan perubahan kecepatan pada turbin yang secara langsung mempengaruhi frekuensi system.

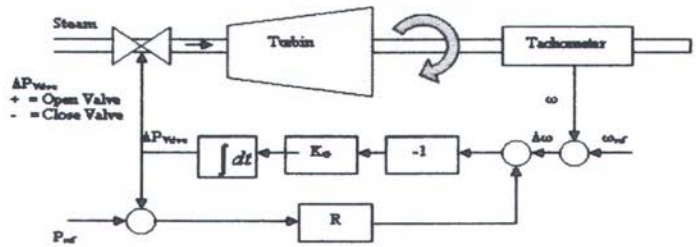


Gambar 2.4 Model governor *Isochronous* (EEL-6266, 2004)

Governor *Isochronous* membuat frekuensi error menjadi nol sesuai dengan perubahan kecepatannya. Kelemahan dari jenis operasi ini adalah tidak dapat digunakan untuk dua generator atau lebih dalam satu sistem elektrik karena akan menyebabkan tabrakan frekuensi dan kesulitan distribusi beban antar generator.

Lain halnya dengan jenis operasi *load reference, "set point"* putaran governor ditentukan berdasarkan kebutuhan daya listrik sistem pada saat itu (*real time*). Kemudian melalui internal proses di dalam governor (sesuai dengan kontrol logic dari manufaktur), governor akan menyesuaikan nilai output daya mekanik turbin supaya sesuai dengan daya listrik yang dibutuhkan sistem.

Untuk mendapatkan frekuensi yang diinginkan (nominal) akibat dari penambahan atau pengurangan beban atau lepasnya pembangkit dari sistem yaitu dengan mengatur pembukaan katup-katup pengatur (*control valve*) bahan bakar berupa *steam* dari penggerak mula (*prime mover*) atau turbin. Penggerak mula atau turbin uap selalu dilengkapi dengan pengatur perputaran (*speed droop governor*). Governor ini alat utama untuk mengatur daya dan frekuensi. Posisi katup governor diatur dengan menggunakan posisi pengubah kecepatan untuk mengatur kecepatan turbin dan daya keluaran. Daya listrik keluaran harus seimbang dengan jumlah total permintaan beban pada sistem supaya dapat beroperasi pada kondisi matap.



Gambar 2.5 Model governor *load reference* (EEL-6266, 2004)

Pada saat terjadi perubahan beban, governor akan menentukan setting point yang baru sesuai dengan aktual beban sehingga dengan pengaturan putaran ini diharapkan frekuensi listrik generator tetap berada di dalam "*acceptable range*" dan generator tidak mengalami "*out of synchronization*". Jenis kontrol governor ini akan selalu pada frekuensi tertentu dimana sistem perubahan dapat dibagi antar generator.

Pada pembangkitan listrik tenaga uap masa sekarang, governor yang digunakan untuk mengatur kecepatan *steam turbine* bukan berupa governor mekanik tapi merupakan governor digital atau berupa mikroprosesor DEHC (*Digital Electro-Hidraulic Controller*) yang didasarkan pada turbin governor. Mikroprosesor ini merupakan dasar pengendalian dengan aplikasi software terintegrasi yang dirancang untuk aplikasi *valve* turbin uap.

2.1.3 Generator

Generator adalah alat untuk membangkitkan listrik, generator terdiri dari *stator* dan *rotor*. *Rotor* dihubungkan dengan shaft turbin sehingga dapat berputar bersama. *Stator bars* di dalam generator membawa arus yang berhubungan dengan output pembangkit. Arus *direct current* (DC) di alirkan melalui *brush gear* yang langsung bersentuhan dengan *slip ring* yang dipasang bersama *rotor* sehingga akan timbul medan magnet (*flux*). Jika rotor berputar, medan magnet akan memotong kumparan di stator sehingga pada ujung-ujung kumparan *stator* akan timbul tegangan listrik. Untuk penyediaan arus listrik generator diambilkan arus DC dari luar. Melalui eksitasi transformer, arus AC akan disearahkan oleh rectifier dan arus DC akan kembali ke generator.

Untuk menjaga kualitas pelayanan tenaga listrik, maka diperlukan pengaturan frekuensi yang bekerja sesuai dengan turun-naiknya beban, sehingga frekuensi sistem dapat dijaga pada harga normalnya. (Aturan Jaringan Sistem Tenaga Listrik Jawa-Madura-Bali, 2007) Dalam pengoperasiannya, frekuensi nominal adalah 50 Hz, diusahakan untuk tidak lebih rendah dari 49,5 Hz atau lebih tinggi dari 50,5 Hz, dan selama waktu keadaan darurat (*emergency*) dan gangguan, frekuensi Sistem diizinkan turun hingga 47.5 Hz atau naik hingga 52.0 Hz sebelum unit pembangkit diizinkan keluar dari operasi. Frekuensi di sistem tidak boleh bekerja lebih rendah daripada frekuensi yang telah ditetapkan. Sedangkan untuk governor yang berpengaruh pada pengatur primer, frekuensi sistem di antara 48,5 Hz hingga 51 Hz.

Perubahan frekuensi yang terlalu besar jika tidak segera ditanggulangi dapat mengakibatkan kerusakan pada peralatan-perlatan listrik yang peka terhadap perubahan frekuensi. Pada generator pembangkit, persamaan frekuensi dapat dilihat pada persamaan :

$$f = \frac{p\,n}{60} \quad \text{.....} \quad (2.1)$$

dengan

f = frekuensi (Hz)
p = jumlah pasang kutub
n = kecepatan (putaran per menit)

Dari persamaaan tersebut frekuensi dapat dirubah kembali dengan merubah jumlah pasang kutub (p) dan kecepatan putar generator (n). Frekuensi berbanding lurus dengan kecepatan putar turbin, jika kecepatan turbin naik maka akan mengakibatkan kenaikan frekuensi. Pada generator, kutub-kutub biasanya sudah terpasang permanen sehingga tidak mungkin dilakukan penamabahan atau pengurangan untuk waktu yang mendadak. Oleh karena itu parameter yang paling mudah dirubah untuk mendapatkan perubahan frekuensi adalah parameter kecepatan putaran turbin-generator.

2.2 Logika Fuzzy

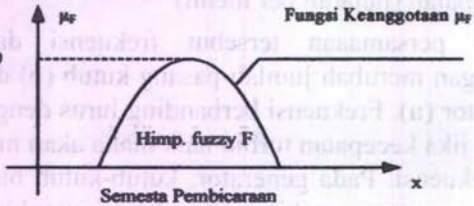
Pada dasarnya, teori himpunan fuzzy merupakan perluasan dari teori himpunan klasik. Pada teori himpunan klasik (*crisp*), keberadaan suatu elemen pada suatu himpunan A, hanya akan memiliki 2 kemungkinan keanggotaan, yaitu anggota A atau tidak menjadi anggota A. Pada tahun 1965 Lotfi A. Zadeh memodifikasi teori himpunan klasik dimana setiap anggotanya memiliki derajat keanggotaan yang bernilai kontinu antara 0 sampai 1. Himpunan ini disebut dengan himpunan kabur (*Fuzzy Set*). Untuk semesta wacana (*Universe of discourse*) U, himpunan fuzzy ditentukan oleh fungsi keanggotaan yang memetakan anggota u ke rentang keanggotaan dalam interval [0,1].

2.2.1 Himpunan Fuzzy

Suatu himpunan fuzzy F dalam suatu semesta pembicaraan U didefinisikan sebagai kumpulan pasangan elemen u dan fungsi keanggotaan $\mu_F(u)$ yang mempunyai nilai dalam interval [0,1] pada tiap u dalam U. (Widodo, 2005) Nilai fungsi keanggotaan menunjukkan tingkat keanggotaan elemen u dalam F, adalah :

$$\mu_F(u) = \begin{cases} 1 & jika\ u \in A \\ 0 & jika\ u \notin A \end{cases} \quad \text{..... (2.2)}$$

Digambarkan sebagai berikut :



Gambar 2.6 Himpunan fuzzy dan fungsi keanggotaannya (Widodo, 2005)

Secara umum himpunan *fuzzy* dinotasikan sebagai:

$$F = \{(u, \mu_F(u)) | u \in U\} \quad \text{..... (2.3)}$$

Bila secara diskrit dengan n elemen F dapat dinyatakan sebagai:

$$F = \mu_1(u)/u_1 + \mu_2(u)/u_2 + ... + \mu_n(u)/u_n$$
$$= \sum_{i=1}^{n} \mu_F(ui)/u_i \quad \text{..... (2.4)}$$

Jika X kontinu, F dapat dinyatakan sebagai:

$$F = \int \mu_F(u)/u \quad \text{..... (2.5)}$$

Tanda "/" : menyatakan hubungan suatu elemen dan nilai keanggotaannya dan bukan pembagian aritmetika.

Tanda "Σ", "+", "$\int$" : menyatakan uni himpunan dan bukan merupakan penjumlahan aritmetika.

2.2.2 Teori Operasi himpunan Fuzzy

Operasi himpunan fuzzy dilakukan dengan manipulasi fungsi keanggotaan. Misalkan A dan B adalah dua himpunan fuzzy bernilai titik di dalam O dengan fungsi keanggotaan μ_A dan μ_B. (Widodo, 2005) Maka operasi himpunan fuzzy berikut didefinisikan:

- Union

  Union dari dua himpunan fuzzy A dan B dengan fungsi keanggotaan $\mu_A(u)$ dan $\mu_B(u)$ adalah himpunan fuzzy C, dengan fungsi keanggotaan μ_C (u) adalah sebagai berikut :

  $$\mu_C(u) = \mu_{A \cup B}(u) = \max\{\mu_A(u), \mu_B(u)\} \quad untuk\ semua\ u \in U$$

  ..... (2.6)

- Interseksi

  Interseksi dari dua himpunan fuzzy A dan B dengan adalah himpunan fuzzy C, dengan fungsi keanggotaan μ_C (u) adalah sebagai berikut :

  $$\mu_C(u) = \mu_{A \cap B}(u) = \min\{\mu_A(u), \mu_B(u)\} \quad untuk\ semua\ u \in U$$

  ..... (2.7)

- Komplemen

  Komplemen dari himpunan fuzzy A, dinotasikan dengan (-A, *NOT* A), dituliskan sebagai :

  $$\mu_{\bar{A}}(u) = 1 - \mu_A(u) \quad untuk\ semua\ u \in \quad .....\quad (2.8)$$

- Produk Kartesian

  Jika A dan B adalah himpunan fuzzy di X dan Y. Produk kartesian (cartesian product) dari A dan B yang dinyatakan dengan C = A x B adalah himpunan fuzzy dalam ruang produk X x Y dengan fungsi keanggotaan sebagai berikut.

  $$\mu_C(x, y) = \min\{\mu_A(x), \mu_B(y)\} \quad ..... \quad (2.9)$$

  dengan A x B dikarakteristikan oleh fungsi keanggotaan 2 dimensi

- Cartesian Co Product

  Cartesian co-product A + B adalah himpunan fuzzy dengan fungsi keanggotaan sebagai berikut.

$$\mu_C(x,y) = \max\{\mu_A(x), \mu_B(y)\} \quad \text{..... (2.10)}$$

  dengan A + B dikarakteristikan oleh fungsi keanggotaan 2 dimensi

2.2.3 Fungsi Keanggotaan Himpunan Fuzzy

Keanggotaan dalam himpunan fuzzy mempunyai bentuk yang berbeda-beda. Pendefinisian suatu anggota dalam fuzzy dapat menggunakan cara numerik maupun bentuk fungsi. Pendefinisian bentuk fungsi digunakan untuk himpunan yang kontinu, sedangkan pendefinisian numerik digunakan untuk himpunan diskrit.

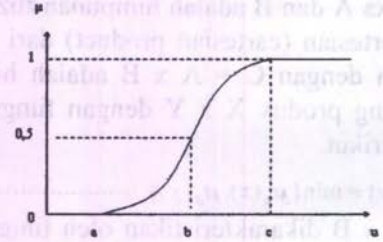
Fungsi keanggotaan yang paling sering digunakan dalam praktik (Widodo, 2005) adalah sebagai berikut :

- Fungsi S

  Fungsi ini berbentuk S dan ditentukan oleh nilai parameter a, b dan c. Fungsi S didefinisikan sebagai berikut.

$$S(u;a,b,c) = \begin{cases} 0 & \rightarrow u < a \\ 2[(u-a)/(c-a)]^2 & \rightarrow a \le u \le b \\ 1-2[(u-c)/(c-a)]^2 & \rightarrow b \le u \le c \\ 1 & \rightarrow u > c \end{cases} \quad \text{..... (2.11)}$$

  Titik persilangan 0,5 terjadi pada $b = \frac{(a+c)}{2}$

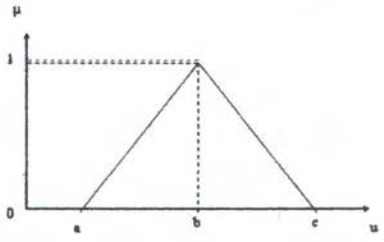


Gambar 2.7 Fungsi keanggotaan S (Widodo, 2005)

- Fungsi Segitiga (Triangular)

  Fungsi ini berbentuk segitiga dengan parameter a, b dan c. Fungsi segitiga didefinisikan sebagai berikut :

$$T(u;a,b,c) = \begin{cases} 0 & \rightarrow u < a \\ (u-a)/(b-a) & \rightarrow a \le u \le b \\ (c-u)/(c-b) & \rightarrow b \le u \le c \\ 1 & \rightarrow u > c \end{cases} \quad \text{.....} \quad (2.12)$$

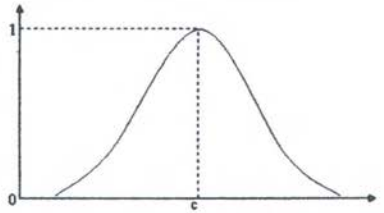


Gambar 2.8 Fungsi keanggotaan segitiga (Widodo, 2005)

- Fungsi Gaussian

  Fungsi ini berguna untuk membuat fungsi keanggotaan dengan kurva *Gaussian.* Ada 2 parameter yang dapat digunakan, yaitu [c a]. Definisi fungsi gaussian adalah sebagai berikut :

$$f(x:c,a) = e^{-\frac{1}{2}\left(\frac{x-c}{a}\right)^2} \quad \text{.....} \quad (2.13)$$

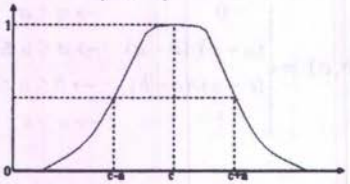


Gambar 2.9 Fungsi keanggotaan gaussian (Widodo, 2005)

- Fungsi Bell Gaussian

  Fungsi ini berbentuk *bell* dan mempunyai tiga parameter yaitu a, b dan c. Definisi fungsi *bell Gaussian* adalah sebagai berikut

$$f(x:a,b,c) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \quad \text{.....} \quad (2.14)$$



Gambar 2.10 Fungsi keanggotaan *bell gaussian* (Widodo, 2005)

2.3 Struktur Dasar Sistem Inferensi Logika Fuzzy

Dalam Sistem Inferensi *Logika Fuzzy* ini sinyal masukan dari plant dikuantisasikan ke dalam beberapa tingkat nilai seperti *big, medium, small* dan *zero*. Secara umum sistem inferensi fuzzy mempunyai empat bagian yaitu : fuzzifikasi, basis pengetahuan, logika pengambil keputusan dan defuzzifikasi.

Gambar 2.11 Struktur Dasar Sistem Inferensi Fuzzy (Widodo, 2005)

(Wang,1994) Keempat bagian tersebut mempunyai fungsi sebagai berikut :

A. Fuzzifikasi

Fuzzifikasi berfungsi untuk mentransformasikan sinyal masukan yang bersifat *crisp* (bukan fuzzy) ke

himpunan fuzzy dengan menggunakan operator fuzzifikasi. Proses ini dinyatakan sebagai :

$$X = Fuzzifier(X_0) \quad \text{..... (2.15)}$$

dengan

X_o = Vektor dari nilai crisp untuk satu variabel input dari proses.

X = Vektor dari himpunan fuzzy terdefinisi untuk variabel input.

B. Basis Pengetahuan

Basis Pengetahuan berisi basis data dan aturan dasar yang mendefinisikan himpunan fuzzy atas daerah – daerah masukan dan keluaran dan menyusunnya dalam perangkat aturan kontrol.

C. Logika Pengambil Keputusan

Logika Pengambil Keputusan merupakan inti dari Inferensi *Logika Fuzzy* yang mempunyai kemampuan seperti manusia dalam mengambil keputusan. Aksi atur fuzzy disimpulkan dengan menggunakan implikasi fuzzy dan mekanisme inferensi fuzzy.

D. Defuzzifikasi

Defuzzifikasi berfungsi untuk mentransformasikan kesimpulan tentang aksi atur yang bersifat fuzzy menjadi sinyal sebenarnya yang bersifat crisp dengan menggunakan operator defuzzifikasi Bentuk umum proses defuzzifikasi dinyatakan dengan:

$$Z_0 = defuzzier(z) \quad \text{..... (2.16)}$$

dimana

z = Aksi pengendalian fuzzy.

Z_0 = Aksi pengendali *crisp*.

Defuzzifier adalah operator defuzzifikasi. Ada beberapa metode defuzzifikasi yang biasa digunakan dalam proses defuzzifikasi, yaitu; metode titik pusat

(*centre of area*), metode kriteria maximum (*criterion max*), dan metode titik tengah maximum (*mean of max*).

a. Metode Titik Pusat (COA)

Metode COA ini adalah metode yang paling sering digunakan. Strategi ini dibangkitkan dari pusat berat (*centre of gravity*) pada aksi kontrol. Dalam kasus diskrit aksi kontrol ditunjukkan sebagai:

$$z_0 = \frac{\sum_{i=1}^{n} \mu_Z(Wi) \bullet Wi}{\sum_{i=1}^{n} \mu_Z(Wi)} \quad \text{..... (2.17)}$$

dimana n adalah banyaknya level kuantisasi dari output.

b. Metode Kriteria Maximum (MAX)

Metode kriteria maximum menghasilkan titik dimana distribusi yang mungkin pada aksi kontrol bernilai maximum.

c. Metode Titik Tengah Maximum (MOM)

Metode MOM merepresentasikan nilai titik tengah dari keluaran yang fungsi anggotanya maximum. Fungsinya ditunjukkan sebagai:

$$z_0 = \sum_{i=1}^{n} \frac{z_i}{l} \quad \text{..... (2.18)}$$

dimana z_i adalah nilai pendukung dengan fungsi keanggotaan bernilai maximum dan l adalah banyaknya nilai pendukung.

Perancangan pengendali logika fuzzy menggabungkan aspek pendefinisian himpunan fuzzy dengan aspek logika fuzzy yang keduanya diterapkan pada masukan dan keluaran untuk memperoleh hasil perancangan yang berbentuk suatu algoritma aturan fuzzy. Pengendali fuzzy menggunakan aturan-aturan linguistik pertama kali diaplikasikan pada tahun 1974 oleh

Mamdani, yang menggunakan metode teoritis Zadeh. Aturan diberikan dalam bentuk:

R_i = "jika x adalah A_i maka y adalah B_i"

dengan $x = (x_1,x_2,x_3,\ldots,x_n)$ adalah variabel keadaan input; y adalah bentuk skalar yang menggambarkan variabel output; dan A_i dan B_i adalah terminologi linguistik yang ditunjukan oleh himpunan fuzzy.

Cara kerja pengendali ini mirip dengan seorang operator kendali, tidak memperhatikan struktur internal *plant*, hanya mengamati error sebagai selisih antara *setpoint* dengan keluaran sistem dan mengubah *setting* panel kendali untuk meminimalkan error tersebut.

2.4 Adaptive Neuro-Fuzzy Inference System (ANFIS)

Adaptive Neuro Fuzzy Inference System (ANFIS) adalah sistem inferensi fuzzy berbasis jaringan saraf adaptif atau sistem inferensi fuzzy berbasis jaringan adaptif. Arsitektur yang secara fungsional sama dengan fuzzy rule base model Sugeno. Arsitektur ANFIS juga sama dengan jaringan syaraf dengan fungsi radial dengan sedikit batasan tertentu. Bisa dikatakan bahwa ANFIS adalah suatu metode yang mana dalam melakukan penyetelan aturan digunakan algoritma pembelajaran terhadap sekumpulan data. Pada ANFIS juga memungkinkan aturan-aturan untuk beradaptasi.

2.4.1 Arsitektur ANFIS

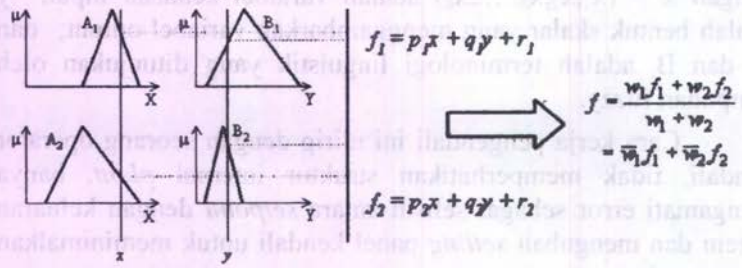
Untuk penyederhanaan, misalkan fuzzy infernsi sistem memiliki ada 2 (dua) input x,y dan 1 (satu) output f. (Widodo, 2005) Untuk model fuzzy sugeno orde pertama, aturan umum dengan 2 aturan, fuzzy Jika-Maka (If-Then) adalah sebagai berikut

If x is A_1 and y is B_1 Then $f_1 = p_1x + q_1y + r_1$

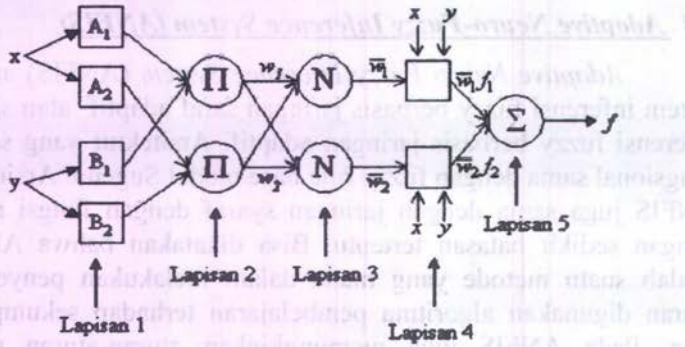
If x is A_2 and y is B_2 Then $f_2 = p_2x + q_2y + r_2$..... (2.19)

Jika w predikat untuk aturan kedua aturan adalah w_1 dan w_2, maka dapat dihitung rata-rata terbobot :

$$f = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2} = \overline{w}_1 f_1 + \overline{w}_2 f_2 \quad \text{.....} \quad (2.20)$$



Gambar 2.12 Sugeno-type Fuzzy Inference System (Adi, 2000)



Gambar 2.13 Arsitektur ANFIS (Adi, 2000)

Gambar 2.12 diatas mengilustrasikan mekanisme penalaran untuk model sugeno, ekivalensi ANFIS pada terlihat gambar 2.13, dimana sampel-sampel pada lapisan yang sama mempunyai fungsi yang sama. Keluaran dari simpul dari jaringan ANFIS yang ditunjukan pada masing-masing lapisan sebagai berikut :

A. Lapisan pertama

Pada lapisan ini merupakan fungsi fuzzyfikasi. Tiap-tiap neuron i pada lapisan pertama adaptif terhadap

parameter suatu fungsi aktivasi. Keluaran dari tiap titik pada lapis ini berupa fungsi keanggotaan dari sinyal masukan. Output dari tiap neuron berupa derajat keanggotaan yang diberikan oleh fungsi keanggotaan input, yaitu : $\mu_{A1}[\mu_1]$, $\mu_{B1}[\mu_2]$, $\mu_{A2}[\mu_1]$ atau $\mu_{B2}[\mu_2]$. Sebagai contoh, misalkan fungsi keanggotaan diberikan sebagai :

$$\mu_F[u] = \frac{1}{1+\left|\frac{u-c}{a}\right|^{2b}} \quad \text{..... (2.21)}$$

Dimana {a,b,c} adalah parameter-parameter. Jika nilai parameter-parameter ini berubah, maka bentuk kurva yang terjadi akan ikut berubah. Parameter-parameter pada lapisan itu biasanya dikenal dengan nama *premise parameters.*

B. Lapisan kedua

Merupakan kaidah operasi perkalian sebagai implikasi dari Fuzzy. Tiap-tiap node pada lapisan kedua berupa node tetap yang outputnya adalah hasil dari masukan. Biasanya digunakan operator AND. Tiap-tiap node merepresentasikan α predikat dari aturan ke-i.

$$wi = \mu_{Ai}(x)\mu_{Bi}(y), \quad i = 1,2. \text{..... (2.22)}$$

C. Lapisan ketiga

Tiap-tiap node pada lapisan ketiga berupa node tetap yang merupakan hasil perhitungan rasio dari w predikat dari aturan ke-i terhadap jumlah dari keseluruhan w predikat.Keluaran lapisan ini merupakan perbandingan antara satu titik dengan jumlah titik yang ada sebelumnya.

$$\overline{w}_i = \frac{wi}{w_1 + w_2}, \textit{ dengan } i = 1,2 \text{ ..... (2.23)}$$

Hasil ini dikenal dengan nama *normalised firing strength* (kuat penyuluhan ternormalisasi).

D. Lapisan keempat

Tiap-tiap node pada lapisan keempat merupakan node adaptif terhadap suatu output. Dimana dilakukan pembobotan atas hasil dari lapis sebelumnya.

$$\overline{w}_i f_i = \overline{w}_i (p_i x + q_i y + r_i), \text{ dengan } i = 1,2 \quad (2.24)$$

dengan

w_i = kuat penyuluhan ternormalisasi dari lapis ketiga

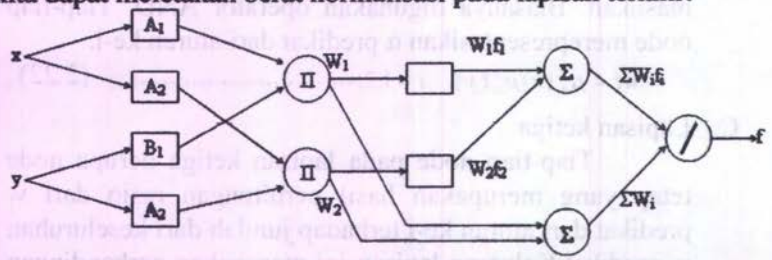
$\{p_i, q_i, r_i\}$ = himpunan parameter dari node ini. Parameter pada lapis ini disebut pameter konsekuensi

E. Lapisan kelima

Tiap-tiap node pada lapisan kelima adalah node tetap yang merupakan jumlahan dari semua masukan. Pada lapisan ini dilakukan defuzzyfikasi nilai yang telah diproses pada lapis sebelumnya.

$$\sum_i \overline{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (2.25)$$

Sejauh ini telah dibangun jaringan adaptif yang secara fungsional sama dengan model fuzzy sugeno. Sebagai catatan struktur dari jaringan adaptif ini tidaklah unik. Dalam jaringan ini kita dapat meletakan berat normalisasi pada lapisan terkhir.



Gambar 2.14 Arsitektur ANFIS model Fuzzy Sugeno (Jang, 1994)

2.4.2 Algoritma Pembelajaran *Hybrid*

Pada ANFIS, parameter adalah fungsi keanggotaan premis dan konsekuensi. (Adi, 2000) Pembelajaran ANFIS adalah pengubahan parameter fungsi keanggotaan masukan dan

keluaran. Pembelajaran ANFIS dapat menggunakan algoritma perambatan balik atau algoritma *hybrid*. Algoritma *hybrid* adalah gabungan antara algoritma perambatan balik dan metode kuadrat terkecil (*Least Squares Estimate*). Metode kuadrat terkecil digunakan untuk menentukan parameter konsekuensi, sedangkan perambatan balik digunakan untuk memperbaharui bobot premis.

Tabel 2.1 Proses Pembelajaran *Hybrid* ANFIS

	Langkah maju	**Langkah mundur**
Parameter Premis	Tetap	Gradient Descent
Parameter konsekuen	RLSE	Tetap
Sinyal	Keluaran Simpul	Laju kesalahan

ANFIS dilatih dengan algoritma pembelajaran *hybrid*. Algoritma pelatihan *hybrid* terdiri dari dua langkah, yaitu langkah maju dan langkah balik.

A. **Langkah Maju**

Pada langkah maju, input jaringan akan merambat maju sampai pada lapisan keempat, parameter premis tetap, sedangkan parameter konsekuensi diidentifikasi dengan metode LSE. Dari arsitektur ANFIS yang terlihat pada gambar 2.14, dengan parameter premis tetap (maka w_i juga tetap), maka keluaran keseluruhan dapat dinyatakan dengan kombinasi linier dari parameter konsekuensi. Input jaringan akan merambat maju sampai pada lapisan keempat, dimana parameter-parameter p, q dan r akan diidentifikasi

$$f = \frac{w_1}{w_1 + w_2} f_1 + \frac{w_2}{w_1 + w_2} f_2$$

$$= \overline{w}_1 (p_1 x + q_1 y + r_1) \overline{w}_2 (p_2 x + q_2 y + r_2) \; .. \qquad (2.26)$$

Yang linier dalam parameter p_1, q_1, r_1, p_2, q_2, dan r_2

$$f_{(1)} = (\overline{w}_1 x_1)p_1 + (\overline{w}_1 y_1)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_1)p_2 + (\overline{w}_2 y_1)p_2 + (\overline{w}_1)r_2$$
$$f_{(2)} = (\overline{w}_1 x_2)p_1 + (\overline{w}_1 y_2)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_2)p_2 + (\overline{w}_2 y_2)p_2 + (\overline{w}_1)r_2$$
.....
$$f_{(p)} = (\overline{w}_1 x_p)p_1 + (\overline{w}_1 y_p)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_p)p_2 + (\overline{w}_2 y_p)p_2 + (\overline{w}_1)r_1$$
..... (2.27)

Untuk pasangan P data pelatihan masukan : x_1, y_1, $x_2, y_2, \ldots, x_p, y_p$ dan keluaran : $f_{(1)}$, $f_{(2)}, \ldots f_{(p)}$, maka diperoleh persamaan linear simultan berikut :
dalam notasi matriks persamaan tersebut dapat ditulis

$$F = A\theta \quad \ldots\ldots (2.28)$$

Dengan F adalah vektor keluaran Px1 :

$$F = \begin{bmatrix} f_{(1)} \\ f_{(2)} \\ . \\ f_{(p)} \end{bmatrix} \quad \ldots\ldots (2.29)$$

A adalah matriks berdimensi P x 6 yang dalam LSE disebut matriks desain :

$$f_{(1)} = (\overline{w}_1 x_1)p_1 + (\overline{w}_1 y_1)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_1)p_2 + (\overline{w}_2 y_1)p_2 + (\overline{w}_1)r_2$$
$$f_{(2)} = (\overline{w}_1 x_2)p_1 + (\overline{w}_1 y_2)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_2)p_2 + (\overline{w}_2 y_2)p_2 + (\overline{w}_1)r_2$$
.....
$$f_{(p)} = (\overline{w}_1 x_p)p_1 + (\overline{w}_1 y_p)q_1 + (\overline{w}_1)r_1 + (\overline{w}_2 x_p)p_2 + (\overline{w}_2 y_p)p_2 + (\overline{w}_1)r_1$$
..... (2.30)

$$A = \begin{bmatrix} \overline{w}_1 x_1 & \overline{w}_1 y_1 & \overline{w}_1 & \overline{w}_2 x_1 & \overline{w}_2 y_1 & \overline{w}_2 \\ \overline{w}_1 x_2 & \overline{w}_1 y_1 & \overline{w}_1 & \overline{w}_1 x_2 & \overline{w}_1 y_2 & \overline{w}_1 \\ . & . & . & . & . & . \\ \overline{w}_1 x_p & \overline{w}_1 y_p & \overline{w}_1 & \overline{w}_p x_p & \overline{w}_1 y_p & \overline{w}_2 \end{bmatrix} \quad \ldots (2.31)$$

Kemudian dicari q yang meminimkan jumlah galat kuadrat yaitu dengan persamaan :

$$\overline{\theta} = (A^T A)^{-1} A^T \quad \ldots\ldots (2.32)$$

B. **Langkah Balik**

Pada langkah balik sinyal galat (error) antara keluaran yang diinginkan dan keluaran aktual dirambatkan mundur sedangkan parameter premis diperbaharui dengan metode penurunan gradien. Pada umumnya menggunakan penurunan gradient.

Dari arsitektur ANFIS yang terlihat pada gambar 2.14 dimisalkan bahwa himpunan data pelatihan mempunyai P pasangan data (*input-output*) yang diinginkan. Ukuran error untuk masukan ke p ($1 \leq p \leq P$) adalah jumlah error kuadrat dari semua node keluaran.

$$E_p = \sum_{k=1}^{P} \frac{1}{2}(y_k{}^r - y_k)^2 \quad \text{.....} \quad (2.33)$$

dengan

y^r = referensi target output sistem

y = output dari sistem fuzzy

karena untuk kasus ANFIS diatas hanya terdapat satu node keluaran maka

$$E_p = \frac{1}{2}(y_k{}^r - y_k)^2 \quad \text{.....} \quad (2.34)$$

Yang menjadi tujuan adalah minimisasi seluruh error untuk seluruh P pasangan data (*input-output*) pelatihan yang didefinisikan sebagai

$$E = \sum_{k=1}^{P} E_p \quad \text{.....} \quad (2.35)$$

Dengan menggunakan metode penurunan gradient Descent sederhana tanpa minimisasi garis, maka formula pembaruan parameter generik α (misalnya parameter premis : a, b, c) adalah sebagai berikut :

$$\Delta\alpha = -\eta \frac{\partial^+ E}{\partial \alpha} \quad \text{.....} \quad (2.36)$$

Dengan derivatif beruntun (*ordered derivative*) terhadap α dari keseluruhan galat E yang didefiniskan oleh :

$$\frac{\partial^+ E}{\partial \alpha} = \sum_{p=1}^{P} \frac{\partial^+ E_p}{\partial \alpha} \quad \text{..... (2.37)}$$

Yang merupakan vektor gradien (derivatif orde satu). Sedangkan η adalah laju pelatihan (digunakan konstanta yang kecil untuk metode tanpa minimisasi garis). Untuk menghitung vektor gradien, bila parameter α mempengaruhi beberapa node digunakan formula sebagai berikut :

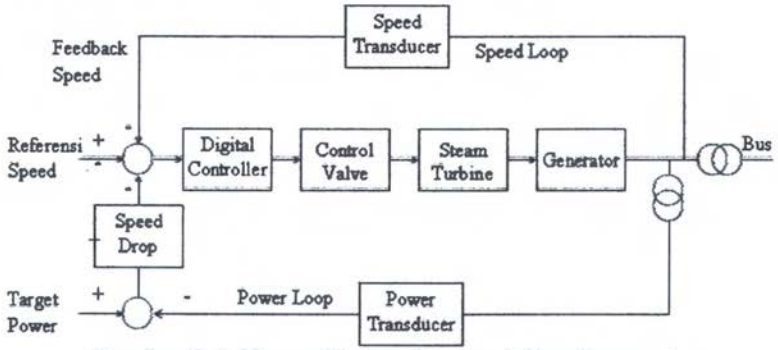
$$\frac{\partial^+ E}{\partial \alpha} = \sum_{O^* \in S}^{P} \frac{\partial^+ E_p}{\partial O^*} \frac{\partial F^*}{\partial \alpha} \quad \text{..... (2.38)}$$

dengan S adalah himpunan node yang dipengaruhi oleh parameter α. Sedangkan O* dan F* adalah keluaran dan fungsi dari node didalam S.

BAB III

PERANCANGAN DAN METODOLOGI

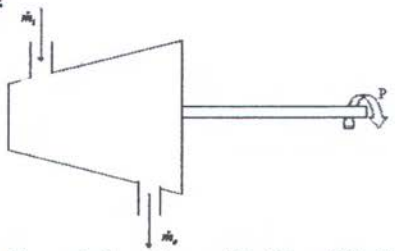
Pada bab ini akan dibahas sistem pengendalian kecepatan pada *steam turbine*, berupa pemodelan matematis dari komponen sistem pengendalian kecepatan *steam turbine*, perhitungan model matematis proses dan perancangan sistem pengendalian ANFIS pada kecepatan *steam turbine*. Secara umum sistem pengendalian kecepatan *steam turbine* adalah sebagai berikut :



Gambar 3.1 Skema Sistem Pengendalian Kecepatan *Steam Turbine* (Tripathy, 1997)

3.1 Pemodelan Matematis

3.1.1 Turbin



Gambar 3.2 Pemodelan *Steam Turbine* (Huddleston, 2006)

Persamaan kesetimbangan massa adalah :

$$\dot{m}_i - \dot{m}_o = \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.1)$$

Persamaan kesetimbangan kalor adalah :

$$\dot{q} - \dot{W} = \dot{m}_o\left(h_o + \frac{v_o^2}{2} + gz_o\right) - \dot{m}_i\left(h_i + \frac{v_i^2}{2} + gz_i\right) \quad \text{..... (3.2)}$$

dengan

$\dot{q}$ = Panas dari lingkungan (kJ/dt)

$\dot{W}$ = Kerja Poros (kJ/dt)

$\dot{m}_i$ dan $\dot{m}_o$ = Laju aliran uap masuk dan keluar (Kg/dt)

h_i dan h_o = *Enthalpi* masuk dan keluar (kJ/Kg)

$\frac{v_i^2}{2}$ dan $\frac{v_o^2}{2}$ = Energi kinetik masuk dan keluar (kJ/dt)

gz_i dan gz_o = Energi Potensial masuk dan keluar (kJ/dt)

Diasumsikan Turbin merupakan proses aliran uap pada kecepatan tinggi sehingga bersifat Reversibel, merupakan proses Adiabatik dan mengabaikan perubahan energi potensial dan energi Kinetik, sehingga persamaan (3.2) diatas menjadi :

$$\cancel{\dot{Q}} - \dot{W} = \dot{m}_2\left(h_2 + \cancel{\frac{v_2^2}{2}} + \cancel{gz_2}\right) - \dot{m}_1\left(h_1 + \cancel{\frac{v_1^2}{2}} + \cancel{gz_1}\right) \quad \text{..... (3.3)}$$

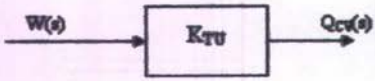
$$\dot{W} = \dot{m}_1 h_1 - \dot{m}_o h_o$$

$$= \dot{m}(h_1 - h_o) \quad \text{..... (3.4)}$$

Persamaan ditransformasikan Laplace, sehingga menjadi :

$$W(s) = Q(s)(h_1 - h_o) \quad \text{..... (3.5)}$$

$$\frac{W(s)}{Q(s)} = (h_1 - h_o) = Ktu \quad \text{..... (3.6)}$$



Gambar 3.3 Fungsi *Steam Turbine*

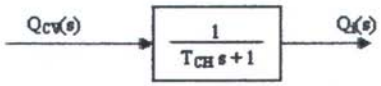
3.1.2 Steam Chest

Steam Chest merupakan ruang penyimpanan uap yang manghubungkan kontrol valve dan stage pertama pada Hp turbin. Digambarkan pada persamaan orde satu yaitu sebagai berikut :

PERPUSTAKAAN ITS

$$Q_{CV}(s) = \frac{1}{\frac{V\rho_o}{wi}s+1} Q_i(s)$$

$$= \frac{1}{T_{CH}s+1} Q_i(s) \quad \text{..... (3.7)}$$



Gambar 3.4 Fungsi Transfer *Steam Chest*

dengan

Q_{CV} = Laju aliran uap control valve (Kg/dt)
Q_i = Laju aliran uap HP turbin (Kg/dt)
T_{CH} = Time Konstan *Steam Chest* (s)

3.1.3 Generator

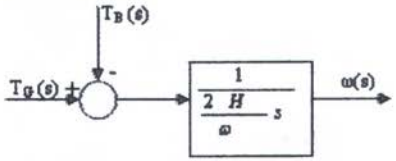
(Kundur, 1994) Menurut persamaan Newton tentang perputaran massa ada hubungannya antara kopel mekanis penggerak generator dengan perputaran generator, yaitu :

$$T_G - T_B = \frac{2H}{\omega_o} \cdot \frac{d\omega}{dt}$$

$$T_G - T_B = M \cdot \frac{d\omega}{dt} \quad \text{..... (3.8)}$$

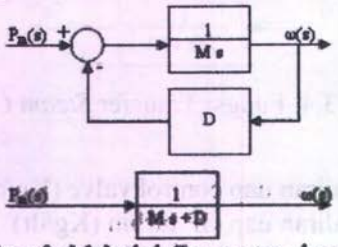
T_G = Kopel penggerak generator (N m)
T_B = Kopel beban yang membebani generator (N m)
ω = Kecepatan sudut perputaran generator (rad/s)
H = Momen inersia dari generator beserta mesin penggeraknya (Kg m^2)

Digambarkan sebagai berikut :



Gambar 3.5 Model Generator

Dengan mengasumsikan kopel penggerak generator sebanding dengan daya mekanik turbin, sedangkan untuk kopel beban adalah dianggap tidak ada. Dalam Generator juga mempunyai faktor peredaman yang yang berpengaruh terhadap perputaran Generator yang dianggap sebagai beban. Sehingga model matematis generator secara lengkap adalah sebagai berikut:



Gambar 3.6 Model Generator dan Beban

dengan

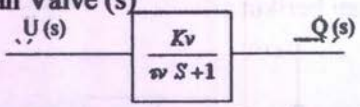
Pm (s) = Daya mekanik Turbin (w)
ω (s) = Kecepatan turbin-generator (pu)
M = Konstanta Inersia (detik)
D = Koefisien peredaman

3.1.4 Control Valve

Diasumsikan mempunyai fungsi transfer orde 1 dengan waktu lag (a first order lag)
Fungsi transfernya adalah :

$$G_v = \frac{Kv}{\tau v S + 1} \quad \text{.....} \quad (3.9)$$

dengan :
Kv = Penguatan Valve
τ_v = Time Konstan Valve (s)

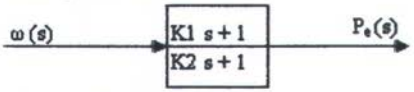


Gambar 3.7 Fungsi Transfer *Control Valve*

3.1.5 Power Transduser

Power Loop dari gambar 3.1 terdiri dari power transduser yang terhubung dengan bus pada output speed. (Tripathy, 1995)

Untuk menyederhanakan analisis, kecepatan turbin-generator merupakan *feedback* untuk daya elektrik dari power transduser. Dengan mengasumsikan bahwa sinyal keluaran kontrol valve sebanding dengan daya yang dihasilkan dari generator. Dengan fungsi transfer sebagai berikut :



Gambar 3.8 Fungsi Transfer Power Transduser

dengan :

K_1 dan K_2 = Time Konstan Power transduser (s)

3.1.6 Speed Drop

Speed Drop merupakan komponen pengatur proporsional untuk mengurangi kesalahan frekuensi dan kecepatan turbin yang terjadi selama kondisi operasi dari sistem pengendalian steam turbin. Speed drop dimodelkan sebagai berikut :

$$(\omega - \omega_o).\frac{1}{Kp} + (P_e - P_o) = 0 \quad \text{..... (3.10)}$$

$$Kp = -\frac{\omega - \omega_o}{P_e - P_o} = \frac{\Delta\omega_p}{P_o - P_e} \text{ pu/Mw} \quad \text{..... (3.11)}$$

dengan :

Kp = Speed Drop (pu/Mw)

ω_o = Kecepatan turbin pada waktu *Steady* (pu)

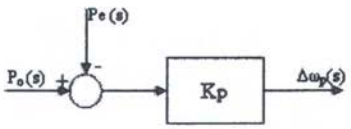
ω = Kecepatan turbin pada waktu terbebani daya listrik (pu)

P_e = Daya Listrik yang dihasilkan power transducer (Mw)

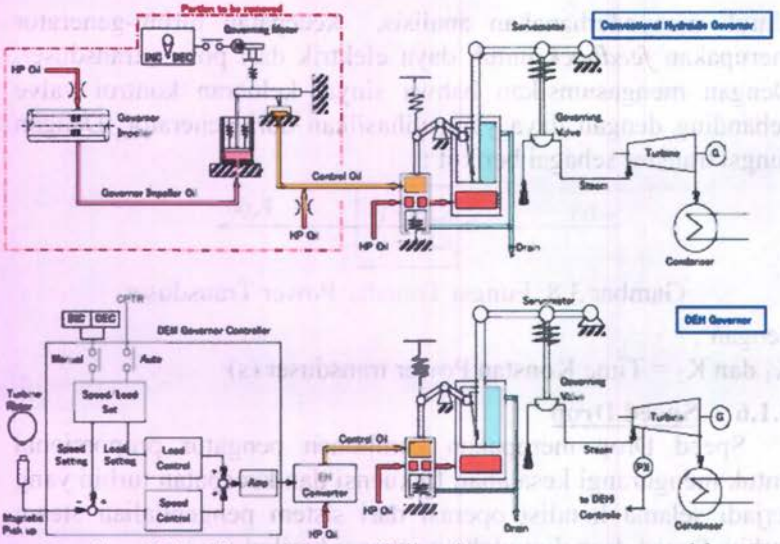
P_o = Daya Listrik yang diinginkan (Mw)

Dalam sistem ini daya listrik yang diinginkan diasumsikan sebagai daya yang membebani sistem.

Dengan fungsi Transfer sebagai berikut

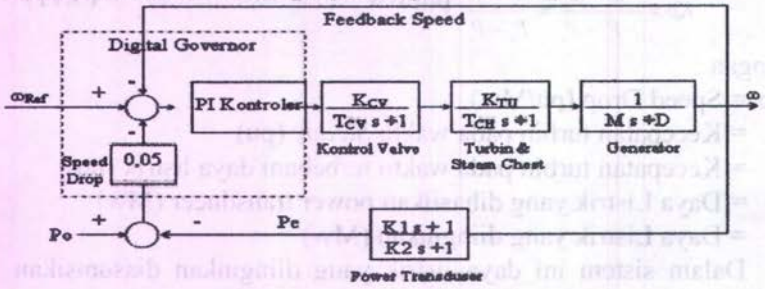


Gambar 3.9 Fungsi Transfer Speed Drop



Gambar 3.10 Sistem Pengendalian kecepatan steam turbine (www.mhi.co.jp)

Desain sistem secara lengkap ditunjukan oleh gambar



Gambar 3.11 Diagram Blok Sistem Pengendalian kecepatan steam turbine

3.1.7 Perhitungan Model Matematis Proses

Kondisi Steam pada steam chest yang masuk ke Turbin

- Laju aliran masuk = 400.000 Kg/jam =111,11 Kg/s
- Tekanan = 88 kg/cm^2 = 8,8 Mpa
- Temperatur = 510 ° C

- Volume penyimpanan uap, V = 3,84 m^3

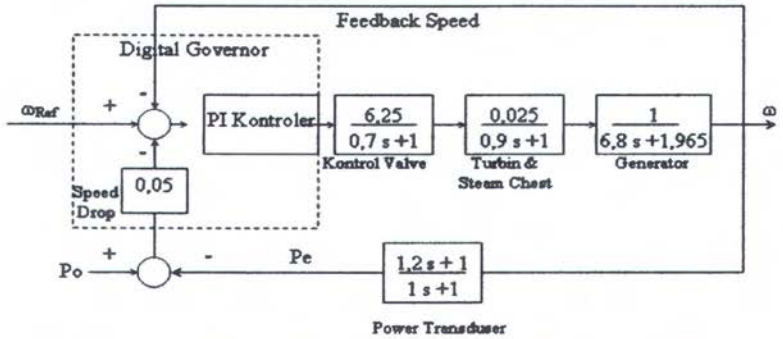
Dari table propertis steam diperoleh hasil :

v = 0,0384 m^3/Kg
h_i = 3704 kJ/Kg
h_o =3679 kJ/Kg
Ktu = 3704 -3679 = 25 kJ/Kg =0,025 MJ/Kg

$$\rho = \frac{1}{v} = \frac{1}{0.0384} Kg/m^3 = 26{,}04 Kg/m^3 \quad \text{.....} \quad (3.12)$$

$$T_{Tu} = \frac{V\rho_o}{w_i} = \frac{3{,}84\, m^3 . 26{,}04\, Kg/m^3}{111{,}11 Kg/s} = 0{,}9\, s \quad \text{.....} \quad (3.13)$$

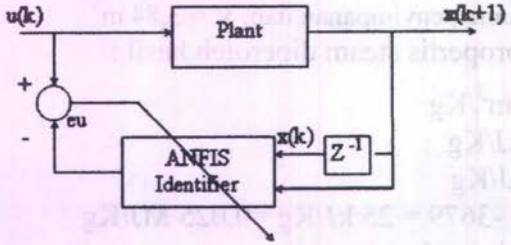
Dengan memasukkan data-data yang diperoleh dari plan, maka fungsi transfer kecepatan diatas menjadi :



Gambar 3.12 Fungsi Transfer Pengendali Kecepatan Steam turbine

3.2 Perancangan Kontroler ANFIS

Pemilihan nilai parameter merupakan awal penting dalam menetukan hasil identifikasi model yang akan diidentifikasi sebelum diimplementasikan, maka perlu dilakukan proses identifikasi awal dari plant yang akan dikendalikan yaitu turbin uap. Sehingga ketika diimplementasikan output identifier, dalam hal ini ANFIS identifier sudah mendekati bidang kendali yang sebenarnya.



Gambar 3.13 Skema dasar identifikasi sistem (Jang, 1994)

Error antara sinyal target yang merupakan sinyal kontroler PI dan *output* (e_u) akan digunakan dalam pembelajaran untuk mendapatkan bobot yang sesuai agar *output* sama dengan data target yang diinginkan. Dalam pemodelan *invers*, error antara data target (sinyal kontroler PI) dengan data *output* yang digunakan untuk pembelajaran *neuro-fuzzy* didapatkan dengan menggunakan RMSE (*Root Mean Square Error*) (Hímer, 2005)

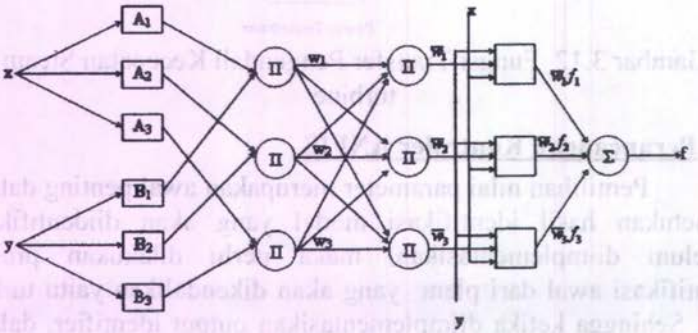
$$RMSE = \sqrt{\frac{1}{K}\sum_{k=1}^{k}(y_{out} - y_{ref})^2} \quad \text{..... (3.14)}$$

dengan

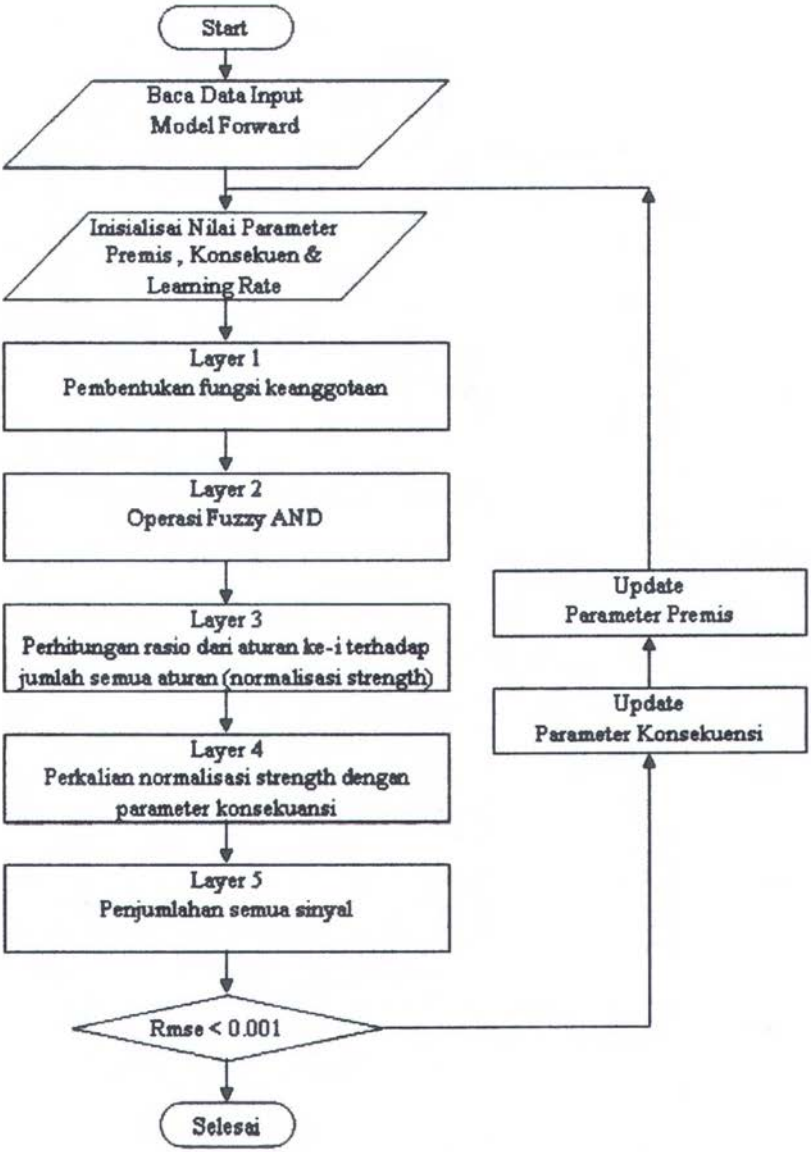
K = banyaknya data yang dilatihkan

y_{out} = data output hasil pembelajaran

y_{ref} = data target yang harus dicapai



Gambar 3.14 Skema ANFIS 2 masukan dengan 3 fungsi keanggotaan (Jang, 1994)



Gambar 3.15 Flowchart ANFIS (Jang, 1994)

Aturan yang digunakan dalam ANFIS ini adalah sebagai berikut :

Ri : IF x_i is A_i and y_i is B_i Then $f_i=p_ix + q_iy + r_i$ (3.15)

Maka persamaan output dari fuzzy dapat dicapai melalui persamaan

$$\mu_1 = \prod_{j=1}^{m} A_{ij}(x_j) \quad \text{.....} \quad (3.16)$$

dan

$$y = \frac{\sum_{i=1}^{n} \mu_i w_i}{\sum_{i=1}^{n} \mu_i} \quad \text{.....} \quad (3.17)$$

- **Menentukan parameter Konsekuan dengan Metode *Least Square Estimator***

Misalkan jaringan adaptif yang ditinjau hanya mempunyai satu keluaran y, maka keluaran ini dapat dinyatakan sebagai berikut. (Widodo, 2005)

$$y = F(U,S) \quad \text{.....} \quad (3.18)$$

dengan

U = vektor variable masukan
S = himpunan parameter
F = fungsi keseluruhan yang diimplementasikan oleh jaringan adaptif.

Bila ada fungsi H sedemikian sehingga fungsi komposisi $H \circ F$ adalah linier dalam beberapa elemen S, maka elemen-elemen ini dapat diidentifikasi dengan metode LSE.
Bila himpunan parameter S dapat dibagi menjadi dua himpunan sebagai berikut.

$$S = S_1 \oplus S_2 \quad \text{.....} \quad (3.19)$$

(dengan $\oplus$: penjumlahan langsung) sedemikian sehingga $H \circ F$ adalah linier dalam beberapa elemen S_2, maka dengan menerapkan H ke persamaan (3.18) diperoleh :

$$H(y) = H \circ F(U,S) = H \circ F(U,S_1 \oplus S_2) \quad \text{.....} \quad (3.20)$$

Yang linier dalam elemen S_2

Bila elemen S_1 tetap sedangkan H(y) dan $F(S_2)$ adalah fungsi identitas, maka :

$$y = \theta_1 f_1(U) + \theta_2 f_2(U) + ... + \theta n(U) \quad \text{.....} \quad (3.21)$$

dengan :

$\theta_1, \theta_2, ... \theta_n$: elemen parameter S_2

$U = (u_1, u_2, ... u_n)^T$: vektor masukan model

$f_1, f_2, ... f_n$: fungsi u yang diketahui

Untuk identifikasi parameter θ_1, maka dilakukan eksperimen untuk memperoleh himpunan data pelatihan yang tersusun atas pasangan data

$$\{(ui; y1), i = 1, ..., m\} \quad dengan \quad m \geq n \quad \text{.....} \quad (3.22)$$

yang menyatakan pasangan masukan-keluaran dari sistem target yang dimodelkan. Substitusi setiap pasangan data ke persamaan (3.21) sehingga diperoleh :

$$\begin{aligned} & f_1(u_1)\theta_1 + f_2(u_1)\theta_2 + ... + f\,n(u_1)\theta n = y_1 \\ & f_2(u_2)\theta_1 + f_2(u2)\theta_2 + ... + f\,n(u_2)\theta n = y_2 \quad \text{.....} \quad (3.23) \\ & \text{.....} \\ & f_1(u_m)\theta_1 + f_2(u_m)\theta_2 + ... + f\,n(u_m)\theta n = y_m \end{aligned}$$

Dengan notasi matriks dapat ditulis :

$$A\theta = y \quad \text{.....} \quad (3.24)$$

dengan

$$A = \begin{bmatrix} f_1(u_1) & ... & fn(u_1) \\ ... & ... & ... \\ f_1(um) & ... & f_n(u_m) \end{bmatrix} \quad \text{.....} \quad (3.25)$$

$$\theta = \begin{bmatrix} \theta_1 \\ ... \\ ... \\ ... \\ \theta n \end{bmatrix} \quad \text{.....} \quad (3.26)$$

$$y = \begin{bmatrix} y_1 \\ \cdots \\ \cdots \\ \cdots \\ y_m \end{bmatrix} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.27)$$

dengan

A = matriks m x n

θ = vektor parameter n x 1 yang tidak diketahui

y = vektor keluaran m x 1

Baris ke-1 dari matriks data {A : y} yang dinotasikan dengan (a_i^T ;y_i) adalah terelasi dengan pasangan data (u_1;y_1) dengan :

$$a_i^T = \{f_1(u_1), \ldots\ldots, fn(u_i)\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.28)$$

Karena hampir semua perhitungan berdasarkan pada matriks A dan vektor y, maka (a_iT, y_i) disebut pasangan data ke-i dari himpunan data pelatihan.

Bila m = n (A persegi) dan A nonsingular, maka θ dapat diperoleh dari persamaan (3.24) sebagai berikut.

$$\theta = A^{-1}y \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.29)$$

Tetapi biasanya m > n, berarti pasangan data pelatihan lebih banyak daripada parameter fitting. Dalam kasus ini solusi eksak yang memenuhi m persamaan tidak selalu mungkin, karena data bisa terkontaminasi dengan derau, atau model tidak sesuai untuk menggambarkan sistem target. Maka persamaan (3.24) harus dimodifikasi dengan menerapkan vektor galat e untuk memperhitungkan derau acak atau galat pemodelan sebagai berikut.

$$A\theta + e = y \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (3.30)$$

dengan jumlah *squared error* ditentukan oleh :

$$E(\theta) = \sum_{i=1}^{m} (y_i - a_i^T)^2 = e^T e = (y - A\theta)^2 (y - A\theta) \quad \ldots\ldots\ldots \quad (3.31)$$

dimana $e = y - A\theta$ adalah error vektor dihasilkan dari pemilihan θ.

- **Menentukan parameter Premis dengan Metode *Gradient Descent***

Dengan menggunakan *input-output* data dan algoritma *Gradient Descent* dan meminimisasi fungsi obyektif error. Dengan menuliskan kembali fungsi error kuadratik sebagai (Widodo, 2005)

$$E = \frac{1}{2}(y^{rp} - y^{p})^2 \quad \text{.....} \quad (3.32)$$

dimana :

y^{rp} = target untuk input data sejumlah p (x_i^p,...,x_m^p)

y^p = output yang telah dihitung dari sistem sesuai dengan jumlah input data.

Fungsi keanggotaan yang digunakan oleh J.-S. R. Jang adalah fungsi Bell, untuk membedakannya maka dalam penelitian ini akan menggunakan fungsi gaussian sebagai pembentuk fungsi keanggotaannya. Dengan menggunakan fungsi keanggotaan gaussian pada parameter premis sehingga parameter adaptif perkalian AND, perhitungan titik pusat output dan konstanta output seperti pada persamaan (3.32) fungsi keanggotaan Gaussian ditentukan oleh :

$$\mu_{ij} = e^{-\frac{1}{2}\left(\frac{(x-c_{ij})}{a_{ij}}\right)^2} \quad \text{.....} \quad (3.33)$$

Dengan menggunakan gradient descent, fungsi keanggotaan dari persamaan premis akan disetel untuk merefleksikan aturan dengan lebih baik. Dalam penelitian ini menggunakan fungsi gaussian sebagai parameter premis ke-j dan aturan ke-i, fungsi keanggotaan ini ditunjukan dengan puncak c_{ij} dan variabel pendukung a_{ij} sebagai keseluruh aturan pada persamaan (3.33) dapat ditetukan oleh nilai puncak c_{ij},dan a_{ij},

Parameter yang akan diadaptasi adalah (a_{ij},b_{ij},) dengan p menunjukan jumlah data pembelajaran dan y_r^p menunjukan output pembelajaran. Dengan menggunakan gradient descent, parameter a_{ij} akan diupdate menurut persamaan :

$$c_{ij}(t+1) = c_{ij}(t) - \frac{\mu_a}{p}\frac{\partial E}{\partial c_{ij}} \quad \text{.....} \quad (3.34)$$
$$= c_{ij}(t) - \frac{\mu_a}{2p}\sum_{p'=1}^{p}\frac{\partial Ep'}{\partial c_{ij}}$$

dimana μ_c adalah kecepatan gradient descent untuk c_{ij}, demikian halnya untuk fungsi keanggotaan aij

$$a_{ij}(t+1) = a_{ij}(t) - \frac{\mu_b}{p}\frac{\partial E}{\partial a_{ij}} \quad \text{.....} \quad (3.35)$$
$$= a_{ij}(t) - \frac{\mu_b}{2p}\sum_{p'=1}^{p}\frac{\partial Ep'}{\partial a_{ij}}$$

dimana μ_a adalah kecepatan gradient descent untuk a_{ij}

Untuk memudahkan perhitungan persamaan diatas digunakan E, y dan y' dari persamaan pada E^p, y^p dan y'^p untuk p input-output data. Dengan menggunakan persamaan diatas untuk mengupdate parameter a_{ij} didapatkan :

$$\frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial O_5}\cdot\frac{\partial O_5}{\partial O_4}\cdot\frac{\partial O_4}{\partial O_3}\cdot\frac{\partial O_3}{\partial O_2}\cdot\frac{\partial O_2}{\partial O_1}\cdot\frac{\partial O_1}{\partial O a_{ij}} \quad \text{.....} \quad (3.36)$$

dengan fungsi keanggotaan gaussian dan produk perkalian AND, turunan parsial dari persamaan adalah

$$\frac{\partial E}{\partial y} = y - y' \quad \text{.....} \quad (3.37)$$

Turunan parsial $\frac{\partial y}{\partial w_i}$ adalah

$$\frac{\partial y}{\partial w_i} = \frac{\mu_i}{\sum_{i'=1}^{n}\mu_i'} \quad \text{.....} \quad (3.38)$$

dan kita dapatkan juga (dengan y' sebagai konstanta)

Penurunan dari $\frac{\partial O_j}{\partial O_{j-1}}, j = 5,...,2$ adalah sebagai berikut

$$\frac{\partial O_5}{\partial O_4} = \frac{\partial y}{\partial O_4} = \frac{\partial(\sum f_i \overline{w}_i)}{\partial f_i \overline{w}_i} = 1 \quad \text{.....} \quad (3.39)$$

dimana w_i adalah *normalised firing strength* (kuat penyulutan ternormalisasi) untuk rule ke-i

$$\frac{\partial O_4}{\partial O_3} = \frac{\partial f_i \overline{w}_i}{\partial \overline{w}_i} = f_i \quad \text{.....} \quad (3.40)$$

Dimana i adalah rule yang bersangkutan

$$\frac{\partial O_3}{\partial O_2} = \frac{\partial}{\partial w_i}\left(\frac{w_i}{\sum_{j=1}^{n} wi}\right) = \frac{\sum_{j=1}^{n} w_j - w_i}{\left(\sum_{j=1}^{n} w_j\right)^2} \quad \text{.....} \quad (3.41)$$

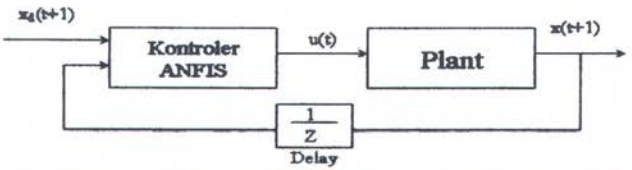
dan

$$\frac{\partial O_2}{\partial O_1} = \frac{\partial}{\partial Am}\left(\prod_{A_j \in \Re(A_m)} A_j\right) = \prod_{A_j \in \Re(A_m), A_j \neq A_m} A_j \quad \text{.....} \quad (3.42)$$

$$\frac{\partial O_1}{\partial c_{ij}} = \frac{u_{ij}(x_j - c_{ij})}{a_{ij}^2} \quad \text{.....} \quad (3.43)$$

$$\frac{\partial O_1}{\partial a_{ij}} = \frac{u_{ij}(x_j - c_{ij})^2}{a_{ij}^3} \quad \text{.....} \quad (3.44)$$

Dari gambar 3.13 diatas tampak bahwa proses identifikasi yang dilakukan sebelumnya diimplementasikan, dimana kontroler ANFIS merupakan hasil implementasi dari ANFIS kontroler. Kontroller ANFIS diterapkan di dalam sistem, digambarkan sebagai berikut



Gambar 3.16 Skema dasar ANFIS sebagai kontroler

----------(HALAMAN INI SENGAJA DIKOSONGKAN)----------

BAB IV

ANALISA DATA DAN PEMBAHASAN

Pada bab ini akan menerangkan dan menampilkan hasil perancangan sistem pengendalian kecepatan steam turbine yang telah dilakukan dengan menggunakan *software* Matlab 7 R14 dalam bentuk M-file dan Simulink kemudian akan dilakukan suatu simulasi yang diikuti dengan analisa data yang diperoleh dan pembahasan dari simulasi tersebut.

Tahap simulasi yang dilakukan dibagi menjadi dua. Yang pertama adalah simulasi pada sistem kontroler PI untuk mengetahui respon sistem pengendalian kecepatan *steam turbine* menggunakan kontroler PI, seperti yang ada pada *plant* yang sebenarnya dan sebagai model proses pembelajaran kontroler *Adaptive Neural-Fuzzy Inference System (*ANFIS) untuk mendapatkan parameter-parameter yang dibutuhkan, kemudian dilanjutkan pada simulasi pada sistem *closed-loop*, dimana pada pengujian ini akan dilakukan dengan mengubah nilai *setpoint,* pemberian nilai beban yang berubah-ubah.

Pemodelan kecepatan turbin uap pada penelitian ini menggunakan metode (ANFIS) dengan struktur 2 input dan 1 output dan proses *Inverse Learning*, dimana variabel *input* ANFIS mengandung *input* (x) diasumsikan output dari kecepatan *steam turbine* masa sekarang x(t+1) dan masa lampau x(t) dan *output* (y) diasumsikan sebagai sinyal kontroler PI. Untuk mendapatkan model yang baik, *ANFIS* diuji cobakan dengan mengubah-ubah nilai *Learning rate*. Proses *training* ANFIS ini dilakukan secara *offline* dengan data set yang diambil dari data kontroler PI yang telah disimulasikan terlebih dulu sejumlah 1.000 data set. Waktu pemodelan proses adalah 500 iterasi. Proses *training* akan berhenti ketika RMSE target bernilai $1x10^{-3}$ telah terpenuhi, jika target tidak terpenuhi proses *training* berkerja sampai jumlah iterasi. Setelah parameter diperoleh akan disimulasikan pada pengujian kontroler ANFIS.

Untuk mengetahui performansi dari sistem pengendalian yang telah dirancang, maka melalui pengujian pada *setpoint* yang berbeda akan didapatkan nilai performansi sistem. Beberapa parameter yang dapat menyatakan performansi sistem dapat dinyatakan baik secara kualitatif maupun kuantitatif. Beberapa parameter kualitatif yang dipergunakan untuk menganalisa performansi sistem pada pengerjaan Tugas Akhir ini adalah *Rise-Time* (Tr), *Settling Time* (Ts) dan *Maximum Overshoot* (Mp)

Beberapa parameter kualitatif antara lain :

a. *Rise Time Tr*
   Didefinisikan sebagai waktu yang diperlukan respon untuk naik dari 10 % sampai 90 % dari harga akhirnya atau steady state nya.

b. *Settling Time* (Ts)
   Didefinisikan sebagai waktu yang diperlukan untuk respon agar mencapai keadaan stabil (keadaan tunak) atau dianggap stabil atau harga dalam kisaran nilai *set point* yang disederhanakan dengan persentase mutlak harga *set point* (5%),

c. Persen maksimum *Overshoot* (Max. Ov)
   Didefinisikan sebagai prosentase maksimum nilai puncak terhadap nilai *set point*. Besarnya persen overshoot maksimum menunjukkan kestabilan relatif dari sistem

$$Mp(\%) = \frac{C(t_p) - C(\infty)}{C(\infty)}$$

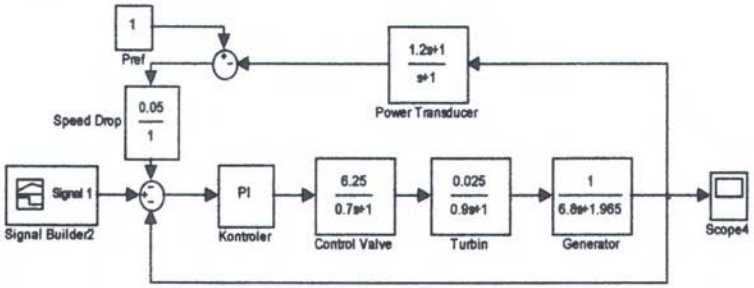
   Dimana C(tp) adalah *output* respon pada waktu puncak (tp) dan C(∞) adalah *output* respon pada saat tak terhingga.

4.1 Analisa Data

4.1.1 Simulasi Sistem Kontroler PI

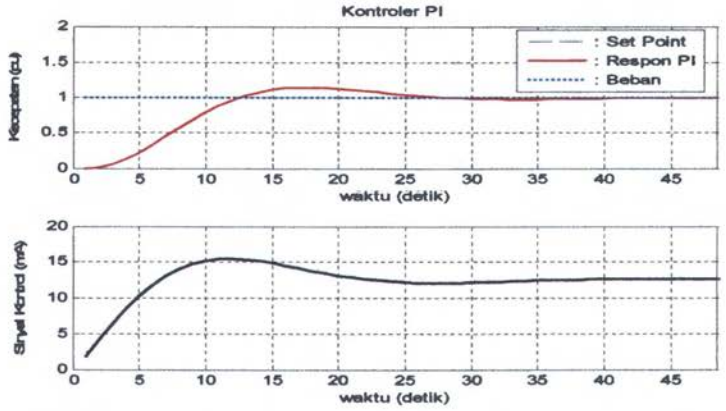
Pada sistem sistem pengendalian kecepatan *steam turbine* menggunakan kontroler PI merupakan simulasi sistem

pengendalian kecepatan turbin uap yang ada pada *plant* yang sebenarnya.



Gambar 4.1 Simulasi sistem Kontroler PI

Kontroler PI ini mempunyai nilai *gain proporsional* sebesar 1,9 dan nilai *integral time constan* sebesar 0,4 , akan di ujikan sinyal step berupa setpoint kecepatan = 1 pu dengan beban elektrik nol atau tanpa beban, didapatkan hasil seperti gambar 4.2 dibawah.



Gambar 4.2 Respons sistem kontroler PI pada pengendalian kecepatan *steam turbine*

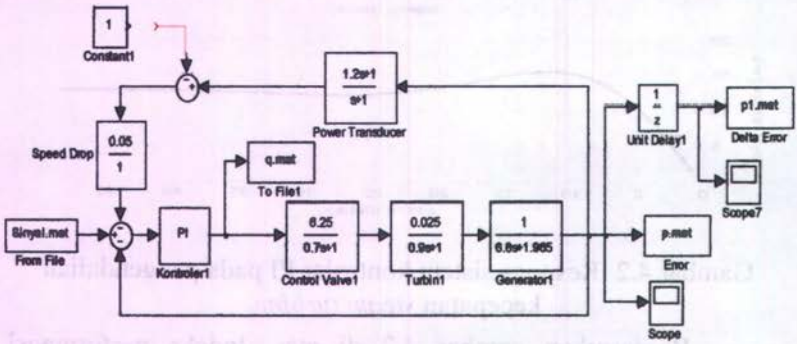
Berdasarkan gambar 4.2 di atas. Indeks performansi sebagai parameter kualitatif dari sistem pengendalian adalah :

- *Rise Time* = 7.4744 detik
- *Settling Time* = 23.5562 detik
- *Max. Overshoot* = 14.5163 %

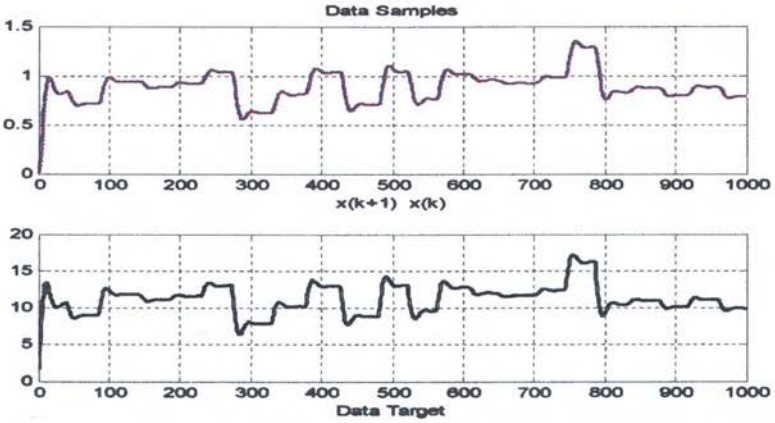
Dari grafik diketahui bahwa plant menghasilkan respons sesuai dengan *set point* yang diinginkan yaitu 1 pu dan mengikuti sinyal kontrol yang ada. waktu yang dibutuhkan respon mencapai kestabilan (*steady state*) = 24 detik, kestabilan terjadi ketika respon berada sekitar 5 % di sekitar set point. Maximum Overshoot terjadi pada sekitar detik ke-18. Respon ini dapat dikatakan cukup baik karena mengalami kestabilan tanpa osilasi.

4.1.2 Training Kontroler ANFIS

Untuk mendapatkan pemodelan plant, dilakukan Simulasi untuk mendapatkan pasangan data input (u(k)) berupa output dari kontroler PI dan output proses yang diasumsikan sebagai output masa lampau x(k) & masa sekarang x(k+1), input yang diberikan berupa *Amplitudo modulated Pseudo Random Binary Signal (APRBS)* dengan nilai rentang 0.6 sampai dengan 1.4 pu. Kemudian akan di ujikan dengan proses *training* untuk mendapatkan nilai parameter yang terbaik. Berikut adalah gambar diagram blok sistem pengambilan data input-output dari pengendalian kecepatan turbin uap.

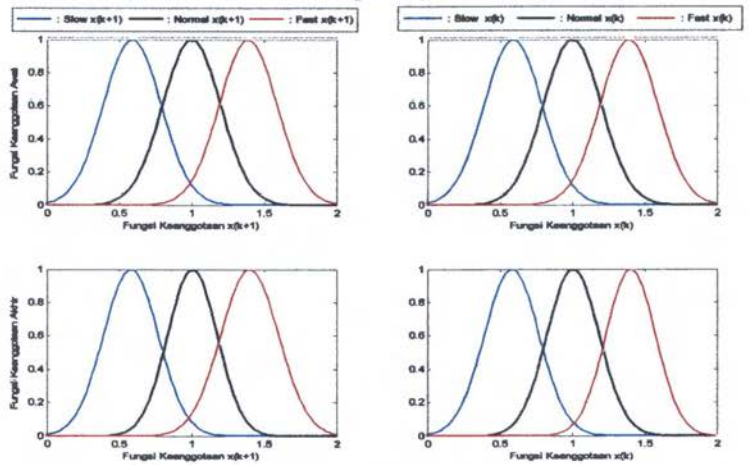


Gambar 4.3 Simulasi sistem pengambilan data input-output



Gambar 4.4 Data Sampel dan Data Target

Proses *training* pada dasarnya adalah untuk menentukan bobot *layer input (*parameter Premis*)* dan bobot *layer output* (parameter konsekuen). Setelah mendapatkan pasangan data input-output, akan di lakukan training untuk mendapatkan nilai parameter premis dan konsekuen yang paling optimal. Grafik hasil *training*-nya dapat dilihat pada gambar dibawah ini :



Gambar 4.5 Fungsi Keanggotaan Awal dan Akhir

Pada grafik baris pertama diatas merupakan inisialisasi grafik x(k+1) dan x(k) sebelum dilakukan training, yang masing-masing mempunyai 3 fungsi keanggotaan berbentuk *gaussian* yaitu *slow, normal* dan *fast*. Fungsi keanggotaan ini mewakili kecepatan turbin. Pada grafik baris kedua merupakan hasil dari prose training ANFIS. Dengan nilai parameter yang diperoleh adalah sebagai berikut :

Tabel 4.1 Nilai parameter Premis awal dan akhir Training

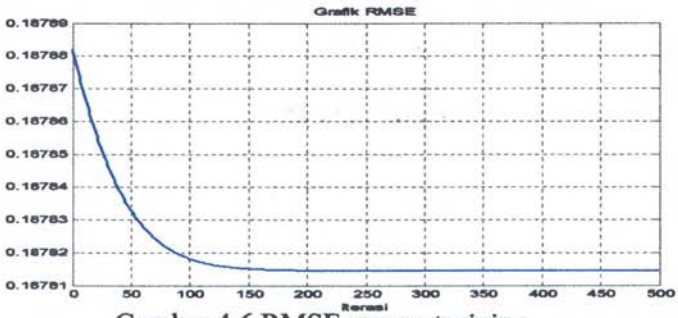
Input	MF	Nilai Awal		Nilai Akhir	
		Cp	Ap	Cp	Ap
Input X x(k+1)	X1 (slow)	0.6	0.2	0.5913	0.18856
	X2 (normal)	1	0.2	1.0062	0.17075
	X3 (fast)	1.4	0.2	1.4112	0.20000
Input Y x(k)	Y1 (slow)	0.6	0.2	0.5913	0.18866
	Y2 (normal)	1	0.2	1.0059	0.18003
	Y3 (fast)	1.4	0.2	1.4313	0.17075

Dapat dilihat untuk masing-masing *input* mempunyai 3 fungsi keanggotaan, yaitu *slow, normal* dan *fast*, dengan tiap-tiap fungsi keanggotaan mempunyai 2 parameter untuk menentukan bentuk dari himpunan keanggotaan yang berupa fungsi *Gaussian*, yaitu titik pusat (tengah) fungsi keanggotaan (Cp) dan lebar fungsi keanggotaan (Ap), sehingga untuk parameter premis mempunyai 12 jumlah nilai. Sebelum proses *training*, diberikan nilai awal untuk parameter-parameter ini. Pada saat *training* terjadi langkah balik sinyal error antara keluaran yang diinginkan dan keluaran dirambatkan mundur sedangkan parameter premis tersebut akan diperbarui dengan metode *gradient descent*.

Tabel 4.2 Nilai parameter Konsekuen

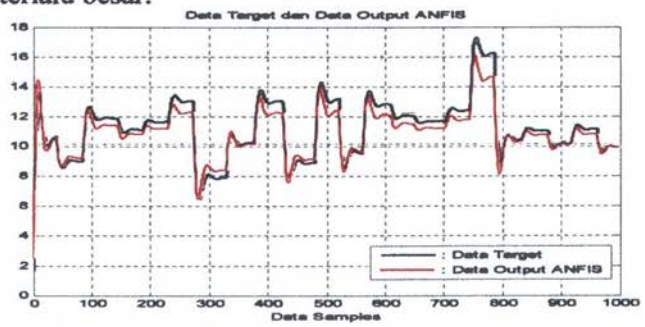
	Nilai Awal			Nilai Akhir		
	p	Q	r	p	q	r
$\overline{w_1}$	0	0	0	61.6573	-52.5824	2.5334
$\overline{w_2}$	0	0	0	68.0665	-58.1990	0.4623
$\overline{w_3}$	0	0	0	79.5322	-70.5437	4.6358

Tabel diatas merupakan tabel nilai parameter konsekuen merupakan nilai pembobotan pada layer keempat pada sistem ANFIS. Untuk semua parameter nilai awal sebelum dilakukan training diberikan nilai 0. Nilai ini diidentifikasi pada saat proses *training* ANFIS secara langkah maju dengan metode *Least Square Estimator Recursive*. Untuk nilai akhir merupakan nilai parameter konsekuen hasil dari proses *training*.



Gambar 4.6 RMSE proses *training*

Dari hasil training dengan proses *inverse learning*, dengan nilai learning rate 0,00001 dan RMSE di set pada nilai 0,001. Dari grafik proses *training* didapatkan nilai RMSE 0.18781. RMSE merupakan jumlah nilai error antara data yang ditargetkan dengan data yang didapatkan selama proses training. RMSE yang terjadi jauh dari nilai yang diharapkan, hal ini mungkin di sebabkan karena rentang nilai fungsi keanggotaan yang terlalu besar.

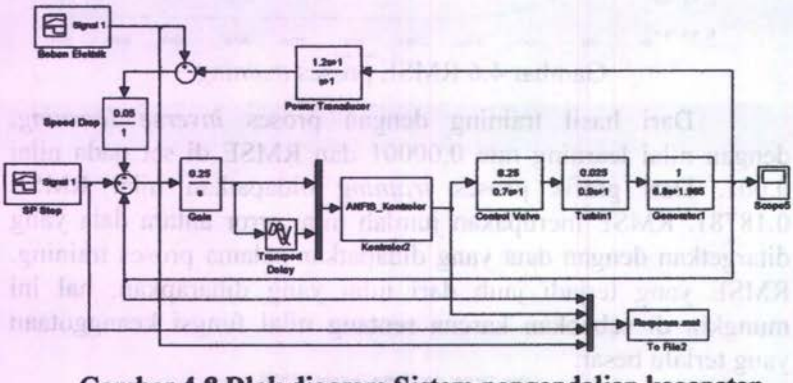


Gambar 4.7 Data Target dan Output pada Proses *training*

Sinyal hasil training menunjukan proses training kurang bisa mengikuti target yang diinginkan, hal ini mungkin disebabkan karena banyaknya parameter yang harus diidentifikasi. Dari grafik 4.7 bahwa data target yang mendekati nilai 9 dan 11, respons yang terjadi semakin baik dan error (RMSE) yang terjadi semakin kecil. Sebaliknya data target yang menjauhi nilai tersebut error (RMSE) nya menjadi lebih besar hal ini disebabkan range pada fungsi keanggotaan yang terlalu besar sehingga data menjadi kurang akurat.

4.1.3 Simulasi Kontroler ANFIS

Blok diagram Blok diagram Sistem pengendalian kecepatan turbin dengan ANFIS kontroler ditunjukkan pada Gambar 4.7.

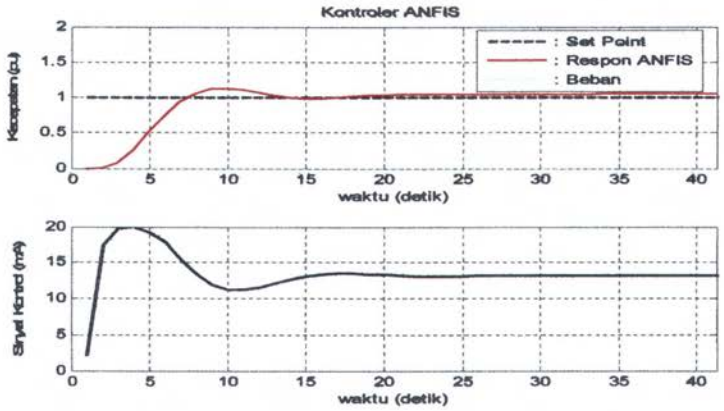


Gambar 4.8 Blok diagram Sistem pengendalian kecepatan turbin dengan ANFIS kontroler

A. Tanpa Beban

Pengujian tanpa beban atau beban nol untuk mengetahui performansi sistem dari kondisi awal ke kondisi siap pemberian beban ke generator, dengan melakukan pengujian pada setpoint sebagai berikut :

Dengan memberikan setpoint kecepatan = 1 pu, didapatkan hasil seperti gambar 4.9 dibawah.



Gambar 4.9 Respon sistem pada pengujian set point 1 pu tanpa beban.

Berdasarkan gambar 4.9 di atas. Indeks performansi sebagai parameter kualitatif dari sistem pengendalian adalah :

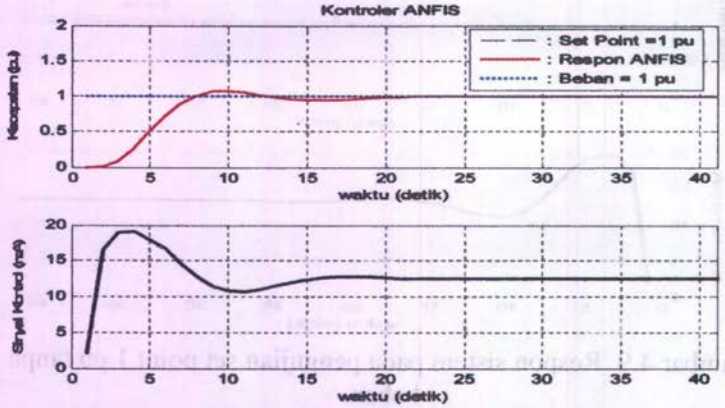
- *Rise Time* = 3.9261 detik
- *Settling Time* = - detik
- *Max. Overshoot* = 7.6298 %

Dari pengujian sistem dengan set point 1 pu tanpa beban, dilihat dari Error setady state nya bahwa sistem tersebut tidak mengalami kestabilan karena melebihi Error yang diijinkan yaitu 5%. Hal ini disebabkan karena ada nya Speed Drop yang terpasang pada sistem sebesar 5 %. Speed drop berguna sebagai toleransi sistem apabila adanya beban daya elektrik yang terpasang sebesar 1 pu. Dengan kata lain apabila pada beban penuh 1 pu (100%) dikehendaki kecepatan turbin adalah 1 pu atau 100 % dan untuk ini kecepatan turbin pada beban nol harus sebesar 105 %.

B. Dengan Beban

Pengujian set point ini merupakan uji sinyal step dengan beban daya listrik sebesar 1, dengan melakukan pengujian pada setpoint sebagai berikut :

Dengan memberikan setpoint kecepatan = 1 pu, didapatkan hasil seperti gambar 4.10 dibawah.

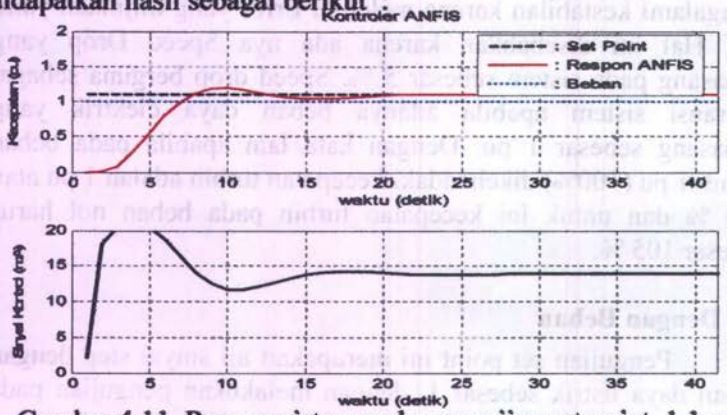


Gambar 4.10 Respon sistem pada pengujian set point 1 pu.

Berdasarkan gambar 4.10 di atas. Indeks performansi sebagai parameter kualitatif dari sistem pengendalian adalah :

- *Rise Time* = 3.9247 detik
- *Settling Time* = 16.0941 detik
- *Max. Overshoot* = 7.4001 %

Dengan memberikan kenaikan *set point* sebesar 1.1 pu, didapatkan hasil sebagai berikut :

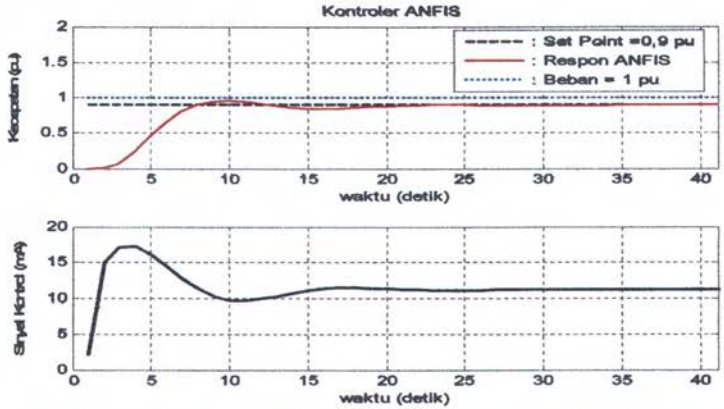


Gambar 4.11 Respon sistem pada pengujian set point 1,1 pu

Berdasarkan gambar 4.11 di atas. Indeks performansi sebagai parameter kualitatif dari sistem pengendalian adalah :

- *Rise Time* = 3.9158 detik
- *Settling Time* = 16.4107 detik
- *Max. Overshoot* = 7.9274 %

Dengan memberikan penurunan set point sebesar 0.9 pu, maka didapatkan hasil sebagai berikut :



Gambar 4.12 Respon sistem pada pengujian set point 0,9 pu

Berdasarkan gambar 4.12 di atas. Indeks performansi sebagai parameter kualitatif dari sistem pengendalian adalah :

- *Rise Time* = 3.878 detik
- *Settling Time* = 16.1126 detik
- *Max. Overshoot* = 6.6533 %

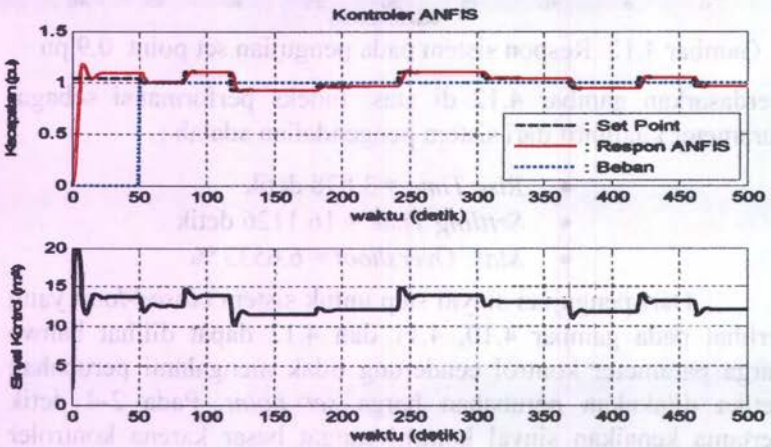
Dari pengujian sinyal step untuk sistem *closed-loop* yang terlihat pada gambar 4.10, 4.11 dan 4.12 dapat dilihat bahwa harga parameter kontrol cenderung tidak mengalami perubahan ketika dilakukan perubahan harga *set point*. Pada 2-4 detik pertama kenaikan sinyal kontrol sangat besar karena kontroler berusaha untuk menggerakan turbin dari posisi diam untuk bergerak dengan membuka penuh *control valve* sehingga *steam* yang masuk semakin besar. Semakin banyak steam yang masuk

akan menyebabkan turbin bergerak akibat *steam* melakukan aksi terhadap sudu (*blades).* Pada detik selanjutnya sinyal kontrol akan turun untuk menyesuaikan kecepatan turbin dengan bukan valve. Dapat dilihat juga bahwa respon sistem mampu mengikuti *set point* yang diberikan dengan cepat dengan nilai *rise time* sekitar 3-4 detik, nilai *settling time* sekitar 16-17 detik dan maksimal *overshoot* dibawah 5 % dan dengan nilai error *steady state* nya yang kecil mendekati nilai nol. Sistem pengendalian kecepatan *steam turbine* menggunakan ANFIS ini dapat mencapai kestabilan setelah mengalami osilasi yang tidak begitu besar dan yang langsung dapat dikendalikan oleh kontroler nya.

4.2 Pengujian *tracking*

4.2.1 Uji *tracking set point* dengan beban tetap

Uji *tracking set point*, diberikan sinyal random untuk set point dengan rentang 0.9 pu sampai dengan 1.1 pu. Uji *tracking set point* untuk mengetahui respon sistem terhadap perubahan set point secara acak didapatkan grafik sebagai berikut :



Gambar 4.13 Respon sistem pada pengujian *tracking set point*

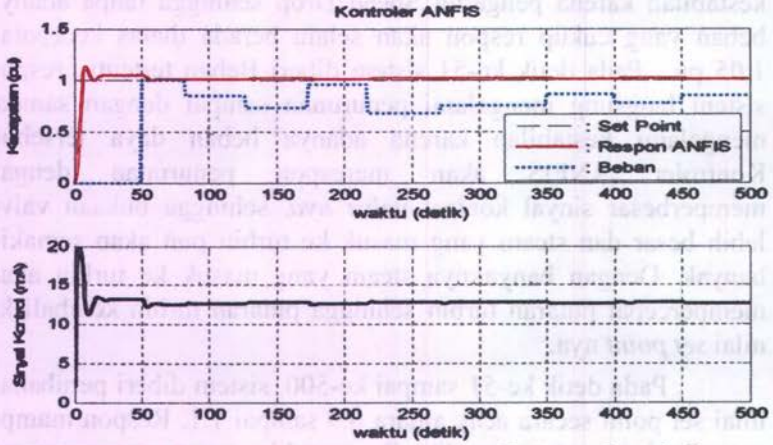
Pada uji *tracking set point* pada gambar 4.13 menunjukan dari detik ke-1 sampai ke-50 merupakan respon sistem dengan set

kestabilan karena pengaruh Speed Drop sehingga tanpa adanya beban yang cukup respon akan selalu berada diatas kecepatan 1.05 pu . Pada detik ke-51 sistem diberi Beban tertentu, respon sistem langsung mengalami penurunan sampai dengan sampai mengalami kestabilan karena adanya beban daya tersebut. Kontroler ANFIS akan merespon penurunan dengan memperbesar sinyal kontrol *valve nya,* sehingga bukaan valve lebih besar dan steam yang masuk ke turbin pun akan semakin banyak. Dengan banyaknya steam yang masuk ke turbin akan mempercepat putaran turbin sehingga putaran turbin kembali ke nilai *set point* nya.

Pada detik ke-51 sampai ke-500, sistem diberi perubahan nilai set point secara acak antara 0.9 sampai 1.1. Respon mampu mengikuti *set point* yang diberikan meskipun sistem mempunyai beban 1 pu dan rata-rata respon stabil pada t = 17 detik. Dari pengujian *set point* kontroler ANFIS mampu mengikuti perubahan *set point* yang diberikan. Apabila *set point* naik kontroler akan menaikan sinyal kontrolnya untuk menambah *steam* yang masuk pada valve sehingga kecepatan *steam turbine* akan ikut naik. Apabila *set point* diturunkan sinyal kontrol akan menurunkan bukaan valve untuk mengurangi steam yang masuk pada *steam turbine* agar kecepatan steam turbine mengikuti *set point* yang diinginkan. Kontroler ANFIS ini dapat mengikuti *set point* nya mungkin disebabkan pada proses pembelajaran yang sebelumnya dilakukan dengan range yang sesuai walaupun range yang ada masih terlalu besar, *set point* masuk dalam *range* tersebut sehingga untuk mencapai kestabilan lebih cepat.

4.2.2 Uji set point tetap dengan tracking beban

Uji gangguan dilakukan untuk mengetahui performansi dari suatu sistem ketika diberi gangguan perubahan beban sistem secara acak dengan nilai *set point* nya tetap. Uji *tracking* beban disimulasikan apabila terjadi adanya perubahan daya elektrik ke generator yang disebabkan karena adanya perubahan permintaan beban daya elektrik dari konsumen.



Gambar 4.14 Respon sistem pada pengujian *tracking* beban

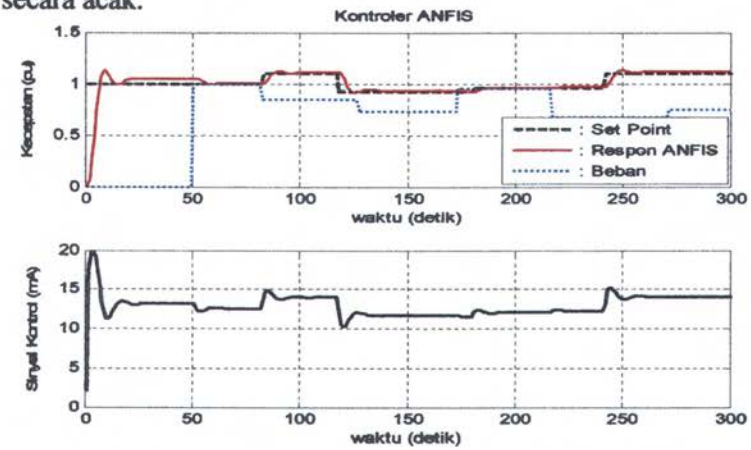
Untuk pengujian terhadap gangguan, Awalnya sistem diberi set point tertentu dan tanpa beban sampai detik ke-50. Setelah respon tidak mengalami osilasi tetapi masih belum mencapai kestabilan karena Error steady state nya diatas 5 %. Mulai pada detik ke-51 sistem diberi perubahan nilai beban secara acak antara 0.7 pu sampai 1 pu, pada awalnya respon mengalami penurunan yang cukup drastis. Namun kontroler ANFIS langung menaikan sinyal kontrol untuk memperbesar bukaan *valve* agar steam yang masuk semakin besar supaya kecepatan turbin kembali ke nilai *set point* nya.

Pada pemberian beban secara acak kecepatan turbin masih mengikuti set point yang diinginkan, walaupun terpengaruh adanya perubahan beban tersebut tetapi masih pada kondisi steady. Semakin besar beban yang diberikan maka penurunan kecepatan turbin juga akan semakin besar, hal ini disebabkan karena dengan pemberian beban akan menyebabkan torsi beban bertambah besar sehingga slip bertambah besar yang akan menyebabkan putaran turbin berkurang. Kecepatan turbin yang berkurang akan direspon dengan menaikan sinyal kontrol sehingga steam yang masuk semakin besar. Demikian juga apabila ada pelepasan beban, torsi beban semakin kecil sehingga

putaran turbin akan lebih cepat, kontroler akan berusaha mengurangi steam yang masuk ke turbin.

4.2.3 Pengujian gabungan (*tracking set point* dan beban)

Untuk lebih mengetahui performansi sistem, akan diberikan uji gabungan perubahan set point dan perubahan beban secara acak.



Gambar 4.15 Respon sistem pada pengujian gabungan

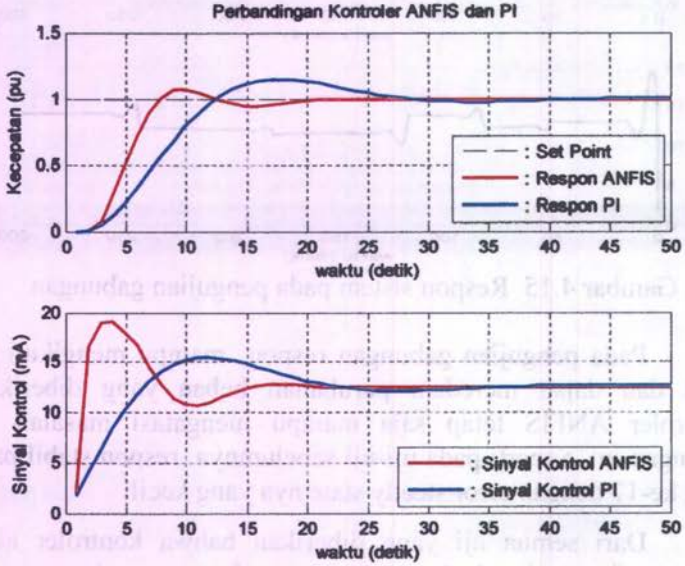
Pada pengujian gabungan respon mampu mengikuti set point dan dapat meredam perubahan beban yang diberikan. Kontroler ANFIS tetap saja mampu mengatasi masalah uji gabungan ini. Seperti pada uji-uji sebelumnya respon stabil pada detik ke-17 dengan error steady state nya yang kecil.

Dari semua uji yang diberikan bahwa kontroler akan menyesuaikan sinyal kontrolnya untuk memperbesar atau memperkecil bukaan *valve* agar steam yang masuk juga akan semakin besar atau semakin kecil untuk menyesuaikan perubahan *set point* atau perubahan beban yang diberikan sedemikian rupa sehingga kecepatan turbin sama dengan nilai *set point* nya.

4.3 Perbandingan Sistem Pengendalian

Pembandingan sistem pengendalian ini digunakan untuk melihat seberapa handal performansi dari sistem pengendalian yang telah dirancang dalam tugas akhir ini. Dalam penelitian ini sebagai pembanding adalah sistem pengendalian PI yang digunakan pada plant PLTU Gresik.

Dengan menggunakan simulasi untuk mendapatkan performasi respon dari sistem pengendalian PI, sehingga dapat diperoleh performansi sistem untuk uji set point dari pengendalian PI pada pengendalian kecepatan turbin sebagai berikut :



Gambar 4.16 Perbandingan Respon sistem dan sinyal kontrol Kontroler ANFIS dan PI tanpa beban

Untuk mempermudah dalam membandingkan sistem pengendalian yang telah dirancang dengan sistem pengendalian PI, maka dapat dilihat dalam hasil untuk pengendalian kecepatan turbin seperti pada tabel dibawah ini :

Tabel 4.3 Perbandingan performansi pengendalian kecepatan turbin

	ANFIS	PI
Rise Time (detik)	3.9247	7.4744
Settling Time (detik)	16.0941	23.5562
Maksimum Overshoot (%)	7.4001	14.5163

Mengacu dari hasil analisa yang telah dilakukan diatas, dapat dilihat bahwa pengendali ANFIS mampu menangani pengendalian kecepatan turbin dengan berbagai gangguan. Dan jika dilihat dari tabel 4.3 maka pengendali ANFIS yang telah dirancang dalam tugas akhir ini mempunyai performansi yang lebih baik daripada pengendali PI.

----------(HALAMAN INI SENGAJA DIKOSONGKAN)----------

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dalam tugas akhir ini telah dilakukan perancangan sistem pengendalian kecepatan *steam turbine* dengan metode *adaptive neuro-fuzzy inference system* (ANFIS) secara simulasi pada matlab 7.0.4. Dari pengujian dan penelitian yang dilakukan dapat disimpulkan beberapa hal sebagai berikut :

- Algoritma pembelajaran ANFIS dari sistem kontroler PI dengan invers learning mampu mendapatkan nilai RMSE 0.18781.
- Indeks performansi parameter ANFIS pada uji set point 1 pu sebagai berikut: Rise Time, Tr = 3,8766 detik; Settling Time, Ts = 16,0941 detik dan Maximum Overshoot, Mp = 7,4001 %
- Dari perbandingan Indeks performansi parameter kontrol Settling Time dan Maximum Overshoot, Kontroler ANFIS mempunyai performansi yang lebih baik daripada Kontroler PI pada sistem pengendalian kecepatan *steam turbine* dan dapat diterapkan sebagai kontroler kecepatan *steam turbine*

5.2 Saran

Dari hasil tugas akhir ini dapat diberikan saran untuk pengembangan penelitian selanjutnya antara lain:

- Agar dalam melakukan *training* ANFIS menggunakan model *learning* yang lain
- Agar dalam penelitian selanjutnya menggunakan proses MIMO dalam Sistem Tenaga Listrik. Sehingga dapat diketahui lebih banyak keunggulan dari beberapa outputan seperti *Frekuensi* dan *tegangan*.
- Agar dalam penggunaan ANFIS controller mendatang dapat diterapkan dalam proses yang lain sehingga dapat diketahui keunggulan kontroler ini.

----------(HALAMAN INI SENGAJA DIKOSONGKAN)----------

DAFTAR PUSTAKA

[1] Jang, J.-S.R. & Sun, C.-T. & Mizutani, E. 1994. ***Neuro-Fuzzy and Soft Computing : a Computational Approach to Learning and Machine Intelligence***. New Jersey : Prentice-Hall International, Inc. Englewood Cliffs.

[2] Marsudi, Djiteng. 2006. **Operasi Sistem Tenaga Listrik.** Yogyakarta : Graha Ilmu.

[3] Robandi, Imam. 2006. **Desain Sistem Tenaga Modern**. Yogyakarta : Andi.

[4] Tripathy, S. C. 1997. **Digital Governor For Use in Computer Control Of Generating Unit**. New Delhi : Indian Institute of Technology.

[5] Wahyunggoro, Oyas., Thomas Sri Widodo, dan Raditya. 2005. **Simulasi Kendali Sistem Elektro-Hidrolik Berbasis *Adaptive Neuro-Fuzzy Inference System* (Anfis).** Yogyakarta : Jurusan teknik Elektro UGM.

[6] Widodo, Thomas S. 2005. **Sistem Neuro Fuzzy : Untuk pengolahan Informasi, Pemodelan dan Kendali**. Yogyakarta : Graha Ilmu.

[7] Adi, Anton, 2000 **Studi dan Penerapan Model Neuro-Fuzzy Dalam Prakiraan Cuaca** , S1 Jurusan Teknik Fisika ITB.

[8] Florida State University. 2004. **EEL 6266. Power System Operation and Control** :Chapter 9 Control of Generation.

[9] Gunterus, frans. 1994. **Falsafah Dasar : Sistem Pengendalian Proses**. Jakarta : PT Elex Media Komputindo.

[10] Hímer, Zoltán Vincent Wertz, Jenő Kovács, Urpo Kortela. 2005. **NEURO- FUZZY MODEL OF FLUE**

GAS OXYGEN CONTENT. University of Oulu, Systems Engineering Laboratory,

[11] Huddleston, Ted. 2006. **Turbines & Compressors.** Dept of Chemical Engineering Univ of South Alabama Mobile, AL

[12] Kundur, P. 1994. **Power System Stability and Control,** McGraw-Hill, Section 12.5.

[13] MENTERI ENERGI DAN SUMBER DAYA MINERAL REPUBLIK INDONESIA. 2007. **PERATURAN MENTERI ENERGI DAN SUMBER DAYA MINERAL NOMOR : 03 TAHUN 2007 TENTANG ATURAN JARINGAN SISTEM TENAGA LISTRIK JAWA-MADURA-BALI**

[14] Pearmine, Ross Stuart. 2006. **Review Of Primary Frequency Control Requirements On The Gb Power System Against A Background Of Increasing Renewable Generation** School of Engineering and Design, Brunel University. UK

[15] Wang, Li-Xin.1994. **Adaptive Fuzzy Systems And Control: Design and Stability.** Prentice Hill International, Inc.

[16] www.mhi.co.jp

DATA TEKNIK PERALATAN

PERALATAN UTAMA PLTU I/II

1. KETEL

Tipe	: IHI SN
Pabrik	: Ishikawajima–Harima Heavy Industries Co,LTD
Kapasitas	: 400.000 Kg/jam
Tekanan Perencanaan	: 110 Kg/Cm2
Temperatur Perencanaan	: 513 derajat Celcius
Bahan Bakar Utama	: Residu
Tahun Pembuatan	: 1980

2. TURBIN

Tipe	: SC – 26
Pabrik	: Toshiba
Putaran	: 3 000 Rpm
Kapasitas Nominal	: 100 Mega Watt (MW)
Ekstraksi	: 5 Buah
Kondisi Uap Masuk	
Tekanan	: 88 Kg/Cm2
Temperatur	: 510 Derajat Celcius
Tekanan Buang (abs)	: 65 mmHg
Tahun Pembuatan	: 1980

3. GENERATOR

Tipe	: TAKS
Pabrik	: Toshiba
Kapasitas	: 125 000 kVA
Arus	: 5.468 ampere
Tegangan	: 13,2 kV
Putaran	: 3.000 Rpm
Frekwensi	: 50 Hz
Faktor Daya	: 0,8
Fasa	: 3 Fasa
Kutub	: 2 Buah
Arus Penguat	: 1.310 A
Kelas Isolasi	: B
Pendingin	: Gas Hidrogen
Tekanan Gas	: 2 Kg/Cm2
Tahun Pembuatan	: 1980

4. TRANSFORMATOR UTAMA

Tipe	: Outdoor use, oil immersed forsedoil forsed air cooled (OFAF), three phases, two windings
Pabrik	: Toshiba
Kapasitas	: 120.000 kVA
Tegangan	: 13,2 kV/154 kV
Frekwensi	: 50 Hz

ITS

Table Data Parameter Turbin & Generator Unit I
PLTU Gresik

No.	Nama Parameter	Besar (pu)
1.	T_A	0,04 detik
2.	K_A	300
3.	M	6,8
4.	T_{do}	6,2 detik
5.	T_{Gu}	0,90 detik
6.	K_{Gu}	20
7.	T_{Tu}	0,025 detik
8.	Wo	6,28
9.	K_{CV}	6.25
10.	T_{CV}	0.7 detik
11.	D_1	1,965
12.	K1	1.2 detik
13.	K2	1 detik
14.	Kp	0,05
15.	K_F	0,04
16.	T_F	0,5 detik
17.	T_E	0,9 detik
18.	K_E	1,0
19.	K_R	1,0
20.	T_R	0,06 detik
21.	R_a	0
22.	X_d	1,810
23.	X_{da}	0,308
24.	X_q	1,5
25.	X_{qa}	0,471
26.	X_t	0,0446
27.	P (base)	100 (MVA)

```
%=========================================================%
        % PROGRAM ANFIS INVERS LEARNING  %
%=========================================================%
clear all
clc
%Inisialisasi Harga Awal%
%----------------------%
load p.mat;
p=p(2,1:1000);
load p1.mat;
p1=p1(2,1:1000);
load q.mat;
q=q(2,1:1000);

k=p(1:1000);
dk=p1(1:1000);
yd=q(1:1000); %Target=Output
[n m] = size (yd); %jumlah data

epoh=0;
MaxEpoh=500;
rmse=1;
rmseMin=1e-3;
Lr=0.000001;   % Learning Rate untuk Update parameter Premis
%=========================================================%

figure(1)
subplot(2,1,1);
plot(k,'b-');
hold on
plot(dk,'r-');
grid
xlabel('x(k+1)  x(k)');
title('Data Samples');
```

```
subplot(2,1,2);
plot(yd,'k-');
grid
xlabel('Data Target')
%=========================================================%
%Inisialisasi parameter premis fungsi keanggotaan 1
%---------------------------------------------------------%
c11=0.6;        ca1=c11;
c12=1;          ca2=c12;
c13=1.4;        ca3=c13;

c21=0.6;        cb1=c21;
c22=1;          cb2=c22;
c23=1.4;        cb3=c23;

a11=0.2;        aa1=a11;
a12=a11;        aa2=a12;
a13=a11;        aa3=a13;

a21=0.2;        ab1=a21;
a22=a21;        ab2=a22;
a23=a21;        ab3=a23;

X=zeros(9,1);

while (epoh<MaxEpoh)& (rmse>rmseMin),
   epoh=epoh+1;
   disp(epoh);
   Eu=0;
   E=0;

for i=1:m
%perhitungan layer 1
%=========================================================%
mfx1(i)=exp(-0.5*((k(i)-c11)/a11)^2);
```

```
mfx2(i)=exp(-0.5*((k(i)-c12)/a12)^2);
mfx3(i)=exp(-0.5*((k(i)-c13)/a13)^2);

mfy1(i)=exp(-0.5*((dk(i)-c21)/a21)^2);
mfy2(i)=exp(-0.5*((dk(i)-c22)/a22)^2);
mfy3(i)=exp(-0.5*((dk(i)-c22)/a22)^2);
%perhitungan layer 2
%=========================================================%
W1(i)=mfx1(i)*mfy1(i);
W2(i)=mfx2(i)*mfy2(i);
W3(i)=mfx3(i)*mfy3(i);
sumW(i)=(W1(i)+W2(i)+W3(i));

%perhitungan layer 3
%=========================================================%
Wbar1(i)=W1(i)/sumW(i);
Wbar2(i)=W2(i)/sumW(i);
Wbar3(i)=W3(i)/sumW(i);
Wbar=[Wbar1' Wbar2' Wbar3'];

a1(i)=(k(i)*Wbar1(i)); b1(i)=(dk(i)*Wbar1(i));
c1(i)=(Wbar1(i));
a2(i)=(k(i)*Wbar2(i)); b2(i)=(dk(i)*Wbar2(i));
c2(i)=(Wbar2(i));
a3(i)=(k(i)*Wbar3(i)); b3(i)=(dk(i)*Wbar3(i));
c3(i)=(Wbar3(i));
end

X=[a1' b1' c1' a2' b2' c2' a3' b3' c3'];
%=========================================================%

% Hitung parameter konsekuan dengan LSE Rekursif
% Untuk mendapatkan parameter pada bagian then
R=inv(X'*X)*X'*yd';
```

```
f1=[(R(1,1)*k)+(R(2,1)*dk)+R(3,1)];
f2=[(R(4,1)*k)+(R(5,1)*dk)+R(6,1)];
f3=[(R(7,1)*k)+(R(8,1)*dk)+R(9,1)];

%perhitungan layer 4
%=========================================%
O1=Wbar1(i)*f1;
O2=Wbar2(i)*f2;
O3=Wbar3(i)*f3;

%perhitungan layer 5
%=========================================%
fz=O1+O2+O3;
data1=[yd fz (fz-yd)];
Eu=sum(data1(3)^2);
Eu=Eu+(yd(i)-fz(i))^2;

data1=[yd fz (yd-fz)];
E(i)=-1*(yd(i)-fz(i));
rmse=sqrt(1/m*Eu);

for i=1:m
F1(i)=((R(1,1)*k(i))+(R(2,1)*dk(i))+R(3,1));
F2(i)=((R(4,1)*k(i))+(R(5,1)*dk(i))+R(6,1));
F3(i)=((R(7,1)*k(i))+(R(8,1)*dk(i))+R(9,1));
end

for j=1:m
   C11(j)=E(i)*F1(j)*(sumW(j)-W1(j))/sumW(j)^2*W1(j)*(k(j)-
c11)/(a11^2);
   A11(j)=E(i)*F1(j)*(sumW(j)-W1(j))/sumW(j)^2*W1(j)*(k(j)-
c11)^2/(a11^3);
   C12(j)=E(i)*F2(j)*(sumW(j)-W2(j))/sumW(j)^2*W2(j)*(k(j)-
c12)/(a12^2);
```

```
   A12(j)=E(i)*F2(j)*(sumW(j)-W2(j))/sumW(j)^2*W2(j)*(k(j)-
c12)^2/(a12^3);
   C13(j)=E(i)*F3(j)*(sumW(j)-W3(j))/sumW(j)^2*W3(j)*(k(j)-
c13)/(a13^2);
   A13(j)=E(i)*F3(j)*(sumW(j)-W3(j))/sumW(j)^2*W3(j)*(k(j)-
c13)^2/(a13^3);

   C21(j)=E(i)*F1(j)*(sumW(j)-W1(j))/sumW(j)^2*W1(j)*(dk(j)-
c21)/(a21^2);
   A21(j)=E(i)*F1(j)*(sumW(j)-W1(j))/sumW(j)^2*W1(j)*(dk(j)-
c21)^2/(a21^3);
   C22(j)=E(i)*F2(j)*(sumW(j)-W2(j))/sumW(j)^2*W2(j)*(dk(j)-
c22)/(a22^2);
   A22(j)=E(i)*F2(j)*(sumW(j)-W3(j))/sumW(j)^2*W2(j)*(dk(j)-
c22)^2/(a22^3);
   C23(j)=E(i)*F3(j)*(sumW(j)-W3(j))/sumW(j)^2*W3(j)*(dk(j)-
c23)/(a23^2);
   A23(j)=E(i)*F3(j)*(sumW(j)-W3(j))/sumW(j)^2*W3(j)*(dk(j)-
c23)^2/(a23^3);
end

SC11=sum(C11);      SC21=sum(C21);
SC12=sum(C12);      SC22=sum(C22);
SC13=sum(C13);      SC23=sum(C23);

SA11=sum(A11);      SA21=sum(A21);
SA12=sum(A12);      SA22=sum(A22);
SA13=sum(A13);      SA23=sum(A23);

%Update nilai c,a baru
%-----------------------------------------------
c11=c11-(Lr*SC11); c21=c21-(Lr*SC21);
c12=c12-(Lr*SC12); c22=c22-(Lr*SC22);
c13=c13-(Lr*SC13); c23=c23-(Lr*SC23);
```

```
a11=a11-(Lr*SA11); a21=a21-(Lr*SA21);
a12=a12-(Lr*SA12); a22=a22-(Lr*SA22);
a12=a23-(Lr*SA13); a23=a23-(Lr*SA23);

RMSE(epoh)=rmse;
End

data1=[yd fz (yd-fz)];
save data1 data1

disp(['c11= ',num2str(c11),';']);
disp(['c12= ',num2str(c12),';']);
disp(['c13= ',num2str(c13),';']);

disp(['c21= ',num2str(c21),';']);
disp(['c22= ',num2str(c22),';']);
disp(['c23= ',num2str(c23),';']);

disp(['a11= ',num2str(a11),';']);
disp(['a12= ',num2str(a12),';']);
disp(['a13= ',num2str(a13),';']);

disp(['a21= ',num2str(a21),';']);
disp(['a22= ',num2str(a22),';']);
disp(['a23= ',num2str(a23),';']);

disp(['R(1,1)= ',num2str(R(1,1)),';']);
disp(['R(2,1)= ',num2str(R(2,1)),';']);
disp(['R(3,1)= ',num2str(R(3,1)),';']);
disp(['R(4,1)= ',num2str(R(4,1)),';']);
disp(['R(5,1)= ',num2str(R(5,1)),';']);
disp(['R(6,1)= ',num2str(R(6,1)),';']);
disp(['R(7,1)= ',num2str(R(7,1)),';']);
disp(['R(8,1)= ',num2str(R(8,1)),';']);
disp(['R(9,1)= ',num2str(R(9,1)),';']);
```

```
disp(['RMSE= ',num2str(rmse),';']);
%-------------------------------------------%

figure(2);
plot(RMSE,'b-');
grid
xlabel('Iterasi');
title('Grafik RMSE');
%-------------------------------------------%

figure(3);
plot(yd,'k-');
hold on
plot (fz,'r-');
grid
legend(': Data Target',': Data Output ANFIS')
xlabel('Data Samples');
title('Data Target dan Data Output ANFIS');
%-------------------------------------------%

figure(4);
subplot(2,2,1);
x=linspace(0,2,100);
mfx1=exp(-0.5*(((x-ca1)/aa1).^2));
mfx2=exp(-0.5*(((x-ca2)/aa2).^2));
mfx3=exp(-0.5*(((x-ca3)/aa3).^2));
plot(x,mfx1,'b-',x,mfx2,'k-',x,mfx3,'r-');
legend(': Slow x(k+1)',': Normal x(k+1)',': Fast x(k+1)')
ylabel ('Fungsi Keanggotaan Awal');
xlabel('Fungsi Keanggotaan x(k+1)');
%-------------------------------------%

subplot(2,2,2);
x=linspace(0,2,100);
mfy1=exp(-0.5*(((x-cb1)/ab1).^2));
```

```
mfy2=exp(-0.5*(((x-cb2)/ab2).^2));
mfy3=exp(-0.5*(((x-cb3)/ab3).^2));
plot(x,mfy1,'b-',x,mfy2,'k-',x,mfy3,'r-');
legend(': Slow  x(k)',': Normal x(k)',': Fast x(k)');
xlabel('Fungsi Keanggotaan x(k)');
%-------------------------------------%

subplot(2,2,3);
x=linspace(0,2,100);
mfx1=exp(-0.5*(((x-c11)/a11).^2));
mfx2=exp(-0.5*(((x-c12)/a12).^2));
mfx3=exp(-0.5*(((x-c13)/a13).^2));
plot(x,mfx1,'b-',x,mfx2,'k-',x,mfx3,'r-');
%legend(': Small x(k)',': Med x(k)',': Big x(k)');
ylabel ('Fungsi Keanggotaan Akhir');
xlabel('Fungsi Keanggotaan x(k+1)');
%-------------------------------------%

subplot(2,2,4);
x=linspace(0,2,100);
mfy1=exp(-0.5*(((x-c21)/a21).^2));
mfy2=exp(-0.5*(((x-c22)/a22).^2));
mfy3=exp(-0.5*(((x-c23)/a23).^2));
plot(x,mfy1,'b-',x,mfy2,'k-',x,mfy3,'r-');
%legend(': Small x(k)',': Med x(k)',': Big x(k)');
xlabel('Fungsi Keanggotaan x(k)');
drawnow
```

```
function [sys,x0,str,ts] = ANFIS(t,x,in,flag)
% Dispatch the flag. The switch function controls the calls to
% S-function routines at each simulation stage.
switch flag,
  case 0
    [sys,x0,str,ts] = mdlInitializeSizes; % Initialization
  case 3
    sys = mdlOutputs(t,x,in); % Calculate outputs
  case { 1, 2, 4, 9 }
    sys = []; % Unused flags
  otherwise
    error(['Unhandled flag = ',num2str(flag)]); % Error handling
end;

%=============================================================%
% Function mdlInitializeSizes initializes the states, sample
% times, state ordering strings (str), and sizes structure.
%=============================================================%
function [sys,x0,str,ts] = mdlInitializeSizes
% Call function simsizes to create the sizes structure.
sizes = simsizes;

% Load the sizes structure with the initialization information.
sizes.NumContStates= 0;
sizes.NumDiscStates= 0;
sizes.NumOutputs=    1;
sizes.NumInputs=     2;
sizes.DirFeedthrough=1;
sizes.NumSampleTimes=1;

% Load the sys vector with the sizes information.
sys = simsizes(sizes);
x0 = [];
str = [];
ts = [1 0]; % Inherited sample time
```

```
% End of mdlInitializeSizes.
% Function mdlOutputs performs the calculations.
%=============================================%

function [fz] = mdlOutputs(x,t,in)
clc;
k=in(1);
dk=in(2);

c11= 0.59127;
c12= 1.0062;
c13= 1.4112;
c21= 0.59132;
c22= 1.006;
c23= 1.413;
a11= 0.18856;
a12= 0.17075;
a13= 0.2;
a21= 0.18866;
a22= 0.18003;
a23= 0.17075;
R(1,1)= 61.6573;
R(2,1)= -52.5824;
R(3,1)= 2.5334;
R(4,1)= 68.0665;
R(5,1)= -58.199;
R(6,1)= 2.4623;
R(7,1)= 79.5322;
R(8,1)= -70.5437;
R(9,1)= 4.6358;

%perhitungan layer 1
%=============================================%
mfx1=exp(-0.5*((k-c11)/a11)^2);
mfx2=exp(-0.5*((k-c12)/a12)^2);
```

```
mfx3=exp(-0.5*((k-c13)/a13)^2);

mfy1=exp(-0.5*((dk-c21)/a21)^2);
mfy2=exp(-0.5*((dk-c22)/a22)^2);
mfy3=exp(-0.5*((dk-c22)/a22)^2);

%perhitungan layer 2
%=========================================================%
W1=mfx1*mfy1;
W2=mfx2*mfy2;
W3=mfx3*mfy3;
sumW=(W1+W2+W3);

%perhitungan layer 3
%=========================================================%
Wbar1=W1/sumW;
Wbar2=W2/sumW;
Wbar3=W3/sumW;

p1=(k*Wbar1); q1=(dk*Wbar1); r1=(Wbar1);
p2=(k*Wbar2); q2=(dk*Wbar2); r2=(Wbar2);
p3=(k*Wbar3); q3=(dk*Wbar3); r3=(Wbar3);

X=[p1' q1' r1' p2' q2' r2' p3' q3' r3'];

f1=[(R(1,1)*k)+(R(2,1)*dk)+R(3,1)];
f2=[(R(4,1)*k)+(R(5,1)*dk)+R(6,1)];
f3=[(R(7,1)*k)+(R(8,1)*dk)+R(9,1)];

%perhitungan layer 4
%=========================================================%
O1=Wbar1*f1;
O2=Wbar2*f2;
O3=Wbar3*f3;
```

```
%perhitungan layer 5
%=========================================================%
fz=O1+O2+O3;

out(1)=fz;
```